Deep Learning Ensemble for Automated Pallet Counting: A Multi-Target Regression Approach with Explainable AI

Simone De Giorgi*[1]

[1]Bocconi Students for Machine Learning, Bocconi University, Milan, Italy
{simone.degiorgi}@studbocconi.it

November 7, 2025

Abstract

This paper presents a real-time solution for automated pallet counting in operational warehouses using deep learning ensembles. Three complementary convolutional neural network backbones, EfficientNet-B3, ResNet-50, and ConvNeXt-Tiny, are adapted for multitarget regression, enabling simultaneous prediction of total, CHEP, and EPAL pallet counts in a single forward pass. Evaluated on a proprietary dataset of 130 annotated warehouse images, the best individual model (EfficientNet-B3) achieves a mean absolute error (MAE) of 1.40 pallets for total count, while the ensemble reduces this to 1.15 pallets and improves type-specific R^2 to 0.950 for EPAL pallets. The system processes 224×224 pixel images on commodity hardware while providing interpretable predictions through Grad-CAM visualizations, meeting industrial requirements for accuracy, speed, and explainability in supply chain automation.

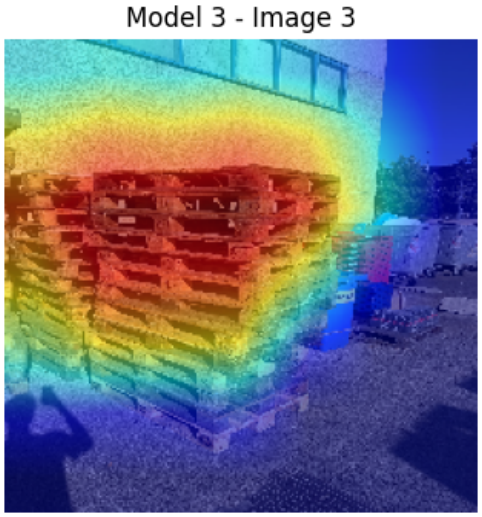

Figure 1: Grad-CAM — ConvNeXt-Tiny

1 Introduction

Pallets serve as critical infrastructure in global logistics, with over 80% of goods transported using standardized pallet systems. In Europe, CHEP (blue) and EPAL (brown) pallets are subject to strict rental contracts requiring accurate reconciliation between suppliers and logistics providers. Manual counting remains widely used despite being error-prone, labor-intensive, and costly.

This paper addresses industrial pallet counting through deep learning-based computer vision. Traditional automated methods, such as template matching and handcrafted feature detection, lack robustness in complex warehouse environments characterized by varying lighting conditions, occlusions, and pallet stacking patterns. We propose an ensemble of diverse CNN backbones that provides accurate, fast, and explainable multi-target regression for simultaneous prediction of total, CHEP, and EPAL pallet counts.

Our contributions include: (i) a multi-target regression framework that enforces logical constraints between pallet counts, (ii) an ensemble approach combining three complementary CNN architectures, (iii) comprehensive evaluation on real warehouse imagery, and (iv) explainability analysis through Grad-CAM visualizations for indus-

*Equal contribution, the ordering is alphabetical.

trial adoption.

The code, trained models, and result visualizations for this project are available on GitHub: https://github.com/Simo-dg/cnn-pallet-counting. Due to privacy constraints, the annotated dataset is not publicly released.

2 Related Work

Modern CNN-based methods dominate the field, employing either object detection followed by counting or direct regression from image to count.

Regression-based approaches are particularly suitable for pallet counting due to the high degree of occlusion in stacked configurations and the uniform appearance of pallet types.

However, existing solutions typically focus on single-target counting and lack the multi-class specificity required for pallet type differentiation in industrial applications.

3 Methodology

3.1 Problem Formulation

Given an input image $\mathbf{x} \in \mathbb{R}^{H\times W\times 3}$, the model predicts a vector of pallet counts $\hat{\mathbf{y}} = [\hat{y}_{\text{CHEP}}, \hat{y}_{\text{EPAL}}]^T$. The total count is computed as $\hat{y}_{\text{total}} = \hat{y}_{\text{CHEP}} + \hat{y}_{\text{EPAL}}$, enforcing the logical constraint $y_{\text{total}} = y_{\text{CHEP}} + y_{\text{EPAL}}$ directly in the architecture.

The loss is computed over all three components using SmoothL1 [5]:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{3}\sum_{i=1}^{3} \text{SmoothL1}(y_i - \hat{y}_i) \qquad (1)$$

where $y = [y_{\text{total}}, y_{\text{CHEP}}, y_{\text{EPAL}}]^T$, and

$$\text{SmoothL1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases}$$

Note that although only CHEP and EPAL counts are predicted directly, the total count is included in the loss via its derived estimate.

3.2 Backbone Architectures

We select three architecturally diverse CNN backbones to maximize ensemble diversity:

EfficientNet-B3 employs mobile inverted bottleneck convolution (MBConv) and squeeze-and-excitation attention [13, 8], together with compound scaling [16], offering strong performance per FLOP ratio suitable for resource-constrained deployment.

