Lightweight Hybrid CNN–Vision Transformer for Real-Time Automated Shipping Container Damage Detection

Adi Kurniawan
PhD Student
King Fahd University of Petroleum and Minerals, Electrical Engineering
Saudi Arabia

Mohamed A. Mohandes
IRC-SES
King Fahd University of Petroleum and Minerals, Electrical Engineering
Saudi Arabia

Naveed Iqbal
IRC-CSS
King Fahd University of Petroleum and Minerals, Electrical Engineering
Saudi Arabia

Shafiqur Rehman
IRC-CSS
King Fahd University of Petroleum and Minerals, Electrical Engineering
Saudi Arabia

Hilal H. Nuha
Telkom University
Bandung, Indonesia

Manual container inspections often lead to inconsistencies and inefficiencies, which can disrupt supply chains and increase operational costs. The time-consuming nature of manual checks makes automation an appealing alternative. This paper presents a lightweight hybrid model combining Convolutional Neural Networks (CNN) and Vision Transformers (ViT), specifically designed for automated container damage classification. The CNN extracts fine-grained local features, while the ViT models global structural patterns, overcoming the limitations of purely convolutional architectures. We evaluate four model variants on a dataset of 2,116 images, collected from container depots near Jakarta Port. Our proposed CNN–ViT hybrid model generalized well with this dataset and achieves 96.57% ± 0.83 accuracy, 0.089 ± 0.015 binary cross-entropy loss, and 64.21 ± 1.47 ms inference latency, peaking at 97.2% accuracy and 62 ms latency in the best trial with only 1 million parameters. Compared to MobileNetV2, our approach improves classification accuracy by about 1% while reducing inference time by approximately 9 ms, demonstrating its efficiency for real-time automated container inspection in resource-constrained environments.

Keywords: *computer vision, lightweight models, binary classification, CNN, ViT, container inspection.*

1. INTRODUCTION

Shipping containers play an essential role in global trade, facilitating the transportation of goods over large distances. However, containers might get damaged by impact [1] during transit or gradual wear over time. To ensure the reliability and security of shipments, they need to be inspected regularly. However, manual inspections are labor intensive, time consuming, and prone to human error. Field observations reveal that inspecting stacked containers is labor-intensive and time-consuming, as they must be grounded for assessment. Automated inspection promises faster, more accurate, and more consistent inspection solutions.

Advances in computer vision and deep learning have made it increasingly feasible to automate visual inspec–tion tasks. Convolutional Neural Networks (CNN) are renowned for their ability to extract detailed local fea–tures, while Vision Transformers (ViT) excel at mode–ling global dependencies and the overall structure of visual data. Over the years, artificial neural networks (ANN) have been widely applied in industrial appli–cations, demonstrating their effectiveness in modeling and optimizing complex processes [2]. In the context of container damage detection, leveraging ANN-based models allows for improved predictive accuracy, making automated inspection systems more robust and efficient for real-world deployment.

However, due to the difficulty of obtaining a suitable dataset, we opted to collect and manually annotate the data ourselves. In this paper, we address the binary classification problem of container damage detection—specifically, distinguishing between **damaged** and **normal** exterior and interior panels of containers. Given the dataset's limited size, we designed an efficient and robust model tailored to dataset's limitation.

To achieve this, a lightweight hybrid model is proposed that merges a CNN fine-grained feature extraction with a miniaturized ViT global self-attention. The CNN uncovers localized damage details—like rust spots and small dents—while the ViT captures broader structural deformities, overcoming CNN inherent locality bias. Despite using only 1 million parameters, our model achieves up to 97.2% accuracy and delivers an average inference time of 62 ms per image, making it perfectly suited for real-time, resource-constrained container inspections.

2.1 Convolutional Neural Network (CNN)

The foundation of Convolutional Neural Networks (CNN) was laid by LeCun et al. [3] through the intro–duction of LeNet-5, one of the earliest successful applications of neural networks to image recognition tasks such as handwritten digit classification. CNNs fundamentally operate by stacking multiple stages, each typically consisting of a convolutional layer followed by a non-linear activation function and a pooling layer. Convolutional layers apply local filters to extract spatial features such as edges, textures, and patterns, while

Received: June 2025, Accepted: August 2025
Correspondence to: Dr Shafiqur Rehman
IRC-SES, King Fahd University of Petroleum & Minerals, Saudi Arabia
E-mail: srehman@kfupm.edu.sa
doi: 10.5937/fme2504537K

pooling layers reduce the spatial dimensions, making the representations progressively more abstract and robust to small translations.

This basic architecture enables CNN to learn incre–asingly complex features, from low-level edges to high level object parts, across deeper layers. The introduction of AlexNet [4] in 2012 dramatically accelerated CNN development by demonstrating the effectiveness of deep CNN trained on large datasets like ImageNet using modern GPU. This success popularized CNNs as a backbone for computer vision tasks. Moreover, CNN have been applied to multi domain problems such as forecasting wind power density [5] and seismic signal denoising [6].

