Université du Québec
à Rimouski

Contribution à la détection intelligente des défauts géométriques des infrastructures de transport par vision transformers et réseaux de neurones convolutifs

Mémoire présenté

dans le cadre du programme de maîtrise en informatique

en vue de l'obtention du grade de maître ès sciences (M. Sc.)

PAR

© Samira Mohammadi

Mars 2025

Composition du jury :

Jean-François Méthot, Président du jury, UQAR

Mehdi Adda, Directeur de recherche, UQAR

Sasan Sattarpanah Kargandroudi, Codirecteur de recherche, UQTR

Haïfa Nakouri, Examinateur Externe, UQAC

Vahid Rahmanian, Membre Externe, UQTR

Dépôt initial le 11 Septembre 2024 Dépôt final le 6 Mars 2025

À mes chers parents qui m'ont soutenu tout au long de mon parcours.

REMERCIEMENTS

Je voudrais profiter de cette occasion pour exprimer ma profonde gratitude à tous ceux qui ont contribué, directement ou indirectement, à l'achèvement de mon projet. Leur soutien, leur expertise et leurs encouragements ont été essentiels pour mener ce travail à bien. Je suis profondément reconnaissante envers mon directeur de thèse, Mehdi Adda, pour son engagement indéfectible et ses conseils inestimables tout au long de ce parcours. Ses connaissances approfondies, son expertise et ses conseils éclairés ont été une source constante d'inspiration. J'apprécie sa disponibilité, sa patience et les discussions intellectuellement stimulantes que nous avons eues.

Je tiens également à remercier chaleureusement mon co-directeur, Sasan Sattarpanah Kargandroudi, dont les idées, le soutien et les contributions ont considérablement enrichi la profondeur et la qualité de ce travail. Je suis reconnaissante pour son soutien et les efforts de collaboration qui ont amélioré le processus de recherche, contribuant ainsi à la richesse de cette dissertation. De plus, j'exprime ma gratitude aux membres de mon jury de thèse. Leurs commentaires constructifs et leurs suggestions ont grandement enrichi mon travail. Leur expertise et leur intérêt pour mon sujet ont été une source supplémentaire de motivation.

Je tiens à exprimer ma plus profonde gratitude à mon mari Mojtaba, et à mes enfants, Dina et Sina, pour leur amour et leur soutien. Merci à tous. Leur amour, leur soutien indéfectible et leurs encouragements ont été une véritable source de force et de motivation.

RÉSUMÉ

La gestion et la maintenance prédictives de l'infrastructure de transport reposent surl'application de techniques de contrôle non destructif et d'imagerie, qui s'avèrent crucialespour identifier les irrégularités sans causer de dommages, prévenant ainsi les accidents potentiels et les interruptions de service. Cette recherche utilise des modèles pré-entraînés et intègre des concepts de d'apprentissage par transfert pour surmonter les contraintes de l'ensemble de données. Elle met en lumière l'inspection géométrique significative de ces modèles dans l'automatisation de la classification d'images pour la maintenance des systems ferroviaires. Cette étude vise à évaluer l'efficacité de divers modèles d'apprentissage automatique, notamment le Vision Transformer (ViT), le Data-efficient Image Transformer (DeiT), VGG19, VGG16 et Resnet50, dans l'amélioration du contrôle non destructif dans les voies ferrées. ViT se démarque comme le meilleur performer en raison de son efficacité d'apprentissage supérieure et de sa capacité de généralisation, augmentée par un ajustement précis des hyperparamètres. DeiT, VGG19, VGG16 et le Resnet50 démontrent des capacités efficaces de détection de défauts, bénéficiant d'un ajustement minutieux des hyperparamètres.

Mots clés : Détection des défauts ferroviaires, Apprentissage automatique, CNN, Transformateurs, Infrastructure de transport.

ABSTRACT

The implementation of Non-Destructive Evaluation (NDE) imaging techniques plays a pivotal role in identifying infrastructure anomalies without causing damage, thereby preventing potential accidents and service disruptions. his study focuses on assessing the efficacy of various machine learning models in enhancing NDE within railway infrastructure. Such an evaluation is fundamental to ensuring operational safety and reliability in rail transport systems. Examined models include the Vision Transformer (ViT), Data-efficient Image Transformer (DeiT), VGG19, VGG16, and ResNet50. ViT emerges as the top performer due to its superior learning efficiency and generalization capability, augmented by precise hyperparameter tuning. DeiT, VGG19, VGG16, and the vanilla CNN demonstrates effective defect detection capabilities, benefiting from careful hyperparameter tuning. The findings highlight the potential of these models to aid automated image classification for railway maintenance applications, emphasizing the crucial role of hyperparameter tuning in optimizing performance. This research not only advances machine learning applications but also contributes to enhancing NDE methodologies in railway safety and maintenance.

Keywords: Railway defect detection, Machine learning, CNN, Transformers, Transport infrastructure

TABLE OF CONTENTES

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

- **AI**
  Artificial Intelligence
- **CNN**
  Convolutional Neural Network
- **DeiT**
  Data-Efficient Image Transformer
- **Faster R-CNN**
  Faster Region-based Convolutional Neural Network
- **GPR**
  Ground-Penetrating Radar
- **IoT**
  Internet of Things
- **Mask R-CNN**
  Mask Region-based Convolutional Neural Network
- **NDE**
  Non-Destructive Evaluation
- **R-CNN**
  Region-based Convolutional Neural Network
- **ResNet**
  Residual Network
- **SGD**
  Stochastic Gradient Descent
- **SHM**
  Structural Health Monitoring
- **ViT**
  Vision Transformer
- **VGG**
  Visual Geometry Group
  (often refers to the family of models such as VGG16 and VGG19)
- **YOLO**
  You Only Look Once (an object detection family, e.g., YOLOv4, YOLOv5)

INTRODUCTION GÉNÉRALE

1. INTRODUCTION

Industry 4.0, also known as the Fourth Industrial Revolution, marks the trend towards automation and data exchange in manufacturing technologies. It involves modern technologies like the Internet of Things (IoT), cloud computing, AI, and cyber-physical systems. Originating from a German government initiative, Industry 4.0 aims to digitalize manufacturing, particularly in defect detection and quality control [1]. Traditional manual inspections are being replaced by smart technologies such as machine learning algorithms, which enhance accuracy and reduce the time and cost associated with defect identification [2]. IoT and cloud computing integration allow real-time monitoring and analysis, optimizing production processes and enabling predictive maintenance to minimize downtime[3]. Cyber-physical systems facilitate the interaction between physical and digital components, allowing real-time adjustments and enhancing manufacturing flexibility [4]. These technologies support the development of smart factories, which leverage big data analytics for efficient production management and customization[5]. HM is crucial in Industry 4.0 for maintaining infrastructure like bridges, buildings, and railways. Using sensors and data systems, SHM monitors structural health in real-time, enabling early detection of potential issues [6]. Integration with IoT provides real-time data transmission and remote monitoring, improving infrastructure management efficiency [7]. SHM aids disaster resilience by identifying structural weaknesses early, which is essential for the safety of railway operations [8, 9]. SHM, combined with cyber-physical systems, enhances infrastructure reliability and safety [10]. NDE techniques inspect materials and systems without causing damage, essential for quality control in Industry 4.0. Innovations in AI, machine learning, robotics, and sensor technology have advanced NDE methodologies, improving flaw detection and inspection efficiency [11]. These services enhance manufacturer productivity, compliance with international standards, and product quality [12].

NDE methods are particularly notable for their ability to detect and characterize issues without damaging materials [13] and are used widely in the aerospace and manufacturing industries [14]. Rail transit is a vital mode of transportation, contributing significantly to

economic growth and connectivity [15]. However, aging infrastructure and inadequate investment pose challenges, highlighting the need for regular maintenance to ensure safety and functionality [16, 17]. Regular checks and upkeep are crucial to lessen hazards and avoid disruptions in the railway system [18]. Technologies like aerial and terrestrial imaging, optical and laser scanning, and robotic inspections have transformed quality control in railways [19,20]. These methods improve the efficiency and effectiveness of track maintenance, crucial for ensuring rail safety and reliability [21-24]. The creation and implementation of innovative techniques and strategies are crucial. Machine learning and AI are increasingly used to analyze vast data from modern monitoring equipment, aiming to improve system reliability and lower maintenance costs and risks [25]. AI systems can continuously track the condition of rail tracks, alerting maintenance teams about minor issues before they develop into larger problems, thereby increasing the overall safety and dependability of rail services [26, 27].

Geometric inspections detect track deviations and irregularities, ensuring track alignment and safety. Utilizing advanced technologies like 3D scanning, these automated systems inspect parts much faster and with greater accuracy, crucial for precision-critical industries [28]. Automated inspections, despite higher initial investments, prove more cost-effective over time due to their efficiency and lower error rates [29]. Railway defect detection is critical for maintaining the safety and reliability of rail systems. NDE methods and advanced imaging technologies are essential for identifying flaws and defects in rail tracks, preventing accidents, and ensuring smooth operations [19, 30]. Machine learning and AI enhance defect detection by analyzing data from inspection technologies, allowing for preventive maintenance and real-time monitoring [26, 31]. This integration ensures that railway infrastructure remains safe, reliable, and efficient.

2. PROBLEMATIC

While several studies have investigated the application of CNNs and transformers in railway defect detection, these methods often struggle to effectively detect missing fasteners—small components that appear in complex and cluttered environments—and to

demonstrate robust generalizability across different railway systems. Furthermore, many existing approaches rely on large annotated datasets—often unavailable in real-world scenarios—hindering their broad practical adoption. As a result, there remains a persistent lack of comparative analyses between these architectures, especially in addressing issues such as domain shifts across varying track conditions and the computational overhead of different models. Bai et al. [32] introduced an improved YOLOv4 method for the efficient detection of railway surface defects, utilizing MobileNetv3 and deep separable convolution. Similarly, Zheng et al. [33] employed CNNs to detect rail surface and fastener defects, employing an improved YOLOv5 framework for localization and Mask R-CNN for surface defect detection. Sresakoolcha et al. [34] track geometry data for the detection of rail switch, crossing, fastener, and rail joint defects, employing supervised machine learning techniques such as deep neural networks and CNNs, as well as unsupervised methods like K-means clustering and association rules. Additionally, Xu et al. [35] used deep learning to recognize railway subgrade defects from GPR data, improving the Faster R-CNN model. Pre-trained models, especially on PASCAL VOC2007 boosted performance. Wei et al. [15] applied transfer learning to improve their fastener detection model, utilizing a pre-trained model trained on ImageNet. Wang et al. [36] a method to detect defects in split pins of high-speed railway catenary devices, utilizing transfer learning and pre-trained models for accuracy. Wu et al. [37] used a pre-trained ResNet-101 model to initialize their fastener defect detection system, which was fine-tuned for detection. Lu [38]-tuned the pre-trained ResNet V2 and compared it to other models like Faster R-CNN, achieving higher accuracy in identifying defective joints. Li et al. [39] developed a method for rail defect detection using transfer learning and pre-trained models, outperforming single architectures like YOLOv5 and Faster R-CNN on an 8-class defect dataset. Jian et al. [40] an approach to railway defect detection utilizing transfer learning and multi-category defect detection. The emergence of transformer-based models, such as ViT and DeiT, has shown superior capabilities in various defect detection applications. While these methods have shown promise, many are predominantly CNN-based and do not fully explore how transformer-based solutions might address the unique challenges of railway defect detection.

Hutten et al. [41] conducted a systematic comparison, demonstrating that ViT can achieve performance equivalent to or better than CNNs, even with limited data. Additionally, recent studies by Alexakos et al. [42] An et al. [43] and Dang et al. [44] applied transformers to diverse defect detection tasks, further highlighting their efficacy. However, despite the potential of transformers, their utilization in railway defect detection remains unexplored. Furthermore, there is a noticeable gap in comparative studies between CNNs and transformers within this domain. Notably, transformer architectures offer interpretability benefits through their attention mechanisms, which could provide deeper insights into detection processes—advantages yet to be fully leveraged in railway applications. Additionally, due to the inherent scarcity of labeled data in railway defect detection, leveraging pre-trained models and transfer learning techniques becomes imperative to enhance model performance. Therefore, this study aims to address these gaps by investigating pretrained CNN and transformer models for missing fastener detection, incorporating transfer learning and hyperparameter tuning with Optuna [45], an open-source hyperparameter optimization framework that employs adaptive algorithms to automate the search for optimal hyperparameters and reduce manual trial-and-error. Through this approach, we seek to systematically optimize learning rates, batch sizes, and other critical parameters to improve accuracy and generalizability. In addition to measuring raw detection performance, we will evaluate the robustness and adaptability of these models under varied operational settings, paving the way for more reliable real-world deployment. By conducting a comparative analysis of both architectures and examining the efficacy of advanced optimization strategies, we seek to provide insights into the optimal approach for railway defect detection, contributing to safer and more efficient railway operations.

3. OBJECTIVE

This research aims to enhance railway track defect detection by leveraging transformer-based models (ViT, DeiT) and Convolutional Neural Networks (CNNs**)** to improve the accuracy, efficiency, and generalization capabilities of defect identification systems. Specifically, we focus on detecting missing fasteners—critical yet often subtle defects that

can compromise track integrity and safety—across diverse and challenging railway environments.

To achieve this, we systematically evaluate the performance of pretrained deep learning architectures, comparing the effectiveness of Vision Transformers (ViT, DeiT) and CNN-based models (ResNet50, VGG16, VGG19) in railway track defect classification. Given the limitations of conventional methods in detecting small-scale defects and adapting to varied conditions, our approach integrates transfer learning and advanced hyperparameter optimization (using Optuna) to refine model performance. This enables improved feature extraction, better adaptation to limited labeled data, and enhanced generalization across different railway conditions.

Additionally, we employ Non-Destructive Evaluation (NDE) techniques to assess track conditions without causing structural damage and incorporate Structural Health Monitoring (SHM) methodologies for continuous surveillance of railway infrastructure. By integrating these approaches, we aim to establish a reliable defect detection system that provides early failure warnings, minimizes maintenance costs, and enhances railway safety.

To ensure rigorous evaluation, we define clear performance metrics (accuracy, precision, recall, loss, and ROC AUC scores) to systematically compare model variations. Our research utilizes a publicly available Kaggle dataset of railway fastener images, divided into training, validation, and test sets using stratified sampling techniques to ensure balanced class representation.

Through a comparative analysis of our optimized models against existing baseline approaches, this research aims to develop a state-of-the-art defect detection framework that meets the evolving demands of modern railway networks, improving operational efficiency and long-term infrastructure safety.

4. HYPOTHESIS

Building on the gaps identified in the literature and our objectives, we propose four specific hypotheses regarding the detection of rail defects—particularly missing fasteners—

using CNNs and Transformer-based models (ViT, DeiT). These hypotheses also reflect the use of pre-trained models and advanced hyperparameter tuning (Optuna) to address the data-scarce context of railway defect detection.

1. Model Performance Hypothesis

    - Which methods? We will compare CNN-based models (ResNet50, VGG16, VGG19) to Transformer-based models (ViT, DeiT).

    - Expected Outcome: We hypothesize that these advanced models, when trained on our Kaggle rail defect dataset (840 defective, 840 non-defective), will significantly outperform simpler or unoptimized baselines in identifying missing fasteners.

    - By How Much? We anticipate at least a 1–2% improvement in accuracy, precision and recall over traditional baseline CNNs (e.g., a basic CNN architecture without transfer learning).

2. Hyperparameter Optimization Hypothesis

    - Which settings? We will specifically tune learning rate, momentum, dropout rate, and weight decay using Optuna. We will also explore variations in batch size and number of epochs (within practical limits) to determine their impact on performance.

    - Expected Outcome: Systematic hyperparameter optimization will yield higher detection accuracy for small-scale defects (i.e., missing fasteners) and better generalization across various railway conditions.

    - By How Much? We expect a 1–2% improvement in precision and recall compared to default hyperparameter settings, owing to more effective regularization and optimal learning rates.

3. Transfer Learning Hypothesis

- Which models? We will use pre-trained ResNet50, VGG16, VGG19 (trained on ImageNet) and pre-trained Transformers (ViT, DeiT) to leverage previously acquired feature representations.

- Expected Outcome: Incorporating transfer learning will enable the models to detect subtle rail defects (e.g., small or occluded fasteners) more effectively than training solely from scratch, particularly given the limited size of our labeled dataset.

- By How Much? We estimate a 1–2% increase in recall—especially for missing fasteners—compared to non-pre-trained variants, due to the richer feature representations learned from large-scale image datasets.

4. CNN vs. Transformer Hypothesis

    - Which methods? We will directly compare CNN architectures (ResNet50, VGG16, VGG19) with Transformer architectures (ViT, DeiT).

    - Expected Outcome: Owing to their global attention mechanisms, Transformer-based models may offer superior performance in complex backgrounds, whereas CNN-based models may excel in more localized feature extraction.

    - By How Much? We hypothesize that Transformers might achieve a 1–2% higher F1-score compared to CNNs under challenging conditions (e.g., cluttered environments, lower image quality), but possibly at the expense of longer training times.

5. SIGNIFICANCE OF STUDY

The study of applying the ViT, DeiT, ResNet50, and VGG16 and VGG19 models to a Kaggle dataset for rail defect detection is highly significant for several reasons. First, this approach can enhance accuracy in identifying defects on railway tracks.We can identify

which model works best for this specific task by applying different models and comparing their performance. We measure these improvements by evaluating key metrics—such as accuracy, precision, recall, and loss relative to simpler baseline models (e.g., basic CNNs without transfer learning) and default hyperparameter settings.This comparative analysis helps refine the detection process, leading to more reliable results. Secondly, studying these models helps in understanding their differences and nuances. Each model architecture has its strengths and weaknesses and exploring them on the dataset provides insights into which model is better suited for rail defect detection. Such understanding guides future model selection for similar tasks, ensuring optimal performance. Moreover, optimizing hyperparameters is crucial for maximizing a model's effectiveness. Researchers can find the optimal settings for each model by comparing different models with varied hyperparameters. We quantify the performance gains by comparing detection metrics against those achieved with untuned or default parameters, thereby demonstrating how hyperparameter tuning contributes to more accurate defect identification. This fine-tuning process enhances detection accuracy and optimizes the models in real-world scenarios. Additionally, leveraging pre-trained models like VGG19 and VGG16 through transfer learning is beneficial, especially in scenarios with limited data. These pre-trained models come with knowledge learned from large datasets, which can be transferred and fine-tuned on the rail defect dataset. This approach improves model performance even with a scarcity of training samples, making it particularly valuable when data is limited. Furthermore, comparing traditional CNNs to newer Transformer-based architectures like ViT and DeiT provides insights into the effectiveness of both approaches for rail defect detection. Any observed performance improvements are assessed by these architectures against each other, offering a clear view of how each method enhances detection capabilities over existing baselines.This analysis helps understand the applicability of transformer models in computer vision tasks compared to CNNs, thus informing future model selection and development efforts.

In our study, we focus on using image-based techniques for railway track defect detection. To address the specific defects we aim to detect, we include representative images of issues such as missing fasteners, and surface irregularities, providing a clear visual reference of the conditions our approach targets. This approach allows us to inspect railway

tracks without causing any damage, ensuring that the structural integrity is maintained while defects are detected. By utilizing images, we can continuously monitor the condition and performance of railway tracks, even in remote and hard-to-reach places. This helps identify potential issues early, preventing major problems and ensuring the safety and reliability of the railway system. By relying on image-based methods, we can develop a comprehensive and effective approach to railway track defect detection, ultimately enhancing the safety, reliability, and efficiency of railway networks.

6. METHODOLOGY

In our study, we focus on railway defect detection using machine learning, leveraging deep learning techniques to classify railway fasteners as defective or non-defective. We used a publicly available dataset from Kaggle, consisting of 700 defective and 700 non-defective images, representing real-world railway conditions. Our approach includes data preprocessing, training multiple models, hyperparameter tuning, and performance evaluation. We applied image transformations such as resizing, normalization, and data augmentation to enhance model generalization. To identify the most effective architecture, we tested various deep learning models, including Convolutional Neural Networks (CNNs) and Vision Transformers (ViT, DeiT). Transfer learning was employed using pre-trained models on ImageNet, replacing the classification layers with task-specific outputs. We also conducted hyperparameter tuning using Optuna, optimizing key parameters like learning rate, dropout rate, and batch size. Each model was evaluated using accuracy, precision, recall, and ROC-AUC, ensuring a robust assessment of their performance in railway defect detection. Below, we describe our steps for preparing the dataset, preprocessing data, selecting models, and tuning their parameters in detail.

6.1 DATASET

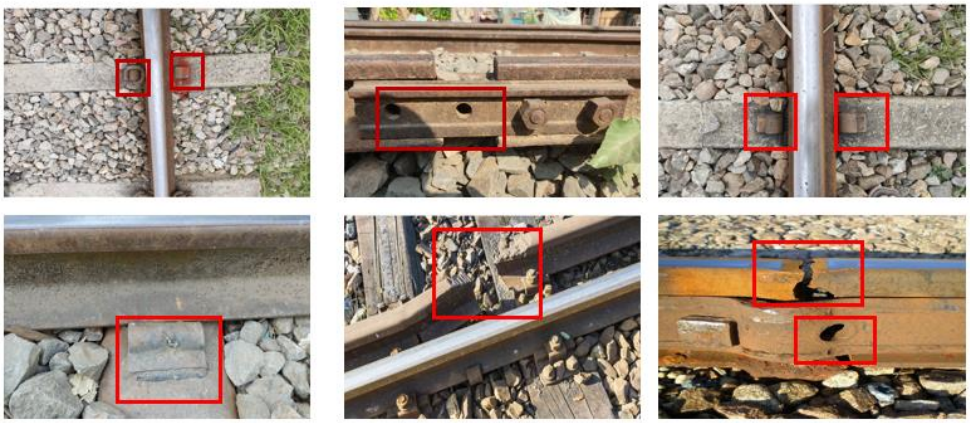

Figure 1. Sample railway track images showing fasteners and rail joints, with red bounding boxes highlighting potential defects or missing components.

In our research, we use a publicly available railway dataset from Kaggle, specifically designed for image classification in defect detection. This dataset consists of images labeled as either defective or non-defective, closely mirroring real-world railway maintenance challenges. Defective samples exhibit various types of fastener-related issues, such as broken, missing, loose, or corroded components, while non-defective samples represent properly secured fasteners. The dataset is balanced, containing an equal number of 700 defective and 700 non-defective images. However, it does not include bounding boxes or segmentation labels to specify the exact defect locations. Instead, each image is categorized at a global level as either defective or non-defective, meaning that the defect may appear anywhere within the image and is not necessarily centered.

Defects in this dataset involve issues with fastening components that secure railway tracks to the underlying infrastructure, including:

- Bolts: Loose, broken, or missing
- Clips : Deformed or incorrectly placed
- Anchors: Rusted, damaged, or improperly fastened

- Plates : Corroded or misaligned

Although we provide examples with bounding boxes in our study to illustrate defect regions, it is important to note that the original dataset does not contain these annotations. This means that models trained on this dataset do not explicitly learn defect localization but instead focus on a binary classification task (defective vs. non-defective). Due to the lack of localized defect annotations, our approach does not rely on object detection or segmentation methods. Instead, we train classification models that learn to differentiate between images containing any type of defect and those that do not, making it a global classification problem rather than a localized detection task. This distinction is crucial in determining the appropriate deep learning techniques for defect detection in railway infrastructure.

6.2 DATA-PREPROCESSING

Data preprocessing is a crucial stage in the machine learning (ML) process, focusing on transforming raw data into a format that is easier to understand and work with for subsequent analysis. In this study, we dealt with a dataset consisting of 1400 images, comprising 700 defective and 700 non-defective images. This step involves rectifying inconsistencies and integrating data from various sources to create a uniform dataset. With this balanced dataset, preprocessing included data transformation for normalization and aggregation, data reduction to minimize volume while preserving relevant results, and data discretization, which converts continuous attributes into categorical ones. The preprocessing phase was essential in ensuring that the dataset was well-structured and suitable for the specific needs of our ML models, providing a solid foundation for accurate and efficient analysis [46]. In our study, the emphasis on data preprocessing was key to enhancing the performance of deep learning models for defect detection. Data cleaning was essential to ensuring accuracy and relevance in the dataset. This step involved removing irrelevant information and correcting errors, which is crucial in defect detection where data precision directly impacts detection accuracy. Feature scaling and image augmentation, utilizing techniques like RandomResizedCrop and RandomHorizontalFlip, are employed to Enhance the model's learning process by introducing variations in the data. RandomResizedCrop

randomly crops the images to a size of 224x224 pixels. This augmentation technique randomly selects a portion of the original image and resizes it to the specified size, Which helps the model learn from different parts of the image.

RandomHorizontalFlip flips the images horizontally with a probability of 0.5. This augmentation technique horizontally flips the images, providing additional variations to the training data. Normalization, achieved by normalizing the pixel values of the images using mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225], standardizes the input data. This makes the optimization process more stable and efficient. For the validation and test datasets, the Resize and Centercorp technique is applied. It resizes the images to 256x256 pixels and then center-crops them to 224x224 pixels. This ensures that the images are consistently sized and centered, facilitating better generalization during validation and testing [47], were also performed. Moreover, to illustrate these transformations, we generated sample outputs showing how an original rail-track image is reshaped, randomly cropped, flipped, and normalized. These examples help demonstrate exactly how images appear before and after each transformation step. The PyTorch transforms pipeline seamlessly applies this sequence of operations during data loading, ensuring consistency and reproducibility. In practice, the preprocessing phase is handled by PyTorch's transforms within our code. This includes composing different transformation steps (e.g., resizing, cropping, flipping, and normalizing) into a pipeline applied to each image. We ensure that the training set gets the augmentations (RandomResizedCrop, RandomHorizontalFlip) while the validation and testing sets only receive resizing and cropping, preserving real-world data distribution for evaluation. Our rational for these specific transformations is threefold: (1) they help correct for minor inconsistencies in image composition (via cropping and flipping), (2) they address variations in scale and orientation (particularly relevant in detecting small fasteners in different positions), and (3) they standardize pixel values (normalization), making training more stable. Finally, splitting the dataset into training and validation sets enabled a comprehensive evaluation of the models. This approach ensured not only effective training but also validation of the model's performance in real-world conditions—a critical aspect of defect detection. In our case, we used the 10-fold cross-validation strategy. By providing justifications and examples for each transformation (RandomResizedCrop,

RandomHorizontalFlip, Resize, CenterCrop, and Normalize), we make the preprocessing phase transparent and replicable, thereby reinforcing the reliability and clarity of our defect detection approach. We began with a balanced dataset of 1,400 images. To ensure a robust evaluation of the model, we divided these images into three subsets: training, validation, and testing. First, we reserved 15% of the dataset—210 images—as a final test set. This left 85% (1,190 images) for training and validation. Within that 85%, we employed a stratified approach to split the data into a training portion, which contains 980 images (70% of the original dataset) and a validation portion with 210 images (15% of the original dataset). The stratification process helped maintain consistent class proportions across each subset, preventing any inadvertent imbalance that might arise from purely random splitting.Using three distinct subsets is important because each one serves a different purpose. The training set is used to fit the model's parameters, allowing the model to learn meaningful representations and patterns. The validation set is then used to monitor how well the model is performing during training and to tune hyperparameters, which helps avoid overfitting. Finally, the test set is kept separate until all model decisions and adjustments are complete; it provides an unbiased measure of performance on unseen data, offering a realistic sense of how the model will generalize in real-world scenarios.

If only two subsets (training and testing) are used, there is a risk of overfitting hyperparameters to the test set due to the lack of a dedicated validation phase, as well as reduced feedback for adjusting the model effectively during training. Conversely, if one relies solely on training and validation subsets (without a separate test set), the validation data may inadvertently guide too many design decisions—such as hyperparameter choices—leading to subtle overfitting. This can yield an overly optimistic assessment of model performance and reduce confidence in how well the model would generalize to real-world, unseen data. By employing three subsets—training, validation, and testing—we ensure that each serves a distinct purpose: the training set is used to learn model parameters, the validation set provides iterative feedback and hyperparameter tuning without compromising the final performance metric, and the test set remains untouched until all model decisions are finalized, providing an unbiased measure of true generalization performance. Table 1 shows the key data preprocessing steps applied in our study for railway defect detection. It

summarizes the techniques used, including data cleaning, augmentation, normalization, and dataset splitting, along with their purposes and the specific subsets they were applied to. These preprocessing steps ensure data consistency, enhance model generalization, and optimize performance for deep learning-based defect classification.

