Original Article

Optimization of the sheet metal inspection process using artificial intelligence

Erik de Souza Couto [1]*
Luis Fernando de Almeida [2]
Wesley Rossi Pimenta [1]
Daniel Costa Trocades [1]
Regis Fonseca de Oliveira [1]
Yuri Lucas de Souza Paropat [1]
Maycon Lacerda de Oliveira [1]
Silvio de Carvalho Sabença [1]

Abstract

This study details the implementation of a computer vision system for automating quality control in the steel industry, with a focus on detecting scratches on steel plates at the CSN cold rolling mill. The objective was to validate the feasibility of an artificial intelligence model as a replacement for manual inspection, which is subjective and prone to error. The methodology involved fine-tuning a model using a custom dataset collected directly from the production process. the system's viability as a proof of concept, achieving a mAP@50 of 0.574, an F1-Score of 0.584, and an inference time of just 3.2 ms per image, indicating the possibility for real-time application. It is concluded that the solution is technically feasible and efficient, indicating potential contributions to quality control accuracy, reduced operational costs, and the generation of data for continuous process optimization.

Keywords: Computer vision; Defect detection; Steel industry; Artificial intelligence.

1 Introduction

The integration of computer vision systems into industrial processes has become increasingly essential in recent years, particularly in the domain of automated quality control [1]. As industries such as steel manufacturing strive for greater efficiency, consistency, and precision in their production lines, the demand for advanced visual inspection systems has grown substantially.

A computer vision system utilizing the OpenCV library for low-cost inspection and defect detection on the surface of rolled steel sheets is presented in [2], where the raw material consists of a metal sheet coil that is uncoiled during the process execution. The experiments were conducted on images containing six different types of surface defects, achieving an 80% detection rate of defects.

Another low-cost approach is proposed in [3]. The authors present an automatic inspection system based on computer vision, low-cost hardware, and Python programming, utilizing open-source libraries. The approach demonstrated promising results, achieving 100% effectiveness in defect detection, with a false detection rate of 21%.

Boff [4] presents a computer vision system to verify whether the dimensions of capacitors in automatic welding equipment comply with the specified parameters. The evaluation was based on the analysis of 500 capacitor images, achieving a reasonable accuracy rate.

The cold rolling sector faces unique challenges in quality control, especially in identifying and classifying surface defects [5]. Manual inspection, traditionally performed by operators, is time-consuming, subjective, and prone to human error caused by fatigue and inconsistency. Such shortcomings can result in production losses and compromised product quality. In this context, Artificial Intelligence (AI) emerges as a promising solution.

Computer vision, combined with Deep Learning techniques, can be applied to automate the quality inspection of automobile floor panels [6]. In this case, the results are analyzed by comparing the manual inspection method with the automated inspection method, achieving an accuracy greater than 95%.

Fernandes [7] presents a Convolutional Neural Network for performing quality inspection tasks, where it is necessary to load a set of known images from different classes, train the model, and then perform the inspection task using real photos and execute inference on the trained model.

[1]*Companhia Siderúrgica Nacional, Volta Redonda, RJ, Brasil.*
[2]Programa de Mestrado Profissional em Engenharia Mecânica, Universidade de Taubaté, SP, Brasil.

**Corresponding author: erikinf@gmail.com*
E-mails: luis.almeida@unitau.br; engwesleyrossi@gmail.com; daniel.trocades@csn.com.br; regis.oliveira@csn.com.br; yuri.paropat.yp1@csn.com.br; maycon.oliveira@csn.com.br; silvio.cs@csn.com.br

The results showed that, even with a small dataset and training performed over a limited number of epochs, it was possible to achieve excellent results.

Deep Learning and Computer Vision are also applied to enhance defect detection tasks in machines used in the production of seatbelts [8]. A case study was conducted at the Manaus Industrial Pole. The proposed approach utilizes computer vision systems to process image data, developed using the Python programming language and the PySimpleGUI library. The evaluation of the trained model was conducted based on loss and accuracy metrics on the test dataset, achieving values of 0% and 100%, respectively.

Therefore, recent advances in deep learning, notably in the field of real-time object detection, have created new opportunities for overcoming these challenges. This article discusses the implementation of a computer vision system for automated quality control in the cold rolling process, with a focus on detecting surface scratches on steel sheets. This study explores the practical implementation of a state-of-the-art model for automated scratch detection. Transfer learning [9] is central to our approach, allowing a pre-trained model on a large dataset to be fine-tuned for this specific industrial task using a custom dataset collected directly from the production process.

The main objectives of this research are:

- Implement a scratch detection system using the model in a real production environment.
- Assess the technical feasibility and practical implications of deploying such a system for quality control in cold rolling.
- Evaluate model performance in terms of accuracy, efficiency, and its potential to generate actionable insights.

While many studies focus on comparing the performance of multiple models in generic scenarios, this work offers a distinctive contribution by presenting an in-depth analysis of the practical application of a single model. It examines the challenges and advantages of integrating an AI-based solution into an industrial environment, serving as a concrete case study of automation in the steel industry.