2.2 Vision Transformer (ViT)

The Vision Transformer (ViT) [7] marked a significant effort to make transformer applicable to work in image classification. ViT splits images into fixed size patches and processes them as sequences using self-attention mechanisms. This design enables ViT to model global dependencies across an image more effectively than conventional CNN. However, ViT models generally require large-scale datasets and extensive pre-training to perform well. Subsequent work like DeiT [8], improved ViT's data efficiency such as pre-trained CNN, heavy augmentation, and regularization, making them more applicable to smaller datasets.

A similar strategy has been adopted in defect detection applications, where pre-trained ViT models are integrated into transformer-based object detection architectures. For instance, YOLOS-PV [9] leverages a ViT backbone trained on large-scale datasets before feeding extracted features into a transformer encoder for improved defect localization. This approach has de–monstrated strong performance in detecting solar panel defects, reinforcing the viability of ViT in automated visual inspection systems, including container damage detection.

2.3 Container Damage Detection

Recent studies have significantly advanced automated visual inspection methods for shipping containers.

Huang, et. al. [10] in 2024, proposed a Vision Transformer (ViT) model to detect container damage. Their dataset consisted of images categorized into three types of container aging damage—Rust, Distort, and Dent. While the total dataset was reported as 3,000 images, only 1,500 were explicitly mentioned in the training and testing process. The dataset was split in a 9:1 ratio. Their approach aimed to improve automated damage classification by leveraging ViT's capability to capture both local and global image details. The experimental results showed an accuracy of 80.6%, a loss function of 0.724, and a learning rate of 0.001, demonstrating the effectiveness of their method in detecting container aging and damage. However, this level of accuracy may not be enough for industrial application, showing a room for improvement.

Li et al. [11] proposed RP-FCN a fully convolutional network based on ResNeXt50. The authors claim to get 85% compared to another model with normal FCN and FCN with fusion Up-sampling respectively 74% and 78%. However, the authors didn't provide more detail about dataset used in the research. The lack of details makes it difficult to verify the results' s generalizability.

Wang et al. [12] previously applied MobileNetV2 for detecting multiple types of container damages. The authors employed 1543 sample images divided to 9 class: 7 types of damages, container, and surrounding environment. Training set and validation set divided according to 9:1. achieving a verification accuracy of 97.99% after retraining, with significant improvements over initial training performance (training accuracy improved from 86.21% to 95.32%, and training loss reduced from 40.59% to 23.31%). The authors tested the model on-site. The accuracy varied across classes due to uneven number of images. However, the author didn't explain clearly how they attempted to mitigate this problem, which can affect model reliability.

Bahrami et al. [13] implemented and optimized several models: Faster R-CNN, SSD-MobileNet, and SSD InceptionV2 for object detection container dama–ge. The authors introduce anchor box optimization that can adapted during training. This technique improves detection accuracy by more than 5%.

Kuo et al. [14] in 2025 introduced the Cad-Trans–former, a hybrid CNN–Transformer architecture for shipping container defect classification. They utilized random image masking and transformer-based recon–struction alongside CNN-based feature extraction speci–fically from visible patches. This combined approach achieves an average accuracy around 85% across multiple defect categories.

The literature shows significant progress in auto–mating container inspection, with methods of combi–nation of CNN and transformer or complex hybrid models. Many of these studies aim to solve the tough problem of classifying multiple types of damage at once. However, this often creates a trade-off, some models require huge dataset of over 16,000 images to achieve moderate accuracy, while others that seem accurate might have higher latency, making them less ideal for real-time use.

We decided to take different approach by focusing on a foundation first step: creating a fast and highly reliable model that simply determines wheter a con–tainer is "damaged" or "normal". This kind of quick, yes-or-no check is incredibly valuable for initial scree–ning in busy port or crowded depot where speed is cru–cial. To address this, we introduce our lightweight hybrid CNN-ViT architecture, designed to provide a robust and practical solution for this essential first step in automated damage detection.

2. HYBRID CNN–VIT ARCHITECTURE

Combining both local feature extraction and global context modeling, we adopt a two-branch network whose outputs are fused before the final decision. Figure 1 depicts the overall structure. The input to the model is an RGB image with a size of 224 × 224.

2.4 Convolutional Stem

A lightweight four-block CNN serves as our local feature extractor. It is made up of four sequential blocks and each block includes:

$$Conv2D \rightarrow BatchNorm \rightarrow ReLU \rightarrow MaxPool \quad (1)$$

with channel widths for the four sequential blocks are considered as 16, 32, 64, and 128; respectively. After four downsampling stages, spatial resolution is reduced by a factor of 16, yielding a 14 × 14 × 128 feature map. A final Global Pooling collapses this into a single 128-dimensional vector v_{CNN}.

2.5 Vision Transformer Branch

In parallel, a compact ViT branch captures long-range dependencies.