Preprocessing Step	Description	Purpose	Applied To
Data Cleaning	Removed irrelevant information and corrected errors in dataset.	Ensures dataset accuracy and relevance.	Entire dataset (1,400 images)
Data Transformation	Standardized dataset structure, ensuring uniform formatting.	Improves consistency and model compatibility.	Entire dataset
Data Augmentation	Applied RandomResizedCrop and RandomHorizontalFlip to increase variability.	Enhances model robustness and prevents overfitting.	Training dataset only
Feature Scaling (Normalization)	Normalized pixel values using mean [0.485, 0.456, 0.406] and std [0.229, 0.224, 0.225].	Standardizes pixel intensities for stable optimization.	Entire dataset
RandomResizedCrop (224x224)	Randomly crops and resizes images to 224x224 pixels.	Helps model generalization by exposing it to different image regions.	Training dataset only
RandomHorizontalFlip (p=0.5)	Flips images horizontally with a 50% probability.	Adds variation to training data for better generalization.	Training dataset only
Resize & CenterCrop (224x224)	Resized images to 256x256 pixels, then center-cropped to 224x224 pixels.	Maintains consistency in image dimensions.	image dimensions. Validation & Test datasets only

Dataset Splitting	Stratified split into Training (70%), Validation (15%), and Test (15%) sets.	Ensures balanced class representation and robust evaluation.	Entire dataset
Cross-Validation (10-Fold)	Divided the training set into 10 stratified folds for model evaluation.	Provides iterative feedback and improves model reliability.	Training & Validation sets
Data Loader (PyTorch Transforms)	Implemented preprocessing pipeline for seamless image transformations.	Ensures reproducibility and automation during model training.	Entire dataset

Table 1. Summary of Data Preprocessing Steps for Railway Defect Detection

6.3 MODELS AND HYPERPARAMETERS

We conducted a series of experiments using various models and configurations. The values for different parameters were chosen based on our experimental setup and the nature of the railway defect detection task. Here's the explanation of how we determined these values and their relevance to our hypotheses: We hypothesized that different models would show varying levels of effectiveness in identifying rail defects. To test this, we trained the models including CNNs and ViT and compared their performance [48]. Additionally, tuning specific hyperparameters could enhance the performance of the models. We used Optuna for hyperparameter optimization, conducting multiple trials to find the best settings. conducting 10 trials for hyperparameter optimization with Optuna provided a reasonable balance between computational expense and the ability to explore different configurations [45]. The choice of 7 epochs was made to avoid overfitting. Training for too many epochs can lead the model to learn the noise and details in the training data, which negatively impacts its performance on new, unseen data. By limiting the number of epochs, we ensure that the model generalizes better and performs more effectively when applied to real-world data [49]. A batch size of 32 was selected to balance computational efficiency and training stability, as commonly used in image processing tasks [50]. We used a weight decay of 0.01 to prevent overfitting by penalizing large weights, a standard practice in deep learning [51] our study, we utilized pre-trained models. These models, already trained on large datasets, provided a

strong starting point for our specific task. This approach was particularly useful given the limited amount of data available for railway defect detection [52]. We used 70% of the data for training, as this is a standard split ensuring enough data for effective model learning. Half of the remaining data after the training split was used for validation, a common practice for fine-tuning models without overfitting. The rest was used for testing to provide an objective evaluation of model performance [53]. In addition, the tuning of these hyperparameters was carried out on the separated training and validation portion (85% of the original dataset, i.e., 1,190 images), where 17.6% (210 images, corresponding to 15% of the entire dataset) is used for validation and 82.4% (980 images, corresponding to 70% of the entire dataset) is used for training. Furthermore, we employed a Stratified K-Fold approach (with a specified number of folds) to ensure that each fold is representative of the overall class distribution, thus further enhancing the reliability of our hyperparameter selection process [54]. The hyperparameters we used in our study are shown in Table 2:

Parameter	**Description**	**Value**
Epochs	Number of full passes through the training data	7
Batch Size	Number of samples processed before the model is updated	32
Train Size	Percentage of data used for training	70% of dataset
Validation Size	Percentage of data used for validation	Half of the remaining after training split
Test Size	Percentage of data used for testing	Rest after training and validation splits
Number of Folds (k)	Number of folds used in Stratified K-Fold; applied on the separated training and validation portion (85% of the original dataset, i.e., 1,190 images). Within this subset, 17.6% (210 images, corresponding to 15% of the entire dataset) is used for validation, and 82.4% (980 images, corresponding to 70% of the entire dataset) is used for training.	10
Weight Decay	adding a penalty to the loss function based on the magnitude of the weights.	0.01

Number of Trials for Optuna	Number of trials to perform in the hyperparameter optimization	10

Table 2. Key Hyperparameters in our study

For tuning hyperparameters such as dropout rate, learning rate, and momentum, we utilized Optuna. Optuna optimizes hyperparameters by searching the parameter space using automated trial-and-error. Optuna's approach involves defining a search space and then evaluating the model performance for each combination of hyperparameters iteratively. The values of the hyperparameters we used in our study, which were tuned using Optuna, are shown in the Table 3 for all models [45, 55]. Our work specifically focuses on tuning learning rate, momentum, and dropout rate [50, 56]. Additionally, we fix weight decay at 0.01 based on best practices in deep learning [51], In all models (ViT and Deit, ResNet50, VGG19, and VGG16) dropout is applied before the final classification layer. It helps prevent overfitting during training by randomly dropping out activations, enhancing the model's generalization ability. Three hyperparameters are tuned using the Optuna framework during the training process in our study which are:

a. Learning Rate (lr): This parameter controls the step size at each iteration while moving toward a minimum of a loss function. It's being tuned within a range from 1e−5 to 1e−1, using a logarithmic scale.

b. Momentum: Momentum helps accelerate gradient vectors in the right directions, thus leading to faster converging. It's being tuned within a range from 0.5 to 0.99.

c. Dropout Rate: This parameter is used in the dropout layers to prevent overfitting. The rate specifies the probability at which outputs of the layer drop out. It's being tuned within a range from 0.1 to 0.5. The hyperparameters that were tuned are shown in 3:

Model	**Learning Rate**	**Momentum**	**Drop Out**
ViT_base_patch16_224	0.0020909	0.7720241	0.3515297
DeiT_base_patch16_224	0.0026888	0.5789481	0.4131130
VGG19	0.0002764	0.7956331	0.2895775

VGG16	0.0134802	0.7690707	0.2988300
Resnet50	0.0071212	0.9045430	0.2345131

Table 3. Hyperparameters tuned by Optuna

In our study, we used an optimizer and a scheduler to enhance model performance. The SGD optimizer updates neural network weights to minimize the loss function, using key hyperparameters like learning rate and momentum. The learning rate determines the step size towards the loss function's minimum, while momentum accelerates gradients in the right direction for faster convergence. We also used the ReduceLROnPlateau scheduler, which adjusts the learning rate when the validation loss plateaus. This fine-tuning helps avoid local minima and ensures effective optimization by reducing the learning rate when performance stops improving. These components were integrated into a Stratified K-Fold cross-validation framework. This method splits the dataset into K folds, maintaining the class distribution in each fold. It allows for adaptive learning rate adjustments for each fold, enhancing model robustness and generalization [57, 58]. Key Components and The functionalities of our project are shown in Table 4:

Phase	**Functionality**	**Details**
Imports	Importing necessary libraries	PyTorch, Timm, NumPy, Matplotlib, Sklearn, Optuna, etc
Dataset Setup	Setting paths, devices, transforms	Path: Location of dataset- Device: CUDA if available, Transforms: Image preprocessing
Data Loading	Loading and splitting the dataset	Load images with labels, random_split: Split into train, validation, and test sets
Model Definition	Creating and modifying the neural network	Uses Timm for loading a pre-trained model- Modifies the model to fit the binary classification t
Hyperparameter Tuning	Hyperparameters tuning using Optuna	- Learning rate, momentum, dropout rate
Training Setup	Prepare data loaders and training environments	- Data Loaders: For handling training and validation data
Optuna Optimization	Optimization of model parameters	- Runs trials to maximize the recall metric on validation data

Stratified K-Fold Training	Cross-validation training	-Uses StratifiedKFold for splitting - Training and validation within each fold
Learning Rate Adjustment	Adjusts learning rate based on performance	- ReduceLROnPlateau: Decreases learning rate when validation loss plateaus
Results Visualization	Visualizing training results	- Plots for loss, accuracy, precision, recall, and ROC curves
Final Parameters	Output best parameters and retrain	-Determines best hyperparameters from Optuna trials - Retrains model with these parameters
Optuna Completion	Optuna study and extracts best trials	- Optuna finds optimal model parameters through defined trials

Table 4. Steps of our study

7. CONTRIBUTION

Our study aims to bridge existing gaps in railway defect detection research by systematically comparing Convolutional Neural Networks (CNNs) and transformer-based models. While prior research has explored CNNs and transformers separately, no comprehensive study directly compares their effectiveness for railway defect detection under the same experimental conditions. Our key contribution is the first in-depth comparative analysis of CNN and transformer-based approaches for railway fastener defect detection, which provides valuable insights into the relative advantages and trade-offs of these architectures.

We build upon existing CNN-based methods and expand our research scope by incorporating transformer models such as Vision Transformer (ViT) and Data-Efficient Image Transformer (DeiT). These transformer models have demonstrated superior performance in various defect detection tasks, yet their potential in railway maintenance remains underexplored. By evaluating these models on a real-world railway fastener dataset from Kaggle, we validate their practical applicability and contribute to the growing body of research on deep learning-based railway inspection.

A major limitation in railway defect detection is the scarcity of large, labeled datasets, which hinders model performance and generalization. To address this challenge, we emphasize the importance of transfer learning and pre-trained models in our study. By

leveraging pre-trained ImageNet models and fine-tuning them for railway defect detection, we demonstrate how transfer learning can significantly enhance performance even with limited labeled data. Furthermore, we systematically investigate the impact of transfer learning on both CNN and transformer models, providing a unique perspective on its effectiveness across different architectures.

This research is significant for several reasons:

- Comprehensive Model Comparison – By evaluating both CNN and transformer-based models under identical preprocessing conditions, transfer learning frameworks, and hyperparameter tuning strategies using Optuna, we ensure a fair and rigorous comparison.

- Optimization of Model Performance – We fine-tune each model to maximize detection accuracy, recall, and robustness, ensuring that our findings contribute to real-world railway defect monitoring applications.

- Integration of NDE and SHM Techniques – Our study aligns with Non-Destructive Evaluation (NDE) and Structural Health Monitoring (SHM) principles, ensuring that defect detection methods are practical, scalable, and cost-effective for railway maintenance operations.

- Ensuring Generalizability – By maintaining a consistent data preprocessing pipeline and a uniform hyperparameter search space, we enhance the reliability and reproducibility of our results, making them applicable beyond our specific dataset.

Ultimately, this research provides critical insights for the railway industry, highlighting the potential of deep learning models—particularly transformers—in defect detection. By advancing the understanding of CNN vs. transformer-based architectures in this domain, we contribute to the development of safer, more efficient, and automated railway maintenance solutions, thereby improving the safety, reliability, and operational efficiency of railway networks.

8. ORGANIZATION

This thesis follows a manuscript-based format, meaning it includes a research paper as a key part of the study. The thesis is divided into four main sections to clearly present the research problem, methods, and results. The Introduction gives an overview of the research topic. It explains why early defect detection in railway tracks is important, defines the research problem, and presents the main hypotheses. It also describes the methods used to solve the problem and introduces the dataset used for testing and evaluation. Chapter 1 is the literature review. It explains past research on railway defect detection, how deep learning models like CNNs and Vision Transformers have been used in this field, and the current gaps in knowledge. This section helps show why this study is needed. Chapter 2 contains the research paper, titled "Toward Smart Railway Maintenance: AI-Enhanced Non-Destructive Evaluation Using Vision Transformers and CNNs for Fastener Defect Detection." The paper explains the methods used, the experiments conducted, and the results of testing different models. It provides the key findings of this study. The final section summarizes the findings and explains what they mean for railway maintenance. It also discusses the study's limitations and suggests ways future research can improve defect detection methods. This structure ensures that the research is presented in a clear and logical way, making it easy to understand how the study was conducted and why it is important.

CHAPITRE 1. REVIEW OF LITERATURE AND CONCEPTUAL FRAMEWORK

1. INDUSTRY 4.0

Industry 4.0, often referred to as the fourth industrial revolution, signifies a transformative approach to industrial production characterized by the integration of advanced technologies such as the Internet of Things (IoT), AI, and big data analytics into manufacturing processes. This integration aims to create smart factories where systems can communicate, analyze data, and make decisions autonomously, leading to increased efficiency and productivity [59]. One of the foundational concepts of Industry 4.0 is interoperability, which allows machines, devices, sensors, and people to connect and communicate with each other via the Internet of Things (IoT) [60]. Information transparency is another critical principle, enabling systems to create a virtual copy of the physical world through sensor data to contextualize information [61]. Technical assistance refers to the ability of systems to support humans in decision-making and problem-solving and to assist in difficult or unsafe tasks [62]. Decentralized decisions in cyber-physical systems enable these systems to autonomously make decisions and perform tasks without centralized control. This capability enhances their flexibility, resilience, and adaptability. Autonomous decision-making in cyber-physical systems is crucial for various applications, including smart manufacturing and logistics, where systems must react in real time to dynamic conditions and optimize operations independently. These systems integrate technologies like AI, the Internet of Things, and big data to support their decision-making processes [63]. The implementation of Industry 4.0 technologies can enhance manufacturing processes. For instance, predictive maintenance powered by AI and machine learning can reduce downtime and maintenance costs by predicting equipment failures before they occur [64].

Moreover, advanced robotics and automation improve precision and efficiency in production lines, leading to higher quality product and reduced human error [65]. However, the transition to Industry 4.0 also presents challenges. One significant barrier is the high initial investment required for adopting these advanced technologies, which can be prohibitive for small and medium-sized enterprises (SMEs) [66]. Additionally, there is a need for a skilled workforce capable of managing and maintaining these sophisticated systems,

which necessitates substantial training and education [67]. Cybersecurity is another critical concern, as the increased connectivity and data exchange between systems raise the risk of cyber-attacks [68]. Despite these challenges, the potential benefits of Industry 4.0 are substantial. Enhanced data analytics can lead to more informed decision-making and optimized operations, while improved flexibility and adaptability allow manufacturers to respond more quickly to market changes and customer demands [69].

Industry 5.0 represents the next phase in the evolution of industrial practices, emphasizing the synergy between human creativity and technological advancements to create more sustainable, resilient, and human-centered systems. While Industry 4.0 focused primarily on interconnected smart factories, Industry 5.0 aims to bring humans back to the center of production, leveraging the power of AI, cognitive computing, and collaborative robotics to enhance—not replace—human capabilities [70]. In contrast to the fully autonomous decision-making prevalent in Industry 4.0, Industry 5.0 promotes a co-creative approach where artisanship and customization coexist with smart automation [71]. A sign of Industry 5.0 is the incorporation of personalization and mass customization at scale, driven by increasingly intelligent systems that respond swiftly to individual customer requirements [72]. This shift expands upon predictive maintenance and data analytics practices established in Industry 4.0 by introducing cognitive intelligence and human-machine collaboration into the design process, enabling more nuanced decision-making and enhancing overall flexibility [73]. Consequently, Industry 5.0 solutions often tackle sustainability and social responsibility by optimizing resource usage, reducing waste, and integrating feedback mechanisms that account for environmental and societal impacts [74]. Despite its promise, Industry 5.0 also faces challenges such as ensuring adequate cybersecurity in more deeply networked systems, overcoming the knowledge gap for workers who must collaborate with advanced robots, and navigating ethical considerations surrounding human-machine cooperation [75]. Nonetheless, by merging human innovation with emergent technologies, Industry 5.0 aspires to transform manufacturing and service sectors into domains where efficiency, personalization, and sustainability coexist, ultimately delivering enhanced value and quality

of life. Figure 2 shows the evolution from Industry 4.0's automation-focused paradigm to the human-centric emphasis of Industry 5.0.

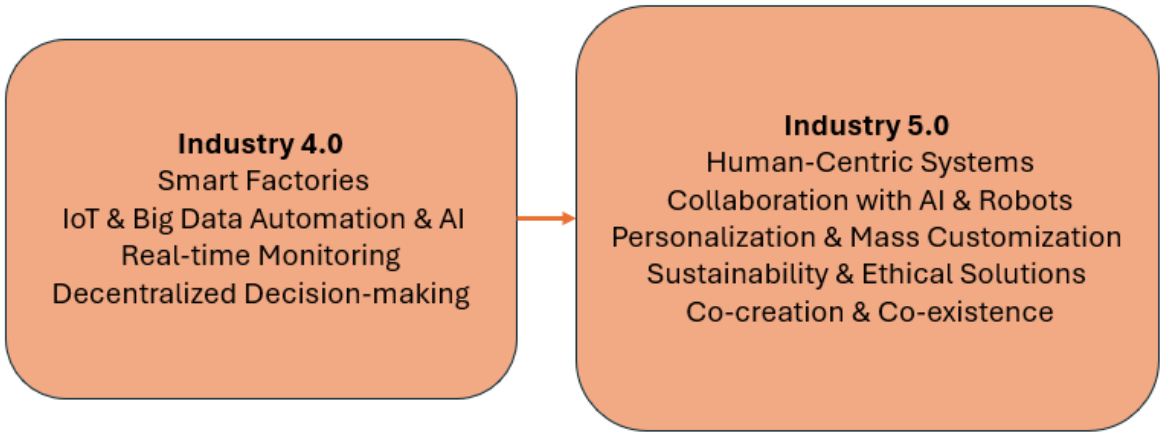

Figure 2. Transition from Industry 4.0's automation focus to Industry 5.0's human-centric approach.

1.1 THE HISTORY OF DEEP LEARNING

1943 Walter Pitts and Warren McCulloch introduced a mathematical representation of a biological neuron in their paper, "A Logical Calculus of the Ideas Immanent in Nervous Activity." Although the McCulloch-Pitts Neuron exhibited limited functionality and did not possess a learning mechanism, it still paved the way for the future development of artificial neural networks and deep learning [76]. Frank Rosenblatt, in 1957, in his paper "The Perceptron: A Perceiving and Recognizing Automaton," demonstrated an enhanced version of the McCulloch-Pitts neuron. He introduced the 'Perceptron,' which boasted authentic learning capacities, allowing it to carry out binary classification independently. This breakthrough ignited a substantial research shift in the field of shallow neural networks [77]. In 1960, Henry J. Kelley introduced a new model called continuous backpropagation. Even though his idea was related to something called Control Theory, it set the stage for improving the model. This model would later be used in things called Artificial Neural Networks (ANN) as time went on [78]. Stuart Dreyfus, in 1962, introduced a new way to use backpropagation with a simple derivative chain rule, instead of using the old method of dynamic programming. This incremental advancement further solidified the forthcoming development of deep

learning [79]. In 1965, Alexey Grigoryevich Ivakhnenko and Valentin Grigor′evich Lapa developed a hierarchical form of a neural network that employs a polynomial activation function and is trained to utilize the Group Method of Data Handling. This is widely regarded as the inaugural multi-layer perceptron, frequently accrediting Ivakhnenko with the title of the father of deep learning [80] . In 1969 Marvin Minsky and Seymour Papert released a book titled "Perceptrons," demonstrating the limitations of Rosenblatt's perceptron in solving complex functions, such as XOR. They showed that handling such functions would require placing perceptrons in multiple hidden layers, which undermined the perceptron learning algorithm [81]. In 1969, Kunihiko Fukushima introduced a seminal concept in the domain of neural networks with the advent of the ReLU (rectified linear unit) activation function [82, 83]. This activation function, known as the rectifier, swiftly ascended to prominence, becoming the most widely adopted activation function for CNNs and deep neural networks more broadly [84].

In 1970 Seppo Linnainmaa came up with a new way to automatically do backpropagation and made it work on computers. Even though there was a lot of research on backpropagation, it wasn't used in neural networks until the next decade [85]. Alexey Grigoryevich Ivakhnenko in 1971, persisted in his explorations within the domain of Neural Networks. He pioneered the development of an 8-layer Deep Neural Network, employing the Group Method of Data Handling (GMDH) as a strategic approach to its creation and functionality. This endeavor reflected a meticulous synthesis of layered neural structures, demonstrating a profound application of GMDH in the realm of deep learning and neural network sophistication [86]. In 1980, Kunihiko Fukushima introduced the Neocognitron, originating the first architecture of a CNN, which possessed the capability to identify visual patterns, notably those found in handwritten characters. This innovative network model became foundational in computer vision, particularly in the recognition and interpretation of visual data, demonstrating a proficient approach to identifying and managing intricate patterns in images. Neocognitron marked a significant step towards sophisticated image recognition by enabling the system to successfully discern handwritten characters, showcasing a new era in neural network design and its applications [87]. In 1982, John Hopfield developed the Hopfield Network, a kind of network that remembers patterns and

can recall them, helping to shape future models in deep learning and recurrent neural networks (RNNs). It was a big step towards networks that could manage memory well and influenced later technological developments in this field [88]. In 1985, Paul Werbos, in his 1982 Ph.D. thesis, introduced the idea of using Backpropagation to spread errors during the training of Neural Networks. His findings would later guide the neural network community to adopt backpropagation as a practical method, helping networks learn more efficiently and laying the groundwork for advancements in modern deep learning and neural network training approaches [89]. In 1985, David H. Ackley, Geoffrey Hinton, and Terrence Sejnowski developed the Boltzmann Machine, a type of neural network that works with probability and only has an input layer and a hidden layer, with no output layer. This network advanced the understanding of neural learning, sparking further research in machine learning and AI [90].

In 1986, Terry Sejnowski developed NeTalk, a neural network designed to learn to pronounce English text. It was trained by being shown written text along with matching phonetic transcriptions to learn from. This innovative approach highlighted the potential of neural networks in language processing and speech synthesis [91, 92]. In 1986, Geoffrey Hinton, Rumelhart, and Williams introduced a successful method for implementing backpropagation in neural networks in their paper. This technique significantly eased the training of complex deep neural networks, solving major challenges faced in the early research stages and paving the way for advancements in the deep learning field [93]. In 1986, Paul Smolensky introduced a variant of the Boltzmann Machine, notably distinguished by the absence of intra-layer connections within the input and hidden layers, subsequently termed the Restricted Boltzmann Machine (RBM). This innovation was not immediately celebrated but, as years progressed, the RBM garnered substantial attention and acclaim, particularly for its efficacy in developing recommender systems, illustrating a gradual but impactful influence in the realm of machine learning and data handling, especially in the context of collaborative filtering and preference prediction [94].

In 1989, Yann LeCun made a groundbreaking advancement in deep learning and computer vision by using backpropagation to train a CNN for recognizing handwritten

numerals. This achievement laid a strong foundation for modern computer vision and influenced many applications across various domains by demonstrating the potential of neural networks to process visual data [95, 96]. In the same year, George Cybenko contributed significantly to deep learning with his paper on the Universal Approximation Theorem. He showed that a feed-forward neural network with a single hidden layer and a finite number of neurons could approximate any continuous function. This insight highlighted the potential and utility of neural networks in diverse computational tasks, enhancing the credibility and applicability of deep learning in various scientific and technological fields [96].

In 1991, Sepp Hochreiter pointed out a big issue in deep learning called the vanishing gradient problem. This problem makes training deep networks very slow and hard. Because of this, many researchers worked to find solutions for years after he highlighted it [97]. In 1997, Sepp Hochreiter and Jürgen Schmidhuber introduced a groundbreaking paper presenting the "Long Short-Term Memory" (LSTM) concept. This design, which is a variation of the recurrent neural network, would later play a pivotal role in advancing the field of deep learning in subsequent years [98]. In 2006, Geoffrey Hinton et al. presented a significant paper. Within this work, they introduced the concept of Deep Belief Networks by layering multiple Restricted Boltzmann Machines (RBMs) on top of each other. Notably, this architecture made the training process considerably more efficient, especially when dealing with vast datasets [99]. In 2008, Andrew NG's team at Stanford University began advocating for utilizing Graphics Processing Units (GPUs) to significantly speed up the training of Deep Neural Networks. This approach enabled more practical and efficient handling of extensive data in the field of Deep Learning, paving the way for more advanced research and applications [100]. Obtaining labeled data has consistently been a tough task for the Deep Learning community. In 2009, Stanford professor Fei-Fei Li launched ImageNet, a massive database containing 14 million labeled images. This database became a crucial resource for deep learning researchers, providing a benchmark through its annual competition, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), where researchers tested and compared their algorithms, sparking numerous advancements in the field [101].

In 2011, Yoshua Bengio, et al. introduced the ReLU activation function to fix the vanishing gradient problem in deep learning. This was important because, while GPUs had already helped train deep neural networks faster, the ReLU function provided another useful tool to make the training of these networks even more efficient and practical, aiding progress in the field [102]. In 2012, AlexNet, designed by Alex Krizhevsky and powered by GPU technology, won ImageNet's image classification contest by achieving an impressive 84% accuracy, significantly surpassing the previous 75% record. This victory didn't just mark a milestone in accuracy but also sparked a global upswing in deep learning research and application, influencing various technological domains [103]. The development of the Generative Adversarial Neural Network (GAN) by Ian Goodfellow in 2014 ushered in a new era of deep learning applications across diverse fields such as fashion, art, and science. GANs, with their ability to create convincing, synthetic data, introduced a range of innovative possibilities in various domains, offering a powerful tool for generating realistic data for various experimental and creative pursuits. This revolutionary technology has not only enriched the machine-learning field but also catalyzed numerous novel applications and developments in both technological and artistic arenas [104].

In 2017, the Google Brain team introduced the modern transformer. This architecture, known for its parallel multi-head attention mechanism, enhanced the handling and Understanding of sequences and contextual information, particularly in natural language processing tasks. The advent of the transformer marked a significant leap in machine-learning, offering sophisticated and scalable solutions for various applications and influencing future AI research and development [105]. In 2018, Google introduced the Bidirectional Encoder Representations from Transformers (BERT) [106], signifying a breakthrough in the field of Natural Language Processing (NLP). According to a survey conducted in 2020, over 150 studies in the domain acknowledged its substantial impact [107] In 2018, OpenAI released an article where it presented the inaugural generative pre-trained transformer (GPT) system [108]. Subsequently, GPT-2 was introduced in 2019, being pre-trained on 40 GB of text, and 8 million documents, from 45 million web pages [109]. A year later, GPT-3 was released. While it bore similarities to GPT-2, modifications were made to enable larger scaling. It was trained using 499 billion tokens from various sources, including

CommonCrawl (570 GB), WebText, English Wikipedia, and two books [110]. GPT-3.5 is a subset of GPT-3 models developed by OpenAI in 2022. Subsequently, GPT-4 was developed as a multimodal large language model by the same organization IN 2023 [111]. In visual tasks, CNNs are often recognized as the core element [112]. However, the success of Transformer models in language tasks has encouraged their use in computer vision and other combined learning tasks. Recently, the Transformer architecture, originally designed for language tasks, is emerging as a promising substitute for CNNs in computer vision [113]. The foundational structure introduced by Dosovitskiy et al. in the 2020 paper utilizes Transformers to analyze the connections between input pairs, typically words in textual data. For images, the primary element under examination is the pixel. Directly mapping relationships between every pixel pair in standard-sized images would be too resource-intensive. To address this, the ViT strategy evaluates pixel associations within smaller segments of the image, such as 16x16 pixel blocks. These segments receive specific positional markers and are organized into sequences. During the learning process, these markers, which are essentially adjustable vectors, are optimized. These image segments are then sequentially organized and combined with an embedding matrix. Once the positional marker is incorporated, this combined data is directed through the Transformer's processing layers [48]. The transformative impact of Transformer architectures, along with their various modifications, has been noticeably evident across a wide spectrum of computer vision tasks. These architectures have shown remarkable proficiency in image recognition and a foundational task that lays the groundwork for many other vision applications [48, 114]. Moreover, they have proven adept at detecting specific objects within images, a critical component in applications like autonomous vehicles and surveillance [115, 116]. When it comes to image segmentation, a task that involves demarcating specific regions or objects in an image, Transformers have also shown promise [117] . Their adaptability doesn't stop there; they've been employed for enhancing the quality of low-resolution images in super-resolution tasks and have displayed competency in video understanding [118, 119].

1.2. HISTORY OF DEFECT DETECTION

Since ancient times, craftsmen have been inspecting their work for defects. In many early industries, such as pottery or blacksmithing, craftsmen visually inspect each product for any imperfections [120]. The early 20th century saw the development of statistical methods for quality control. Walter A. Shewhart introduced the control chart in the 1920s, leading to the birth of statistical process control (SPC) [121]. This period marked the beginning of systematic approaches to detecting and managing defects in manufacturing. During and after World War II, there was a significant rise in non-destructive testing (NDT) methods, including X-ray, ultrasonic, and magnetic particle inspections [122]. These techniques allowed for the inspection of materials and products without causing damage, making them essential for industries like aerospace and military manufacturing. With the rise of electronics manufacturing in the 1980s, automated optical inspection (AOI) systems were developed to inspect printed circuit boards (PCBs) for defects [123]. AOI systems use cameras and image processing algorithms to detect flaws such as missing components, misalignments, and soldering issues. Starting in the late 20th and early 21st centuries, machine learning techniques began to be applied to defect detection. These systems can be trained to recognize and classify defects in a wide variety of manufacturing contexts [124] . The use of neural networks, support vector machines, and other machine learning algorithms has improved the accuracy and speed of defect detection processes. The current trend of automation and data exchange in manufacturing technologies is called Industry 4.0. This includes the Internet of Things (IoT), cloud computing, and cognitive computing. Industry 4.0 represents a new phase in the industrial revolution that focuses heavily on interconnectivity, automation, machine learning, and real-time data, all of which can be applied to defect detection [125]. Smart sensors and IoT devices can continuously monitor production processes and send data to cloud-based platforms where advanced analytics and machine learning algorithms can identify defects in real time [126].