The successful implementation of an automated quality control system carries significant implications for the industry. By reducing dependence on manual inspection, such a system can increase productivity, enhance consistency in defect detection, and promote more efficient resource allocation. Furthermore, the capability to detect defects quickly and accurately enables a more agile response to root causes, thereby minimizing quality deviations and reducing operational costs. Consequently, this research not only contributes to the body of knowledge in computer vision and industrial technology but also provides insights into the adoption of Industry 4.0 solutions within the steel sector [10].

An essential aspect of this research is the proposal to validate the feasibility of an artificial intelligence model for defect detection using restrictive infrastructure, including conventional surveillance cameras and the absence of controlled lighting, while considering an implementation scenario with minimal investment.

2 Development

The study focuses on the implementation and evaluation of an object detection model for automated quality control in the steel industry, with a particular emphasis on detecting and classifying surface scratches on steel sheets. The model's capabilities are explored through the application of transfer learning techniques, which aim to optimize its performance for the specific industrial context. The research methodology is structured into four main components: dataset preparation and preprocessing, model architecture and implementation, training strategy, and performance evaluation.

2.1 Dataset and preprocessing

For this study, we used a custom dataset collected directly from the production process of Cold Strip Mill 1 at CSN. The dataset was organized with 540 images for training and 49 images for validation. It contains a single defect class of interest: "Scratch." To standardize the model input and enhance its robustness, the following steps were applied:

- Resizing: All images were resized to 640 × 640 pixels.
- Data Augmentation: During training, data augmentation techniques were applied, including horizontal flipping, rotation between -8° and +8°, horizontal and vertical shearing of ±10°, and blur up to 0.8 px.

The dataset construction followed a two-stage supervised annotation protocol. In the first stage, a quality specialist in the cold rolling process monitored the camera image stream in real time. Upon visually identifying scratches, the specialist recorded the corresponding timestamp for subsequent image collection. This manual process resulted in approximately 50 initial images containing confirmed defects.

In the second stage, these images were used to train a preliminary model with reduced accuracy. This model was then employed as an automatic pre-selection tool for defect candidates, generating detections that included both true positives and false positives.

The pre-selected images were subsequently reviewed and validated by the specialist, who manually annotated the bounding boxes using the Roboflow software.

This bootstrapping approach significantly accelerated the dataset construction process, reducing the waiting time for spontaneous defect occurrences during production.

All final annotations were validated by a professional with experience in the cold rolling process, ensuring the reliability of the class labels.

2.2 Dataset analysis

Figure 1 presents the analysis of the bounding box characteristics in the dataset, containing 2,175 annotations distributed across 522 images. The distribution of the aspect ratio (Figure 1a) shows that scratch-type defects predominantly exhibit an elongated shape along the horizontal axis, with a mean aspect ratio of 2.97 and a median of 2.64. Approximately 93% of the annotations have a width greater than their height, which is consistent with the linear nature of scratches along the rolling direction. The spatial distribution of the bounding box centers (Figure 1b) indicates that defects occur across the entire image surface; however, more pronounced concentrations are observed in certain regions, possibly associated with characteristics of the rolling process, such as roll geometry or non-uniform stresses in the sheet. The scatter plot of the bounding box dimensions (Figure 1c) confirms the predominance of defects with a width greater than height (points above the diagonal). In contrast, the area distribution (Figure 1d) shows that most defects are minor in size, with a mean normalized area of 0.0028.

2.3 Model architecture and transfer learning

In this study, we implemented a computer vision model. The model architecture comprises 181 layers and a total of 2,590,035 parameters, optimized to strike a balance between speed and accuracy.

We employed a training methodology to ensure model optimization. The main aspects of our training process were:

- Optimizer: The optimization model used in this work was AdamW. The main innovation of AdamW compared to the traditional Adam optimizer lies in its handling of weight decay. Instead of incorporating weight decay as part of the gradient (L2 regularization), AdamW decouples it, applying it directly to the model weights. This results in more effective and stable regularization [11].