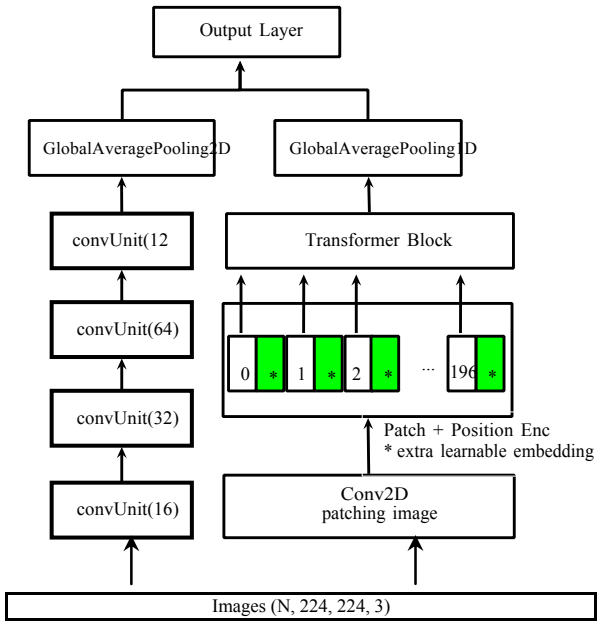

Figure 1. Hybrid CNN–ViT architecture: a lightweight CNN stem extracts fine-grained local features (edges, textures), a compact ViT head captures global context via self-attention, and their fused 256-D feature vector is classified by an MLP for binary damage detection.

Patch Embedding. We use a straightforward method to convert image patches into embeddings by applying a single Conv2D layer. The kernel size and stride are set to the same value as the patch size, so the layer splits the image into non-overlapping patches and projects them directly into the embedding space. This avoids the usual method in ViT, which first slices the image and then uses a linear layer. Our method is inspired by the ViT-Lite model [15], which showed that this approach can reduce complexity and still perform well, especially when there isn't a lot of training data. Unlike other models that use extra convolution or pooling layers before the transformer, we maintained a clean design aligned with the original ViT approach, with fewer built-in assumptions from CNN.

A Conv2D layer (kernel, k = 16, stride, s = 16, filters, f = 128) is applied to the input image, $X \in R^{224\times224\times3}$. The output tensor, $X_p \in R^{14\times14\times128}$, represents 196 patches, each mapped to a-128-dimen–sional vector.

$$X_p = conv2D\left(X; k=16, s=16, f=128\right) \quad (2)$$

We reshape these patches into tokens with dimension, (196, 128).

$$X_{pr} = Reshape\left(X_p\right) \in R^{196\times128} \quad (3)$$

Thus, patches become tokens arranged sequentially, ready for processing by the transformer.

Positional Encoding. Learned embeddings $P \in R^{196\times128}$ are added to each patch vector:

$$Z = X_{pr} + P \quad (4)$$

Transformer Blocks. Stack L = 4 blocks, each perfor–ming Multi Head Self-Attention with set to 2 with residual block then feed forward MLP with residual block.

$$Z' = Z + MHSA\left(LayerNorm\left(Z\right)\right) \quad (5)$$

$$Z'' = Z' + \mathrm{MLP}\left(LayerNorm\left(Z'\right)\right) \quad (6)$$

where the MLP is a two-layer feed-forward network with inner dimension 256, GELU activation, and drop–out 0.1.

Global Pooling. A final GlobalAveragePooling1D re–duces the sequence of 196 tokens to a single 128-dim vector v_{ViT}. Instead of using a special [CLS] token to represent the whole image, we keep things simpler by applying global average pooling [16]. This averages all patch embeddings into a single vector. It's lightweight, doesn't add extra parameters, and works well - espe–cially when training data is limited. While the [CLS] token can sometimes help by learning to focus on im–portant parts of the image, it also makes the model a bit more complex. For a compact and efficient design, global pooling is often the better choice.

2.6 Feature Fusion and Classification

We concatenate the two branch outputs,

$$v = \left[v_{\mathrm{CNN}}; v_{\mathrm{ViT}}\right] \in R^{256} \quad (7)$$

and feed v into a two-layer MLP with sigmoid activation for binary classification. We insert two Dropout layers to mitigate overfitting during training.

The proposed hybrid architecture effectively com–bines the inductive biases and efficiency of a light–weight CNN with the global context modeling of a compact Vision Transformer. By concatenating the fea-ture vectors from both branches before classification, the model captures complementary local and global in–formation in a unified representation, leading to imp–roved robustness and accuracy on small datasets while maintaining low computational cost - ideal for real-time and resource-constrained applications such as container damage classification.

3. EXPERIMENTAL SETTING AND ENVIRONMENT

In this section we discuss dataset preparation, image pre-processing, and experimental setup on Kaggle.

2.7 Dataset preparation

One of the primary challenges in applying computer vision to container damage detection is the limited availability of relevant datasets. To address this, we constructed a custom dataset comprising images of both the exterior and interior panels of shipping containers, which are commonly subject to damage from impact, prolonged use, or environmental exposure. The dataset was collected from real-world shipping container depots near Jakarta Port (Tanjung Priok) between 2023 and 2025. The photos were captured on-site by depot surveyors and third-party independent surveyors under natural daylight with varying lighting conditions. Most images were captured using smartphones or compact digital cameras, later compressed to reduce storage requirements - mimicking the quality constraints of actual field inspections. Both interior and exterior container panels were included, with attention to areas most susceptible to damage. The images were then resized or cropped to get focused on damage areas.