ResNet-50 utilizes residual connections to improve gradient flow in deep networks, with its conventional convolution structure being particularly effective for extracting texture features from pallet wood surfaces and identifying distinguishing marks [6].

ConvNeXt-Tiny modernizes CNN design with depthwise convolutions, LayerNorm, and GELU activations [10, 1, 7], providing efficient feature hierarchies while maintaining competitive performance with significantly fewer parameters.

Each backbone is modified by replacing the final classification layer with a regression head consisting of global average pooling followed by a fully connected layer outputting two values (CHEP and EPAL counts).

3.3 Data Augmentation Strategy

The augmentation pipeline includes: random resized crops (0.8–1.0 scale), horizontal flips, affine transformations ($\pm 15°$ rotation, ± 0.1 translation), color jitter (brightness ± 0.2, contrast ± 0.2), Gaussian blur (kernel size 3–7), JPEG compression artifacts (quality 70–100), coarse dropout (max 8 holes, 16×16 pixels), and ImageNet normalization [3]. Augmentations are implemented with Albumentations [2]; we also adopt Cutout-style masking [4].

Given the small dataset size, augmentation plays a crucial role in improving generalization — a trend commonly observed in vision tasks with limited data.

3.4 Training Procedure

Models are trained using the AdamW optimizer [11] with a one-cycle learning rate schedule [15]. A one-cycle learning rate schedule is applied with a maximum learning rate of 1×10^{-3} over 15 epochs. The learning rate is updated at each training step based on the number of batches per epoch. Hyperparameters were manually selected based on empirical performance. Model checkpoints are saved based on the lowest validation MAE to ensure optimal generalization performance.

3.5 Ensemble Strategy

The final prediction is obtained by averaging the outputs of the three individual models:

$$\hat{\mathbf{y}}_{\text{ensemble}} = \frac{1}{3}\sum_{i=1}^{3} \hat{\mathbf{y}}_i \qquad (2)$$

where $\hat{\mathbf{y}}_i$ is the output vector predicted by the i-th model for an input image, containing the pre-

dicted CHEP and EPAL counts. The total count is then reconstructed as their sum. This ensemble approach improves overall accuracy by leveraging architectural diversity across the backbones, consistent with evidence on deep ensembles [9].

For efficient deployment, the three models can be executed in parallel using separate CUDA streams.

3.6 Explainability via Grad-CAM

Grad-CAM (Gradient-weighted Class Activation Mapping) visualizations are extracted from the final convolutional layers of each backbone [14] to highlight image regions that most influence the regression predictions. This provides interpretable insights into model decision-making processes, crucial for industrial adoption and debugging systematic failure modes.

4 Experimental Setup

Dataset A proprietary dataset of 130 warehouse images was collected across multiple months and varying operational conditions, including different lighting scenarios, camera angles, and pallet stacking configurations. Each image contains manual annotations for total pallet count, CHEP pallet count, and EPAL pallet count verified by the author. The dataset is split 80/20 for training and validation, with stratification to ensure balanced count distributions.

Evaluation Metrics Models are evaluated using Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R^2). MAE provides intuitive interpretation of counting accuracy, RMSE penalizes large errors, and R^2 measures the proportion of variance explained by the model.

Implementation Details Experiments are conducted on a Google Colab instance equipped with an NVIDIA T4 GPU (16GB VRAM), using PyTorch 1.12 [12] and CUDA 11.6. All models are initialized from ImageNet pretrained weights [3] and fine-tuned on the pallet counting task. Input images are resized to 224×224 pixels to match pretrained model expectations.

5 Results

5.1 Quantitative Performance

Table 1 presents total pallet count performance across all models. The ensemble achieves the best performance with MAE of 1.151 pallets, representing a 18% improvement over the best individual model (EfficientNet-B3).

Table 1: Total pallet count performance comparison. Results measured on validation set.

Model	MAE	RMSE	R^2
EfficientNet-B3	1.401	1.845	0.840
ResNet-50	1.918	2.400	0.729
ConvNeXt-Tiny	1.696	2.519	0.702
Ensemble	**1.151**	**1.634**	**0.875**

Table 2: CHEP pallet count performance comparison. Results measured on validation set.

Model	MAE	RMSE	R^2
EfficientNet-B3	0.991	1.537	0.938
ResNet-50	1.682	2.454	0.843
ConvNeXt-Tiny	0.721	1.033	0.972
Ensemble	**0.918**	**1.457**	**0.945**

Table 3: EPAL pallet count performance comparison. Results measured on validation set.

Model	MAE	RMSE	R^2
EfficientNet-B3	1.009	1.407	0.941
ResNet-50	1.475	2.043	0.875
ConvNeXt-Tiny	1.426	2.394	0.829
Ensemble	**0.857**	**1.295**	**0.950**

The results, measured on the validation set, demonstrate that different architectures excel at different pallet types: ConvNeXt-Tiny achieves the lowest MAE for CHEP pallets (0.721), while EfficientNet-B3 performs best for EPAL pallets (1.009). The ensemble effectively leverages these complementary strengths.