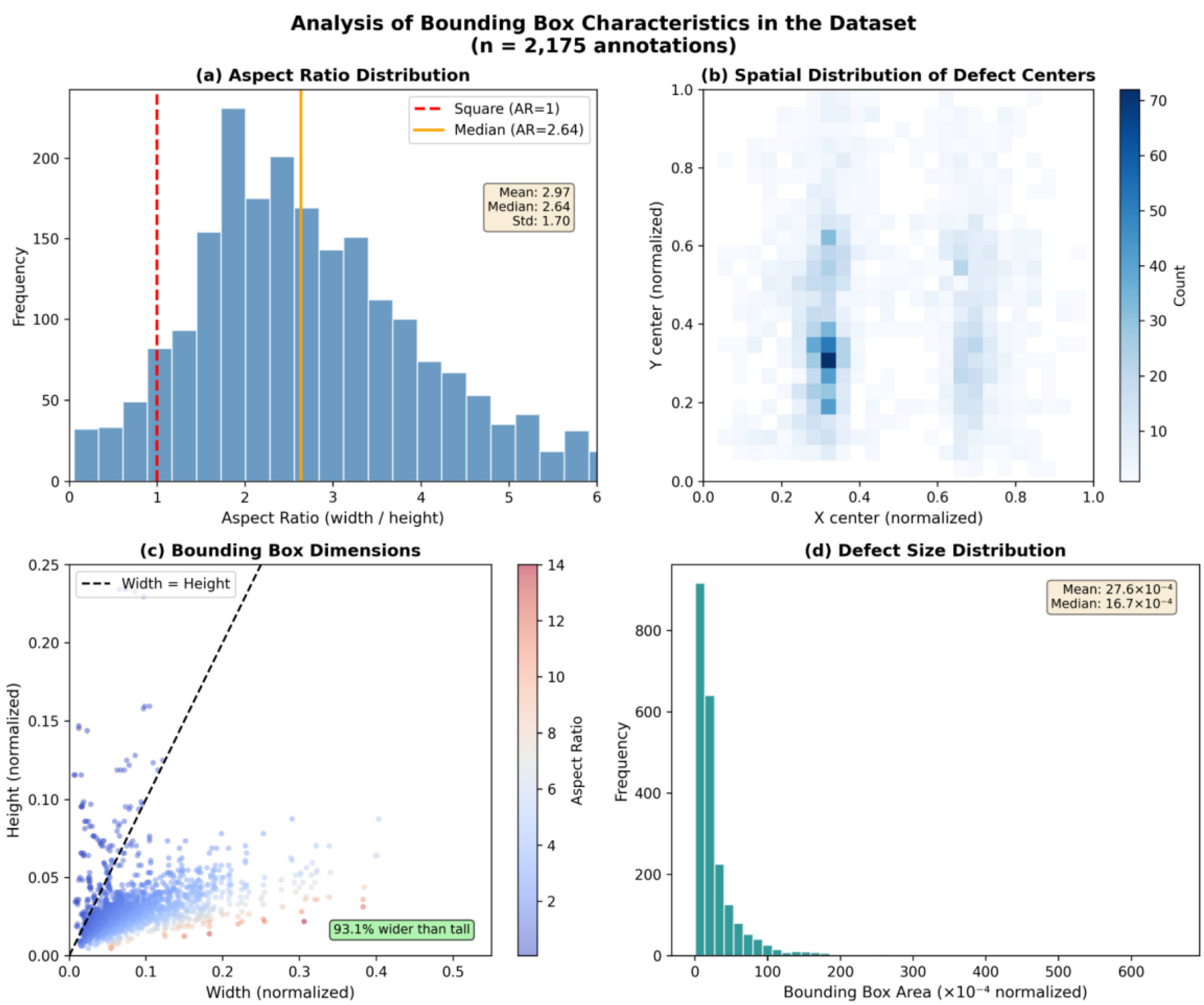

Figure 1. Analysis of Class Distribution and Bounding Box Characteristics in the Dataset.

- Training Parameters:
  - Batch Size: 100
  - Number of Epochs: Training was set for 500 epochs, but an Early Stopping mechanism was employed.
  - Early Stopping: To prevent overfitting, training was automatically halted after the model showed no improvement for 30 consecutive epochs.

This study employed the YOLOv8n (nano) model, the most compact variant of the YOLOv8 family developed by Ultralytics [12]. YOLOv8 represents the state-of-the-art in real-time object detection, incorporating architectural improvements over previous versions while maintaining computational efficiency suitable for industrial deployment.

The YOLOv8 architecture follows a three-component design: backbone, neck, and head. The backbone employs CSPDarknet (Cross Stage Partial Darknet) with C2f (Cross Stage Partial with two convolutions) modules for multi-scale feature extraction. This design reduces computational redundancy while maintaining gradient flow during training. The neck implements a PANet (Path Aggregation Network) structure with Feature Pyramid Network (FPN) capabilities, enabling effective fusion of features at different spatial resolutions—critical for detecting defects of varying sizes. The detection head utilizes an anchor-free, decoupled approach, separating classification and regression tasks into independent branches, which improves convergence during training [13]. Table 1 presents the architectural specifications of the YOLOv8n model employed in this study.

The choice of the nano variant was motivated by the trade-off between detection accuracy and computational efficiency, aiming to enable deployment on edge computing hardware without requiring high-performance GPUs. This consideration is particularly relevant for industrial applications where inference must occur in real-time at the production line.

Table 1. YOLOv8n Model Specifications

Specification	Value
Architecture	YOLOv8n (nano)
Total Layers	181
Parameters	2,590,035 (2.59M)
Model Size	5.5 MB
Input Resolution	640 × 640 pixels
GFLOPs	8.1

Transfer learning was employed by initializing the model with weights pre-trained on the COCO dataset [14], which contains over 330,000 images across 80 object categories.

This approach leverages the feature extraction capabilities learned from a large-scale dataset, enabling effective fine-tuning for the specific task of scratch detection with a relatively small custom dataset. Table 2 compares the YOLOv8n model with other architectures reported in the literature for surface defect detection.

The comparison demonstrates that YOLOv8n offers the most compact architecture with the fastest inference time, making it particularly suitable for real-time industrial applications where computational resources may be limited.

2.4 Evaluation metrics

To comprehensively evaluate the performance of our model, we used the following evaluation metrics:

- Mean Average Precision (mAP): Calculated at different Intersection over Union (IoU) thresholds, including mAP@0.5 and mAP@0.5:0.95.
- Precision and Recall: Used to measure the proportion of correct predictions and the model's ability to identify all relevant defects, as defined in Equations (1) and (2), where TP are True Positives, FP are False Positives, and FN are False Negatives.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

- F1-Score: The harmonic mean of precision and recall, providing a balanced measure of performance, as defined in Equation (3).