NDE 4.0 is an extension of Industry 4.0 principles applied to non-destructive evaluation. It involves the integration of AI, machine learning, IoT, and big data analytics to improve inspections [127]. By leveraging these technologies, NDE 4.0 aims to create

intelligent inspection systems capable of performing real-time data analysis and decision-making, thus enhancing defect detection and prevention capabilities [128]. Recent advancements in deep learning and computer vision have further revolutionized defect detection. ViTs and CNNs are now being used to enhance defect detection in complex industrial environments [129]. These models can analyze high-resolution images and identify subtle defects that might be missed by traditional inspection methods. Additionally, the integration of digital twins—virtual replicas of physical systems—allows for the simulation and analysis of manufacturing processes in real time, providing a proactive approach to defect detection and prevention. This technology leverages big data and AI to predict potential defects before they occur, thus enhancing overall product quality and efficiency. The incorporation of advanced analytics and predictive maintenance strategies has further improved defect detection capabilities. By analyzing historical and real-time data, predictive maintenance models can forecast potential failures and schedule timely interventions [130].

1.3. IOT

The Internet of Things (IoT) refers to the network of physical devices embedded with sensors, software, and other technologies to connect and exchange data with other devices and systems over the Internet. IoT has the potential to transform various industries by enabling real-time monitoring, automation, and data-driven decision-making [131]. IoT devices are used in various applications, from smart homes and wearable devices to industrial automation and smart cities. In smart homes, IoT devices such as thermostats, lights, and security systems can be controlled remotely, providing convenience and energy savings [132]. In industrial settings, IoT enables the creation of smart factories where machines and equipment can communicate and coordinate with each other to optimize production processes. For instance, IoT sensors can monitor the condition of machinery in real time, predicting maintenance needs and preventing equipment failures [133].

In agriculture, IoT devices can monitor soil moisture, weather conditions, and crop health, enabling precision farming and improving yield. Smart cities represent another significant application of IoT, where interconnected systems enhance urban living by

improving infrastructure, reducing energy consumption, and enhancing public services. For example, smart traffic management systems can analyze data from sensors and cameras to optimize traffic flow and reduce congestion [134]. IoT-enabled waste management systems can monitor the fill levels of waste bins and optimize collection routes, reducing costs and environmental impact [135, 136]. Interoperability is another challenge, as devices from different manufacturers need to communicate and work together seamlessly [137]. Additionally, the management and analysis of the large volumes of data generated by IoT devices require advanced analytics capabilities and significant computing resources [138]. IoT has the potential to revolutionize various industries by enabling real-time monitoring, automation, and data-driven decision-making. As technology continues to advance, IoT will play an increasingly vital role in enhancing efficiency, productivity, and quality of life [131].

2. NDE 4.0

NDE 4.0 applies the principles of Industry 4.0 to nondestructive testing (NDT), enhancing the detection, characterization, and monitoring of material defects using advanced technologies. NDE 4.0 aims to transform traditional NDT methods through the integration of AI, machine learning, IoT, and big data analytics, thus improving the accuracy, reliability, and efficiency of inspections. The primary objective of NDE 4.0 is to create intelligent inspection systems that can perform real-time data analysis and decision-making. For instance, AI algorithms can be used to analyze large datasets from ultrasonic inspections, identifying patterns and anomalies that human inspectors might miss [139][140]. IoT plays a crucial role in NDE 4.0 by enabling the collection and transmission of inspection data from various sensors and devices to a central database. This interconnected network allows for remote monitoring and analysis, reducing the need for on-site inspections and enabling timely interventions [141]. Additionally, big data analytics can process vast amounts of inspection data, providing insights into material conditions and predicting potential failures [142]. Enhanced data accuracy and analysis capabilities lead to more reliable inspection results, while the automation of routine tasks increases efficiency and reduces human error [143]. Furthermore, the ability to monitor equipment in real-time allows for proactive maintenance, minimizing downtime and extending the lifespan of assets [144].

However, the implementation of NDE 4.0 also presents challenges. The integration of advanced technologies requires significant investment in infrastructure and training, which can be a barrier for some organizations [145]. Additionally, the reliance on digital systems increases the risk of cyber threats, necessitating robust cybersecurity measures [146]. There is also a need for standardized protocols and guidelines to ensure consistency and reliability in inspections across different industries. Despite these challenges, the potential benefits of NDE 4.0 are substantial. By leveraging advanced technologies, NDE 4.0 can enhance the accuracy, reliability, and efficiency of nondestructive inspections, ultimately improving safety and performance in various industries [147].

2.1. SMART DIGITAL TWINS

Smart digital twins are virtual replicas of physical assets, systems, or processes that use real-time data and advanced simulations to mirror and predict the behavior of their real-world counterparts. These digital twins integrate data from IoT devices, AI, and machine learning algorithms to provide insights into the performance, maintenance needs, and optimization opportunities of physical systems [148]. The concept of digital twins originated in the aerospace industry but has since expanded to various sectors, including manufacturing, healthcare, and urban planning. By creating a digital Counter part of a physical asset, organizations can monitor its condition in real-time, simulate different scenarios, and predict potential issues before they occur [149-151]. This predictive capability is particularly valuable for maintenance and operational optimization, as it allows for proactive interventions that can prevent failures and reduce downtime [152]. AI and machine learning play a critical role in the functionality of smart digital twins. These technologies enable the analysis of vast amounts of data collected from sensors and IoT devices, identifying patterns and correlations that can inform decision-making [153]. For example, in a manufacturing context, a digital twin of a production line can analyze data from various stages of the process to identify bottlenecks, optimize resource allocation, and improve overall efficiency [154]. The integration of digital twins with IoT and big data analytics provides a comprehensive view of the asset's lifecycle. This holistic approach enables organizations to optimize the design, production, operation, and maintenance phases, leading to improved performance

and cost savings [61]. Furthermore, the ability to simulate different scenarios allows for better risk management and informed decision-making [155]. Smart digital twins represent a transformative technology with the potential to revolutionize various industries by providing real-time insight and predictive capabilities. As technology continues to evolve, it will play a crucial role in enhancing the efficiency, reliability, and sustainability of physical systems [148].

2.2 STRUCTURAL HEALTH MONITORING (SHM) IN INDUSTRY 4.0

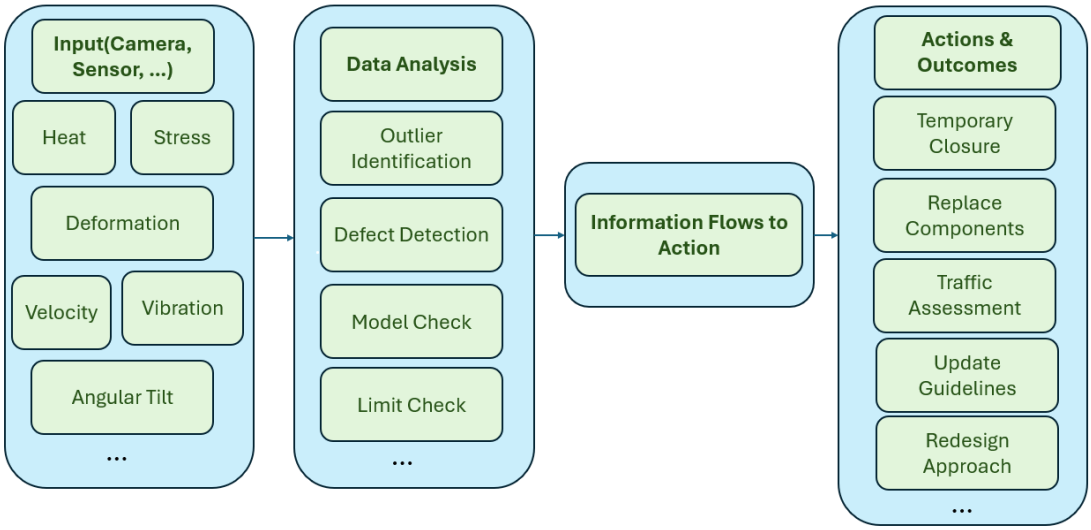

Figure 3. Flow diagram illustrating how various camera/sensor inputs undergo data analysis procedures, generating actionable insights that inform critical decisions and outcomes.

Figure 3 shows a SHM workflow, starting with various inputs (e.g., camera data, sensor readings on heat, stress, deformation, etc.), proceeding through data analysis steps (outlier identification, defect detection, model checks), and culminating in actionable outcomes such as temporary closure, component replacement, or traffic assessment [156]. SHM is an integral part of Industry 4.0, particularly in maintaining and ensuring the safety of infrastructure such as bridges, buildings, and railway tracks. SHM involves using various sensors and data acquisition systems to monitor the health and performance of structures in real-time. The integration of SHM with advanced analytics and machine learning allows for the continuous assessment of structural integrity, enabling early detection of potential issues

before they become critical [157]. This proactive approach not only enhances safety but also optimizes maintenance schedules and reduces operational costs [158]. SHM systems can also integrate with IoT technologies to provide real-time data transmission and remote monitoring capabilities, significantly improving the efficiency and accuracy of infrastructure management [129]. Additionally, SHM can aid in disaster resilience by providing early warnings of structural weaknesses that could lead to catastrophic failures [159]. The application of SHM in the Railway industry, for example, involves continuous monitoring of rail tracks and components, which is essential for preventing accidents and ensuring smooth operations [9]. Furthermore, SHM Facilitates the integration of cyber-physical systems with structural health monitoring, enhancing the reliability and safety of critical infrastructure [160].

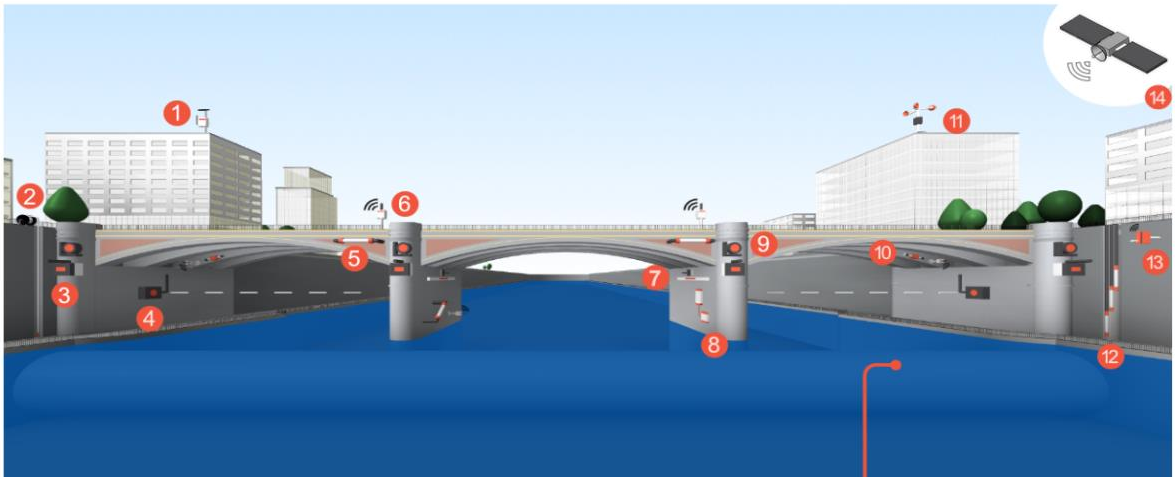

Figure 4. Automated systems and geotechnical instruments that can be used for structural monitoring. 1. Gateway with Solar Panel 2. Water Level Meter 3. Tiltmeter 4. LaserTilt90 5. Vibrating Wire crackmeter 6. Single Channel Data Logger 7. Electrolevel Beam Sensors 8. Vibration Monitor 9. Optical Survey Prism 10. Strain Gauges 11. Meteorological Station 12. Piezometer 13. Five Channel Data Logger 14. InSAR[1]

Figure 4 shows a typical urban bridge integrated with an SHM (Structural Health Monitoring) system, highlighting various sensor placements (e.g., on the bridge deck, piers, and surrounding infrastructure) for real-time data gathering. These sensors monitor factors like deformation, vibration, and environmental conditions, helping assess the structure's

[1] https://www.geomotion.com.au/structural-monitoring.html

integrity and ensure public safety. Reliable assessment methods for railways and other infrastructure components are critical for ensuring long-term safety and functionality [161]. The development of smart sensing technologies and their application in SHM represents a significant advancement in the field, offering new opportunities and challenges [162]. The use of machine learning in SHM is becoming increasingly important, offering enhanced capabilities for detecting and predicting structural issues [163]. SHM systems integrated with Industry 4.0 technologies can provide comprehensive and continuous monitoring, ensuring the safety and reliability of railway infrastructure [164].

3. RAILWAY AND DEFECT DETECTION

Over the past 20 years, rail transit has significantly advanced worldwide and has become a crucial mode of modern transportation. This fast-paced growth in rail systems has led to heightened demands for safety in transport. Ensuring the good condition of railway tracks is vital for the safe and reliable running of trains [165]. Transportation is essential in today's world, linking people and commerce and enabling the movement of goods and services. Railways stand out in this sector, offering a dependable and eco-friendly way of transporting people and goods, and are a crucial part of modern transport systems [166]. Railways drive economic growth, connect communities, support trade and tourism, and offer a sustainable option for long-distance travel and freight movement. They help ease urban congestion and are more environmentally friendly than many other transport modes, playing a vital role in reducing the transportation industry's environmental footprint. The global rail sector generates over a trillion dollars annually and employs over five Million individuals. Railways are a key economic force in numerous countries, contributing up to 3 percent of some nations' Gross Domestic Product (GDP). Besides their economic impact, railways are essential in linking communities and promoting trade and tourism. They offer a dependable and effective means of transportation for both long-distance Journeys and goods transport and can aid in alleviating traffic congestion and enhancing urban mobility [167]. In developing nations, railways play a vital role in linking remote and less-served regions with key economic hubs, thus promoting economic growth and enhancing the availability of markets and services. Despite its numerous advantages, the railway industry encounters

various obstacles. A lack of sufficient investment in railway infrastructure by many nations has resulted in outdated and inefficient systems [168]. Rails tend to wear down over time due to ongoing strain from frequent train travel, the rapid pace of trains on the railway network, the pressure of axle loads, and varying climatic factors [17].

Rail track damage can cause train derailments, putting the safety of passengers and rail workers at risk. Over time and with frequent use, tracks, like any mechanical system, become vulnerable to defects and breakdowns. In the year 2009, track faults were responsible for 658 out of the 1890 recorded railway accidents [169]. During the past decade, about one in every three train accidents in the United States has been due to issues related to the tracks [170]. The significant risks associated with rail track defects emphasize the need for diligent maintenance and repair of rail lines. Regular checks and upkeep are crucial to lessen hazards and avoid disruptions in the railway system [18].

Innovative inspection technologies such as aerial and terrestrial imaging, optical and laser scanning, and robot-assisted inspection have revolutionized quality control in various sectors [171]. Aerial imaging, conducted with drones or aircraft, is particularly effective for inspecting large-scale infrastructures [172]. Terrestrial imaging, which uses ground-based cameras, is better suited for thorough inspections of easily accessible areas [172]. Optical and laser scanning techniques are used to create detailed 3D models and high-resolution surface mappings [173], whereas robotic inspections are employed to reach areas that are either hazardous or difficult to access [174]. Maintaining railway tracks is notably one of the most expensive aspects of railway engineering. In cases like the Netherlands, it's estimated that over fifty percent of the annual maintenance budget is dedicated solely to track upkeep. To lower the costs and risks related to rail track issues and to enhance both safety and maintenance effectiveness, the creation and implementation of innovative techniques and strategies are crucial [175]. Moreover, the railway industry is increasingly incorporating connected devices, sensors, and big data to upgrade maintenance practices. Machine learning, having already transformed areas like computer vision and speech recognition, is now being leveraged in railways. This is due to the vast data from modern monitoring equipment like sensor networks and HD cameras, aiming to improve system reliability and

lower maintenance costs and risks [176]. The impact of AI in safety monitoring for railway operations is significant. These AI systems are capable of continuously tracking the condition of rail tracks and alerting maintenance teams about minor issues before they develop into larger problems. This preventive approach is key in minimizing accident risks, thereby increasing the overall safety and dependability of rail services. With real-time data and alerts from AI, potential risks are addressed quickly, ensuring that railway operations consistently adhere to high safety standards [177, 178].

Geometric inspection of high-speed railway tracks specifically refers to the process of detecting vertical and lateral deviations, as well as irregularities of the track, especially when there is no load from trains. The results of this inspection are crucial as they form the primary basis for any necessary adjustments to the track. This inspection process involves calculating the center mileage of the railway line by measuring specific coordinates, which are crucial for ensuring the track's alignment and overall integrity. The accuracy and precision of geometric inspections are vital for maintaining the safety and efficiency of high-speed railway operations. Geometric inspection in manufacturing significantly outperforms manual methods in efficiency and precision. Utilizing advanced technologies like 3D scanning, these automated systems inspect parts much faster and with greater accuracy, a crucial factor in precision-critical industries [179]. From a cost perspective, automated inspections, despite higher initial investments, prove more cost-effective over time due to their efficiency and lower error rates. In contrast, Manual methods, while cheaper initially, may incur higher long-term costs due to less efficiency and higher error margins [180]. Railway defect detection is a critical aspect of maintaining the safety and reliability of rail systems. NDE methods are increasingly adopted in this field due to their ability to inspect materials and components without causing damage. These methods are essential for identifying flaws and defects in rail tracks, which can prevent accidents and ensure smooth operations. Innovative inspection technologies such as aerial and terrestrial imaging, optical and laser scanning, and robot-assisted inspection have revolutionized quality control in the railway sector [181]. Aerial imaging, conducted with drones or aircraft, is particularly effective for inspecting large-scale rail infrastructures. This method allows for rapid data collection over extensive areas, providing comprehensive overviews of track conditions [182].

Terrestrial imaging, which uses ground-based cameras, is better suited for thorough inspections of easily accessible track sections. This approach enables detailed examination of specific track areas, identifying defects that might be missed by broader aerial surveys [183]. Optical and laser scanning techniques are used to create detailed 3D models and high-resolution surface mappings of rail tracks. These technologies provide precise measurements of track geometry, allowing for the detection of even minor deviations and irregularities. The high-resolution data collected can be analyzed to identify cracks, and other defects that could compromise track integrity [184]. Robotic inspections are employed to reach areas that are either hazardous or difficult to access [185]. Robots equipped with advanced sensors can navigate complex track environments, performing detailed inspections in locations that would be unsafe or impractical for human inspectors. In railway defect detection, these imaging technologies are preferred for their ease of use, cost-effectiveness, comprehensive coverage, high-resolution details, flexibility, and data integration capabilities. The advantages of these methods, coupled with their non-destructive nature, make them ideal for ensuring railway safety and reliability. By allowing for thorough inspection and maintenance while preserving the integrity of the railway infrastructure, these technologies help maintain continuous and safe rail operations. Machine learning and AI further enhance railway defect detection by analyzing the vast amounts of data generated by these advanced inspection technologies. AI systems can continuously monitor the condition of rail tracks, identifying potential issues before they develop into significant problems. This preventive approach is crucial in minimizing accident risks, thereby increasing the overall safety and dependability of rail services. Real-time data and alerts from AI-driven systems enable maintenance teams to address potential risks promptly, ensuring that railway operations consistently adhere to high safety standards [186]. In summary, the integration of NDE methods and advanced imaging technologies in railway defect detection represents a significant advancement in the field. These innovations provide a framework for maintaining rail infrastructure, ensuring that it remains safe, reliable, and efficient. Figure 5 shows railway inspection technologieswhich are using optical and laser scanning systems.

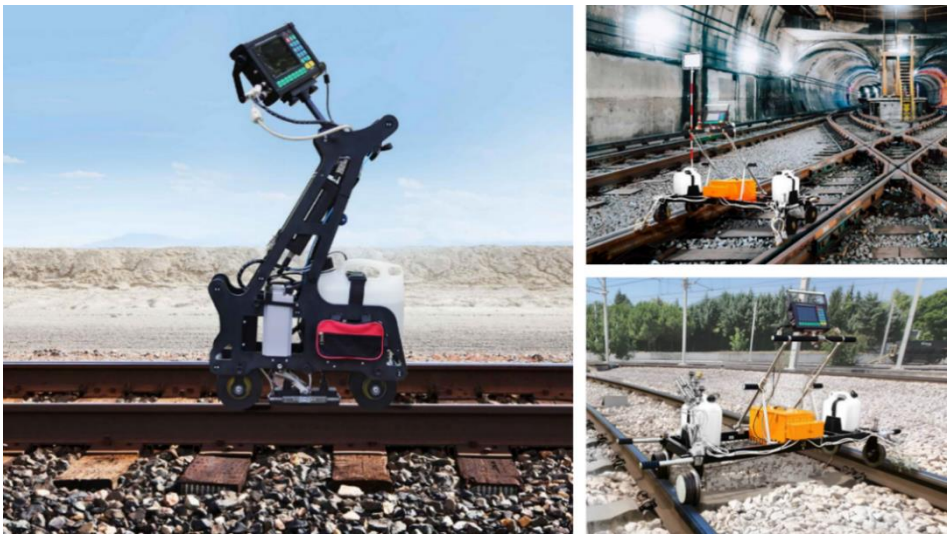

Figure 5. Advanced Railway Inspection Technologies: Utilizing Optical and Laser Scanning Systems[2]

4. SMART COMPUTER VISION IN SHM

In this section, we discuss several important concepts related to our study. These include transfer learning, CNNs, specific CNN architectures like ResNet50, VGG16, and VGG19, the ViT, and the DeiT. We will also cover K-fold cross-validation and key performance metrics such as accuracy, precision, recall, and loss. Additionally, we will outline the workflow of our machine-learning project. This information will provide a clear understanding of the techniques and methods used in our study.

4.1 TRANSFER LEARNING

Transfer learning is a machine learning concept that involves leveraging knowledge gained from solving one problem and applying it to a different, but related, problem [187]. In traditional machine learning, models are typically trained from scratch for a specific task. However, transfer learning acknowledges that knowledge acquired from solving one task can be beneficial for solving a different, yet related, task [188]. For example, in image recognition, a model pre-trained on a vast dataset for object recognition can be fine-tuned for

[2] https://www.railway-technology.com/contractors/training/okondt-group/

a specific task, such as facial recognition or detecting specific objects in medical images. Transfer learning facilitates faster convergence and often requires less annotated data for the target task [189-191]

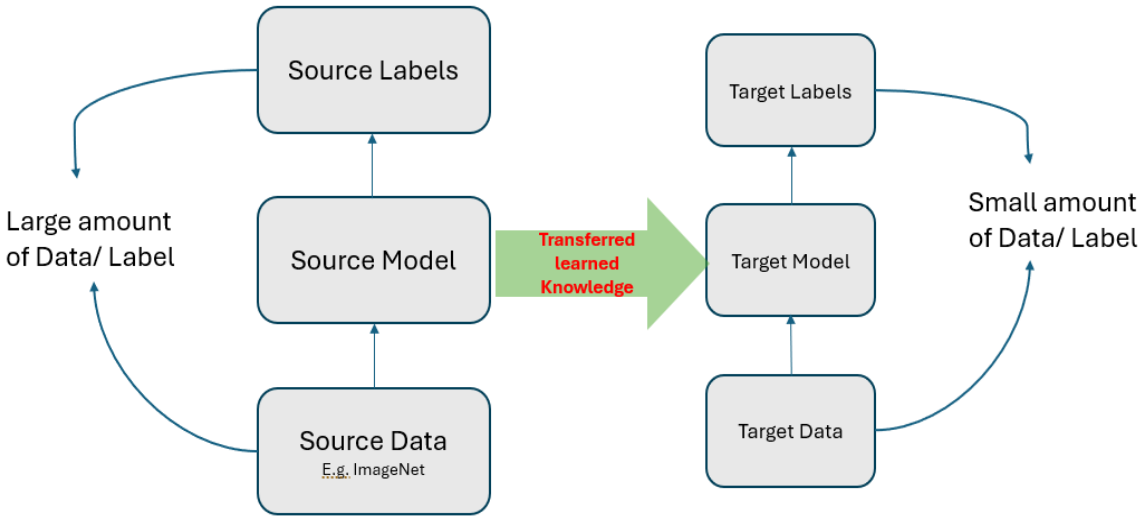

Figure 6. The idea of Transfer Learning[3]

Pretrained models are neural networks that have been trained on massive datasets for a specific task, such as image classification, natural language processing, or speech recognition. These models, already equipped with learned features and representations, serve as powerful starting points for new tasks [190]. The idea behind pre-trained models is to capture general features and patterns in the data, which can then be fine-tuned or adapted for a specific task. These models serve as a starting point for transfer learning. Common pre-trained models include architectures like VGG, ResNet, BERT, GPT, etc., which have been trained on large-scale datasets for tasks like image classification, object detection, natural language understanding, etc [191]. Figure 6 shows a typical transfer learning pipeline, where a large, labeled dataset (e.g., ImageNet) is used to train a source model. Knowledge gained from this source model is then transferred to a target model, which is fine-tuned using a much smaller labeled dataset. This approach leverages the previously learned features to improve

[3] https://www.slideshare.net/slideshow/transfer-learning-d2l4-insightdcu-machine-learning-workshop-2017/75523347

performance on tasks with limited data.Transfer learning and pre-trained models offer a practical solution to the challenges of resource-intensive model training and data scarcity in machine learning. By reusing pre-trained models, these approaches significantly reduce the computational burden and time required for training new models, promoting efficiency [192]. Additionally, pre-trained models demonstrate data efficiency, excelling in tasks with limited labeled data. Their ability to capture high-level features and representations results in improved performance compared to models trained from scratch, particularly in scenarios with constrained training data. Transfer learning and pre-trained models serve as valuable tools, enhancing the overall efficiency and effectiveness of machine learning systems across diverse applications [193].

4.2 CONVOLUTIONAL NEURAL NETWORK (CNN)

Based on Figure 7, which illustrates the architecture of a CNN, we can delve into a more detailed explanation of how CNNs function. CNN is a class of deep neural networks, most applied to analyzing visual imagery. The architecture of CNN is designed to automatically learn spatial hierarchies of features from input images. Here's a step-by-step breakdown of the CNN process is explained below:

a. **Image Input:** The process starts with an input image. This image is represented in the form of a matrix of pixel values.

b. **Convolutional Layers**: The first stage of a CNN is a series of convolutional layers. These layers use filters or kernels to perform convolution operations that detect various features such as edges, corners, or other visual elements. The filters slide over the image and compute the dot product of the filter values and the original pixel values of the image. Each filter produces a different feature map.

c. **Pooling Layers:** Following convolution, the network applies pooling layers, typically max pooling, to reduce the dimensionality of each feature map. Pooling helps to make the detection of features invariant to scale and orientation changes and also reduces the computational load for the network.

d. **Fully Connected Layers:** After pooling, the network flattens the pooled feature maps and feeds them into a series of fully connected layers. These dense layers perform classification based on the features extracted in previous layers. Each neuron in a fully connected layer has full connections to all activations in the previous layer.

e. **Output:** The final layer in a CNN is the output layer. In a classification task, this output layer will often consist of a SoftMax activation function that converts the outputs into probability scores for each class. The class with the highest probability is the network's prediction.

The entire CNN consists of two parts: feature extraction (convolution and pooling) and classification (fully connected layers). The feature extraction part uses convolutional layers to automatically identify important patterns and features in the input images, followed by pooling layers to reduce the dimensionality of these features while preserving essential information. This process eliminates the need for manual feature selection. The classification part then takes the extracted features and uses fully connected layers to map them to the output classes, determining the probability that the input image belongs to each class. This automated approach of learning feature representations directly from images is one of the biggest advantages of CNNs, making them highly effective for tasks like image recognition, medical image analysis, and other areas of computer vision. [194- 195].