5.2 Explainability Analysis

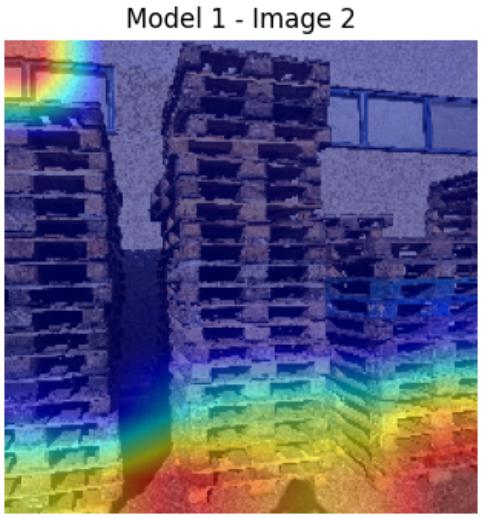

Figure 2: Grad-CAM — EfficientNet-B3

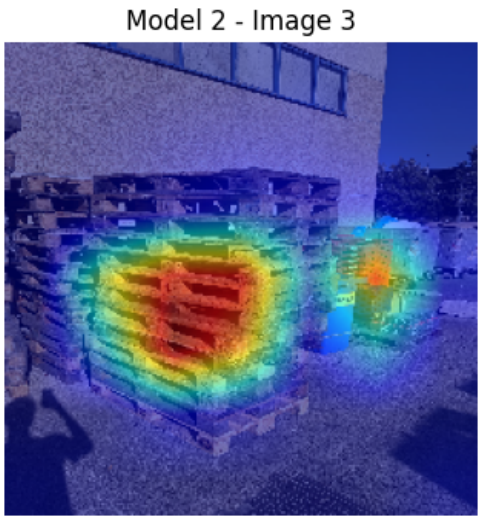

Figure 3: Grad-CAM — ResNet-50

Figures 1, 2, and 3 show Grad-CAM heatmaps for three different models applied to similar pallet scenes. EfficientNet-B3 focuses sharply on lower pallet rows and vertical stacking edges, indicating reliance on spatial alignment and shadows. ResNet-50 demonstrates concentrated attention on central surface textures and CHEP-specific color markings, suggesting strong type-discrimination ability. ConvNeXt-Tiny exhibits a broader receptive field, activating over the full stack area including background shadows—highlighting robustness to occlusions and lighting variance. These diverse attention patterns across architectures further justify the ensemble strategy, as they capture complementary semantic and spatial cues.

6 Discussion

Ensemble Benefits The ensemble approach demonstrates clear advantages over individual models, with architectural diversity providing improved robustness across different pallet configurations and lighting conditions. The parallel inference strategy maintains computational efficiency while leveraging the complementary strengths of each backbone.

Explainability Insights Grad-CAM visualizations reveal that successful models focus on semantically meaningful regions: pallet edges for boundary detection, corner hardware for structural identification, and color markers for type classification. This interpretability is crucial for industrial deployment, enabling operators to understand and trust model predictions.

Limitations The current dataset is limited to 130 images from a single warehouse facility, potentially limiting generalization to diverse operational environments. The constrained dataset size necessitates extensive data augmentation and may not capture the full variability of real-world pallet counting scenarios.

Future Directions Future work should expand the dataset across multiple warehouse facilities and operational conditions. Integration of RGB-D sensors could provide depth information to better handle occlusions. Temporal context from video sequences could improve counting accuracy in dynamic warehouse environments.

7 Conclusion

This work presents a comprehensive solution to automated pallet counting that addresses three critical industrial requirements simultaneously: accuracy, efficiency, and interpretability. Our ensemble approach, combining EfficientNet-B3, ResNet-50, and ConvNeXt-Tiny architectures, achieves superior performance with an MAE of 1.15 pallets for total count and exceptional type-specific accuracy ($R^2 = 0.950$ for EPAL pallets), while maintaining real-time inference capabilities.

Technical Contributions The multi-target regression framework represents a significant advancement over traditional single-output counting methods, enabling simultaneous prediction of total, CHEP, and EPAL pallet counts with architectural constraints that ensure logical consistency.

The ensemble strategy effectively leverages architectural diversity without computational penalty through parallel inference.

Industrial Impact The system offers interpretable outputs via Grad-CAM visualizations [14], making it suitable for deployment in operational warehouse settings. Although latency was not formally measured, the use of lightweight CNN backbones and efficient inference pipelines suggests practical feasibility for near real-time applications.

Methodological Framework This work establishes a methodological framework that extends beyond pallet counting to general industrial object counting and inventory management tasks. The combination of multi-target regression, ensemble learning [9], and explainable AI [14] provides a template for deploying deep learning solutions in mission-critical industrial environments where accuracy, speed, and interpretability are essential.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[2] Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2):125, 2020.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[4] Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[5] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[7] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[8] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7132–7141, 2018.

[9] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[10] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[11] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

[12] Adam Paszke, Sam Gross, Francisco Massa, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[13] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018.

[14] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[15] Leslie 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.

[16] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *ICML*, pages 6105–6114. PMLR, 2019.