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{3}$$

- Confusion Matrix: Used to visualize the model's classification errors, particularly between defects and the background.
- Inference Time: The average time required to process a single image, which is crucial for real-time applications.

Table 2. Comparison with Other Detection Architectures

Model	Parameters (M)	Model Size (MB)	Inference Time (ms)	Application	Reference
YOLOv8n	2.59	5.5	3.2	Scratches in cold rolling	this work
YOLOv5s	7.2	14.1	6.4	NEU Dataset	[15]
YOLOv8s	11.2	22.5	4.5	Steel surface defects	[16]
Faster R-CNN	41.1	165	120	Metal surfaces	[17]
SSD300	23.7	95	25	Surface inspection	[18]

2.5 Experimental setup

Our experiments were conducted using PyTorch 2.6.0 and Python 3.11.13 frameworks. The hardware consisted of a GPU with 15 GB of memory. The experimental protocol followed a standardized procedure: initializing the model with pre-trained weights, training it with the specific dataset, conducting continuous validation to monitor progress, and finally testing the best saved model to evaluate its generalization performance.

Data collection and system validation were carried out at Cold Strip Mill No. 1 at CSN (Figure 2). The physical configuration of the vision system, as shown in Figure 3, was designed to operate in a continuous production environment. The capture camera was strategically positioned between the coiler and uncoiler of the line.

This location was chosen because it is compatible with the camera's capture speed, as the line speed at this point reaches a maximum of 180 meters per minute (MPM). To ensure synchronization and data accuracy, signals from the Programmable Logic Controller (PLC) were integrated into the system. These signals served two critical functions: validating the presence of a sheet in the inspection area and recording the exact section of material being monitored at the moment the image was captured.

3 Results and analysis

After conducting our experiments with the model on the steel sheet scratch dataset, we obtained results that provide insights into the model's performance. In this section, we present and analyze these results in detail.

Figure 2. Cold Strip Mill No. 1 at CSN.

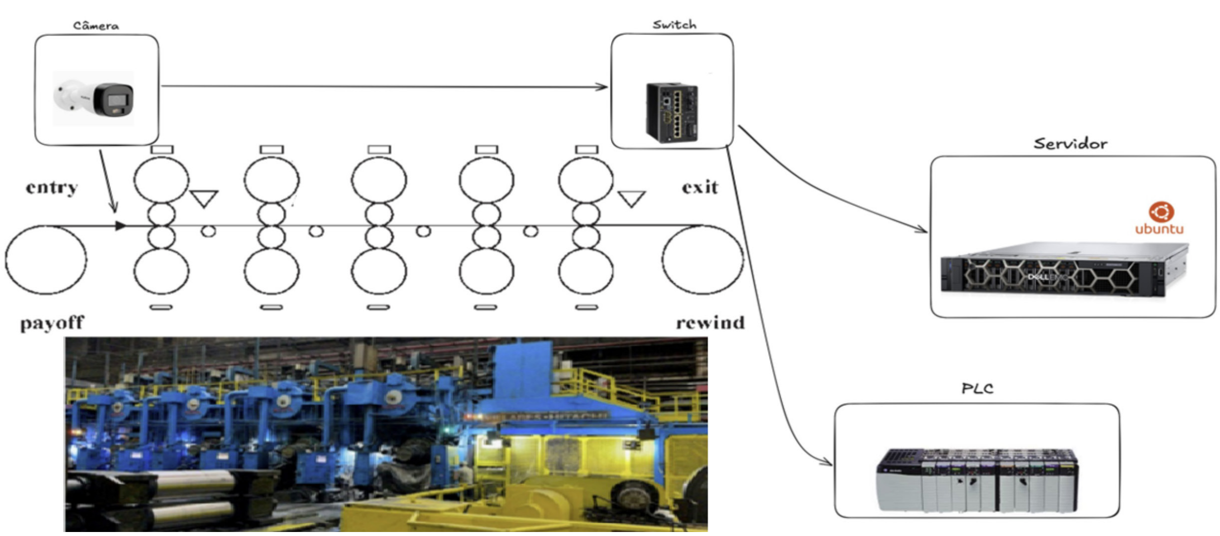

Figure 3. The physical configuration of the vision system.

3.1 Training performance

Figure 4 shows the model's training curves, illustrating the progression of various metrics over the training epochs.

Key Observations from the Training Curves:

- The model demonstrated consistent improvement throughout the training period, with loss metrics (box_loss, cls_loss, dfl_loss) progressively decreasing.
- Training was halted at epoch 106 by an Early Stopping mechanism, which prevented overfitting by detecting no performance improvement over the last 30 epochs. The best performance was achieved at epoch 76.
- Validation metrics, such as mAP_0.5, showed an increase and stabilization, indicating that the model generalized well to unseen data.

3.1.1 Model performance on validation images

Figure 5 illustrates the analysis of detection results on validation images, highlighting the following aspects of the model's performance.

- Detection Accuracy: The model successfully detected scratches with varying confidence levels.
- Robustness: The model demonstrated robustness to variations in lighting conditions and captured angles, detecting defects across different areas of the sheet.