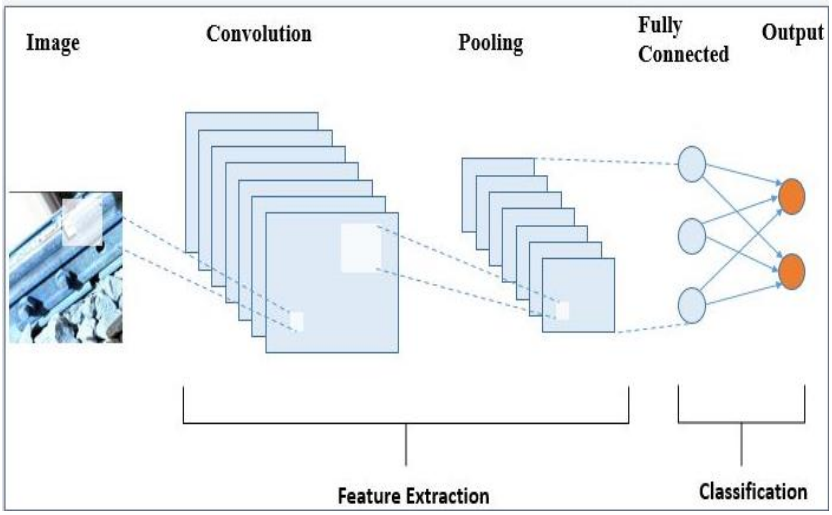

Figure 7. Simple CNN architecture

3.2.1 Resnet50:

ResNet50, with 50 layers, is a deep CNN designed to solve the vanishing gradient problem that often occurs when training very deep networks. This problem arises when gradients become very small during backpropagation, making it difficult for the network to learn effectively as more layers are added. ResNet50 introduces a novel approach called residual learning to overcome this challenge. Residual learning uses shortcut connections, also known as skip connections, which allow gradients to flow more easily through the network. These shortcuts bypass one or more layers by connecting the output of a layer directly to the output of a deeper layer. This direct path helps to preserve the gradient during backpropagation, ensuring that the network can learn even as it becomes deeper. By facilitating the flow of gradients, these shortcut connections address the vanishing gradient problem, enabling the training of much deeper networks than was previously possible.

The architecture of ResNet50 includes several key components. Convolutional layers apply filters to the input data to extract relevant features. Batch normalization layers normalize the output of the previous layers, stabilizing and accelerating the training process. Activation layers introduce non-linearity into the network, allowing it to Learn complex patterns. Pooling layers reduce the spatial dimensions of the data, making the network more

computationally efficient. Finally, fully connected layers perform the final classification based on the features extracted by the convolutional layers. ResNet50 is part of the broader ResNet family, which includes other variants like ResNet18, ResNet34, ResNet101, and ResNet152, each with different numbers of layers and complexity. ResNet50 has proven highly effective in various image recognition tasks and is widely used in both academic research and industry applications. It is known for its strong performance on benchmark datasets such as ImageNet, achieving state-of-the-art results. The ability of ResNet50 to train very deep networks without suffering from the vanishing gradient problem has made it a significant advancement in the field of deep learning and image recognition. [196- 97].

3.2.2 VGG6 and VGG19:

VGG16 and VGG19 are deep CNN architectures developed by the VGG at the University of Oxford. These models have gained significant prominence due to their depth and capability in handling large-scale image recognition tasks, such as those seen in the ImageNet competition. VGG16, as the name suggests, is composed of 16 weight layers. These include 13 convolutional layers, which are tasked with extracting features from the input images. Each convolutional layer utilizes small 3x3 filters, an approach that helps capture fine details while keeping the number of parameters manageable. Following these convolutional layers are 3 fully connected layers. The first two fully connected layers have 4096 nodes each, while the final layer has 1000 nodes, corresponding to the number of classes in the ImageNet dataset. This structure allows VGG16 to effectively learn and classify high-dimensional data. On the other hand, VGG19 extends the architecture of VGG16 by adding three more convolutional layers, bringing the total to 19 weight layers. These additional convolutional layers enable VGG19 to capture more complex patterns and finer details within the images. While this added depth can lead to improved performance in recognizing intricate features, it also makes VGG19 more computationally intensive, requiring more resources for both training and inference. Despite the increased computational load, VGG19's enhanced capability to learn detailed features makes it a valuable tool for tasks demanding high accuracy. The primary distinction between VGG16 and VGG19 is the additional three convolutional layers in VGG19, which result in greater computational demands. However,

these extra layers also allow VGG19 to learn more complex features from the input data, which can be particularly advantageous in tasks requiring detailed image analysis. Both models have proven highly effective in large-scale image recognition challenges and have influenced the development of subsequent deep-learning architectures. The VGG architectures have not only excelled in their own but have also served as foundational models for further advancements in the field of computer vision. Their impact is evident in various applications, ranging from image recognition to medical image analysis, demonstrating their versatility and robustness in handling complex visual data [198].

4.3 VISION TRANSFORMER (VIT)

The ViT represents a novel approach in computer vision, leveraging the transformer architecture, traditionally used in natural language processing. In the ViT framework, an image is treated similarly to a sequence of words or tokens. Typically, the following steps are performed.

a. **Linear Projection of Flattened Patches:** The process begins with the division of an image into a grid of non-overlapping patches. Each patch is then "flattened", meaning its pixel values are unrolled into a single vector. These vectors are subsequently subjected to a linear projection to transform them into embeddings, making them compatible with processing by the transformer.

b. **Transformer Encoder:** These patch embeddings are then fed into a transformer encoder. The encoder consists of multiple layers that feature multi-head attention mechanisms and feed-forward networks, interspersed with normalization layers. The strength of the transformer lies in its ability to capture long-range dependencies within the input. In the context of ViT, this means it can detect relationships between different patches of the image, which is crucial for tasks like image recognition.

c. **Class Token and MLP Head:** A unique aspect of the ViT is the introduction of a 'Class' token alongside the patch embeddings. This token gathers global contextual information about the image as it passes through the transformer layers. After the

encoder, the transformed representation of this class token is then passed through a Multi-Layer Perceptron (MLP) head. This final step is where the model makes its prediction, determining the category or class that the input image belongs to. This approach allows the Vision Transformer to effectively handle image recognition tasks by understanding both the local features of image patches and their global arrangement, thereby providing a comprehensive analysis of the image as a whole [199].

3.3.1 ViT-base-patch16-224:

ViT-base-patch16-224 is a specific configuration of the ViT. The term "ViT-base" indicates that this is a medium-sized transformer model. In the context of transformer models, "base" denotes a particular architecture that includes 12 layers, also known as transformer blocks. Each layer contains 768 hidden units and 12 attention heads. This configuration strikes a balance between complexity and performance, making it suitable for many standard tasks without being as resource intensive as larger models. The "patch16" part of the name refers to how the model processes input images. Instead of analyzing entire images directly, the Vision Transformer divides each image into smaller patches. In this case, each patch is 16x16 pixels in size. These patches are then treated as individual tokens, like how words are treated in natural language processing models. This approach allows the model to capture detailed information from various parts of the image, which is crucial for accurate image classification. Finally, the "224" in the model's name specifies the input image size. Before being fed into the model, images must be resized to 224x224 pixels. This standardization ensures consistency in the input data, which is essential for training and evaluating the model effectively. By using images of this specific size, the model can efficiently process and classify images, leveraging its transformer-based architecture. Overall, ViT-base-patch16-224 is designed for standard image classification tasks and is particularly effective when large-scale datasets are available for training. As part of the broader Vision Transformer family, this model configuration aims to provide a balance between computational efficiency and predictive performance, making it a versatile choice for various image recognition applications [200-201].

3.3.2 DeiT

The DeiT is a specialized version of the ViT designed for image classification tasks. It is trained using a teacher-student strategy, where a "teacher" model, typically a more complex and well-trained network, guides a "student" model. This method incorporates a unique "distillation token" that interacts with the class and patch tokens through self-attention layers. The distillation token helps the student model focus and learn effectively from the teacher's knowledge, thereby improving its performance with less data and computational resources. DeiT Models are notable for their efficiency and effectiveness, achieving competitive accuracy on standard benchmarks like ImageNet without the need for extensive datasets or large-scale computational resources used by traditional transformers [114].

4.4 K-FOLD CROSS-VALIDATION

The k-fold cross-validation strategy involves partitioning the dataset into K equal segments or folds. In each iteration of the training/validation process, one-fold is Used as the validation set while the remaining K-1 folds are combined to form the training set. This cycle is repeated until each fold has been used once as the validation set, ensuring that every data point contributes to both the training and validation phases [202]. In our study, we used 10-fold cross-validation to evaluate machine learning models for defect detection. We chose K=7 to balance bias and variance, offering a thorough assessment without overfitting. This setting is also computationally efficient compared to more folds.

4.5 METRICS

Metrics provide a quantitative measure of how well a model performs. Different metrics can be used to assess different aspects of a model's performance, such as its accuracy, precision, recall, and loss [201].

Loss: In machine learning, the loss is a mathematical function that measures the difference between the predicted output and the actual output (or target). The loss function measures how closely the model's Predictions match the actual values. The goal during training is to

minimize this loss. Given that we are distinguishing between two classes - defective and non-defective - we employ the binary loss formula as below. The binary loss formula effectively penalizes wrong predictions, especially those that are confidently incorrect. The use of logarithms in this formula amplifies the penalty for predictions that are far off from the actual value. The main objective in the model's training process is to minimize this loss value, thus enhancing the accuracy of the model's predictions. Here is the formula of loss: $loss = -(y\log(\hat{y}) + (1-y)\log(1-\hat{y}))$

Confusion Matrix: A confusion matrix is a tool used in classification tasks to visualize the performance of an algorithm. As shown in Figure 8, the confusion matrix is a table that displays the number of correct and incorrect predictions, categorized by each class. Binary classification, like distinguishing between defective and non-defective items, consists of four parts: True Positives, True Negatives, False Positives, and False Negatives. This matrix helps in calculating key performance metrics such as accuracy}, precision, and recall, providing a clear picture of the model's strengths and weaknesses in predicting the two classes.

Accuracy: When we wish to evaluate the effectiveness of a binary classifier, accuracy is the statistic that is often used. In our case, it represents the number of times a model has correctly predicted the class of an image divided by the total number of predictions made across the two classes (defective and non-defective). The formula for determining accuracy is the ratio between the number of true predictions (TP + TN) and the total number of predictions (TP + TN + FP + FN). Here is the formula for calculating accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

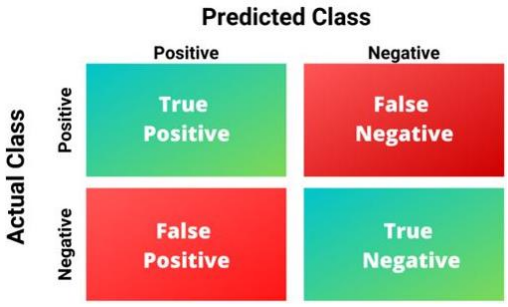

Figure 8. Confusion Matrix

Precision and Recall: Confusion matrix is used to introduce the concepts of precision and recall. Precision is just the accuracy that is calculated only for positive predictions. It is also called specificity because it shows how sensitive an instrument is when it needs to recognize an output. The metric tells us the percentage of times we are right when we label a class as positive. Recall, on the other hand, helps us figure out the percentage of positive samples that were correctly identified. Here is the formula for calculating precision and recall:

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

In an image classification project distinguishing between defective and non-defective items, choosing the right metrics is crucial. While the cross-entropy loss function guides learning, it doesn't fully reflect practical performance. Metrics like accuracy, recall, and precision are essential. Accuracy shows overall performance but can be misleading with imbalanced datasets, such as having more non-defective items. Recall measures the percentage of actual defects correctly identified and is crucial when missing defects is costly or dangerous, even if it flags some non-defective items as defective. Precision ensures that predicted defective items are indeed defective, minimizing unnecessary waste or costs.

4.6 WORKFLOW OF MACHINE LEARNING PROJECT

Figure 9 outlines the machine learning workflow, starting with problem intuition understanding the issue, and proceeding to data collection for gathering relevant data. The next steps involve data preprocessing to clean, normalize, and encode the data, followed by feature engineering to select, and transform features. After this, model selection involves

choosing the appropriate algorithm, and model training fits the data to this model. Once trained, the model is evaluated to assess its performance, and hyperparameter tuning optimizes this performance. The model is then tested on unseen data, followed by deployment into systems. Ongoing monitoring and maintenance ensure the model remains effective, and a feedback loop refines the model based on real-world feedback.

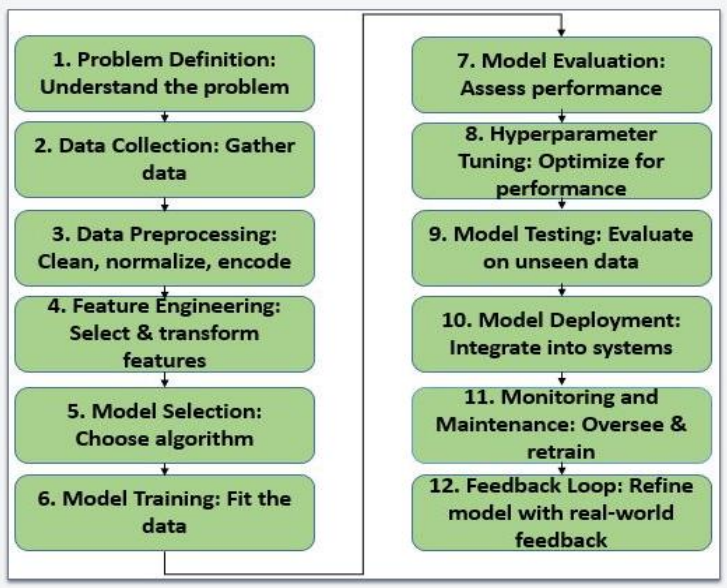

Figure 9. Workflow of Machine Learning Project

CHAPITRE 2

VERS UNE MAINTENANCE INTELLIGENTE DES VOIES FERRÉES : ÉVALUATION NON DÉSTRUCTIVE AMÉLIORÉE PAR L'IA AVEC DES TRANSFORMERS DE VISION ET DES CNNS POUR LA DÉTECTION DE DÉFAUTS DES ATTACHES

1. RÉSUMÉ EN FRANÇAIS DU PREMIER ARTICLE

Cette recherche porte sur la maintenance prédictive des infrastructures ferroviaires en utilisant des techniques d'évaluation non destructive (NDE) et l'imagerie pour identifier les défauts des fixations de voies ferrées. En exploitant des modèles d'apprentissage automatique, y compris les réseaux de neurones convolutifs (CNNs) et les architectures basées sur les transformeurs, l'étude identifie Vision Transformer (ViT) et Data-efficient Image Transformer (DeiT) comme les modèles les plus performants en raison de leur excellente généralisation et efficacité d'apprentissage. L'intégration de l'IA, du machine learning et des technologies de l'Industrie 4.0 améliore la maintenance ferroviaire en automatisant la détection des défauts, augmentant ainsi la fiabilité et réduisant les coûts.

L'étude passe en revue différentes techniques de détection des défauts, en commençant par les méthodes de traitement d'image traditionnelles, comme la détection des contours et les opérations morphologiques, qui sont limitées face aux motifs de défauts complexes. Elle explore ensuite les approches d'apprentissage automatique, notamment les méthodes supervisées (ex. : SVMs, Random Forests) et non supervisées (ex. : Clustering, Autoencodeurs), qui exploitent les données historiques mais nécessitent une ingénierie des caractéristiques poussée. L'adoption de modèles d'apprentissage profond, tels que ResNet, AlexNet, Inception et YOLO, a permis d'améliorer la classification des défauts en extrayant automatiquement les caractéristiques pertinentes. Les avancées les plus récentes se concentrent sur les modèles basés sur les transformeurs, y compris ViT, Swin Transformer, TransUNet, Rail-Former et TrackNet, qui surpassent les CNNs grâce à leur capacité à capturer à la fois des caractéristiques locales et globales grâce aux mécanismes d'auto-attention.

Pour valider ces modèles, l'étude utilise un jeu de données public disponible sur Kaggle, comprenant 1400 images de voies ferrées (700 défectueuses, 700 non défectueuses), représentant des fixations telles que des boulons, des clips, des ancrages et des plaques. Le prétraitement des données inclut le redimensionnement des images (224x224 pour l'entraînement, 256x256 pour la validation/test), la normalisation, l'augmentation des données (recadrage aléatoire, retournement horizontal) et une validation croisée stratifiée à 10 plis afin d'assurer un équilibre entre les classes. L'étude évalue des architectures CNN (ResNet50, VGG16, VGG19) aux côtés des modèles basés sur les transformeurs (ViT, DeiT). Alors que les CNNs sont fiables, ils peinent à capturer le contexte global des images, tandis que ViT et DeiT traitent les images sous forme de séquences de patches, améliorant considérablement la détection des défauts. L'étude applique l'apprentissage par transfert à l'aide de modèles pré-entraînés sur ImageNet et utilise Optuna pour l'optimisation des hyperparamètres, ajustant notamment le taux d'apprentissage, le taux de dropout, le momentum et la taille des lots.

L'évaluation des performances repose sur des métriques clés, notamment la fonction de perte, l'exactitude, la précision, le rappel et l'aire sous la courbe ROC (AUC ROC). Les résultats montrent que ViT et DeiT obtiennent les meilleurs scores en AUC ROC (0.97 & 0.98), exactitude (98 % & 95.7 %) et rappel (98.46 % & 98.62 %), ce qui en fait les modèles les plus performants. VGG16 et VGG19 affichent des performances modérées (~93 % d'exactitude), tandis que ResNet50 est moins efficace (~85 % d'exactitude, 0,89 AUC ROC). L'importance de l'optimisation des hyperparamètres est évidente, Optuna permettant d'améliorer significativement le rappel et l'exactitude, réduisant ainsi le risque de faux négatifs. Sur un jeu de test distinct de 210 images, ViT (98.09 % d'exactitude) et DeiT (95,71 % d'exactitude) surpassent les modèles basés sur les CNNs, confirmant leur supériorité pour la classification des défauts.

Cette étude conclut que les transformeurs de vision (ViT et DeiT) sont les modèles les plus efficaces pour la détection des défauts des fixations ferroviaires, surpassant les CNNs grâce à leur capacité à analyser globalement les images et à se focaliser sur plusieurs régions simultanément. Alors que les modèles VGG offrent des performances acceptables, ils sont

dépassés par les transformeurs en matière d'efficacité d'apprentissage et de précision, tandis que ResNet50 est le moins performant dans ce contexte. La recherche met également en avant le rôle crucial de l'optimisation des hyperparamètres, qui influence considérablement les performances des modèles et souligne la nécessité d'une optimisation systématique.

Malgré ses points forts, l'étude reconnaît certaines limites, notamment le fait que la répartition équilibrée du jeu de données ne reflète pas nécessairement la répartition réelle des défauts et que le coût computationnel élevé des transformeurs peut limiter leur déploiement dans des environnements à ressources restreintes. Les recherches futures se concentreront sur l'extension des jeux de données, l'optimisation de l'efficacité computationnelle et le développement d'architectures hybrides CNN-Transformers afin d'équilibrer performance et efficacité. Ces résultats renforcent le rôle croissant des transformeurs de vision dans la détection des défauts ferroviaires, posant ainsi les bases pour de futures avancées dans la maintenance ferroviaire automatisée et l'analyse prédictive.

Mots-clés : Détection de défauts sur attaches ferroviaires, Apprentissage automatique, CNN, Transformers.

2. TOWARD SMART RAILWAY MAINTENANCE: AI-ENHANCED NON-DESTRUCTIVE EVALUATION USING VISION TRANSFORMERS AND CNNS FOR FASTENER DEFECT DETECTION

Title: Toward Smart Railway Maintenance: AI-Enhanced Non-Destructive Evaluation Using Vision Transformers and CNNs for Fastener Defect Detection

Samira Mohammadi*, Samira.Mohammadi2@uqar.ca[1]

Sasan Sattarpanah Karganroudi, sattarpa@uqtr.ca[2,3]

Mehdi Adda, Mehdi.Adda@uqar.ca[1]

Hussein Ibrahim, Hussein.Ibrahim@uqtr.ca[3]

Abstract

Predictive health management and maintenance of transport infrastructure are critical for preventing accidents and service disruptions. Applying Non-Destructive Evaluation (NDE) and imaging techniques is essential for identifying irregularities without causing harm. This research utilizes pre-trained models and incorporates transfer learning concepts to overcome dataset constraints. This study assesses the effectiveness of various machine learning models, including the Vision Transformer (ViT), Data-efficient Image Transformer (DeiT), VGG19, VGG16, and ResNet50, in enhancing NDE for railway track fasteners. ViT and DeiT, both transformer-based models, emerged as the top performers due to their superior learning efficiencies and generalization capabilities, augmented by precise hyperparameter tuning. VGG models are a reliable alternative, while ResNet50 is better suited for applications prioritizing computational efficiency over accuracy.

Keywords: Railway Fastener Defect detection, Machine learning, CNN, Transformers

1. Introduction

In recent years, rail transit has rapidly advanced globally, becoming a key mode of modern transportation. This growth has increased safety demands, making the condition of railway tracks crucial for the safe and reliable operation of trains [1]. Transportation is vital today, connecting people and commerce. Railways offer a reliable, eco-friendly way to move goods and people, making them essential to modern transport systems [2]. Railways fuel economic growth, connect communities, support trade, and offer a sustainable option for travel and freight. They reduce urban congestion, are eco-friendly, and generate over a trillion dollars annually, employing millions globally. Railways also contribute significantly to national GDPs while promoting

[1] Department of Mathematics, Computer Science and Engineering, Université du Québec à Rimouski, 1595 Bd Alphonse-Desjardins, Lévis, G6V 0A6, QC, Canada

[2] Department of Mechanical Engineering, Université du Québec à Trois-Rivières, 575 Boul de l'Université, Drummondville, J2C 0R5, QC, Canada

[3] Centre national intégré du manufacturier intelligent, Université du Québec à Trois-Rivières, 575 Boul de l'Université, Drummondville, J2C 0R5, QC, Canada

tourism and efficient transport [3]. In developing nations, railways connect remote regions to economic hubs, promoting growth. However, insufficient investment has led to outdated and inefficient systems despite their many benefits [4]. Rails wear down over time due to frequent train use, high speeds, heavy loads, and changing weather. Track damage can lead to derailments, risking passenger and worker safety. In 2009, track faults caused 658 of 1,890 railway accidents. [5]. During the past decade, about one in every three train accidents in the United States has been due to issues related to the tracks [6]. The significant risks associated with rail track defects emphasize the need for diligent maintenance and repair of rail lines. Regular checks and upkeep are crucial to lessen hazards and avoid disruptions in the railway system [7]. Track maintenance is one of the costliest aspects of railway engineering, consuming over 50% of maintenance budgets in places like the Netherlands. Innovative strategies are essential to reduce costs, risks, and improve safety [8]. The railway industry is adopting connected devices, sensors, and big data for better maintenance. Machine learning uses data from sensors and cameras to boost reliability and reduce costs and risks [9]. AI significantly enhances railway safety by monitoring tracks, detecting minor issues, and preventing accidents. Real-time alerts enable quick responses, ensuring consistent safety and reliability [10]. Industry 4.0, the Fourth Industrial Revolution, incorporates automation and data exchange in manufacturing using technologies like IoT, cloud computing, AI, and cyber-physical systems [11]. Machine learning algorithms in defect detection and quality control offer greater accuracy and cost savings than manual inspections [12]. Geometric inspection of high-speed railway tracks, aided by 3D scanning technologies, ensures precision and efficiency, crucial for safety [13]. Non-Destructive Evaluation (NDE) is a vital technique in Industry 4.0, allowing the inspection of materials, components, or systems without causing damage [14]. Advancements in Industry 4.0 have led to the integration of AI, machine learning, robotics, and advanced sensors into NDE, enhancing the accuracy and efficiency of flaw detection [15]. Automated inspection technologies, such as aerial and terrestrial imaging, optical/laser scanning, and robotized inspection, have improved quality control across industries [16]. Aerial imaging is ideal for large-scale checks, while terrestrial imaging is suited for detailed inspections [17]. Optical and laser scanning create 3D models and high-resolution maps, and robotized inspection reaches hard-to-access areas [18]. Imaging methods offer ease of use, cost-effectiveness, comprehensive coverage, high resolution, flexibility, and data integration capabilities, making them ideal for railway safety and reliability.

In this research, we used NDE methods for imaging and explored the strengths of pre-trained CNNs and transformers in enhancing railway fastener defect detection, especially with limited data. We compared the effectiveness of CNNs and transformer models, utilizing ResNet50, VGG16, VGG19, ViT, and DeiT models, highlighting the superior performance of transformer models through NDE methods. This article is organized into seven sections. Section 2 introduces the background. Section 3 conducts a thorough literature review. Section 4 outlines our methodology and materials. Results are presented in Section 5, followed by a detailed discussion in Section 6. Finally, Section 7 concludes our research.

2. Literature Review

Defect detection improves processes by providing insights into root causes [19]. Leveraging data analytics and machine learning, organizations gain deeper insights into production processes for informed decision-making [20]. Defect detection has been a prominent field of research in computer vision and image processing, with many studies exploring various techniques and algorithms. Traditional image processing methods, such as edge detection, thresholding, and morphological operations, have been widely utilized for defect detection tasks [21], [22]. These techniques are efficient in terms of computation and rely on basic image features. However, they often encounter difficulties when dealing with complex or varied defect patterns. Machine learning algorithms, both supervised and unsupervised, have gained significant traction in defect detection due to their capability to learn from data and adapt to different conditions. Supervised techniques, including support vector machines (SVMs) [23], random forests [24], and boosting algorithms [25], have shown strong performance in numerous applications. On the other hand, unsupervised methods, such as clustering algorithms [26] and autoencoders [27], are valued for their ability to detect anomalies or defects without the need for labeled data. In recent years, deep learning approaches, particularly convolutional neural networks (CNNs), have set new standards in image-based defect detection [28], [29]. These models excel in automatically extracting distinguishing features from raw image data, enabling them to detect and classify intricate defect patterns with high accuracy. Furthermore, transfer learning and domain adaptation techniques have broadened the applicability of deep learning models by allowing the use of pre-trained networks from related fields [30], [31]. Building upon these advancements, several studies have demonstrated the effectiveness of CNNs and Vision Transformer models in defect detection across diverse applications. For instance, Jamshidi et al. [32] used basic CNN architectures with real-world image datasets for squat flaw classification, incorporating ultrasonic measurement data. Shams et al. demonstrated that CNN models enhance accuracy across complex datasets like Fashion-MNIST, MNIST, and EMNIST Digits, significantly improving image recognition performance [33], [34]. Faghih-Roohi et al. [35]35] trained CNNs on Dutch rail track datasets for automatic squat flaw detection and classification. Dai et al. [36] employed AlexNet and ResNet for fastener defect detection, demonstrating the generalization capabilities of pre-trained ResNet models. Gibert et al. [37] developed CNNs for rail track material analysis and rail fastener condition classification. Ritika et al. [38] used a pre-trained Inception V3 CNN for detecting sun kink faults. Soukup et al. [39] constructed a two-layer CNN using photometric stereo images to overcome dataset size limitations with data augmentation. Chandran et al integrated deep learning with image processing for missing clamp detection on the Borlange-Avesta rail line in [40] . Bai [41] introduced an improved YOLOv4 method for railway surface defect detection using MobileNetv3 and deep separable convolution. Zheng et al. [42] used CNN to detect rail surface and fastener defects. Sresakoolcha et al. [43] used track geometry data for defect detection, developing models with supervised machine learning and analyzing defect relationships with unsupervised methods. Xu et al. [44] used deep learning for railway subgrade defect recognition from ground-penetrating radar (GPR) data. Wei et al. [45] applied transfer

learning to improve fastener detection. Wang et al. [46] proposed a method to detect defects in split pins of high-speed railway catenary devices using transfer learning and pre-trained models. Wu et al. [47] used a pre-trained ResNet-101 model for fastener defect detection. Lu [48] fine-tuned the pre-trained ResNet V2 for defect detection. Li et al. [49] used transfer learning and pre-trained models for rail defect detection. Jian et al. [50] employed transfer learning for multi-category defect detection.