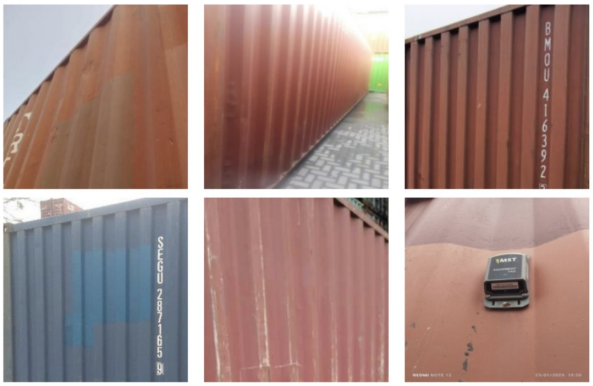

Figure 2. Sample dataset container labeled as *normal*

All images were manually labeled into two classes: **normal** and **damaged**, based on visual signs such as dents, rust, cut, broken, chemical contamination, surface deformation, or holes. The total number of images acquired was 2116 images: 1081 labeled as damage and 1035 labeled as normal. The dataset was then split into three subsets: training (80%), validation (10%), and testing (10%). Annotations were performed by a single annotator, and although care was taken, some subjectivity may remain. No formal inter-annotator agreement process was conducted. To enhance annotation reliability and reduce bias, in the future, multiple annotators will be involved to improve labeling consistency, along with a formal inter-annotator reliability measure like Cohen's Kappa. Expert validation and collaborative labeling will further increase dataset accuracy and quality, ensuring ambiguous cases are addressed.

2.8 Pre-processing Images

Before feeding the images into the network, all images were resized to 224×224 pixels. This convention ensures compatibility with to ImageNet pre-trained models [4, 17]. Since our dataset has limited variability, extensive data augmentation is applied to enhance diversity and better reflect real-world scenarios. There isn't a clear theoretical guideline for choosing the best augmentation techniques to maximize dataset benefits [18], so we focused on approaches that would streng–then generalization and help the model perform well in different conditions. In this project, we implemented a strong data augmentation pipeline during the training images, ensuring diversity while preserving their essential features. The augmentation techniques used included horizontal flipping, rotations, zooming, contrast, and brightness each randomly up to 20% adjustments.

Figure 3. Sample dataset container labeled as *damage*

2.9 Environment Setting

The experiment uses GPU P100 provided by Kaggle, running TensorFlow 2.18 and Python 3.11. Batch size is set to 64. We use Adam [19] as the optimizer with a learning rate of 0.001 and train for 50 epochs. Given the simplicity of our task - binary classification of container damage with a relatively small dataset - we primarily rely on CNN for feature extraction and classification due to its efficiency. To enhance the model's ability to capture long-range dependencies, we integrate ViT, which improves contextual understanding without adding excessive complexity.

We use *ReduceLROnPlateau* to dynamically adjust the learning rate when validation loss plateaus, stabilizing training and reducing overfitting. Additionally, *EarlyStopping* is implemented to halt training when further improvements cease, reducing unnecessary computations and refining model generalization. These techniques collectively optimize the training process, allowing CNN-based architecture to perform more effectively with ViT integration.

4. RESULTS AND DISCUSSION

In this section we present the best performance of our proposed model based on ten independent runs to see its stability and use SHAP to analyze the performance of our model.

Model Performance Evaluation

Our experiments demonstrate that integrating a lightweight Vision Transformer (ViT) with the Base CNN yields superior results for container damage classification. This hybrid model achieved a peak accuracy of 97.2% while maintaining computational efficiency with a modest 1 M parameters and a rapid inference time of

62 ms, as detailed in Table 1. This highlights the model's balance between spatial detail and contextual understanding.

Ablation Study

To evaluate the performance of adding a lightweight ViT branch, we benchmarked four variants: Base CNN alone, MobileNetV2 alone, and each fused with our ViT module. As shown in Table 1, the Base CNN alone achieved an accuracy of 95.8%. Integrating the ViT module boosted its accuracy to 97.2%, while also lowe–ring loss by 36%, with only a slight latency increase of 6 ms.

Table 1. Integrating the ViT branch into our Base CNN not only boosts accuracy by 1.4 points but also lowers validation loss significantly, all with just a 6 ms latency increase.