It is essential to emphasize that this study was conducted as a proof of concept in a real industrial environment, utilizing low-cost equipment and existing infrastructure. The images were captured by conventional surveillance cameras, without a dedicated or controlled lighting system, resulting in visible reflections and illumination variations, as shown in Figure 5.

3.2 Model performance analysis

Table 3 presents a summary of the main performance metrics for the model on the validation dataset.

Key conclusions from the performance analysis:

- The model achieved an mAP50 of 0.574, indicating promising detection capability for a proof-of-concept system at an IoU threshold of 50%.
- The F1-Score of 0.584 demonstrates a reasonable balance between precision and recall.
- The inference time of 3.2 ms is exceptionally low, confirming the model's feasibility for real-time applications on a production line.

Table 3. Key Performance Metrics for the Model

Metric	Value
mAP@0.5 (B)	0.574
mAP@0.5:0.95 (B)	0.216
Precision (B)	0.555
Recall (B)	0.616
F1-Score (B)	0.584
Inference Time (ms)	3.2

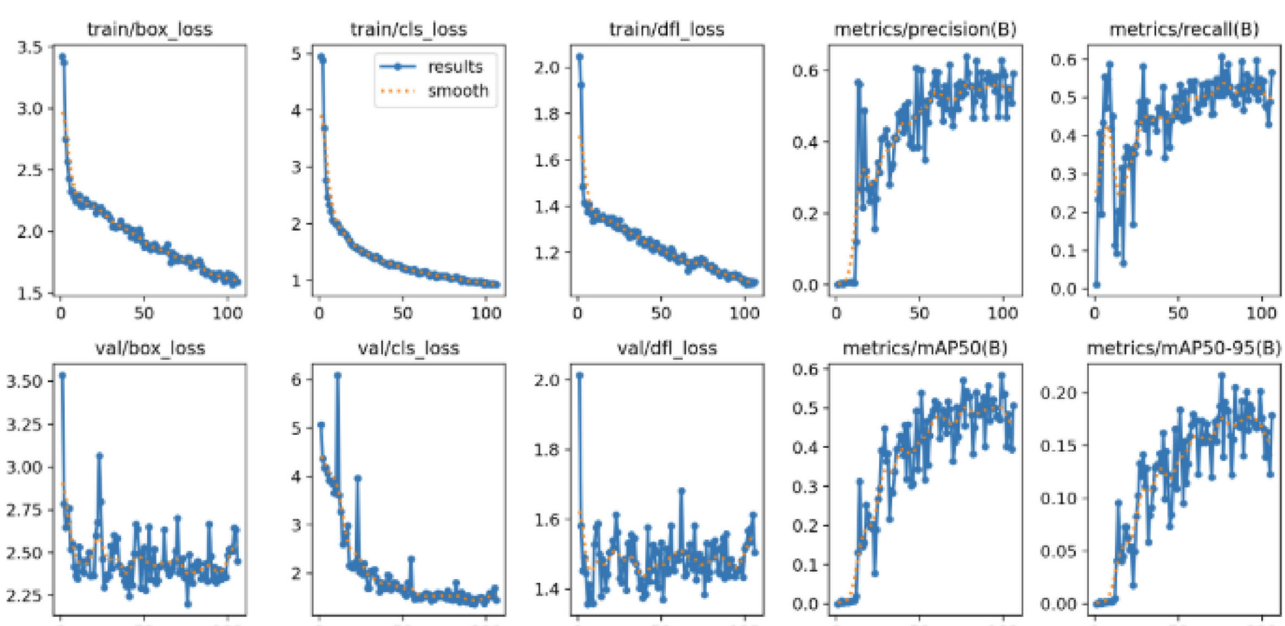

Figure 4. Model Training Curves.