Vision Transformers have demonstrated capabilities in a wide range of applications, from steel plate defect detection to medical diagnostics, indicating a shift towards more sophisticated and efficient analytical techniques [51] [52]. Notably, studies by Hütten et al. have shown that Vision Transformers can perform as well as or better than CNNs, even with limited data availability [53]. In 2021, Alexakos et al. [54] used an image classification transformer for classifying vibration images. Dang et al. [55] used DETR for sewer pipe defect detection. Tang et al. [56] and Li et al. [57] used Swin Transformers for steel plate surface defect detection. Lu et al. [58] used ViT for classifying ultrasonic flaw detection B-scan images. In the field of concrete and structural applications, Wan et al. introduced the BR-DETR model, a pioneering transformer-based architecture designed for surface damage detection on concrete bridges. This model achieved a high mean average precision (mAP) of 91.9%, but its deployment is limited by the need for extensive datasets and high computational demands [59]. Similarly, Shamsabadi et al. advanced the field with the TransUNet, a hybrid model combining ViT with CNNs to enhance crack detection on various surfaces, demonstrating robustness against noise and a significant improvement over traditional CNNs [60]. The industrial sector has also seen notable advancements with Shang et al. [64] developing the Defect-Aware Transformer Network (DAT-Net), which specializes in detecting tiny and irregular surface defects like tool wear, achieving an mIoU score of over 90%. However, its high computational needs restrict its application in real-time environments [61]. Guidong Yang et al. highlighted the effectiveness of the Swin Transformer in defect detection tasks, achieving 88% accuracy and demonstrating high performance in segmentation tasks, though the scarcity of publicly available datasets for training these models remains a challenge [62]. In addition to these models, hybrid approaches combining the strengths of CNNs and Transformers are emerging as powerful tools for defect detection. Wang et al. proposed a hybrid model that integrates CNN, Transformer, and Multi-Layer Perceptron (MLP) components for defect classification in reinforced concrete bridges, achieving an accuracy of 85.5%, recall of 85.0%, and an F1-score of 85.2% [63]. Similarly, Mohammad Shahin et al. combined a Visual Transformer with a CNN to enhance detection of concrete cracks, achieving an accuracy of 99%, albeit with increased training times and computational demands [64]. Mingchao Li et al. introduced CrackTrNet, a CNN-Transformer hybrid model designed for mobile devices, achieving 97.6% segmentation accuracy in concrete dam inspections. This model combines CNN's local feature extraction capabilities with the Transformer's global context awareness, yet its use is currently limited to specific structures [65]. Expanding the scope to other applications, Liu et al. (2022) developed a transformer-based framework for defect detection in cylinder liners, integrating a Swin Transformer with a block

division and mask mechanism for enhanced accuracy, especially with small defects [57]. Toqa Alaa et al. used a Vision Transformer-based model for defect detection on metal surfaces, achieving 93.5% classification accuracy and effective localization with an MAE of 3.2 pixels and IOU of 0.72 [66]. An et al. proposed LPViT, a transformer-based model for PCB classification and defect detection, which achieved state-of-the-art results with 98.8% mean average precision on defect detection tasks [67]. In industrial applications, Ding et al. (2023) introduced a novel vision transformer architecture that incorporates Hybrid Window Attention (HWA) and Dynamic Token Normalization (DTN) for balancing local and global information processing, achieving high classification accuracies of 96.8% on the NEU dataset and 98.5% on the DAGM dataset [68]. Recent developments in rail surface defect detection emphasize the critical role of maintaining railway safety and efficiency. The adoption of transformer-based models marks a significant shift towards overcoming limitations of traditional convolutional neural networks (CNNs).

Feng Guo et al. introduced Rail- Former, a transformer-based semantic segmentation network specifically designed for detecting rail surface defects. This model addresses the shortcomings of CNN-based methods, particularly their challenges in preserving small-scale details and capturing hierarchical features. Employing an encoder-decoder architecture with self-attention mechanisms and a Criss-Cross attention module, Rail-Former excels at integrating local and global features. Tested on both public rail datasets and customized datasets, it demonstrated superior performance in terms of mean Intersection over Union (mIoU) and defect visualization, showcasing its ability to detect finer details that are often missed by CNNs [69]. Shiyao Lu et al. utilized the Vision Transformer (ViT) to classify rail defects using B-scan images from ultrasonic flaw detectors. By focusing on the global structure of images, the ViT model achieved high classification accuracy, significantly reducing the manual labor and time traditionally required for rail defect detection. Despite its success, there is potential for future enhancements through dataset expansion and model fine-tuning [58]. TrackNet, another transformer-based model, was designed to detect defects in ballastless tracks on high-speed railways. Using multi-head self-attention, TrackNet improves global feature extraction, effectively overcoming CNN limitations. It also incorporates transfer learning to reduce dependence on large, high-quality datasets, enhancing its adaptability. The model not only showed improved accuracy and F1-score over competitors like Swin Transformer but also provided interpretability through heatmap visualizations, aiding in understanding the decision-making process. However, challenges in generalizing to unseen environments and scaling for real-time use remain, especially in resource-limited settings [70]. Also, Jin He et al. introduced the CNN-Transformer Bridge Mode (CTBM-DAHD) network to detect defects in arcing horns, crucial components of the overhead contact system. This network blends CNNs for local feature capture with Transformers for global context, enhancing detection under diverse conditions such as rain and nighttime. Effectively deployed on over 500 high-speed trains in China, CTBM-DAHD improved recall and precision significantly, though adapting to different railway environments might require further adjustments [71]. Luo et al. enhanced the YOLOv5s model for rail surface defect detection by

integrating a Swin Transformer, which improved scale detection, and a global attention mechanism in PANet for better feature integration, achieving a 96.9% mAP. Despite its success, there are ongoing challenges with real-time processing efficiency and adaptability to complex environments [72]. Chenghai Yu et al. advanced hybrid modeling with the FHB-DETR model, designed to enhance RT-DETR for railway turnout defect detection. Integrating CNN feature extraction with a modified transformer using Hilo attention, this model reduces computational demands and improves real-time performance, outperforming RT-DETR by 3.5% in mAP50 while reducing computational complexity by 6%, promising better efficiency for high-speed rail applications [73]. Suli Bai and colleagues developed a vision-based non-destructive detection network that combines CNN and Transformer models to identify rail surface defects. Featuring an enhanced Res2Net for detailed local context capture and a Transformer block for global context awareness, this network achieved excellent results on various datasets, indicating its potential despite its high computational demands [74].

These advancements highlight the significant potential of Vision Transformers to enhance defect detection capabilities across various domains. Despite the computational demands and extensive data requirements, ongoing research and optimization are expected to mitigate these challenges, enhancing the practicality of Vision Transformers in real-time applications across diverse industries.

3. Materials and Methods

Railways play a vital role in economic development, offering efficient and eco-friendly transportation. Detecting rail track defects is essential for safety and operational integrity, with regular inspection, maintenance, and machine learning integration improving safety and reducing costs [37]. AI is assisting railway track inspection, improving safety and efficiency. Through visual inspection, AI is a tool to analyze images or video footage to detect defects not easily visible to humans, using sophisticated computer vision algorithms proven effective in identifying track problems [75]. AI's role extends to predictive maintenance, where machine learning algorithms, trained on past inspections and maintenance records, forecast when maintenance on a track section is likely. This proactive scheduling minimizes unexpected failures, ensuring the integrity of railway infrastructure and reliable operations [76]. AI enhances safety in railway operations by monitoring track conditions in real time. It proactively alerts maintenance teams to emerging issues, reducing the risk of accidents and ensuring the safety and reliability of railway operations. Real-time data and alerts from AI systems address potential hazards promptly, maintaining high safety standards in railway operations [77].

3.1.Dataset

Figure 1. Samples of Defects on railway tracks studied in this article [69]

For this study, we utilize a public railway dataset accessible on Kaggle, which is particularly suited to image classification for fastener defect detection. Some samples are shown in Figure 1. The dataset utilized in the study consists of images collected from railway tracks in Bangladesh, indicating that it comprises field samples and reflects real-world data. dataset is balanced, containing an equal number of defective and non-defective images. It comprises 700 defective images and 700 non-defective images. The dataset itself contains images featuring defects in the fastening components—bolts, clips, anchors, and plates—that secure rail tracks to the underlying infrastructure.

3.2.Data preprocessing

Data preprocessing is a crucial step in the training of deep learning models. In our study, several preprocessing techniques are employed to prepare the images for effective training, validation, and testing of a convolutional neural network. The transforms module from the torchvision library is utilized to apply a series of transformations that standardize and augment the input data. For the training set, RandomResizedCrop is used to randomly crop the images to 224x224 pixels—a common practice for maintaining size consistency while introducing randomness to the dataset, which helps the model generalize better [78]. Additionally, RandomHorizontalFlip applies a random horizontal flip to the images, effectively doubling the dataset size with mirrored images, which adds more variability and prevents the model from overfitting [79]. Both validation and test datasets undergo a series of non-random transformations to ensure consistency in evaluation and testing. Resize is used to scale the images up to 256x256 pixels before they are cropped to 224x224 pixels using CenterCrop. This ensures that during validation and testing, the focus remains on the image's central region, where important features are most likely to be concentrated. All datasets—training, validation, and testing—are normalized using the Normalize function with the mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225], which are standard ImageNet dataset statistics. This normalization aligns the data distribution of the input with that of the data used to train pre-established models, facilitating

better model convergence and performance [80]. The DataLoader component of PyTorch is employed to automate the retrieval of data in batches, which is essential for training neural networks efficiently. By processing data in batches, the computation is spread out, reducing memory overhead and improving training speed. Shuffling of the data is specifically used in the training phase to ensure that each training batch helps the model learn without biasing its learning to the order of the data. We began with a balanced dataset of 1,400 images. To ensure a robust evaluation of the model, we divided these images, employing a stratified approach, into two subsets: training andvalidation subset (85% - 1,190 images), and testing subset (15% - 210 images). Within that the first subset representing the 85% of the dataset, we employed a 10-fold stratified approach to split the data into a training portion, which contains 980 images (70% of the original dataset) and a validation portion with 210 images (15% of the original dataset) at each fold. The stratification process helped maintain consistent class proportions across each subset, preventing any inadvertent imbalance that might arise from purely random splitting.

Using three distinct subsets is important because each one serves a different purpose. The training set is used to fit the model's parameters, allowing the model to learn meaningful representations and patterns. The validation set is then used to monitor how well the model is performing during training and to tune hyperparameters, which helps avoid overfitting. Finally, the test set is kept separate until all model decisions and adjustments are complete; it provides an unbiased measure of performance on unseen data, offering a realistic sense of how the model will generalize in real-world scenarios.

3.3.Models

Convolutional Neural Network (CNN) Architecture: A Convolutional Neural Network (CNN) is designed for visual processing and automatically extracts features from images. CNNs are particularly effective in image recognition and other computer vision tasks due to their ability to detect various features at different levels of abstraction. ResNet50, VGG16, and VGG19 are popular CNN models, which are shown in Table 1 [81]. The architecture typically consists of several types of layers:

- Convolutional Layers: These layers apply convolution operations to the input, detecting features such as edges, textures, and patterns.
- Pooling Layers: These layers reduce the dimensionality of the feature maps, retaining essential information while decreasing computational complexity.
- Fully Connected Layers: These layers act as classifiers, using the features detected by the convolutional and pooling layers to make predictions.

CNNs, like ResNet50, VGG16, and VGG19, are popular choices due to their proven track record in achieving high performance in image classification tasks. These models are pre-trained on large datasets such as ImageNet, enabling them to generalize well to new tasks with similar data.

They have been extensively validated in various benchmarks and real-world applications, making them reliable and effective for fastener defect detection in railways [82], [83], [84]. Table 1 provides an overview of the models used in the study, summarizing their architectures and key characteristics. The models listed include ResNet50, VGG16, and VGG19, all of which are pre-trained on the ImageNet dataset. The Table 1 highlights the depth of each model, the number of layers, and their distinctive features, such as residual connections in ResNet50 and the structured weight layers in VGG models. This comparison establishes the foundation for evaluating the models' performance on specific tasks.

Model	Description
ResNet50 (pre-trained)	A deep convolutional neural network with 50 layers and residual connections is designed to learn complex data features and enhance deep network training. Pre-trained on the ImageNet dataset.
VGG16 (pre-trained)	16 weight layers, including 13 convolutional layers and 3 fully connected layers. It is pre-trained on the ImageNet dataset.
VGG19 (pre-trained)	19 weight layers, with 16 convolutional layers and 3 fully connected layers. It is pre-trained on the ImageNet dataset.

Table 1. Specifications of Neural Network Models: Resnet50, VGG16, and VGG19 [85]

Vision Transformer: The Vision Transformer (ViT) applies the transformer architecture to computer vision, treating images as sequences of patches. These patches are processed to capture long-range dependencies, with a final layer for prediction. ViT models excel in image recognition by analyzing both local features and global arrangements of image patches. ViT-base-patch16-224 and DeiT are shown in Table 2 [86], [87]. Vision Transformers (ViT and DeiT) were chosen because they excel at capturing global features and contextual information across an entire image. This capability is particularly beneficial for tasks requiring detailed analysis, such as fastener defect detection in railways, where defects may not be confined to specific regions. Additionally, Vision Transformers have shown superior performance in recent image classification benchmarks, demonstrating their effectiveness in understanding complex visual data [88].

Model	Description
ViT-base-patch16-224 (pre-trained)	& Splits images into 16x16 patches and uses a transformer encoder to analyze their relationships. Pre-trained on ImageNet, it learns a wide range of visual features, making it effective for image classification.
DeiT-base-patch16-224 (pre-trained)	& Optimized for image classification, with advanced data efficiency and pre-trained knowledge. Effective for tasks with limited data and trained on ImageNet.

Table 2. Vision Transformer Models [86], [87]

3.4.Transfer Learning

Transfer learning leverages knowledge from solving one problem to addressing a related problem [89]. It speeds up convergence and reduces the need for large, annotated [90], [91]. Pretrained models, trained on large datasets for tasks like image classification or natural language processing, serve as starting points for new tasks [92], [93]. These models capture general features and patterns, which can be fine-tuned for specific tasks, improving efficiency and performance in scenarios with limited data [94], [95].

3.5.Metrics

Metrics in machine learning serve as quantifiable measures that play a crucial role in assessing a model's performance during both training and evaluation. In binary classification, pivotal metrics, including accuracy, precision, recall, and loss, are essential tools for assessing how effectively a model distinguishes between classes. The importance of these metrics is underscored by their multifaceted roles. Firstly, they enable a thorough assessment of a model's task-specific performance, quantifying its effectiveness. Secondly, metrics, with a focus on the loss function, guide training by unveiling disparities between predicted and actual outputs. Lastly, metrics play a significant role in decision-making processes, aiding in informed choices regarding model selection, hyperparameter tuning, and overarching improvements to enhance overall model performance. The metrics shown in Table 3 [96].

Metric	Description
Loss	Mathematical function measuring disparity between predicted and actual outputs; guides the training process.
Accuracy	The ratio of correct predictions to total instances gives overall performance. Precision & Proportion of true positives among all positives; important when false positives are costly.
Precision	The proportion of true positives among all positives; important when false positives are costly.
Recall	The proportion of true positive predictions among all actual positive instances is important for tasks where missing positives is costly.

Table 3. Performance Metrics Overview for Binary Classification

In our evaluation step, Loss, accuracy, recall, and precision are used to have a complete overview of the performance of our models.

3.6.Hyperparameters

In machine learning, hyperparameters are critical configurations that control the training process of models. These parameters, which are not learned during training, must be pre-set and significantly influence model performance. Common examples of hyperparameters include the number of epochs, batch size, and the sizes of the training, validation, and testing datasets. We chose specific hyperparameter optimization techniques to enhance model performance by systematically searching and evaluating various hyperparameter combinations, which is crucial

for achieving optimal results in machine learning models. In addition, the tuning and finetuning of these hyperparameters were carried out on the training and validation portion of the dataset (85% of the original dataset, i.e., 1,190 images), where 17.6% (210 images, corresponding to 15% of the entire dataset) is used for validation and 82.4% (980 images, corresponding to 70% of the entire dataset) is used for training. Furthermore, we employed a Stratified 10-Fold approach to ensure that each fold is representative of the overall class distribution, thus further enhancing the reliability of our hyperparameter selection process. The hyperparameters we used in our study are shown in Table 4.

Parameter	Description	Value
Epochs	Number of full passes through the training data	7
Batch Size	Number of samples processed before the model is updated	32
Train Size	Percentage of data used for training at each fold	70%
Validation Size	Percentage of data used for validation at each fold	15%
Test Size	Percentage of data used for testing	15%
Number of Folds (k)	Number of folds used in Stratified K-Fold (Applied on the separated training and validation portion (85% of the original dataset, i.e., 1,190 images). Within this subset, 17.6% (210 images, corresponding to 15% of the entire dataset) is used for validation, and 82.4% (980 images, corresponding to 70% of the entire dataset) is used for training.)	10
Weight Decay	Regularization term to prevent overfitting	0.01
Number of Trials for Optuna	Number of trials to perform in the hyperparameter optimization	10

Table 4. Essential Hyperparameters for Model Optimization

For tuning hyperparameters such as dropout rate, learning rate, and momentum, we utilized Optuna, a hyperparameter optimization framework [97]. Optuna optimizes hyperparameters by searching for the parameter space using automated trial-and-error. We selected Optuna because it uses advanced techniques like Bayesian optimization to efficiently explore the hyperparameter space, making it more effective than traditional grid or random search [97]. Optuna's approach involves defining a search space and then evaluating the model performance for each combination of hyperparameters iteratively. In this study, we focused on maximizing recall because it is critical for ensuring that defective fasteners are detected, thus minimizing the risk of undetected faults. Table 5 highlights the importance of maximizing recall in safety-critical applications like railway maintenance.

Focus Area	Description
Safety and Reliability	Ensures identification of all potential defects, reducing risks and preventing accidents in critical systems.

Minimizing False Negatives	Reduces undetected defects, prioritizing detection of all defective fasteners over false positives.
Critical Applications	Emphasizes the high cost of missing defects compared to minor inconvenience caused by false positives.
Improving Inspection	Ensures thorough inspection processes, maintaining the integrity and safety of railway infrastructure.

Table 5. Importance of Maximizing Recall in Defective Fastener Detection

Optuna's optimization capabilities allow us to prioritize this metric effectively, enhancing the model's ability to identify as many defective fasteners as possible [98]. The values of the hyperparameters we used in our study, which were tuned using Optuna, are shown in Table 6 for all models. [97], [99]. In all models (ViT and Deit, ResNet50, VGG19, and VGG16) dropout is applied before the final classification layer. It helps prevent overfitting during training by randomly dropping out activations, enhancing the model's generalization ability. Table 6 shows the learning rate, momentum, and dropout values used for training different models, including ViT, DeiT, VGG19, VGG16, and ResNet50. These hyperparameters highlight the configurations applied to optimize the models' performance during training.

Model	Learning Rate	Momentum	Drop Out
vit_base_patch16_224	0.0020909	0.7720241	0.3515297
deit_base_patch16_224	0.0026888	0.5789481	0.4131130
VGG19	0.0002764	0.7956331	0.2895775
VGG16	0.0134802	0.7690707	0.2988300
Resnet50	0.0071212	0.9045430	0.2345131

Table 6. Optuna Tuned Hyperparameters for Various Models

In our study, we utilized optimizer and scheduler for optimizing model performance. The Stochastic Gradient Descent (SGD) optimizer updates neural network weights to minimize the loss function, with key hyperparameters of learning rate and momentum. The ReduceLROnPlateau scheduler adjusts the learning rate when the validation loss plateaus, helping to fine-tune the model and avoid local minima. These components, integrated into a Stratified K-Fold cross-validation framework, ensure adaptive learning rate adjustments for each fold, enhancing model robustness and generalization [100].

3.7.Receiver Operating Characteristic (ROC) Curve

The Receiver Operating Characteristic (ROC) curve is a crucial tool used to evaluate the performance of binary classification models. It illustrates the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR) at various threshold settings, providing a comprehensive view of the model's ability to discriminate between classes [101].

$$TPR = True\ Positives / (True\ Positives + False\ Negatives)$$

$$FPR = False\ Positives / (False\ Positives + True\ Negatives)$$

The Area Under the ROC Curve (AUC) is a single scalar value that summarizes the overall performance of the model. A higher AUC indicates better model performance, with values ranging from 0.5 (no discrimination) to 1.0 (perfect discrimination) [102]. We used the roc_curve function from the sklearn—metrics package to generate the ROC curve and calculate the AUC for our model's predictions. The function computes the TPR and FPR for various threshold values, providing a detailed view of the model's performance [103].

3.8.Methodology and Experimental Setup

Based on Table 7, the study used the Railway Track Fault Detection dataset from Kaggle, splitting it into training, validation, and test sets with stratified splitting. Data transformations included resizing, cropping, and normalization. The models DeiT, ViT, VGG19, VGG16, and ResNet50, all pre-trained on ImageNet, were fine-tuned by replacing the final classification layer with two output neurons. Optuna was used to optimize hyperparameters such as learning rate, momentum, and dropout rate over 10 trials to maximize recall. Stratified KFold cross-validation with 10 folds was employed for training and evaluation, using metrics like accuracy, precision, recall, F1 score, and ROC AUC. Table 7 summarizes the study's methodology, including data preparation, model architectures (DeiT, ViT, VGG19, VGG16, ResNet50), fine-tuning with pre-trained weights, hyperparameter optimization using Optuna, and evaluation metrics such as accuracy, recall, and ROC AUC. It also outlines the use of stratified cross-validation and training details. Table 7 summarizes the study's methodology, including data preparation, model architectures (DeiT, ViT, VGG19, VGG16, ResNet50), fine-tuning with pre-trained weights, hyperparameter optimization using Optuna, and evaluation metrics such as accuracy, recall, and ROC AUC. It also outlines the use of stratified cross-validation and training details.

Methodological Component	Details
Data Preparation	Dataset: Railway Track Fault Detection from Kaggle Classes: Defective, non-defective Transformations: Training: RandomResizedCrop (224), RandomHorizontalFlip, Normalize ([0.485, 0.456, 0.406], [0.229, 0.224, 0.225]) Validation/Test: Resize (256), CenterCrop (224), Normalize ([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
Data Splitting	Training: 70%, Validation: 15%, Test: 15%, Stratified split to maintain class proportions
Model Architectures	DeiT_base_patch16_224, Vit_base_patch16_224, VGG19, VGG16, and ResNet50, all models are pre-trained on ImageNet.
Fine-Tuning Process	Replace final classification layer with two output neurons \newline Random initialization of new layer \newline Fine-tuning of pre-trained layers with a lower learning rate.

Hyperparameter Optimization	Tool: Optuna, Parameters optimized: Learning rate, momentum, dropout rate, Objective: Maximize recall on validation set, Number of trials: 10
Training and Evaluation	Cross-validation: Stratified KFold, 10 folds Epochs: 7 Optimizer: SGD with learning rate scheduler Metrics: Accuracy, Precision, Recall, F1 Score, ROC AUC
Results	Average metrics across all folds: Loss, Accuracy, Precision, Recall ROC curves plotted for performance visualization

Table 7. Summary of Methodology

4. Results

In this section, we present the outcomes of our experiments and analyses in detail. First, we provide the ROC AUC values and corresponding ROC curves (Table 8 and Figure 1) to illustrate each model's ability to distinguish between defective and non-defective classes. We then showcase the progression of core performance metrics—accuracy, precision, recall, and loss—across multiple training epochs and folds (Figure 2, Table 9). Next, , we offer insights into how different hyperparameter configurations impact recall (Table 10), underscoring the importance of careful tuning to optimize model performance. Finally, we highlight the final test results for the five models (Table 11), demonstrating their comparative performance on test dataset.

4.1. **ROC Curves:** Table 8 shows the ROC AUC values for different models during training and validation, highlighting their performance in distinguishing between classes.

Model	Train ROC AUC	Val ROC AUC
ViT	0.99	0.97
DeiT	0.99	0.98
VGG16	0.96	0.95
VGG19	0.95	0.95
ResNet50	0.91	0.89

Table 8. Train and validation ROC AUC values for different models

Figure 1 shows the ROC curves for training and validation for five models: (a) ViT, (b) DeiT, (c) VGG19, (d) VGG16, and (e) ResNet50, illustrating their class discrimination performance with corresponding ROC AUC scores for both training and validation datasets.

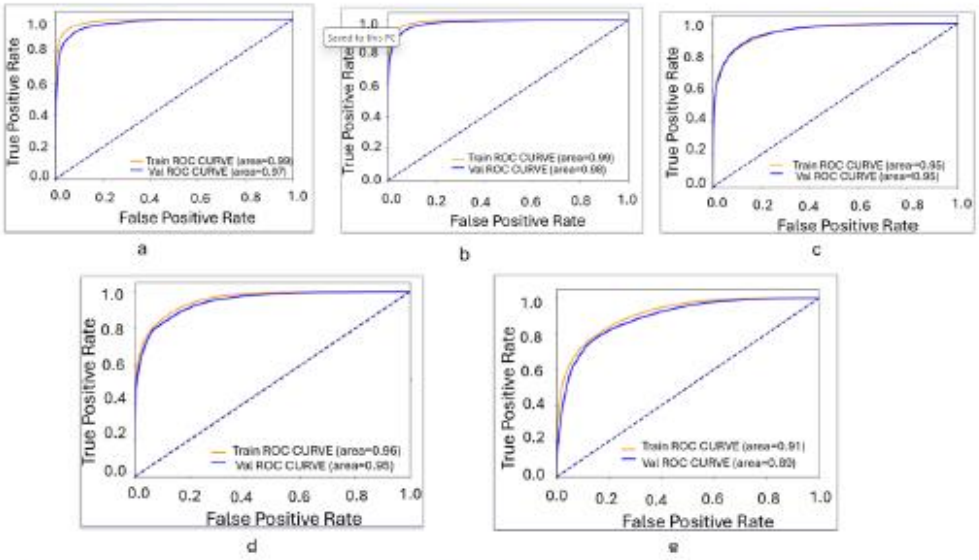

Figure 1. ROC curves for different models. (a) ROC for ViT, (b) ROC for DeiT, (c) ROC for VGG19, (d) ROC for VGG16, (e) ROC for ResNet50

Figure 2 illustrates the average training and validation trends over 10 folds for five models—ViT, DeiT, VGG16, VGG19, and ResNet50—over 7 epochs. The subplots show: (a) Accuracy, (b) Precision, (c) Recall, and (d) Loss. These plots highlight the progression of each metric, averaged across folds, demonstrating the models' performance improvements and convergence over epochs.

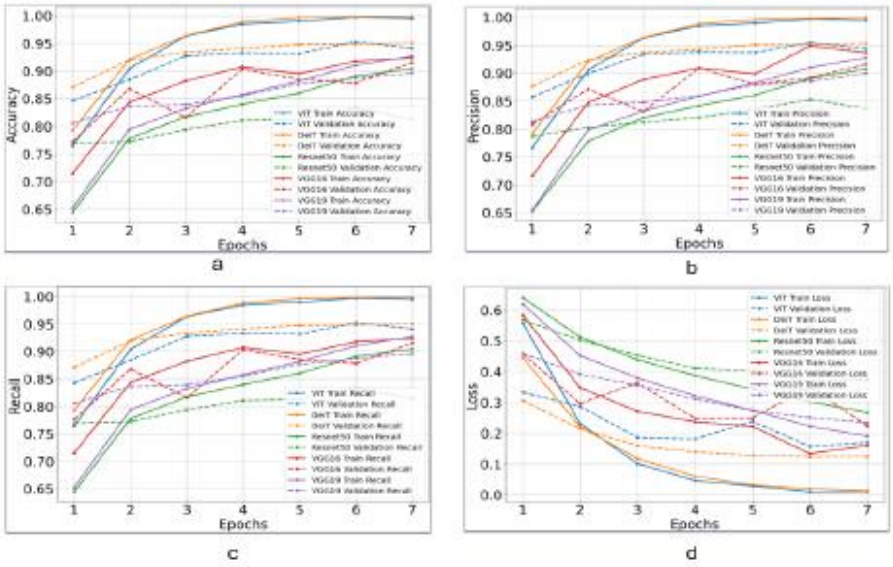

Figure 2. Average metrics for 10 folds: Performance metrics for all models over 10 folds. (a) Average Accuracy, (b) Average Precision, (c) Average Recall, (d) Average Loss

Table 9 shows the average performance metrics—accuracy, precision, recall, and loss—for five models (ViT, DeiT, VGG16, VGG19, and ResNet50) during training and validation across the 1st and 7th epochs, averaged over 10 folds.

Model	Accuracy (Avg: 1st | 7th (Epoch))	Precision (Avg: 1st | 7th (Epoch))	Recall (Avg: 1st | 7th (Epoch))	Loss (Avg: 1st | 7th (Epoch)
ViT (Train)	76.47% |99.04%	76.67% | 98.46%	76.47% | 98.86%	0.5576 | 0.0101
ViT(Val)	84.64% | 94.14%	85.76% | 94.36%	84.26% | 94.14%	0.3324 | 0.1695
DeiT (Train)	79.28% | 99.27%	79.17% | 98.62%	79.28% | 99.17%	0.4456 | 0.0122
DeiT(Val)	87.09% | 95.04%	87.68% | 95.37%	87.09% | 95.4%	0.3061 | 0.1241
VGG16 (Train)	71.48% | 92.40%	71.60% | 93.68%	71.48% | 92.40%	0.5835 | 0.1597
VGG16(Val)	77.18% | 91.54%	80.84% | 91.61%	77.46% | 91.54%	0.4580 | 0.2233
VGG19 (Train)	65.23% | 92.70%	65.44% | 92.78%	65.23% | 92.70%	0.6191 | 0.1899
VGG19(Val)	80.56% | 89.60%	81.34% | 90.13%	80.56% | 89.60%	0.4590 | 0.2338
ResNet50 (Train)	64.46%| 90.43%	65.28% | 90.78%	64.46% | 90.43%	0.6403 | 0.2669
ResNet50 (Val)	76.92% | 81.35%	78.66% | 83.65%	76.92% | 81.35%	0.5680 | 0.3751

Table 9. Average performance metrics (accuracy, precision, recall, and loss) over 10 folds for various models across the 1st and 7th epochs during training and validation.