Model	Params (M)	Acc (%)	Val Loss	Inference Time
Base CNN	0.1	95.8	0.1064	**56 ms**
MobileNetV2	2.3	94.8	0.1321	73 ms
Base CNN + ViT	**1.0**	**97.2**	**0.0679**	62 ms
MobileNetV2 + ViT	3.2	95.3	0.1251	80 ms

Additionally, we further analysed model perfor–mance using precision, recall, and F1-score metrics (Table 2). The Base CNN–ViT model consistently de–monstrated higher precision (96%–98%), recall (96%–98%), and F1-score (97%) compared to Base CNN and MobileNetV2 variants. This enhancement indicates a stronger, more reliable capability in accurately detecting container damage. Particularly notable is the improved recall, essential in practical inspection scenarios where minimizing missed damage detections is critical. In contrast, integrating ViT with MobileNetV2 yielded smaller improvements, highlighting the Base CNN–ViT combination as the most balanced and efficient choice among the evaluated models.

Table 2. F1-scores for the damage and normal classes across all four model variants, including macro and weighted average F1, highlighting the improvement achieved by integrating the ViT branch.

Model	Class	Prec.	Recall	F1-score	Sup.
Base CNN	damage	0.94	0.98	0.96	109
	normal	0.98	0.93	0.96	104
MobileNet V2	damage	0.93	0.97	0.95	109
	normal	0.97	0.92	0.95	104
Base CNN + ViT	damage	0.96	0.98	0.97	109
	normal	0.98	0.96	0.97	104
MobileNet V2 + ViT	damage	0.94	0.97	0.95	109
	normal	0.97	0.93	0.95	104

Error Analysis

The confusion matrix (Figure 4) predominantly occurring near the decision threshold of 0.5. In Figure 5, we isolated six misclassifications and categorized them according to the visual cues present.

Low-Visibility Defects (Figs. 5a–5b): Two damage examples are simply too faint or partially hidden to stand out. In Fig. 5a, a thin rust patch (and overlaid chalk) barely alters the panel's texture, while in Fig. 5b a larger rust streak is broken by a shadow or marking. We initially underestimated how often inspectors' chalk markings confused the model, revealing the need for context-aware attention mechanisms.

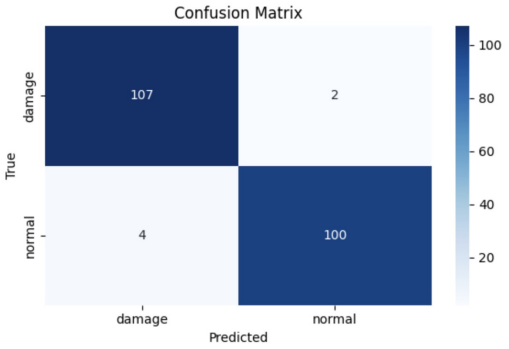

Figure 4. Confusion matrix for the CNN-ViT model: 107/109 damaged and 100/104 normal accurately classified.

High-Contrast Distractors (Figs. 5c–5f): Four normal panels contain bold, reflective, or textured elements that mimic damage. A yellow sign's glare (5c), dirt streaks and weld seams (5d), a sensor box shadow (5e), and thick container lettering (5f) all produce high-intensity or irregular patterns that can be mistaken for corrosion.

For further analysis of model behavior, we conducted a SHAP analysis presented in the Section 5.5.

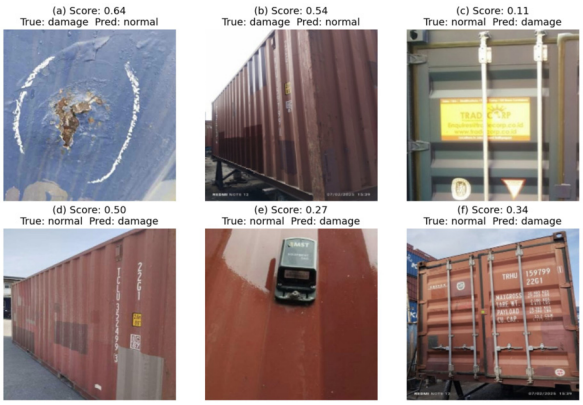

Figure. 5: Error analysis at the 0.5 threshold. (a–b) False negatives occur when corrosion is low-contrast or occluded. (c–f) False positives predominantly occur due to misleading surface artifacts or challenging lighting, which resemble damage patterns. Each panel lists true label, predicted label, and confidence.

Robustness and Stability Analysis

For robustness assessment, we conducted 10 indepen–dent training runs. The average results are summarized in Table 3. The Base CNN-ViT model consistently delivered the highest average accuracy (96.57%±0.83) and lowest loss (0.089±0.015), while maintaining competitive inference latency (64.21ms/sample).

Quantitative Attribution Analysis via SHAP

A comprehensive analysis using explainable AI with SHAP value was conducted to quantify the contribu–tions of the CNN and ViT branches in our hybrid model for container damage classification. The SHAP analysis

employed Kernel SHAP, which approximates Shapley values to explain the model's predictions. We quanti–tatively analyzed the contributions of CNN and ViT branches using mean absolute SHAP values at the final block level, computed across three independent runs (500 background samples, 40 test images per run). These settings were chosen to balance computational efficiency with analysis stability, ensuring reliable SHAP value estimates while keeping processing costs manageable. As shown in Figure 6, the CNN block exhibits higher attri–bution values (0.34±0.02) compared to ViT (0.27±0.015), indicating that CNN features have a stronger and more consistent influence on the model's output.