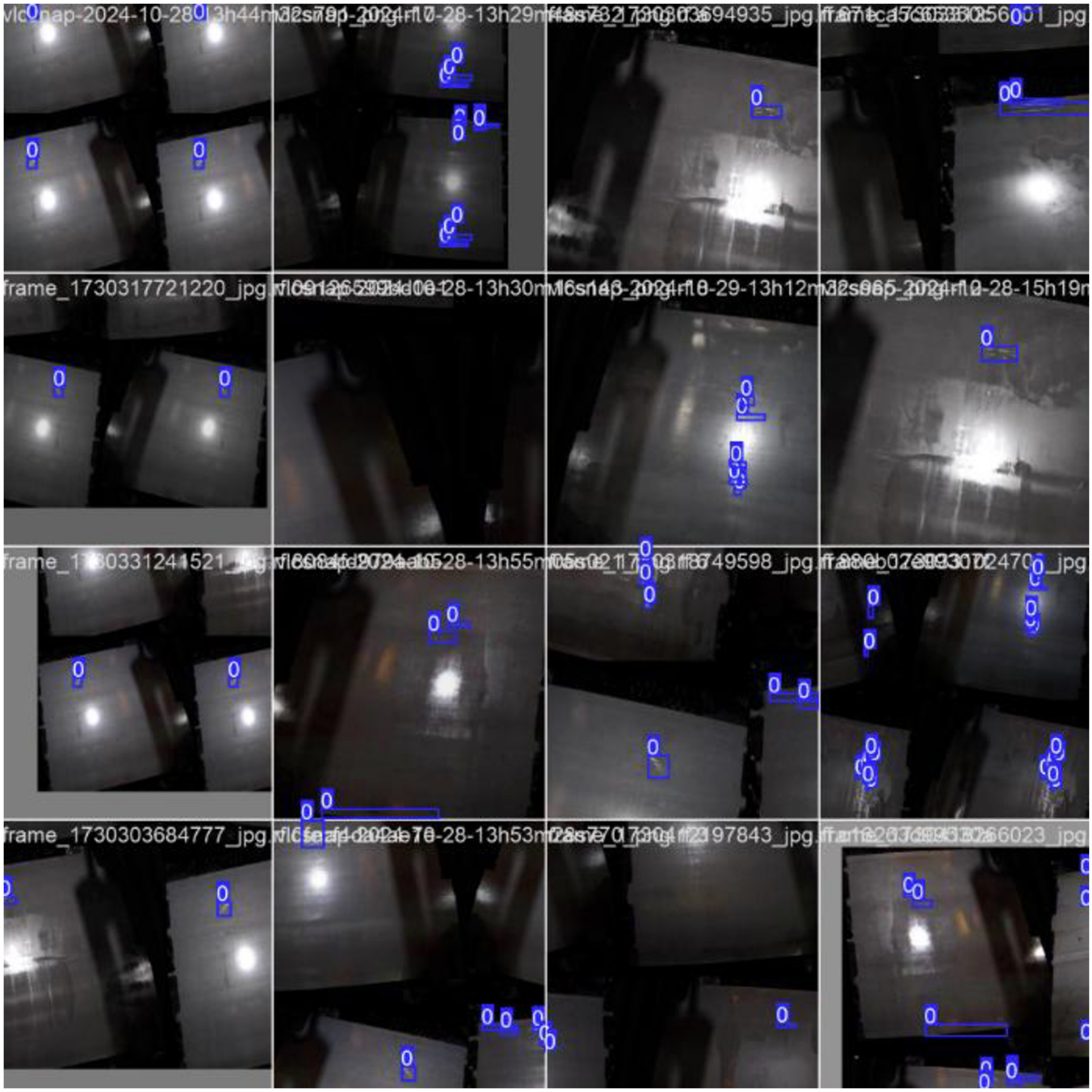

Figure 5. Analysis of Detection Results on Validation Images.

3.3 Class-wise performance analysis

As this study focuses on a single defect class (“Scratch”), the class-wise performance analysis evaluates how well the model performed on this specific task. Figure 6 presents the F1-Confidence curve.

The F1-Score peak (0.57) was achieved at a confidence threshold of 0.467, indicating the model’s optimal operating point for balancing false positives and false negatives.

3.4 Confusion matrix analysis

Figure 7 presents the normalized confusion matrix, providing insights into the classification errors.

Key Insights from the Confusion Matrix: The model correctly classified the majority of “Scratch” instances.

The evaluation of results must consider the experimental context of this study, characterized as a proof of concept conducted under deliberately restrictive conditions: non-specialized capture equipment (surveillance cameras), absence of controlled lighting, and limited-scale dataset.

Although the Precision (0.555) and Recall (0.616) values are below thresholds frequently reported in the literature for consolidated detection systems, it is essential to contextualize these metrics considering two factors: (i) the restrictive experimental conditions of this proof of concept, and (ii) the statistical nature of defect occurrence in the real environment.