Table10 Table 9presents sample results showing the impact of different hyperparameters on recall values for various models. It includes configurations for learning rate, momentum, and dropout rate, along with the corresponding recall values. The models compared are ViT, DeIT, VGG19, VGG16, and ResNet50. Each row represents a specific set of hyperparameters, highlighting the variability in model performance based on these settings.

Model	Learning Rate	Momentum	Drop Out	Recall
vit_base_patch16_224	0.00209	0.7720	0.3515	0.9593
	0.00498	0.7034	0.39620	0.9157
	0.00168	0.7725	0.44753	0.8847
deit_base_patch16_224	0.00268	0.57894	0.41312	0.9654
	0.00060	0.74018	0.12964	0.9027
	0.00010	0.73390	0.17594	0.8765
VGG19	0.00027	0.79563	0.28957	0.9056
	0.00130	0.59047	0.46899	0.8879
	0.00588	0.69252	0.36573	0.8348
VGG16	0.01348	0.76907	0.29883	0.8995
	0.00015	0.64065	0.33684	0.8871
	5.44e-05	0.59010	0..35461	0.7985
Resnet50	0.00712	0.90454	0.23451	0.8179
	0.00621	0.90351	0.20410	0.7954
	0.00539	0.85797	0.19745	0.7841

Table 10. Performance of various models with different hyperparameter configurations, showing learning rate, momentum, dropout, and corresponding recall values

Table 11 demonstrates the comparative performance of five different architectures—Vision Transformer (ViT), Data-Efficient Image Transformer (DeiT), VGG16, VGG19, and ResNet50—when tested on a set of 210 images. The metrics reported include accuracy, precision, recall, and loss.

Model	Accuracy	Precision	Recall	Loss
ViT	0.98095	0.981123	0.980952	0.13507
DeiT	0.95714	0.959338	0.957142	0.05007
VGG16	0.947619	0.947717	0.9476190	0.18558
VGG19	0.938095	0.938403	0.938095	0.11087
ResNet50	0.847619	0.854149	0.8476190	0.39728

Table 11. Summary of each model's performance on the test set of 210 images, including loss, accuracy, precision, recall

5. Discussion

In this study, the performance of five models—ViT, DeiT, VGG16, VGG19, and ResNet50—was evaluated based on key metrics, including ROC AUC, accuracy, precision, recall, and loss, with results summarized in Table 8, Table 9, Figure 1 and Figure 2. The analysis provides a detailed comparison of the models' learning capabilities and generalization performance over 10-fold cross-validation, highlighting the impact of architecture and hyperparameter tuning. Table 8 and Figure 1 presents the train and validation ROC AUC values, reflecting the models' ability to distinguish between classes. The transformer-based models, ViT and DeiT, achieved the highest ROC AUC values, with training scores of 0.99 and validation scores of 0.97 and 0.98, respectively. These results demonstrate their excellent discriminative ability and robust performance. In comparison, the convolutional models, VGG16 and VGG19, achieved slightly lower validation ROC AUC scores of 0.95, indicating effective but less flexible classification capabilities. ResNet50 had the lowest scores, with training and validation ROC AUC values of 0.91 and 0.89, respectively, suggesting limitations in its ability to capture complex patterns in the data. Table 9 and Figure 2 provides the averaged performance metrics, including accuracy, precision, recall, and loss, across the 1st and 7th epochs during training and validation. The transformer models, ViT and DeiT, demonstrated the best performance, achieving over 99% validation accuracy and recall by the 7th epoch. DeiT slightly outperformed ViT, with a final validation recall of 98.62% compared to 98.46% for ViT. These models also showed the lowest validation loss values, with DeiT reaching 0.1241 and ViT 0.1695, reflecting efficient learning and strong convergence.

The VGG models, VGG16 and VGG19, exhibited solid performance but lagged behind the transformer models. VGG16 achieved a validation accuracy of 92.40% and a recall of 93.68%, while VGG19 achieved a validation accuracy of 92.70% and a recall of 92.78%. Both models showed steady improvement across epochs, but their higher loss values (0.2233 for VGG16 and 0.2338 for VGG19) suggest that they were less efficient in learning compared to ViT and DeiT. ResNet50 showed the weakest performance, with a validation accuracy of 90.43% and a recall of 90.78% by the 7th epoch, alongside the highest loss value of 0.3751. These results indicate

slower convergence and less robust learning compared to the other models. The importance of hyperparameter tuning is evident from the results.

Using Optuna, we conducted trials to identify the best hyperparameter configurations for each model. The best-performing configurations are presented in the first row of Table 10 for each model, while two additional rows with suboptimal configurations are included to highlight the impact of tuning. For example, both ViT and DeiT showed significant improvements in accuracy, precision, and recall when optimally tuned, whereas suboptimal configurations led to markedly lower performance. This underscores the necessity of careful hyperparameter tuning to achieve optimal results. Additionally, each model was evaluated on a separate test set of 210 images, with the results summarized in Table 11. ViT delivered the highest performance, achieving an accuracy of 98.09%, closely followed by DeiT with 95.71%. The VGG models exhibited moderate accuracy (94.76% for VGG16 and 93.81% for VGG19), while ResNet50 trailed at 84.76%. These test results further validate the superiority of transformer-based approaches (ViT and DeiT) over traditional convolutional architectures for this specific classification task.

In conclusion, transformer-based models, particularly DeiT, demonstrated superior performance across all metrics, establishing them as the most effective choice for this classification task. While VGG models performed well, they were surpassed by the transformers in both learning efficiency and final outcomes. ResNet50 exhibited the lowest performance, suggesting that its architecture may not be as well-suited for this dataset or application. The study further emphasizes the critical importance of hyperparameter tuning, as it significantly impacts model performance, reinforcing its necessity in the development of high-performing classification systems. The exceptional performance of Vision Transformers (ViT and DeiT) in fastener defect detection can be attributed to their self-attention mechanisms. Unlike traditional convolutional models that focus on localized features, transformers can capture global context across an image. This capability is crucial for detecting defects that may be subtle, distributed, or not easily identifiable through localized features alone. By attending to multiple regions of the input simultaneously, Vision Transformers create comprehensive representations of the fasteners, resulting in superior detection outcomes. The self-attention mechanism in Vision Transformers also allows the models to focus on critical features across the entire image, regardless of their spatial location. This is particularly beneficial for railway fastener detection, where defects can vary widely in appearance and position. By effectively capturing these diverse features, Vision Transformers enhance both accuracy and reliability in defect detection tasks. Furthermore, the ability of Vision Transformers to extract global features contributes to their remarkable generalization capabilities. These models can transfer learned knowledge to new and unseen data more effectively than models relying on localized features. This makes them highly advantageous for tasks like fastener detection, where robust performance on diverse and unpredictable data is essential. The combination of global feature extraction, reliable detection accuracy, and superior generalization underscores the value of Vision Transformers in safety-critical applications like railway maintenance.

6. Conclusion

This study compared the performance of five machine learning models—ViT, DeiT, VGG16, VGG19, and ResNet50—for the critical task of fastener defect detection in railway maintenance. The results demonstrated the superior performance of transformer-based models, particularly DeiT, which consistently outperformed other models across all evaluated metrics, including accuracy, precision, recall, and ROC AUC. The robust performance of DeiT and ViT can be attributed to their self-attention mechanisms, enabling them to capture global context and effectively identify defects that may not be easily detectable through localized features alone. The convolutional models, VGG16 and VGG19, performed competitively but were limited in their ability to match the global feature extraction capabilities of the transformers. ResNet50, while demonstrating solid baseline performance, lagged behind the other models, suggesting that its architecture may not be optimally suited for this application. The study further highlighted the critical role of hyperparameter tuning, with models demonstrating significant variations in performance depending on the tuning configurations. The use of Optuna for hyperparameter optimization proved invaluable in identifying the best-performing configurations, reinforcing the importance of systematic tuning in achieving optimal results. In addition to performance metrics, the global feature extraction capability of Vision Transformers provides a significant advantage for applications requiring high generalization and reliability. Their ability to transfer learned knowledge to unseen data ensures robust performance, making them ideal for safety-critical tasks like railway defect detection. The findings of this study underline the potential of transformer-based architectures to enhance defect detection tasks, where accuracy, reliability, and generalization are paramount. Additionally, the evaluation on a separate test set of 210 images (see Table 11) further validated the superior performance of transformer-based models, with ViT achieving the highest accuracy of 98.09%.

Our study has some limitations. The dataset used for training and validation was balanced, but the real-world distribution of fastener defects may differ, potentially affecting model performance. Additionally, while Transformer models showed higher accuracy and robustness, their computational requirements are higher than those of CNNs, which may limit their deployment in resource-constrained environments.

Future work will focus on addressing these limitations by incorporating more diverse and extensive datasets to improve model generalization. Additionally, we plan to explore hybrid models that combine the strengths of CNNs and Transformers to achieve better performance with optimized computational efficiency. Further hyperparameter tuning and experimentation with different architectures will also be conducted to enhance model performance and applicability in real-world railway maintenance systems.

Acknowledgments

The authors acknowledge the support of Natural Sciences and Engineering Research Council of Canada (NSERC).

Declarations

- Funding: The authors declare that the research conducted in this project was supported by Sciences and Engineering Research Council of Canada (NSERC) grants with reference numbers RGPIN-2018-06351, RGPIN-2023-05578, and DGECR-2023-00336.
- Conflict of interests: The authors declare that there is no conflict of interest.
- Ethics approval: Not applicable
- Consent for publication: The authors declare that they consented to publish this paper and agreed with the publication.

Availability of data and materials: All data, materials, and codes used in this paper are available.

The authors declare that the data used in this paper is publicly available and can be accessed through the following link: https://www.kaggle.com/datasets/ashikadnan/railway-track-fault-detection-dataset2fastener. All necessary materials and datasets required to reproduce the findings of this study are accessible at the provided link.

Authors' contributions: SSK, MA, and HI conceived the idea; SM, MA, and SSK developed the theory; SM and MA conducted computations. SSK and MA verified analytical methods and with HI supervised the findings. SM explored various methods and parameters. All authors actively participated in result discussions and manuscript development, ultimately reviewing and approving the final version.

References

[1] J. C. Gaskell *et al.*, "Breaking down the barriers to regional agricultural trade in Central Africa," 2018. [Online]. Available: http://documents.worldbank.org/curated/en/233071535650013216/Breaking-down-the-barriers-to-regional-agricultural-trade-in-Central-Africa

[2] R. Anand, R. Perrelli, and B. Zhang, "South Africa's Exports Performance: Any Role for Structural Factors?," *IMF Working Papers*, vol. 16, no. 24, p. 1, 2016, doi: 10.5089/9781475594003.001.

[3] *Breaking Down the Barriers to Regional Agricultural Trade in Central AfricaBriser les Obstacles au Commerce Agricole Regional en Afrique Centrale*. World Bank, Washington, DC, 2018. doi: 10.1596/30397.

[4] "Integrated Intervention Tool : Integration Strategies for Urban Poor Areas and Disadvantaged Communities," Washington, DC, Jan. 2013. Accessed: Oct. 15, 2023. [Online]. Available: http://hdl.handle.net/10986/24492

[5] F. Peng, S. Kang, X. Li, Y. Ouyang, K. Somani, and D. Acharya, "A heuristic approach to the railroad track maintenance scheduling problem," *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 2, pp. 129–145, 2011.

[6] A. Lasisi and N. Attoh-Okine, "Principal components analysis and track quality index: A machine learning approach," *Transp Res Part C Emerg Technol*, vol. 91, pp. 230–248, 2018.

[7] I. Durazo-Cardenas *et al.*, "An autonomous system for maintenance scheduling data-rich complex infrastructure: Fusing the railways' condition, planning and cost," *Transp Res Part C Emerg Technol*, vol. 89, pp. 234–253, 2018.

[8] A. Zoeteman, R. Dollevoet, and Z. Li, "Dutch research results on wheel/rail interface management: 2001–2013 and beyond," *Proc Inst Mech Eng F J Rail Rapid Transit*, vol. 228, no. 6, pp. 642–651, 2014.

[9] L. Huang, C. Wu, B. Wang, and Q. Ouyang, "Big-data-driven safety decision-making: A conceptual framework and its influencing factors," *Saf Sci*, vol. 109, pp. 46–56, Nov. 2018, doi: 10.1016/j.ssci.2018.05.012.

[10] O. Alshorman *et al.*, "A Review of Artificial Intelligence Methods for Condition Monitoring and Fault Diagnosis of Rolling Element Bearings for Induction Motor," 2020, doi: 10.1155/2020/8843759.

[11] K. Schwab, *The Fourth Industrial Revolution*. New York: Currency, 2017.

[12] J. Villalba-Diez, D. Schmidt, R. Gevers, J. Ordieres-Meré, M. Buchwitz, and W. Wellbrock, "Deep learning for industrial computer vision quality control in the printing industry 4.0," *Sensors*, vol. 19, no. 18, p. 3987, 2019.

[13] C. Chao, L. Liang, S. U. N. Yu, Z. Wan-ming, W. Kai-yun, and W. Gui-dong, "Dynamics performance of new type of fully automatic track inspection vehicle," *Journal of Traffic and Transportation Engineering*, vol. 21, no. 6, pp. 194–208, 2021.

[14] P. Trampus, V. Krstelj, and G. Nardoni, "NDT integrity engineering–A new discipline," *Procedia Structural Integrity*, vol. 17, pp. 262–267, 2019.

[15] H. Taheri, M. Gonzalez Bocanegra, and M. Taheri, "Artificial intelligence, machine learning and smart technologies for nondestructive evaluation," *Sensors*, vol. 22, no. 11, p. 4055, 2022.

[16] S. Agnisarman, S. Lopes, K. C. Madathil, K. Piratla, and A. Gramopadhye, "A survey of automation-enabled human-in-the-loop systems for infrastructure visual inspection," *Autom Constr*, vol. 97, pp. 52–76, 2019.

[17] F. Outay, H. A. Mengash, and M. Adnan, "Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges," *Transp Res Part A Policy Pract*, vol. 141, pp. 116–129, 2020.

[18] F. Fassi, L. Fregonese, S. Ackermann, and V. De Troia, "Comparison between laser scanning and automated 3d modelling techniques to reconstruct complex and extensive cultural heritage areas," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, pp. 73–80, 2013.

[19] N. Md Nor, C. R. Che Hassan, and M. A. Hussain, "A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems," *Reviews in Chemical Engineering*, vol. 36, no. 4, pp. 513–553, 2020.

[20] F. Provost and T. Fawcett, *Data Science for Business: What you need to know about data mining and data-analytic thinking*. Berlin/New York: " O'Reilly Media, Inc.," 2013.

[21] X. Xie, "A Review of Recent Advances in Surface Defect Detection using Texture analysis Techniques," *ELCVIA: electronic letters on computer vision and image analysis*, pp. 1–22, 2008, Accessed: Dec. 11, 2024. [Online]. Available: https://raco.cat/index.php/ELCVIA/article/view/150223

[22] A. A. Hamdi, M. S. Sayed, M. M. Fouad, and M. M. Hadhoud, "Unsupervised patterned fabric defect detection using texture filtering and K-means clustering," *Proceedings of 2018 International Conference on Innovative Trends in Computer Engineering, ITCE 2018*, vol. 2018-March, pp. 130–144, Mar. 2018, doi: 10.1109/ITCE.2018.8316611.

[23] A. J. Chittilappilly and K. Subramaniam, "SVM based defect detection for industrial applications," *2017 4th International Conference on Advanced Computing and Communication Systems, ICACCS 2017*, Aug. 2017, doi: 10.1109/ICACCS.2017.8014696.

[24] X. Dong, C. J. Taylor, and T. F. Cootes, "A Random Forest-Based Automatic Inspection System for Aerospace Welds in X-Ray Images," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 4, pp. 2128–2141, Oct. 2021, doi: 10.1109/TASE.2020.3039115.

[25] "IEEE Xplore Full-Text PDF:" Accessed: Dec. 11, 2024. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8758813

[26] V. H. Pham and B. R. Lee, "An image segmentation approach for fruit defect detection using k-means clustering and graph-based algorithm," *Vietnam Journal of Computer Science 2014 2:1*, vol. 2, no. 1, pp. 25–33, Aug. 2014, doi: 10.1007/S40595-014-0028-3.

[27] A. Mujeeb, W. Dai, M. Erdt, and A. Sourin, "Unsupervised surface defect detection using deep autoencoders and data augmentation," *Proceedings - 2018 International Conference on Cyberworlds, CW 2018*, pp. 391–398, Dec. 2018, doi: 10.1109/CW.2018.00076.

[28] L. Jiao and J. Zhao, "A Survey on the New Generation of Deep Learning in Image Processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019, doi: 10.1109/ACCESS.2019.2956508.

[29] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognit Lett*, vol. 141, pp. 61–67, Jan. 2021, doi: 10.1016/J.PATREC.2020.07.042.

[30] J. Zhu, C. Zhang, H. Qi, and Z. Lu, "Vision-based defects detection for bridges using transfer learning and convolutional neural networks," *Structure and Infrastructure Engineering*, vol. 16, no. 7, pp. 1037–1049, Jul. 2020, doi: 10.1080/15732479.2019.1680709.

[31] M. Ferguson, R. Ak, Y. T. T. Lee, and K. H. Law, "Detection and Segmentation of Manufacturing Defects with Convolutional Neural Networks and Transfer Learning," *Smart Sustain Manuf Syst*, vol. 2, no. 1, pp. 137–164, Feb. 2018, doi: 10.1520/SSMS20180033.

[32] A. Jamshidi *et al.*, "A Big Data Analysis Approach for Rail Failure Risk Assessment," *Risk Analysis*, vol. 37, no. 8, pp. 1495–1507, Mar. 2017, doi: 10.1111/risa.12836.

[33] A. Shams, D. Becker, K. Becker, S. Amirian, and K. Rasheed, "Evolving Efficient CNN Based Model for Image Classification," *Proceedings - 2023 Congress in Computer Science, Computer Engineering, and Applied Computing, CSCE 2023*, pp. 228–235, 2023, doi: 10.1109/CSCE60160.2023.00041.

[34] A. Shams, K. Becker, D. Becker, S. Amirian, and K. Rasheed, "Evolutionary CNN-based architectures with attention mechanisms for enhanced image classification," *Artif Intell*, pp. 107–132, Jul. 2024, doi: 10.1515/9783111344126-006/PDF.

[35] S. Faghih-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, and B. De Schutter, "Deep convolutional neural networks for detection of rail surface defects," in *2016 International Joint Conference on Neural Networks (IJCNN)*, Berlin/New York: IEEE, Mar. 2016, pp. 2584–2589. doi: 10.1109/IJCNN.2016.7727522.

[36] S. Wang, P. Dai, X. Du, Z. Gu, and Y. Ma, "Rail fastener automatic recognition method in complex background," in *Tenth International Conference on Digital Image Processing (ICDIP 2018)*, Berlin/New York: SPIE, Mar. 2018, p. 314. doi: 10.1117/12.2503323.

[37] X. Gibert, V. M. Patel, and R. Chellappa, "Deep Multitask Learning for Railway Track Inspection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 1, pp. 153–164, Mar. 2017, doi: 10.1109/TITS.2016.2568758.

[38] S. Ritika and D. Rao, "Data augmentation of railway images for track inspection," *arXiv preprint arXiv:1802.01286*, 2018.

[39] D. Soukup and R. Huber-Mörk, "Convolutional Neural Networks for Steel Surface Defect Detection from Photometric Stereo Images," in *International Symposium on Visual Computing*, 2014, pp. 668–677. doi: 10.1007/978-3-319-14249-4_64.

[40] P. Chandran, J. Asber, F. Thiery, J. Odelius, and M. Rantatalo, "An investigation of railway fastener detection using image processing and augmented deep learning," *Sustainability*, vol. 13, no. 21, p. 12051, Mar. 2021, doi: 10.3390/su132112051.

[41] T. Bai, J. Gao, J. Yang, and D. Yao, "A study on railway surface defects detection based on machine vision," *Entropy*, vol. 23, no. 11, p. 1437, 2021.

[42] D. Zheng *et al.*, "A defect detection method for rail surface and fasteners based on deep convolutional neural network," *Comput Intell Neurosci*, vol. 2021, pp. 1–15, 2021.

[43] J. Sresakoolchai and S. Kaewunruen, "Railway defect detection based on track geometry using supervised and unsupervised machine learning," *Struct Health Monit*, vol. 21, no. 4, pp. 1757–1767, 2022.

[44] X. Xu, Y. Lei, F. Yang, and others, "Railway subgrade defect automatic recognition method based on improved faster R-CNN," *Sci Program*, vol. 2018, 2018.

[45] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, and Y. Li, "Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study," *Eng Appl Artif Intell*, vol. 80, pp. 66–81, 2019.

[46] J. Wang, L. Luo, W. Ye, and S. Zhu, "A defect-detection method of split pins in the catenary fastening devices of high-speed railway based on deep learning," *IEEE Trans Instrum Meas*, vol. 69, no. 12, pp. 9517–9525, 2020.

[47] Y. Wu, Y. Qin, Y. Qian, and F. Guo, "Automatic detection of arbitrarily oriented fastener defect in high-speed railway," *Autom Constr*, vol. 131, p. 103913, 2021.

[48] A. Lu, "Detecting Defective Rail Joints on the Swiss Railways with Inception ResNet V2: Simplifying Predictive Maintenance of Railway Infrastructure," 2022, *KTH Royal Institute of Technology*.

[49] H. Li, F. Wang, J. Liu, H. Song, Z. Hou, and P. Dai, "Ensemble model for rail surface defects detection," *PLoS One*, vol. 17, no. 5, p. e0268518, 2022.

[50] Z. Jian, S. He, S. Liu, J. Liu, and Y. Fang, "A multiple species railway defects detection method based on sample generation," *IEEE Trans Instrum Meas*, 2024.

[51] V. Engeln, "Defect detection in metal objects using a pre-trained Vision Transformer model.," Tilburg University, 2022.

[52] R. Fan *et al.*, "Detecting glaucoma from fundus photographs using deep learning without convolutions: Transformer for improved generalization," *Ophthalmology science*, vol. 3, no. 1, p. 100233, 2023.

[53] N. Hütten, R. Meyes, and T. Meisen, "Vision Transformer in Industrial Visual Inspection," *Applied Sciences*, vol. 12, no. 23, p. 11981, 2022.

[54] C. T. Alexakos, Y. L. Karnavas, M. Drakaki, and I. A. Tziafettas, "A combined short time fourier transform and image classification transformer model for rolling element bearings fault diagnosis in electric motors," *Mach Learn Knowl Extr*, vol. 3, no. 1, pp. 228–242, 2021.

[55] L. M. Dang, H. Wang, Y. Li, T. N. Nguyen, and H. Moon, "DefectTR: End-to-end defect detection for sewage networks using a transformer," *Constr Build Mater*, vol. 325, p. 126584, 2022.

[56] B. Tang, Z.-K. Song, W. Sun, and X.-D. Wang, "An end-to-end steel surface defect detection approach via Swin transformer," *IET Image Process*, vol. 17, no. 5, pp. 1334–1345, 2023.

[57] Y. Li, Y. Xiang, H. Guo, P. Liu, and C. Liu, "Swin Transformer Combined with Convolution Neural Network for Surface Defect Detection," *Machines*, vol. 10, no. 11, p. 1083, 2022.

[58] S. Lu, J. Wang, G. Jing, W. Qiang, and M. M. Rad, "Rail Defect Classification with Deep Learning Method," *Acta Polytechnica Hungarica*, vol. 19, no. 6, pp. 2022–225.

[59] H. Wan *et al.*, "A novel transformer model for surface damage detection and cognition of concrete bridges," *Expert Syst Appl*, vol. 213, p. 119019, Mar. 2023, doi: 10.1016/J.ESWA.2022.119019.

[60] E. Asadi Shamsabadi, C. Xu, A. S. Rao, T. Nguyen, T. Ngo, and D. Dias-da-Costa, "Vision transformer-based autonomous crack detection on asphalt and concrete surfaces," *Autom Constr*, vol. 140, p. 104316, Aug. 2022, doi: 10.1016/J.AUTCON.2022.104316.

[61] H. Shang, C. Sun, J. Liu, X. Chen, and R. Yan, "Defect-aware transformer network for intelligent visual surface defect detection," *Advanced Engineering Informatics*, vol. 55, p. 101882, Jan. 2023, doi: 10.1016/J.AEI.2023.101882.

[62] G. Yang *et al.*, "Datasets and processing methods for boosting visual inspection of civil infrastructure: A comprehensive review and algorithm comparison for crack classification, segmentation, and detection," *Constr Build Mater*, vol. 356, p. 129226, Nov. 2022, doi: 10.1016/J.CONBUILDMAT.2022.129226.

[63] W. Wang and C. Su, "Automatic Classification of Reinforced Concrete Bridge Defects Using the Hybrid Network," *Arab J Sci Eng*, vol. 47, no. 4, pp. 5187–5197, Apr. 2022, doi: 10.1007/S13369-021-06474-X/METRICS.

[64] M. ; Chen *et al.*, "Improving the Concrete Crack Detection Process via a Hybrid Visual Transformer Algorithm," *Sensors 2024, Vol. 24, Page 3247*, vol. 24, no. 10, p. 3247, May 2024, doi: 10.3390/S24103247.

[65] M. Li, J. Yuan, Q. Ren, Q. Luo, J. Fu, and Z. Li, "CNN-Transformer hybrid network for concrete dam crack patrol inspection," *Autom Constr*, vol. 163, p. 105440, Jul. 2024, doi: 10.1016/J.AUTCON.2024.105440.

[66] T. Alaa, M. Kotb, A. Zakaria, M. Diab, and W. Gomaa, "Automated Detection of Defects on Metal Surfaces using Vision Transformers," Oct. 2024, Accessed: Nov. 05, 2024. [Online]. Available: https://arxiv.org/abs/2410.04440v1

[67] K. An and Y. Zhang, "LPViT: A Transformer Based Model for PCB Image Classification and Defect Detection," *IEEE Access*, vol. 10, pp. 42542–42553, 2022, doi: 10.1109/ACCESS.2022.3168861.

[68] C. Ding, D. Teng, X. Zheng, Q. Wang, Y. He, and Z. Long, "DHT: Dynamic Vision Transformer Using Hybrid Window Attention for Industrial Defect Images Classification," *IEEE Instrum Meas Mag*, vol. 26, no. 2, pp. 19–28, Apr. 2023, doi: 10.1109/MIM.2023.10083000.

[69] F. Guo, J. Liu, Y. Qian, and Q. Xie, "Rail surface defect detection using a transformer-based network," *J Ind Inf Integr*, vol. 38, p. 100584, Mar. 2024, doi: 10.1016/J.JII.2024.100584.

[70] W. Ye, J. Ren, C. Li, W. Liu, Z. Zhang, and C. Lu, "Intelligent Detection of Surface Defects in High-Speed Railway Ballastless Track Based on Self-Attention and Transfer Learning," *Struct Control Health Monit*, vol. 2024, no. 1, p. 2967927, Jan. 2024, doi: 10.1155/2024/2967927.

[71] J. He, R. Duan, M. Dong, Y. Kao, G. Guo, and J. Liu, "CNN-Transformer Bridge Mode for Detecting Arcing Horn Defects in Railway Sectional Insulator," *IEEE Trans Instrum Meas*, vol. 73, pp. 1–16, 2024, doi: 10.1109/TIM.2024.3373084.

[72] H. Luo, L. Cai, and C. Li, "Rail Surface Defect Detection Based on An Improved YOLOv5s," *Applied Sciences 2023, Vol. 13, Page 7330*, vol. 13, no. 12, p. 7330, Jun. 2023, doi: 10.3390/APP13127330.

[73] C. Yu and X. Chen, "Railway rutting defects detection based on improved RT-DETR," *J Real Time Image Process*, vol. 21, no. 4, pp. 1–13, Aug. 2024, doi: 10.1007/S11554-024-01530-9/METRICS.

[74] S. Bai, L. Yang, and Y. Liu, "A vision-based nondestructive detection network for rail surface defects," *Neural Comput Appl*, vol. 36, no. 21, pp. 12845–12864, Jul. 2024, doi: 10.1007/S00521-024-09781-0/METRICS.

[75] L. E. Mansuri and D. A. Patel, "Artificial intelligence-based automatic visual inspection system for built heritage," *Smart and Sustainable Built Environment*, vol. 11, no. 3, pp. 622–646, Mar. 2022, doi: 10.1108/SASBE-09-2020-0139.

[76] J. Dalzochio *et al.*, "Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges," *Comput Ind*, vol. 123, p. 103298, Mar. 2020, doi: 10.1016/j.compind.2020.103298.