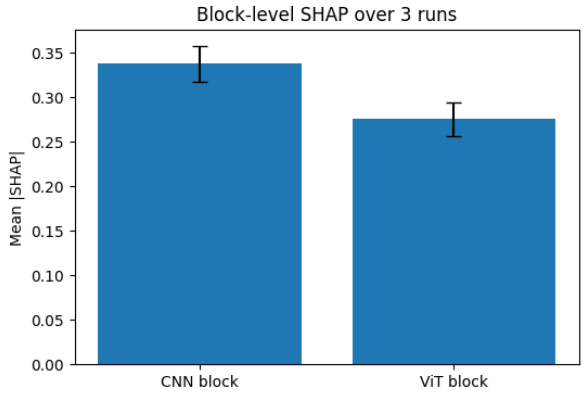

Figure 6. Mean absolute SHAP values for the CNN and ViT branches at the final block level, averaged over three runs. Error bars denote one standard deviation.

In our experiments, both Figures 7 and 8 analyze the same two test images - a heavily scratched-dented panel and a localized rust spot - but at different levels of granularity. Figure 7 compares the CNN's Grad-CAM and the ViT's attention-rollout. On the dented panel (top row), Grad-CAM predominantly highlights the conta–iner's longitudinal ribs and only sparsely overlaps the true dent, whereas the ViT rollout spans the broader scratched region, capturing context but lacking boundary precision. On the rust defect (bottom row), both methods fail to focus on the small corrosion: Grad-CAM is misled by the inspector's white chalk marks, and the ViT rollout produces a diffuse, unfocused heatmap.

Table 3. Ten-run average performance: the CNN-ViT model demonstrates superior stability and accuracy with reasonable latency.

Model	Avg. Acc (%)	Avg. Loss	Latency (ms)
Base CNN	95.82 ± 0.75	0.115 ± 0.020	58.08
MobileNetV2	94.74 ± 1.10	0.142 ± 0.025	73.21
Base CNN - ViT	96.57 ± 0.83	0.089 ± 0.015	64.21
MobileNetV2 - ViT	95.31 ± 0.95	0.108 ± 0.018	80.05

Figure 8 presents the block-level SHAP analysis conducted with our complete hybrid model. Input images for testing were segmented into 100 superpixels using the SLIC algorithm, and SHAP values were computed based on the model predictions against a black baseline. For visualization, we focused on the top 20% most impactful superpixels. This figure provided more granular insights.

On the heavy scratched-dented panel (left), SHAP accurately identified heavily impacted superpixels, validating CNN's effectiveness in capturing pronounced local damage. In contrast, for the localized rust defect (right), SHAP correctly highlighted the rust spot in red, indicating accurate recognition of the true damage.

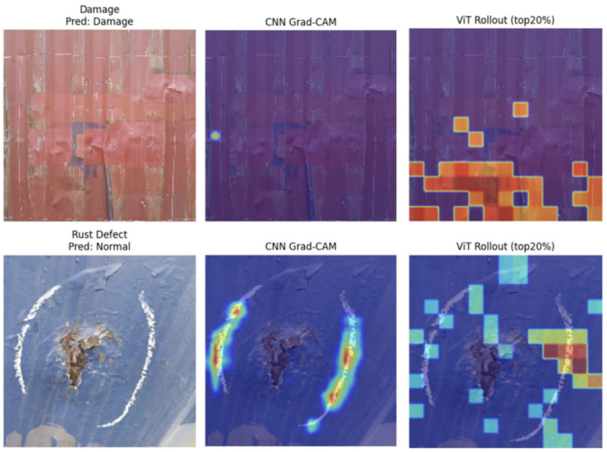

Figure 7. Spatial visualization of model attention. Each row: (left) input image, (middle) CNN Grad-CAM overlay, (right) ViT attention-rollout (top 20%). Top row: correct damage detection. Bottom row: rust defect misclassified as normal.

However, it was evident that the CNN misin–terpreted inspector-made white markings as significant damage indicators as highlighted in adjacent regions. Simultaneously, intact regions appeared prominently in blue, strongly contributing to the incorrect normal classification. This conflict highlights the model's susceptibility to false signals caused by visual artifacts.

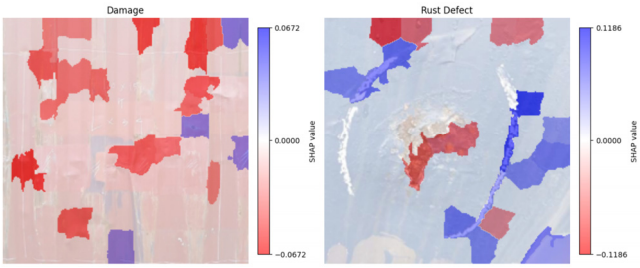

Figure 8. Block level SHAP explanations on two damage examples. Left: a heavy scratched-dented container panel with SHAP highlighting the indented superpixels in red. Right: a localized rust with red shading over the corroded area and blue shading on intact panel sections. Only the top 20 % most impactful superpixels are shown; blue regions contribute toward the "normal" class, red regions toward the "damage" class.