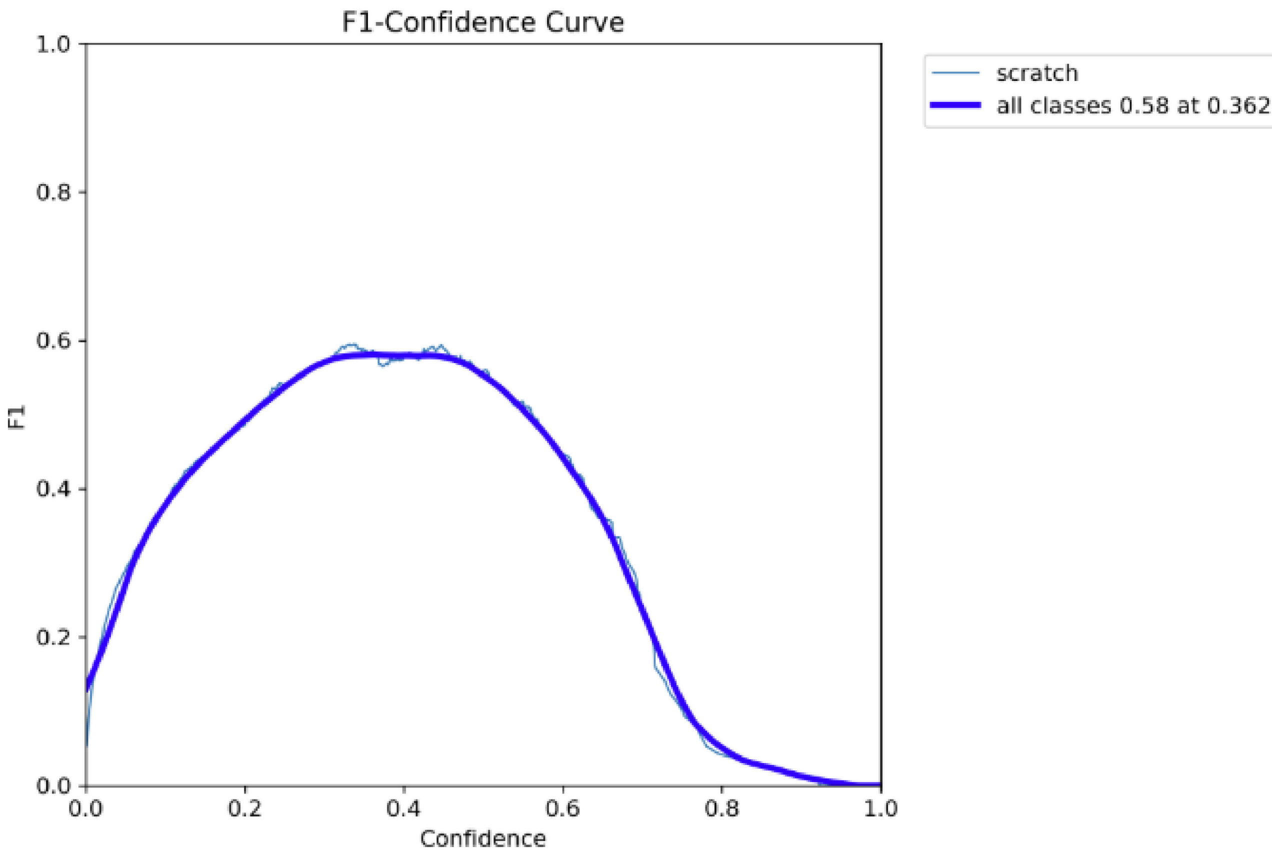

Figure 6. F1-Confidence Curve of the model.

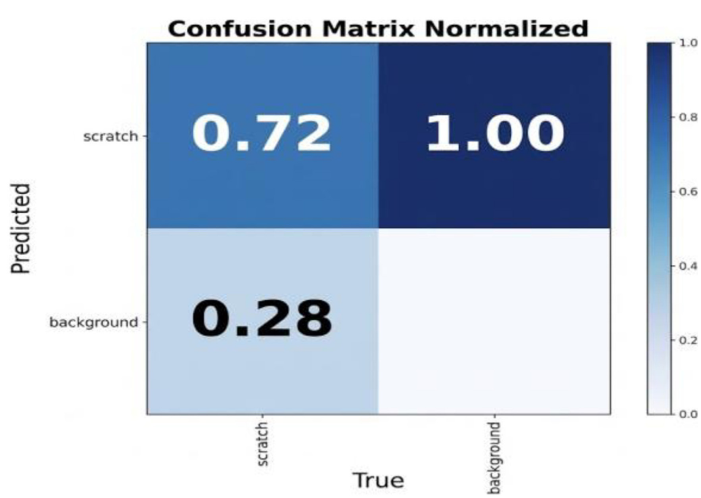

Figure 7. Normalized Confusion Matrix.

Regarding the first factor, the system was deliberately implemented with low-cost infrastructure (surveillance cameras and absence of controlled lighting), aiming to evaluate technical feasibility with minimal investment. Under these conditions, the metrics obtained represent a lower bound of the expected performance with adequate equipment.

Regarding the second factor, the morphological characteristic of scratches in the cold rolling process implies that this type of defect rarely occurs in isolation. When present, multiple instances are observed in sequence along the affected section of the strip. This natural redundancy significantly raises the probability of effective detection of the defective section. Considering an individual Recall of 0.616 and assuming independence between consecutive detections, the probability of detecting at least one defect in a section containing n instances is given by:

$$P(detection) = 1 - (1 - R)^n \quad (4)$$

In Equation (4), R represents Recall. For n = 3, 5, and 10 consecutive instances, this probability reaches 94.3%, 99.2%, and 99.99%, respectively. This analysis demonstrates that, in the real operational context, the effective reliability of the system is substantially higher than the individual metrics per bounding box.

Additionally, it is relevant to compare the proposed system with the current state of the process. The current manual inspection is subject to human variability, fatigue, and the impossibility of covering 100% of production. The automated system, even with current metrics, offers continuous and consistent monitoring, representing an advancement over the traditional method.

Qualitative analysis of detection errors allowed the identification of the main sources of false positives and false negatives in the system.

False positives are predominantly associated with structural elements of the rolling mill that remain visible in the image capture region. These components, manufactured from metallic materials, present surface marks resulting from continuous contact with the strip during the production process. The visual appearance of these marks is morphologically similar to strip scratches, inducing the model to erroneously classify them as defects. From an operational standpoint, this problem is mitigated by integration with PLC (Programmable Logic Controller) signals, which disables detection during intervals between strips, when only the structural background is visible.

False negatives, in turn, are concentrated in the peripheral regions of the image, where lighting is insufficient. The current lighting system, not specifically designed for this application, generates a luminous intensity distribution concentrated in the central region of the strip, resulting in low contrast at the edges. Scratches located in these regions present reduced visibility, hindering their detection by the model. The replacement of the spot lighting system with diffuse lighting, which provides homogeneous light distribution over the entire inspected surface, is recommended to mitigate this limitation in future implementations.