[77] M. Seisenberger *et al.*, "Safe and secure future AI-driven railway technologies: challenges for formal methods in railway," in *International Symposium on Leveraging Applications of Formal Methods*, 2022, pp. 246–268. doi: 10.1007/978-3-031-19762-8_20.

[78] Jeremy. Howard and Sylvain. Gugger, "Deep Learning for Coders with fastai and PyTorch", Accessed: Jul. 09, 2024. [Online]. Available: https://books.google.com/books/about/Deep_Learning_for_Coders_with_fastai_and.html?hl=fr&id=yATuDwAAQBAJ

[79] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning", doi: 10.1186/s40537-019-0197-0.

[80] "A Recipe for Training Neural Networks." Accessed: Jul. 07, 2024. [Online]. Available: https://karpathy.github.io/2019/04/25/recipe/

[81] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[82] X. Feng, X. Gao, and L. Luo, "A ResNet50-Based Method for Classifying Surface Defects in Hot-Rolled Strip Steel," *Mathematics 2021, Vol. 9, Page 2359*, vol. 9, no. 19, p. 2359, Sep. 2021, doi: 10.3390/MATH9192359.

[83] S. Kumaresan, K. S. J. Aultrin, S. S. Kumar, and M. D. Anand, "Deep learning-based weld defect classification using VGG16 transfer learning adaptive fine-tuning," *International Journal on Interactive Design and Manufacturing*, vol. 17, no. 6, pp. 2999–3010, Dec. 2023, doi: 10.1007/S12008-023-01327-3/TABLES/3.

[84] X. Wan, X. Zhang, and L. Liu, "An Improved VGG19 Transfer Learning Strip Steel Surface Defect Recognition Deep Neural Network Based on Few Samples and Imbalanced Datasets," *Applied Sciences 2021, Vol. 11, Page 2606*, vol. 11, no. 6, p. 2606, Mar. 2021, doi: 10.3390/APP11062606.

[85] S. Sharma and K. Guleria, "Deep Learning Models for Image Classification: Comparison and Applications," *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2022*, pp. 1733–1738, 2022, doi: 10.1109/ICACITE53722.2022.9823516.

[86] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *ICLR 2021 - 9th International Conference on Learning Representations*, Oct. 2020, Accessed: Jul. 10, 2024. [Online]. Available: https://arxiv.org/abs/2010.11929v2

[87] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," *Proc Mach Learn Res*, vol. 139, pp. 10347–10357, Dec. 2020, Accessed: Jul. 10, 2024. [Online]. Available: https://arxiv.org/abs/2012.12877v2

[88] H. Touvron, M. Cord, and H. Jégou, "DeiT III: Revenge of the ViT," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13684 LNCS, pp. 516–533, 2022, doi: 10.1007/978-3-031-20053-3_30.

[89] J. Zhang, W. Li, P. Ogunbona, and D. Xu, "Recent advances in transfer learning for cross-dataset visual recognition: A problem-oriented perspective," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[90] S. Shao, S. McAleer, R. Yan, and P. Baldi, "Highly accurate machine fault diagnosis using deep transfer learning," *IEEE Trans Industr Inform*, vol. 15, no. 4, pp. 2446–2455, 2018.

[91] P. Kora *et al.*, "Transfer learning techniques for medical image analysis: A review," *Biocybern Biomed Eng*, vol. 42, no. 1, pp. 79–107, 2022.

[92] P. Marcelino, "Transfer learning from pre-trained models," *Towards data science*, vol. 10, p. 23, 2018.

[93] X. Han *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[94] M. L. Hutchinson, E. Antono, B. M. Gibbons, S. Paradiso, J. Ling, and B. Meredig, "Overcoming data scarcity with transfer learning," *arXiv preprint arXiv:1711.05099*, 2017.

[95] A. Kolides *et al.*, "Artificial intelligence foundation and pre-trained models: Fundamentals, applications, opportunities, and social impacts," *Simul Model Pract Theory*, vol. 126, p. 102754, 2023.

[96] B. Juba and H. S. Le, "Precision-recall versus accuracy and the role of large data sets," in *Proceedings of the AAAI conference on artificial intelligence*, 2019, pp. 4039–4048.

[97] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2623–2631.

[98] J. P. Winkler, J. Grönberg, and A. Vogelsang, "Optimizing for recall in automatic requirements classification: An empirical study," *Proceedings of the IEEE International Conference on Requirements Engineering*, vol. 2019-September, pp. 40–50, Sep. 2019, doi: 10.1109/RE.2019.00016.

[99] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 847–855.

[100] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[101] K. Hajian-Tilaki, "Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation," *Caspian J Intern Med*, vol. 4, no. 2, pp. 627–635, 2013.

[102] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit Lett*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/J.PATREC.2005.10.010.

[103] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. Berlin/New York: " O'Reilly Media, Inc.," 2022.

CONCLUSION GÉNÉRALE

In this section, we present a concise overview of the experimental results on the test set, which comprises 15% (210 samples) of the complete 1,400-image dataset. This portion was separated from the dataset at the outset to ensure that the model would be evaluated on truly unseen data.This balanced distribution ensures that both categories are adequately represented during evaluation, allowing for a more reliable assessment of each model's performance. Table 5 shows summary of each model's performance on the test set of 210 images, including loss, accuracy, precision, recall, F1-score, and the confusion matrix.

Model	**Loss**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Confusion Matrix**
DeiT	0.13507	0.95714	0.959338	0.957142	0.95717689	$\begin{bmatrix} 98 & 1 \\ 8 & 103 \end{bmatrix}$
ViT	0.05007	0.98095	0.981123	0.980952	0.9809489	$\begin{bmatrix} 101 & 3 \\ 1 & 105 \end{bmatrix}$
VGG19	0.18558	0.938095	0.938403	0.938095	0.9380572	$\begin{bmatrix} 94 & 8 \\ 5 & 103 \end{bmatrix}$
VGG16	0.11087	0.947619	0.947717	0.9476190	0.9476466	$\begin{bmatrix} 111 & 6 \\ 5 & 88 \end{bmatrix}$
ResNet50	0.39728	0.847619	0.854149	0.8476190	0.8481182	$\begin{bmatrix} 83 & 10 \\ 22 & 95 \end{bmatrix}$

Table 5. Summary of each model's performance on the test set of 210 images, including loss, accuracy, precision, recall, F1-score, and the confusion matrix

Table 6 provides standard definitions of the four fundamental metrics—true positives, true negatives, false positives, and false negatives—used to evaluate classification performance.

Metrics	**Explanations**
True Positive (TP)	The actual label is Defective (1), and the model correctly predicts Defective (1).
True Negative (TN):	The actual label is Non-Defective (0), and the model correctly predicts Non-Defective (0).

False Positive (FP):	The actual label is Non-Defective (0), but the model incorrectly predicts Defective (1).
False Negative (FN):	The actual label is Defective (1), but the model incorrectly predicts Non-Defective (0).

Table 6. Definitions of confusion matrix metrics used in this study, detailing the various ways a model's predictions can align—or misalign—with the actual labels.

Figure 10 presents the confusion matrices for each model—DeiT, ViT, VGG19, VGG16, and ResNet50—showing how many instances were correctly identified as defective (TP), mistakenly missed (FN), incorrectly flagged (FP), or correctly recognized as non-defective (TN).

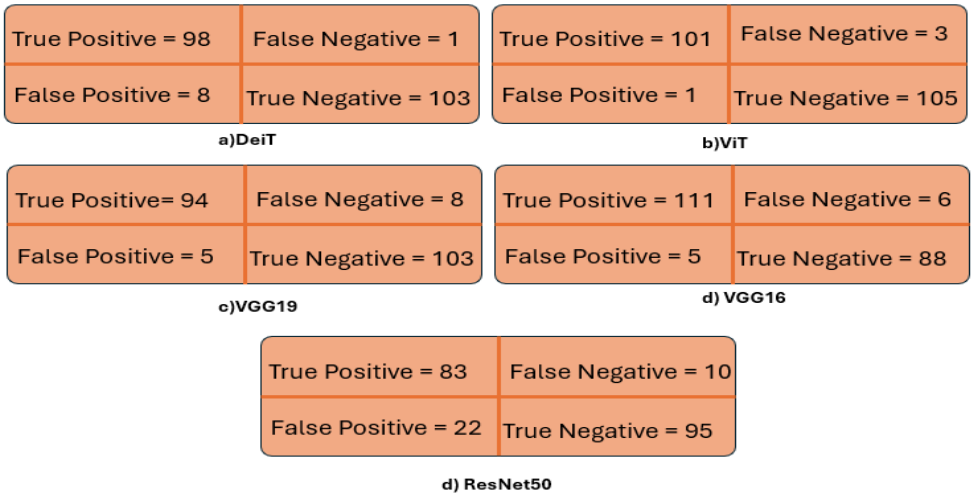

Figure 10. Confusion matrices illustrating the distribution of True Positives (TP), False Negatives (FN), False Positives (FP), and True Negatives (TN) for each model: (a) DeiT, (b) ViT, (c) VGG19, (d) VGG16, and (e) ResNet50.

In this application, we define a "positive" as a defective rail fastener (labeled as 1) and a "negative" as a non-defective fastener (labeled as 0). Consequently, a false negative (FN) represents a missed defect—an outcome we especially want to avoid in safety-critical scenarios—while a false positive (FP) is a case where a good fastener is incorrectly flagged as defective. Given this definition, recall (TP / [TP + FN]) takes center stage when the primary concern is ensuring that no defects slip by undetected. However, precision (TP / [TP + FP]) also matters if we want to reduce the unnecessary cost and downtime of investigating too many healthy fasteners. The F1 score, which harmonically balances recall and precision, is a useful single metric for comparing overall effectiveness across models.

Baed on Table 5 and Figure 10 and Turning to the individual models, ViT stands out as the top performer on almost every metric presented. Its extremely low loss and high accuracy (≈ 98%) coincide with the best reported precision (≈ 98%) and recall (≈ 98%). This means it flags very few good fasteners as defective (low FP) and misses only a handful of actual defects (low FN). Its confusion matrix reflects these strengths, showing minimal misclassifications overall. As a result, ViT achieves the highest F1 score, making it a superb choice if you need both strong detection of actual defects and minimal false alarms. Close behind is DeiT, which delivers approximately 96% recall and 96% precision. It shows a similarly strong ability to catch most defects and avoid too many incorrect flags, though it does fall slightly below ViT's metrics. Practically, this means DeiT may let a few more defective fasteners go unnoticed compared to ViT, and its confusion matrix confirms slightly higher false negatives (missed defects). If you can handle that small trade-off, DeiT remains a robust option, especially given its relatively high accuracy and balanced precision-recall profile.

For those favoring a CNN-based approach, VGG16 and VGG19 offer decent performance in the mid-90% range for recall and precision. While they do not match the top-tier results of ViT and DeiT, their confusion matrices still indicate a respectable balance: missed defects are comparatively few, and false alarms remain manageable. The slight edge goes to VGG16 over VGG19, but both remain viable if the infrastructure or preference leans toward well-established convolutional architectures. ResNet50, however, shows a notable drop in both recall and precision—around the mid-80% range. In practical terms, this means it misses more defective fasteners (higher FN) and also flags more good ones (higher FP). Its higher loss aligns with these relatively weaker outcomes. Given the safety implications of missing defects in railway systems, ResNet50 would require further training, tuning, or architectural refinements before being considered for critical real-world deployment.

Overall, if catching every defect (minimizing FN) is most important, ViT is the strongest across the board—offering stellar recall without sacrificing precision. DeiT also delivers a well-rounded performance, though with slightly lower recall. If you are

especially concerned about false positives (unnecessary maintenance checks), you might lean more toward ViT, as it has the highest precision and a superb F1 score. Meanwhile, VGG16 or VGG19 can serve as acceptable CNN-based alternatives for those comfortable with a modest performance trade-off. Finally, ResNet50 appears less suited for a high-risk, zero-tolerance environment unless significantly improved.

This study aimed to enhance railway fastener defect detection by comparing the performance of various deep learning models, specifically transformer-based models (ViT and DeiT) and traditional CNNs (ResNet50, VGG16, and VGG19). The transformer-based models, ViT and DeiT, showed better performance across all evaluated metrics, including accuracy, precision, recall, and loss. These models achieved the highest accuracy and the lowest loss, with ROC AUC values indicating better classification capabilities. The advanced performance of transformer-based models might be attributed to their self-attention mechanisms, which could capture the global context more effectively than traditional CNNs. VGG models, especially VGG16, also performed well, showing stable training progress and consistent improvement. These models demonstrated reliable accuracy and precision, making them viable alternatives for fastener defect detection tasks. The performance of VGG16 might be due to its deep architecture, which allows for a detailed extraction of features from the input images. However, it did not reach the performance levels of ViT and DeiT, which could be due to the inherent differences in architecture and feature extraction methods. ResNet50 showed the lowest performance among the models tested, with more fluctuations in metrics and less stability during training. This could be due to its residual connections, which, while useful in many contexts, might not be as effective for the specific task of fastener defect detection in railway infrastructure. The instability during training and the higher loss values suggest that ResNet50 might not be as suitable for this application.

The results of this study suggest that transformer-based models are more effective for fastener defect detection in railway maintenance, offering improved accuracy and reliability. This might be due to their advanced ability to capture global context through self-attention mechanisms, which could be more adept at identifying defects in the complex and varied visual environment of railway tracks. VGG models remain reliable alternatives, providing

stable and consistent performance, although they did not match the top performance of the transformer-based models in this study. Overall, the findings indicate that the choice of model could significantly impact the effectiveness of defect detection systems in railway maintenance. Transformer-based models might offer superior performance due to their ability to handle complex visual patterns and contextual information. Meanwhile, VGG models, with their proven track record and stability, could be a dependable choice where slightly lower performance is acceptable. ResNet50, despite its capabilities in other areas, might not be the best fit for this specific task due to its lower stability and higher loss during training.

NDE allows for inspecting railway components without causing damage, which is especially useful for hard-to-reach or remote areas where traditional inspection methods might be difficult or unsafe. This capability is crucial for maintaining the integrity of railway infrastructure, as it enables thorough inspections without disrupting operations or causing further wear and tear. For instance, advanced NDE techniques, such as ultrasonic testing, electromagnetic testing, and laser scanning, can detect internal and surface defects that might not be visible through conventional methods. These methods ensure that even the most minor defects are identified early, thereby preventing potential failures and ensuring the longevity of the railway components. SHM methods enable continuous monitoring of railway tracks, helping to identify potential issues early and reduce maintenance costs. By integrating sensors and IoT devices, SHM systems provide real-time data on the condition of railway tracks, such as stress, strain, and temperature changes. This continuous flow of information allows for predictive maintenance, where maintenance activities are scheduled based on the actual condition of the infrastructure rather than fixed intervals. This approach not only reduces the risk of unexpected failures but also optimizes maintenance schedules, leading to significant cost savings.

These techniques provide a practical approach to maintaining the safety and integrity of railway infrastructure. By leveraging NDE and SHM, railway operators can achieve a higher level of safety and efficiency. NDE ensures that inspections are comprehensive and non-invasive, preserving the condition of critical components. SHM, on the other hand, offers

a dynamic and proactive maintenance strategy, where potential problems are addressed before they escalate into major issues. Together, these methods represent a significant advancement in railway maintenance, combining thorough inspection capabilities with continuous monitoring to ensure the optimal performance of railway systems. In conclusion, the integration of NDE and SHM in railway maintenance practices might revolutionize how inspections and maintenance are conducted. These techniques could enhance the reliability of railway operations, minimize downtime, and ensure the safety of both the infrastructure and the passengers. As the railway industry continues to adopt these advanced methods, we might see a substantial improvement in the overall efficiency and safety of railway systems worldwide.

However, there are several limitations to consider. The dataset used was balanced, but real-world distributions might vary, potentially affecting model performance. This could lead to a decrease in accuracy when the models are applied to different or more complex datasets. Additionally, the dataset size and diversity might not fully represent all possible defect types, which could limit the generalizability of the findings. The higher computational needs of transformer models could limit their use in resource-limited settings. These models require significant processing power and memory, which might not be available in all railway maintenance environments. This could make it challenging to deploy these models in field conditions, especially in remote or less developed areas. Another limitation is the potential for overfitting, where models perform well on training data but poorly on new, unseen data. Despite using techniques to mitigate this, it remains a concern, especially with complex models like transformers. Additionally, while transfer learning can help improve performance with limited data, it might not fully address the variability and uniqueness of different railway environments. These limitations suggest that while the study's findings are promising, they might not fully translate to real-world applications without further refinement. The variations in data distributions, the computational demands, and the risk of overfitting could all impact the effectiveness of these models in practical settings. Future research might need to focus on addressing these challenges to ensure that the models can be effectively used in diverse and resource-limited railway maintenance environments.

Future research should address these limitations by using more diverse datasets to improve generalization. It could also explore hybrid models that combine the strengths of CNNs and transformers for better performance and efficiency. Further tuning of model settings and experimenting with different architectures will be important to enhance the effectiveness of these models in real-world railway maintenance systems. Additionally, expanding the use of NDE and SHM techniques in conjunction with advanced deep-learning models could provide even more comprehensive solutions for railway defect detection and maintenance. This combined approach might offer a robust framework for ensuring the ongoing safety and efficiency of railway infrastructure, particularly in challenging environments where traditional methods fall short. Using a mix of NDE for thorough, non-invasive inspections and SHM for continuous monitoring, combined with advanced models, could lead to a more proactive and effective maintenance strategy. In conclusion, this study highlights the potential of transformer-based models to improve the detection of railway fastener defects. This could lead to better safety, less downtime, and more efficient maintenance schedules in the railway industry. By leveraging the strengths of transformer-based models and CNNs, the study contributes to the development of more reliable railway maintenance systems. Addressing the current limitations and refining these models further could significantly enhance their real-world application, ensuring they perform well across various environments and conditions. Future efforts might focus on creating more adaptable models, optimizing computational resources, and improving generalization to diverse datasets. This way, advanced deep learning models could be more widely deployed, even in resource-constrained settings, ultimately leading to safer and more efficient railway systems worldwide.

RÉFÉRENCES BIBLIOGRAPHIQUE

[1] A. Jamwal, R. Agrawal, M. Sharma, and A. Giallanza, "Industry 4.0 technologies for manufacturing sustainability: A systematic review and future research directions," *Applied Sciences*, vol. 11, no. 12, p. 5725, 2021.

[2] T. V. Andrianandrianina Johanesa, L. Equeter, and S. A. Mahmoudi, "Survey on AI Applications for Product Quality Control and Predictive Maintenance in Industry 4.0," *Electronics (Basel)*, vol. 13, no. 5, p. 976, 2024.

[3] L. Da Xu, E. L. Xu, and L. Li, "Industry 4.0: state of the art and future trends," *Int J Prod Res*, vol. 56, no. 8, pp. 2941–2962, 2018.

[4] L. Monostori *et al.*, "Cyber-physical systems in manufacturing," *Cirp Annals*, vol. 65, no. 2, pp. 621–641, 2016.

[5] A. Gilchrist, *Industry 4.0: the industrial internet of things*. Springer, 2016.

[6] D. A. Tibaduiza Burgos, R. C. Gomez Vargas, C. Pedraza, D. Agis, and F. Pozo, "Damage identification in structural health monitoring: A brief review from its implementation to the use of data-driven applications," *Sensors*, vol. 20, no. 3, p. 733, 2020.

[7] C. Scuro, F. Lamonaca, S. Porzio, G. Milani, and R. S. Olivito, "Internet of Things (IoT) for masonry structural health monitoring (SHM): Overview and examples of innovative systems," *Constr Build Mater*, vol. 290, p. 123092, 2021.

[8] H. Hao, K. Bi, W. Chen, T. M. Pham, and J. Li, "Towards next generation design of sustainable, durable, multi-hazard resistant, resilient, and smart civil engineering structures," *Eng Struct*, vol. 277, p. 115477, 2023.

[9] A. Armijo and D. Zamora-Sánchez, "Integration of Railway Bridge Structural Health Monitoring into the Internet of Things with a Digital Twin: A Case Study," *Sensors*, vol. 24, no. 7, p. 2115, 2024.

[10] W. Doghri, A. Saddoud, and L. Chaari Fourati, “Cyber-physical systems for structural health monitoring: sensing technologies and intelligent computing,” *J Supercomput*, vol. 78, no. 1, pp. 766–809, 2022.

[11] J. Vrana and R. Singh, “NDE 4.0 From Design Thinking to Strategy,” *arXiv preprint arXiv:2003.07773*, 2020.

[12] Markets and Markets, “Market Leadership—Testing, Inspection and Certification Market,” 2022. [Online]. Available: https://www.marketsandmarkets.com/ResearchInsight/testing-inspection-certification-market.asp

[13] S. K. Dwivedi, M. Vishwakarma, and A. Soni, “Advances and researches on non destructive testing: A review,” *Mater Today Proc*, vol. 5, no. 2, pp. 3690–3698, 2018.

[14] Z. Shao *et al.*, “A review of Non-Destructive evaluation (NDE) techniques for residual stress profiling of metallic components in aircraft engines,” *Aerospace*, vol. 9, no. 10, p. 534, 2022.

[15] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, and Y. Li, “Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study,” *Eng Appl Artif Intell*, vol. 80, pp. 66–81, 2019.

[16] J. C. Gaskell *et al.*, “Breaking down the barriers to regional agricultural trade in Central Africa,” 2018. [Online]. Available: http://documents.worldbank.org/curated/en/233071535650013216/Breaking-down-the-barriers-to-regional-agricultural-trade-in-Central-Africa

[17] S. Sharma, Y. Cui, Q. He, R. Mohammadi, and Z. Li, “Data-driven optimization of railway maintenance for track geometry,” *Transp Res Part C Emerg Technol*, vol. 90, pp. 34–58, 2018.

[18] I. Durazo-Cardenas *et al.*, "An autonomous system for maintenance scheduling data-rich complex infrastructure: Fusing the railways' condition, planning and cost," *Transp Res Part C Emerg Technol*, vol. 89, pp. 234–253, 2018.

[19] T. Rakha and A. Gorodetsky, "Review of Unmanned Aerial System (UAS) applications in the built environment: Towards automated building inspection procedures using drones," *Autom Constr*, vol. 93, pp. 252–264, 2018.

[20] C. Mineo and Y. Javadi, "Robotic non-destructive testing," 2022, *MDPI*.

[21] F. Outay, H. A. Mengash, and M. Adnan, "Applications of unmanned aerial vehicle (UAV) in road safety, traffic and highway infrastructure management: Recent advances and challenges," *Transp Res Part A Policy Pract*, vol. 141, pp. 116–129, 2020.

[22] S. P. Bemis *et al.*, "Ground-based and UAV-Based photogrammetry: A multi-scale, high-resolution mapping tool for structural geology and paleoseismology," *J Struct Geol*, vol. 69, pp. 163–178, 2014.

[23] F. Fassi, L. Fregonese, S. Ackermann, and V. De Troia, "Comparison between laser scanning and automated 3d modelling techniques to reconstruct complex and extensive cultural heritage areas," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, pp. 73–80, 2013.

[24] S. Halder and K. Afsari, "Robots in inspection and monitoring of buildings and infrastructure: A systematic review," *Applied Sciences*, vol. 13, no. 4, p. 2304, 2023.

[25] A. Thaduri, D. Galar, and U. Kumar, "Railway assets: A potential domain for big data analytics," *Procedia Comput Sci*, vol. 53, pp. 457–467, 2015.

[26] H. Alawad, S. Kaewunruen, and M. An, "A Deep Learning Approach Towards Railway Safety Risk Assessment," *IEEE Access*, vol. 8, pp. 102811–102832, 2020, doi: 10.1109/ACCESS.2020.2997946.

[27] A. Mosleh, D. Ribeiro, A. Malekjafarian, and M. D. Mart\'\inez-Rodrigo, “Advances in Condition Monitoring of Railway Infrastructure,” 2024, *MDPI*.

[28] H. Cui, Q. Hu, and Q. Mao, “Real-time geometric parameter measurement of high-speed railway fastener based on point cloud from structured light sensors,” *Sensors*, vol. 18, no. 11, p. 3675, 2018.

[29] R. Samsami, “A Systematic Review of Automated Construction Inspection and Progress Monitoring (ACIPM): Applications, Challenges, and Future Directions,” *CivilEng*, vol. 5, no. 1, pp. 265–287, 2024.

[30] S. Agnisarman, S. Lopes, K. C. Madathil, K. Piratla, and A. Gramopadhye, “A survey of automation-enabled human-in-the-loop systems for infrastructure visual inspection,” *Autom Constr*, vol. 97, pp. 52–76, 2019.

[31] M. Seisenberger *et al.*, “Safe and secure future AI-driven railway technologies: challenges for formal methods in railway,” in *International Symposium on Leveraging Applications of Formal Methods*, 2022, pp. 246–268. doi: 10.1007/978-3-031-19762-8_20.

[32] T. Bai, J. Gao, J. Yang, and D. Yao, “A study on railway surface defects detection based on machine vision,” *Entropy*, vol. 23, no. 11, p. 1437, 2021.

[33] D. Zheng *et al.*, “A defect detection method for rail surface and fasteners based on deep Volutional neural network,” *Comput Intell Neurosci*, vol. 2021, pp. 1–15, 2021.

[34] J. Sresakoolchai and S. Kaewunruen, “Railway defect detection based on track geometry using supervised and unsupervised machine learning,” *Struct Health Monit*, vol. 21, no. 4, pp. 1757–1767, 2022.

[35] X. Xu, Y. Lei, F. Yang, and others, “Railway subgrade defect automatic recognition method based on improved faster R-CNN,” *Sci Program*, vol. 2018, 2018.

[36] J. Wang, L. Luo, W. Ye, and S. Zhu, “A defect-detection method of split pins in the catenary fastening devices of high-speed railway based on deep learning,” *IEEE Trans Instrum Meas*, vol. 69, no. 12, pp. 9517–9525, 2020.

[37] Y. Wu, Y. Qin, Y. Qian, and F. Guo, “Automatic detection of arbitrarily oriented fastener defect in high-speed railway,” *Autom Constr*, vol. 131, p. 103913, 2021.

[38] A. Lu, “Detecting Defective Rail Joints on the Swiss Railways with Inception ResNet V2: Simplifying Predictive Maintenance of Railway Infrastructure,” 2022, *KTH Royal Institute of Technology*.

[39] H. Li, F. Wang, J. Liu, H. Song, Z. Hou, and P. Dai, “Ensemble model for rail surface defects detection,” *PLoS One*, vol. 17, no. 5, p. e0268518, 2022.

[40] Z. Jian, S. He, S. Liu, J. Liu, and Y. Fang, “A multiple species railway defects detection method based on sample generation,” *IEEE Trans Instrum Meas*, 2024.

[41] N. Hütten, R. Meyes, and T. Meisen, “Vision Transformer in Industrial Visual Inspection,” *Applied Sciences*, vol. 12, no. 23, p. 11981, 2022.

[42] C. T. Alexakos, Y. L. Karnavas, M. Drakaki, and I. A. Tziafettas, “A combined short time fourier transform and image classification transformer model for rolling element bearings fault diagnosis in electric motors,” *Mach Learn Knowl Extr*, vol. 3, no. 1, pp. 228–242, 2021.

[43] K. An and Y. Zhang, “LPViT: a transformer based model for PCB image classification and defect detection,” *IEEE Access*, vol. 10, pp. 42542–42553, 2022.

[44] L. M. Dang, H. Wang, Y. Li, T. N. Nguyen, and H. Moon, “DefectTR: End-to-end defect detection for sewage networks using a transformer,” *Constr Build Mater*, vol. 325, p. 126584, 2022.

[45] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A Next-generation Hyperparameter Optimization Framework,” *Proceedings of the 25th ACM SIGKDD*

International Conference on Knowledge Discovery & Data Mining, pp. 2623–2631, 2019.

[46] F. Gorunescu, *Data Mining: Concepts, models and techniques*, vol. 12. Berlin/New York: Springer Science & Business Media, 2011.

[47] X. Mao *et al.*, "Towards Robust Vision Transformer."

[48] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[49] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1– learning rate, batch size, momentum, and weight decay," *arXiv preprint arXiv:1803.09820*, 2018.

[50] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Berlin/New York: MIT press, 2016.

[51] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[52] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans Knowl Data Eng*, vol. 22, no. 10, pp. 1345–1359, 2009.

[53] B. Bischl *et al.*, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 13, no. 2, p. e1484, 2023.

[54] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.

[55] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013, pp. 847–855.

[56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[57] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *International conference on machine learning*, 2013, pp. 1139–1147.

[58] S. Prusty, S. Patnaik, and S. K. Dash, "SKCV: Stratified K-fold cross-validation on ML classifiers for predicting cervical cancer," *Frontiers in Nanotechnology*, vol. 4, p. 972421, 2022.