In summary, while the CNN block significantly contributes to accurate classifications by effectively capturing local damage features, it remains susceptible to misinterpretations caused by non-damage markings, such as inspection signs. This highlights the need for enhanced feature disambiguation strategies, such as guided attention mechanisms or refined fusion logic.

In misclassifications, CNN's Grad-CAM activates on high-contrast markings (e.g., chalk) rather than the central rust defect, suggesting overfitting to irrelevant details. The ViT rollout, however, shows diffuse attention without clear focus on the defect, highlighting its difficulty in identifying subtle localized features.

Table 4: Comparison of recent container damage detection methods showing problem type, dataset size, model architecture, accuracy, number of parameters, and latency. The proposed method is included for reference.

Aspect	[10]	[11]	[12]	[13]	[14]	Our work
Domain Problem	Multi-Class Classification	Object Detection	Multi-Class Classification	Object Detection	Multi-Class Classification	Binary Classification
Dataset	3,000 images, 3 classes	Not explicitly detailed	1,543 images; 9 classes	Not explicitly detailed	16,000 images; 8 classes	2,116 images; 2 classes
Models	Vision Transformer (ViT)	RP-FCN (ResNeXt50)	MobileNetV2	Faster R-CNN, MobileNet, InceptionV2 optimized with box anchor	Cad-transformer	CNN–ViT
Results (Accuracy)	80.6%	85%	95.32% (train), 97.99% (validation)	66%	~85% accuracy (averaged)	97.2%
Key Techniques	Transformer-based self-attention	Fusion Upsampling, Pyramid Pooling, ResNeXt50, FCN structure	Transfer Learning, Image Augmentation, Weak Supervision (WESPE enhancement)	Anchor Box Optimization	CNN + Transformer MAE with CFE-VP module	Compact CNN–ViT integration
Class Detected	Rust, Distortion, Dent	Door damage, deformation, distortion, concave, convex, hole, scratch, number losses	Surrounding, damage, hole, rusty, bent, dent, open, collapse, normal	Corrosion	Broken, Cut, Dent, Hole, Rust, Distorted, Normal, Others	Damage, Normal

This behavior highlights the complementary roles of both models: while CNN excels in precise localization though prone to misinterpreting irrelevant details), ViT provides contextual insights but struggles with subtle defects.

These observations emphasize the advantage of integrating CNN's detailed spatial sensitivity with ViT's broader contextual understanding, particularly useful in scenarios with ambiguous visual cues. Future enhan–cements might explore methods to better harmonize these complementary strengths. A summary of this work alongside related studies is presented in Table 4.

5. CONCLUSIONS

This work proposes a compact hybrid architecture that integrates a lightweight convolutional stem with a vision ViT head to effectively capture both local and global features for container damage classification.

Our primary contribution to the state-of-the-art is the demonstration that a minimalist hybrid model can outperform more complex architectures on a specialized industrial vision task, particularly under the common constraint of a small dataset. Through extensive experiments on a small, real-world dataset, the model achieved high accuracy (up to 97.2%) while maintaining low computational cost with only 1M parameters and 62 ms inference time. Our lightweight CNN-ViT hybrid demonstrates effectiveness as a practical, and scalable solution for real-time inspections. Specifically, this performance perfectly suited for deployment at the automated gates of a port or depot. It can function as an efficient screening tool, performing a quick "yes-or-no" damage assessment as containers pass through. This allows operators to immediately pull aside only containers that need a more detail secondary inspection, keeping traffic moving smoothly.

Despite its promising results, our study is constrained by the limited dataset and diversity, potential annotation inconsistencies due to single-annotator labeling, and the model's limitation to binary damage detection. Future work aims to address these limitations by expanding and diversifying the image collection, incorporating multi-annotator consensus for more reliable labels, and extending the framework to multiclass damage typologies.

REFERENCES

[1] Z. Németh and P. Böröcz, "Impact shock events in multimodal container transshipment for packaging testing," FME Transactions, vol. 51, no. 2, pp. 161–168, 2023. DOI: 10.5937/fme2302161N.

[2] M. J. Madić, M. R. Radovanović, "Optimal selec–tion of ANN training and architectural parameters using Taguchi method: A case study," FME Transactions, vol. 39, no. 2, pp. 79–86, 2011. DOI: 10.5937/fme1102079M.

[3] Y. LeCun et al., "Backpropagation applied to hand–written zip code recognition," Neural Computation, vol. 1, no. 4, pp. 541–551, 1989. DOI: 10.1162/neco.1989.1.4.541.

[4] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, no. 6, pp. 84–90, 2017. DOI: 10.1145/3065386.

[5] D. Gupta et al., "Short-term prediction of wind power density using convolutional LSTM network," FME Transactions, vol. 49, no. 3, pp. 653–660, 2021. DOI: 10.5937/fme2103653G.

[6] N. Iqbal, "DeepSeg: Deep segmental denoising neural network for seismic data," IEEE

Transactions on Neural Networks and Learning Systems, vol. 34, no. 7, pp. 3397–3404, 2023. DOI: 10.1109/TNNLS.2022.3205421.

[7] A. Dosovitskiy et al., “An image is worth 16×16 words: Transformers for image recognition at scale,” arXiv preprint, arXiv:2010.11929, 2021. DOI: 10.48550/arXiv.2010.11929.

[8] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, “Training data-efficient image transformers: Distillation through attention,” arXiv preprint, arXiv:2012.12877, 2020.

[9] H. Tella, M. A. Mohandes, B. Liu, A. Al-Shaikhi, S. Rehman, “A novel cost-function for transformer-based YOLO algorithm to detect photovoltaic panel defects,” FME Transactions, vol. 52, no. 4, pp. 639–646, 2024. DOI: 10.5937/fme2404639T.

[10] X.-R. Huang, G.-Z. Huang, S.-Y. Kuo, L.-B. Chen, “A vision transformer model-based shipping conta–iner damage inspection scheme,” in Proc. 2024 Int. Conf. on Consumer Electronics – Taiwan (ICCE–Taiwan), pp. 817–818, 2024. DOI: 10.1109/ICCE-Taiwan62264.2024.10674300.

[11] X. Li, X. Huang, Q. Liu, “Container damage iden–tification based on RP-FCN,” in Proc. 39th Chinese Control Conf. (CCC), pp. 7031–7034, 2020. DOI: 10.23919/CCC50068.2020.9189392.

[12] Z. Wang, J. Gao, Q. Zeng, Y. Sun, “Multitype damage detection of container using CNN based on transfer learning,” Mathematical Problems in Engineering, vol. 2021, pp. 1–12, 2021. DOI: 10.1155/2021/5395494.

[13] Z. Bahrami, R. Zhang, R. Rayhana, T. Wang, Z. Liu, “Optimized deep neural network architectures with anchor box optimization for shipping container corrosion inspection,” in Proc. 2020 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1328–1333, 2020. DOI: 10.1109/SSCI 47803.2020.9308472.

[14] S.-Y. Kuo et al., “Cad-transformer: A CNN-transformer hybrid framework for automatic appearance defect classification of shipping containers,” IEEE Transactions on Instrumentation and Measurement, vol. 74, 2025. DOI: 10.1109/TIM.2025.3548214.

[15] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, H. Shi, “Escaping the big data paradigm with compact transformers,” arXiv preprint, arXiv:2104.05704, 2021.

[16] L. Beyer, X. Zhai, A. Kolesnikov, "Better plain ViT baselines for ImageNet-1k," arXiv preprint, arXiv:2205.01580, 2022.

[17] K. He, X. Zhang, S. Ren, J. Sun, “Deep residual learning for image recognition,” arXiv preprint, arXiv:1512.03385, 2015.

[18] S. Yang, W. Xiao, M. Zhang, S. Guo, J. Zhao, F. Shen, “Image data augmentation for deep learning: A survey,” arXiv preprint, arXiv:2204.08610, 2023.

[19] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint, arXiv: 1412.6980, 20

ЛАГАНИ ХИБРИДНИ CNN-VISION ТРАНСФОРМАТОР ЗА АУТОМАТИЗОВАНО ОТКРИВАЊЕ ОШТЕЋЕЊА КОНТЕЈНЕРА У РЕАЛНОМ ВРЕМЕНУ

А. Курниаван, М.А. Мохандес, Н. Икбал, С. Рехман, Х.Х. Нуха

Ручне инспекције контејнера често доводе до недос–ледности и неефикасности, што може пореметити ланце снабдевања и повећати оперативне трошкове. Временски захтевна природа ручних провера чини аутоматизацију привлачном алтернативом. Овај рад представља лаган хибридни модел који комбинује конволуционе неуронске мреже (CNN) и визуелне трансформаторе (ViT), посебно дизајниране за аутоматизовану класификацију оштећења контеј–нера. CNN издваја фино зрнасте локалне карак–теристике, док ViT моделира глобалне структурне обрасце, превазилазећи ограничења чисто конво–луционих архитектура. Процењујемо четири вари–јанте модела на скупу података од 2.116 слика, прикупљених из контејнерских депоа у близини луке Џакарта. Наш предложени CNN-ViT хибридни модел добро се генерализовао са овим скупом пода–така и постиже тачност од 96,57% ± 0,83, губитак бинарне унакрсне ентропије од 0,089 ± 0,015 и латенцију инференције од 64,21 ± 1,47 ms, дости–жући врхунац од 97,2% тачности и латенције од 62 ms у најбољем испитивању са само 1 милион параметара. У поређењу са MobileNetV2, наш прис–туп побољшава тачност класификације за око 1% уз истовремено смањење времена закључивања за при–ближно 9 ms, демонстрирајући његову ефикасност за аутоматизовану инспекцију контејнера у реалном времену у окружењима са ограниченим ресурсима.