[59] H. Kagermann, W. Wahlster, J. Helbig, and others, "Recommendations for implementing the strategic initiative INDUSTRIE 4.0," *Final report of the Industrie*, vol. 4, no. 0, p. 82, 2013.

[60] M. Hermann, T. Pentek, and B. Otto, "Design principles for industrie 4.0 scenarios," in *2016 49th Hawaii international conference on system sciences (HICSS)*, 2016, pp. 3928–3937.

[61] R. Söderberg, K. Wärmefjord, J. S. Carlson, and L. Lindkvist, "Toward a Digital Twin for real-time geometry assurance in individualized production," *CIRP annals*, vol. 66, no. 1, pp. 137–140, 2017.

[62] F. Rosin, P. Forget, S. Lamouri, and R. Pellerin, "Enhancing the decision-making process through industry 4.0 technologies," *Sustainability*, vol. 14, no. 1, p. 461, 2022.

[63] M. I. Hossain, Dr. T. Steigner, M. I. Hussain, and A. Akther, "Enhancing Data Integrity and Traceability in Industry Cyber Physical Systems (ICPS) through Blockchain Technology: A Comprehensive Approach," May 2024, Accessed: Jun. 18, 2024. [Online]. Available: https://arxiv.org/abs/2405.04837v1

[64] G. M. Sang, L. Xu, and P. de Vrieze, "A predictive maintenance model for flexible manufacturing in the context of industry 4.0," *Front Big Data*, vol. 4, p. 663466, 2021.

[65] R. D. S. G. Campilho and F. J. G. Silva, “Industrial Process Improvement by Automation and Robotics,” *Machines 2023, Vol. 11, Page 1011*, vol. 11, no. 11, p. 1011, Nov. 2023, doi: 10.3390/MACHINES11111011.

[66] A. Cotrino, M. A. Sebastián, and C. González-Gaya, “Industry 4.0 Roadmap: Implementation for Small and Medium-Sized Enterprises,” *Applied Sciences 2020, Vol. 10, Page 8566*, vol. 10, no. 23, p. 8566, Nov. 2020, doi: 10.3390/APP10238566.

[67] M. Rüßmann *et al.*, “Industry 4.0: The future of productivity and growth in manufacturing industries,” *Boston consulting group*, vol. 9, no. 1, pp. 54–89, 2015.

[68] J. Pochmara and A. Świetlicka, “Cybersecurity of Industrial Systems—A 2023 Report,” *Electronics 2024, Vol. 13, Page 1191*, vol. 13, no. 7, p. 1191, Mar. 2024, doi: 10.3390/ELECTRONICS13071191.

[69] G. Aceto, V. Persico, and A. Pescapé, “A survey on Information and Communication Technologies for Industry 4.0: state of the art, taxonomies, perspectives, and challenges,” 2019, doi: 10.1109/COMST.2019.2938259.

[70] M. Ghobakhloo, M. Iranmanesh, B. Foroughi, E. Babaee Tirkolaee, S. Asadi, and A. Amran, “Industry 5.0 implications for inclusive sustainable manufacturing: An evidence-knowledge-based strategic roadmap,” *J Clean Prod*, vol. 417, p. 138023, Sep. 2023, doi: 10.1016/J.JCLEPRO.2023.138023.

[71] B. Nicoletti, “Industrial Revolutions and Supply Network 5.0,” *Supply Network 5.0*, pp. 43–101, 2023, doi: 10.1007/978-3-031-22032-6_3.

[72] H. Singh, U. Chauhan, S. P. S. Chauhan, A. Saxena, and P. Kumari, “Adaptive and Personalized Learning in Industry 5.0 Education,” *https://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/979-8-3693-0782-3.ch001*, pp. 1–19, Jan. 1AD, doi: 10.4018/979-8-3693-0782-3.CH001.

[73] M. Ghobakhloo, H. A. Mahdiraji, M. Iranmanesh, and V. Jafari-Sadeghi, “From Industry 4.0 Digital Manufacturing to Industry 5.0 Digital Society: a Roadmap

Toward Human-Centric, Sustainable, and Resilient Production," *Information Systems Frontiers 2024*, pp. 1–33, Feb. 2024, doi: 10.1007/S10796-024-10476-Z.

[74] A. Rehman and T. Umar, "Literature review: Industry 5.0. Leveraging technologies for environmental, social and governance advancement in corporate settings," *Corporate Governance (Bingley)*, vol. ahead-of-print, no. ahead-of-print, 2024, doi: 10.1108/CG-11-2023-0502/FULL/XML.

[75] B. Martini, D. Bellisario, and P. Coletti, "Human-Centered and Sustainable Artificial Intelligence in Industry 5.0: Challenges and Perspectives," *Sustainability 2024, Vol. 16, Page 5448*, vol. 16, no. 13, p. 5448, Jun. 2024, doi: 10.3390/SU16135448.

[76] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull Math Biophys*, vol. 5, no. 4, pp. 115–133, Mar. 1943, doi: 10.1007/BF02478259.

[77] F. Rosenblatt, "The perceptron, a perceiving and recognizing automaton Project Para," *Cornell Aeronautical Laboratory*, 1957.

[78] H. J. Kelley, "Gradient Theory of Optimal Flight Paths," *ARS Journal*, vol. 30, no. 10, pp. 947–954, Mar. 1960, doi: 10.2514/8.5282.

[79] S. Dreyfus, "The numerical solution of variational problems," *J Math Anal Appl*, vol. 5, no. 1, pp. 30–45, Mar. 1962, doi: 10.1016/0022-247X(62)90004-5.

[80] S. J. Farlow, *Self-organizing methods in modeling: GMDH type algorithms*. Berlin/New York: CrC Press, 2020.

[81] M. Minsky and S. Papert, "An introduction to computational geometry," *Cambridge tiass., HIT*, vol. 479, no. 480, p. 104, 1969.

[82] K. Fukushima, "Visual Feature Extraction by a Multilayered Network of Analog Threshold Elements," *IEEE Transactions on Systems Science and Cybernetics*, vol. 5, no. 4, pp. 322–333, 1969, doi: 10.1109/TSSC.1969.300225.

[83] J. Schmidhuber, “Annotated history of modern AI and Deep learning,” *arXiv preprint arXiv:2212.11279*, 2022.

[84] P. Ramachandran, B. Zoph, and Q. V Le, “Searching for activation functions,” *arXiv preprint arXiv:1710.05941*, 2017.

[85] S. Linnainmaa, “Alogritmin kumulatiivinen pyöristysvirhe yksittäisten pyöristysvirheiden Taylor-kehitelmänä,” Master’s thesis, University of Helsinki, 1970.

[86] A. G. Ivakhnenko, “ Polynomial Theory of Complex Systems,” *IEEE Trans Syst Man Cybern*, vol. 4, pp. 364–378, 1971.

[87] K. Fukushima, “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biol Cybern*, vol. 36, no. 4, pp. 193–202, Mar. 1980, doi: 10.1007/BF00344251.

[88] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities.,” *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, Mar. 1982, doi: 10.1073/pnas.79.8.2554.

[89] P. Werbos, “Applications of advances in nonlinear sensitivity analysis,” *ystem modeling and optimization. Springer*, pp. 762–770, 1982.

[90] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, “A Learning Algorithm for Boltzmann Machines*,” *Cogn Sci*, vol. 9, no. 1, pp. 147–169, Mar. 1985, doi: 10.1207/s15516709cog0901_7.

[91] G. Hinton, *Connectionist Symbol Processing*, 1st ed. The MIT Press, 1991.

[92] T. Dutoit, *An Introduction to Text-to-Speech Synthesis*. Berlin/New York: Springer Science & Business Media, 2001.

[93] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, “Learning representations by back-propagating errors,” *Nature*, vol. 323, no. 6088, pp. 533–536, Mar. 1986, doi: 10.1038/323533a0.

[94] D. E. Rumelhart, J. L. McClelland, and PDP Research Group, *Parallel distributed processing: Explorations in the microstructure of cognition*, vol. 1. Foundations. MIT press, 1986.

[95] “Backpropagation Applied to Handwritten Zip Code Recognition,” *Neural Comput*, vol. 1, no. 4, pp. 541–551, Mar. 1989, doi: 10.1162/neco.1989.1.4.541.

[96] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303–314, Mar. 1989, doi: 10.1007/BF02551274.

[97] S. Hochreiter, “ Untersuchungen zu dynamischen neuronalen Netzen,” Institut f. Informatik, Technische Univ. Munich, 1991.

[98] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Mar. 1997, doi: 10.1162/neco.1997.9.8.1735.

[99] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” *Neural Comput*, vol. 18, no. 7, pp. 1527–1554, Mar. 2006, doi: 10.1162/neco.2006.18.7.1527.

[100] “https://en.wikipedia.org/wiki/Andrew_Ng.”

[101] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Mar. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[102] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.

[103] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun ACM*, vol. 60, no. 6, pp. 84–90, Mar. 2017, doi: 10.1145/3065386.

[104] I. Goodfellow *et al.*, "Generative adversarial nets," *Adv Neural Inf Process Syst*, vol. 27, 2014.

[105] A. Vaswani *et al.*, "Attention is all you need," *Adv Neural Inf Process Syst*, vol. 30, 2017.

[106] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018. [Online]. Available: https://github.com/tensorflow/tensor2tensor

[107] A. Rogers, O. Kovaleva, and A. Rumshisky, "A Primer in BERTology: What We Know About How BERT Works," *Trans Assoc Comput Linguist*, vol. 8, pp. 842–866, Mar. 2021, doi: 10.1162/tacl_a_00349.

[108] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[109] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[110] T. Brown *et al.*, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[111] OpenAI, "GPT-4 Technical Report," Mar. 2023.

[112] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Computer Society Conference on Computer*

Vision and Pattern Recognition, vol. 2016-December, pp. 770–778, Dec. 2015, doi: 10.1109/CVPR.2016.90.

[113] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in Vision: A Survey," *ACM Computing Surveys (CSUR)*, vol. 54, no. 10, Mar. 2022, doi: 10.1145/3505244.

[114] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," Dec. 2020.

[115] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, 2020, pp. 213–229.

[116] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[117] L. Ye, M. Rochan, Z. Liu, and Y. Wang, "Cross-Modal Self-Attention Network for Referring Image Segmentation," 2019.

[118] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5791–5800.

[119] C. Sun, A. Myers, C. Vondrick, K. Murphy, C. Schmid, and G. Research, "VideoBERT: A Joint Model for Video and Language Representation Learning," 2019.

[120] R. J. Mikulak, R. McDermott, and M. Beauregard, *The Basics of FMEA*. Berlin/New York: Productivity Press, 2017. doi: 10.1201/b16656.

[121] W. A. Shewhart, "Control of quality of manufactured product," 1929.

[122] D. E. Bray and R. K. Stanley, *Nondestructive evaluation: a tool in design, manufacturing and service*. CRC press, 1996.

[123] M. Moganti, F. Ercal, C. H. Dagli, and S. Tsunekawa, "Automatic PCB inspection algorithms: a survey," *Computer vision and image understanding*, vol. 63, no. 2, pp. 287–313, 1996, doi: 10.1006/CVIU.1996.0020.

[124] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv Neural Inf Process Syst*, vol. 25, 2012.

[125] K. Schwab, *The Fourth Industrial Revolution*. New York: Currency, 2017.

[126] E. Cumbajin *et al.*, "A Real-Time Automated Defect Detection System for Ceramic Pieces Manufacturing Process Based on Computer Vision with Deep Learning," *Sensors 2024, Vol. 24, Page 232*, vol. 24, no. 1, p. 232, Dec. 2023, doi: 10.3390/S24010232.

[127] P. N. Mahalle *et al., Industry 4.0 Convergence with AI, IoT, Big Data and Cloud Computing: Fundamentals, Challenges and Applications*. Bentham Science Publishers, 2023.

[128] H. Taheri, M. G. Bocanegra, and M. Taheri, "Artificial Intelligence, Machine Learning and Smart Technologies for Nondestructive Evaluation," *Sensors 2022, Vol. 22, Page 4055*, vol. 22, no. 11, p. 4055, May 2022, doi: 10.3390/S22114055.

[129] X. W. Ye, T. Jin, and C. B. Yun, "A review on deep learning-based structural health monitoring of civil infrastructures," *Smart Struct. Syst*, vol. 24, no. 5, pp. 567–585, 2019.

[130] R. T. Reisch *et al.*, "Context awareness in process monitoring of additive manufacturing using a digital twin," *International Journal of Advanced Manufacturing Technology*, vol. 119, no. 5–6, pp. 3483–3500, Mar. 2022, doi: 10.1007/S00170-021-08636-5/FIGURES/15.

[131] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[132] R. Chataut, A. Phoummalayvane, and R. Akl, “Unleashing the Power of IoT: A Comprehensive Review of IoT Applications and Future Prospects in Healthcare, Agriculture, Smart Homes, Smart Cities, and Industry 4.0,” *Sensors 2023, Vol. 23, Page 7194*, vol. 23, no. 16, p. 7194, Aug. 2023, doi: 10.3390/S23167194.

[133] M. Pech, J. Vrchota, and J. Bednář, “Predictive Maintenance and Intelligent Sensors in Smart Factory: Review,” *Sensors 2021, Vol. 21, Page 1470*, vol. 21, no. 4, p. 1470, Feb. 2021, doi: 10.3390/S21041470.

[134] A. Soussi, E. Zero, R. Sacile, D. Trinchero, and M. Fossa, “Smart Sensors and Smart Data for Precision Agriculture: A Review,” *Sensors 2024, Vol. 24, Page 2647*, vol. 24, no. 8, p. 2647, Apr. 2024, doi: 10.3390/S24082647.

[135] A. ; Gallegos *et al.*, “Waste Management in the Smart City: Current Practices and Future Directions,” *Resources 2023, Vol. 12, Page 115*, vol. 12, no. 10, p. 115, Sep. 2023, doi: 10.3390/RESOURCES12100115.

[136] L. Tawalbeh, F. Muheidat, M. Tawalbeh, and M. Quwaider, “IoT Privacy and Security: Challenges and Solutions,” *Applied Sciences 2020, Vol. 10, Page 4102*, vol. 10, no. 12, p. 4102, Jun. 2020, doi: 10.3390/APP10124102.

[137] M. Noura, M. Atiquzzaman, and M. Gaedke, “Interoperability in Internet of Things: Taxonomies and Open Challenges,” *Mobile Networks and Applications*, vol. 24, no. 3, pp. 796–809, Jun. 2019, doi: 10.1007/S11036-018-1089-9/FIGURES/5.

[138] E. Adi, A. Anwar, Z. Baig, and S. Zeadally, “Machine learning and data analytics for the IoT,” *Neural Comput Appl*, vol. 32, no. 20, pp. 16205–16233, Oct. 2020, doi: 10.1007/S00521-020-04874-Y/METRICS.

[139] H. Taheri, M. Gonzalez Bocanegra, and M. Taheri, “Artificial intelligence, machine learning and smart technologies for nondestructive evaluation,” *Sensors*, vol. 22, no. 11, p. 4055, 2022.

[140] K. Taha, “Observational and Experimental Insights into Machine Learning-Based Defect Classification in Wafers,” Oct. 2023, Accessed: Jun. 18, 2024. [Online]. Available: https://arxiv.org/abs/2310.10705v4

[141] N. Meyendorf, N. Ida, R. Singh, and J. Vrana, “Handbook of Nondestructive Evaluation 4.0,” *Handbook of Nondestructive Evaluation 4.0*, pp. 1–1282, Jan. 2022, doi: 10.1007/978-3-030-73206-6.

[142] G. Sariyer, S. K. Mangla, Y. Kazancoglu, C. Ocal Tasar, and S. Luthra, “Data analytics for quality management in Industry 4.0 from a MSME perspective,” *Ann Oper Res*, pp. 1–29, Aug. 2021, doi: 10.1007/S10479-021-04215-9/METRICS.

[143] “Home - ASNT Pulse.” Accessed: Jun. 16, 2024. [Online]. Available: https://blog.asnt.org/

[144] “Industry 4.0 and predictive technologies for asset maintenance | Deloitte Insights.” Accessed: Jun. 16, 2024. [Online]. Available: https://www2.deloitte.com/us/en/insights/focus/industry-4-0/using-predictive-technologies-for-asset-maintenance.html

[145] V. Kumar, P. Vrat, and R. Shankar, “Prioritization of strategies to overcome the barriers in Industry 4.0: a hybrid MCDM approach,” *OPSEARCH*, vol. 58, no. 3, pp. 711–750, Sep. 2021, doi: 10.1007/S12597-020-00505-1/TABLES/16.

[146] A. M. Alnajim, S. Habib, M. Islam, S. M. Thwin, and F. Alotaibi, “A Comprehensive Survey of Cybersecurity Threats, Attacks, and Effective Countermeasures in Industrial Internet of Things,” *Technologies 2023, Vol. 11, Page 161*, vol. 11, no. 6, p. 161, Nov. 2023, doi: 10.3390/TECHNOLOGIES11060161.

[147] J. Vrana and R. Singh, “NDE 4.0—A Design Thinking Perspective,” *J Nondestr Eval*, vol. 40, no. 1, pp. 1–24, Mar. 2021, doi: 10.1007/S10921-020-00735-9/FIGURES/7.

[148] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, “Digital Twin in Industry: State-of-the-Art,” *IEEE Trans Industr Inform*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019, doi: 10.1109/TII.2018.2873186.

[149] L. Li, S. Aslam, A. Wileman, and S. Perinpanayagam, “Digital twin in aerospace industry: A gentle introduction,” *IEEE Access*, vol. 10, pp. 9543–9562, 2021.

[150] P. Armeni, I. Polat, L. M. De Rossi, L. Diaferia, S. Meregalli, and A. Gatti, “Digital twins in healthcare: is it the beginning of a new era of evidence-based medicine? a critical review,” *J Pers Med*, vol. 12, no. 8, p. 1255, 2022.

[151] G. Caprari, G. Castelli, M. Montuori, M. Camardelli, and R. Malvezzi, “Digital twin for urban planning in the green deal era: A state of the art and future perspectives,” *Sustainability*, vol. 14, no. 10, p. 6263, 2022.

[152] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital twin: Enabling technologies, challenges and open research,” *IEEE access*, vol. 8, pp. 108952–108971, 2020.

[153] D. Jones, C. Snider, A. Nassehi, J. Yon, and B. Hicks, “Characterising the Digital Twin: A systematic literature review,” *CIRP J Manuf Sci Technol*, vol. 29, pp. 36–52, 2020.

[154] F. Emmert-Streib, “What Is the Role of AI for Digital Twins?,” *AI 2023, Vol. 4, Pages 721-728*, vol. 4, no. 3, pp. 721–728, Sep. 2023, doi: 10.3390/AI4030038.

[155] A. ; Khdoudi *et al.*, “A Deep-Reinforcement-Learning-Based Digital Twin for Manufacturing Process Optimization,” *Systems 2024, Vol. 12, Page 38*, vol. 12, no. 2, p. 38, Jan. 2024, doi: 10.3390/SYSTEMS12020038.

[156] G. T. Webb, P. J. Vardanega, and C. R. Middleton, “Categories of SHM Deployments: Technologies and Capabilities”, doi: 10.1061/(ASCE)BE.1943-5592.0000735.

[157] “REAL-TIME STRUCTURAL HEALTH MONITORING FOR CONCRETE BEAMS: A COST-EFFECTIVE ‘INDUSTRY 4.0’ SOLUTION USING PIEZO SENSORS”.

[158] C. R. Farrar and N. A. J. Lieven, “Damage prognosis: the future of structural health monitoring,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 623–632, 2007.

[159] G. Wang and J. Ke, “Literature Review on the Structural Health Monitoring (SHM) of Sustainable Civil Infrastructure: An Analysis of Influencing Factors in the Implementation,” *Buildings 2024, Vol. 14, Page 402*, vol. 14, no. 2, p. 402, Feb. 2024, doi: 10.3390/BUILDINGS14020402.

[160] E. Ozer and M. Q. Feng, “Structural Reliability Estimation with Participatory Sensing and Mobile Cyber-Physical Structural Health Monitoring Systems,” *Applied Sciences 2019, Vol. 9, Page 2840*, vol. 9, no. 14, p. 2840, Jul. 2019, doi: 10.3390/APP9142840.

[161] D. V Achillopoulou, S. A. Mitoulis, S. A. Argyroudis, and Y. Wang, “Monitoring of transport infrastructure exposed to multiple hazards: A roadmap for building resilience,” *Science of the total environment*, vol. 746, p. 141001, 2020.

[162] S. Sony, S. Laventure, and A. Sadhu, “A literature review of next-generation smart sensing technology in structural health monitoring,” *Struct Control Health Monit*, vol. 26, no. 3, p. e2321, 2019.

[163] A. Malekloo, E. Ozer, M. AlHamaydeh, and M. Girolami, “Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights,” *Struct Health Monit*, vol. 21, no. 4, pp. 1906–1955, 2022.

[164] R. Hartung, L. Senger, and K. Klemt-Albert, “Linking building information modeling and structural health monitoring for reliable railway infrastructure,” in *Proceedings of the 29th European Safety and Reliability Conference (ESREL)*, 2019, pp. 596–603.

[165] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, and Y. Li, “Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study,” *Eng Appl Artif Intell*, vol. 80, pp. 66–81, 2019.

[166] R. Anand, R. Perrelli, and B. Zhang, “South Africa’s Exports Performance: Any Role for Structural Factors?,” *IMF Working Papers*, vol. 16, no. 24, p. 1, 2016, doi: 10.5089/9781475594003.001.

[167] *Breaking Down the Barriers to Regional Agricultural Trade in Central AfricaBriser les Obstacles au Commerce Agricole Regional en Afrique Centrale*. World Bank, Washington, DC, 2018. doi: 10.1596/30397.

[168] “Integrated Intervention Tool : Integration Strategies for Urban Poor Areas and Disadvantaged Communities,” Washington, DC, Jan. 2013. Accessed: Oct. 15, 2023. [Online]. Available: http://hdl.handle.net/10986/24492

[169] F. Peng, S. Kang, X. Li, Y. Ouyang, K. Somani, and D. Acharya, “A heuristic approach to the railroad track maintenance scheduling problem,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 2, pp. 129–145, 2011.

[170] A. Lasisi and N. Attoh-Okine, “Principal components analysis and track quality index: A machine learning approach,” *Transp Res Part C Emerg Technol*, vol. 91, pp. 230–248, 2018.

[171] C. Wu, Y. Yuan, Y. Tang, and B. Tian, “Application of Terrestrial Laser Scanning (TLS) in the Architecture, Engineering and Construction (AEC) Industry,” *Sensors 2022, Vol. 22, Page 265*, vol. 22, no. 1, p. 265, Dec. 2021, doi: 10.3390/S22010265.

[172] “[2401.02343] AERIAL-CORE: AI-Powered Aerial Robots for Inspection and Maintenance of Electrical Power Infrastructures.” Accessed: Jun. 22, 2024. [Online]. Available: https://ar5iv.labs.arxiv.org/html/2401.02343v1

[173] D. D. Lichti, S. J. Gordon, and T. Tipdecho, “Error Models and Propagation in Directly Georeferenced Terrestrial Laser Scanner Networks,” *Journal of Surveying*

Engineering, vol. 131, no. 4, pp. 135–142, Nov. 2005, doi: 10.1061/(ASCE)0733-9453(2005)131:4(135).

[174] H. Azpúrua *et al.*, "Towards Semi-autonomous Robotic Inspection and Mapping in Confined Spaces with the EspeleoRobô," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 101, no. 4, pp. 1–27, Apr. 2021, doi: 10.1007/S10846-021-01321-5/METRICS.

[175] A. Zoeteman, R. Dollevoet, and Z. Li, "Dutch research results on wheel/rail interface management: 2001–2013 and beyond," *Proc Inst Mech Eng F J Rail Rapid Transit*, vol. 228, no. 6, pp. 642–651, 2014.

[176] L. Huang, C. Wu, B. Wang, and Q. Ouyang, "Big-data-driven safety decision-making: A conceptual framework and its influencing factors," *Saf Sci*, vol. 109, pp. 46–56, Nov. 2018, doi: 10.1016/j.ssci.2018.05.012.

[177] O. Alshorman *et al.*, "A Review of Artificial Intelligence Methods for Condition Monitoring and Fault Diagnosis of Rolling Element Bearings for Induction Motor," 2020, doi: 10.1155/2020/8843759.

[178] Y. Zhao *et al.*, "A Review on Rail Defect Detection Systems Based on Wireless Sensors," *Sensors 2022, Vol. 22, Page 6409*, vol. 22, no. 17, p. 6409, Aug. 2022, doi: 10.3390/S22176409.

[179] H. Cui, Q. Hu, and Q. Mao, "Real-Time Geometric Parameter Measurement of High-Speed Railway Fastener Based on Point Cloud from Structured Light Sensors," *Sensors 2018, Vol. 18, Page 3675*, vol. 18, no. 11, p. 3675, Oct. 2018, doi: 10.3390/S18113675.

[180] M. Achouch *et al.*, "On Predictive Maintenance in Industry 4.0: Overview, Models, and Challenges," *Applied Sciences 2022, Vol. 12, Page 8081*, vol. 12, no. 16, p. 8081, Aug. 2022, doi: 10.3390/APP12168081.

[181] L. Xiong, G. Jing, J. Wang, X. Liu, and Y. Zhang, “Detection of Rail Defects Using NDT Methods,” *Sensors 2023, Vol. 23, Page 4627*, vol. 23, no. 10, p. 4627, May 2023, doi: 10.3390/S23104627.

[182] M. Shafiee, Z. Zhou, L. Mei, F. Dinmohammadi, J. Karama, and D. Flynn, “Unmanned Aerial Drones for Inspection of Offshore Wind Turbines: A Mission-Critical Failure Analysis,” *Robotics 2021, Vol. 10, Page 26*, vol. 10, no. 1, p. 26, Feb. 2021, doi: 10.3390/ROBOTICS10010026.

[183] C. Wu, Y. Yuan, Y. Tang, and B. Tian, “Application of Terrestrial Laser Scanning (TLS) in the Architecture, Engineering and Construction (AEC) Industry,” *Sensors 2022, Vol. 22, Page 265*, vol. 22, no. 1, p. 265, Dec. 2021, doi: 10.3390/S22010265.

[184] Z. Xiong, Q. Li, Q. Mao, and Q. Zou, “A 3D Laser Profiling System for Rail Surface Defect Detection,” *Sensors 2017, Vol. 17, Page 1791*, vol. 17, no. 8, p. 1791, Aug. 2017, doi: 10.3390/S17081791.

[185] K. A. Kas and G. K. Johnson, “Using unmanned aerial vehicles and robotics in hazardous locations safely,” *Process safety progress*, vol. 39, no. 1, p. e12066, 2020.

[186] C. Yang, Y. Sun, C. Ladubec, and Y. Liu, “Developing Machine Learning-Based Models for Railway Inspection,” *Applied Sciences 2021, Vol. 11, Page 13*, vol. 11, no. 1, p. 13, Dec. 2020, doi: 10.3390/APP11010013.

[187] M. Iman, H. R. Arabnia, and K. Rasheed, “A review of deep transfer learning and recent advancements,” *Technologies (Basel)*, vol. 11, no. 2, p. 40, 2023.

[188] F. Zhuang *et al.*, “A comprehensive survey on transfer learning,” *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[189] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio, “Transfusion: Understanding transfer learning for medical imaging,” *Adv Neural Inf Process Syst*, vol. 32, 2019.

[190] P. Huilgol, “Top 4 Pre-Trained Models for Image Classification—With Python Code,” *Analytics Vidhya: Gurgaon, India*, 2020.

[191] X. Han *et al.*, "Pre-trained models: Past, present and future," *AI Open*, vol. 2, pp. 225–250, 2021.

[192] Y. Shu, Z. Cao, Z. Zhang, J. Wang, and M. Long, "Hub-pathway: transfer learning from A hub of pre-trained models," *Adv Neural Inf Process Syst*, vol. 35, pp. 32913–32927, 2022.

[193] L. Alzubaidi *et al.*, "Novel transfer learning approach for medical imaging with limited labeled data," *Cancers (Basel)*, vol. 13, no. 7, p. 1590, 2021.

[194] "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J Big Data*, vol. 8, no. 1, pp. 1–74, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[195] I. Rakhmatulin, M.-S. Dao, A. Nassibi, and D. Mandic, "Exploring Convolutional Neural Network Architectures for EEG Feature Extraction," *Sensors*, vol. 24, no. 3, p. 877, 2024.

[196] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[197] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 2261–2269, Aug. 2016, doi: 10.1109/CVPR.2017.243.

[198] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, pp. 611–629, 2018.

[199] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[200] A. Dosovitskiy *et al.*, “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale,” Oct. 2020.

[201] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” in *International conference on machine learning*, 2021, pp. 10347–10357.

[202] Y. Jung, “Multiple predicting K-fold cross-validation for model selection,” *J Nonparametr Stat*, vol. 30, no. 1, pp. 197–215, 2018.