3.5 Scope and extensibility

The present study focused exclusively on the “scratch” class, as this defect represented a critical and priority demand for the quality control of the industrial unit at the time of the research. This scope delimitation allowed an in-depth analysis of the model’s performance for this specific defect type.

The YOLOv8 architecture natively supports multi-class detection, allowing the inclusion of additional categories of surface defects (such as scale, roll marks, under-pickling, among others) by expanding the training dataset.

The incorporation of new classes requires only the collection and annotation of representative samples of each defect type, followed by model retraining. This architectural flexibility enables the incremental evolution of the system toward a more comprehensive defect detector in future work.

3.6 Precision-recall curve analysis

Figure 8 illustrates the Precision-Recall curve, providing a view of the model’s performance across different confidence thresholds.

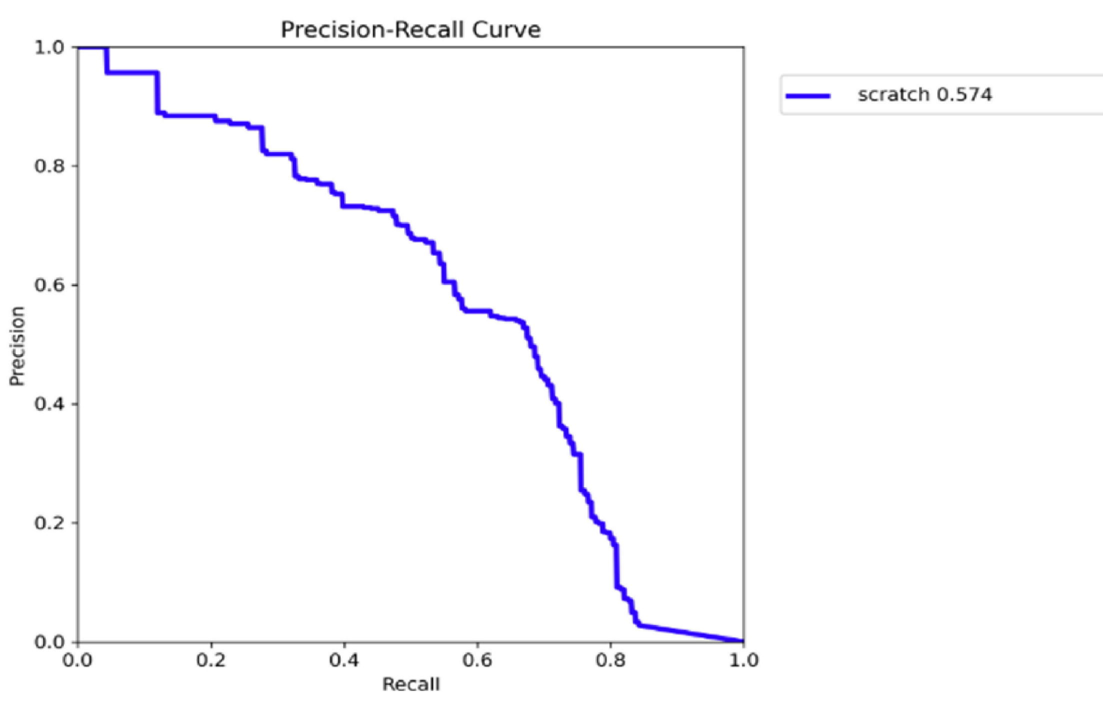

Figure 8. Precision-Recall Curve.

Table 4. Model Size and Inference Time

Parameters (M)	Model Size (MB)	Inference Time (ms)
2.59	5.5	3.2

The area under the curve represents the mean Average Precision (mAP) value, confirming the model's overall performance. The curve illustrates the inherent trade-off: achieving higher precision (fewer false positives) tends to reduce recall (more false negatives), and vice versa.

3.7 Computational efficiency

It is crucial to consider the computational requirements for practical implementation. Table 4 compares the model size and inference time.

The low number of parameters and compact model size, combined with speedy inference time, suggest that the model is a viable and highly efficient option for real-time applications in industrial environments, without requiring high-end computational hardware for deployment.

4 Conclusion

A comprehensive study on the implementation of a computer vision system for automated quality control in manufacturing processes, focusing on the detection of scratches on steel sheets, yielded insights into the capabilities of a state-of-the-art object detection model. Through systematic implementation and evaluation, the work demonstrates the potential of this advanced model to transform quality control processes in the steel industry. The model achieved good performance, with an mAP50 of 0.574 and an F1-Score of 0.584.

The study was specifically conducted to analyze performance in a real, restrictive environment, particularly with respect to lighting and camera-related limitations. Surveillance cameras were used to validate the feasibility of real-time defect detection using AI models.

This methodological choice was deliberate, aiming to evaluate the technical feasibility of automated defect detection under adverse conditions and with minimal investment in infrastructure. The fact that the model achieved a mAP@5 of 0.574 even under these restrictive conditions demonstrates the robustness of the proposed approach. It suggests that significantly superior results could be obtained with the implementation of an appropriate lighting system and dedicated industrial cameras.

Future work will focus on optimizing the image acquisition system, including the use of diffuse lighting to eliminate specular reflections, which is expected to lead to a substantial improvement in detection metrics.

These results, combined with remarkable computational efficiency - a 3.2 ms inference time per image and a model size of 5.5 MB - indicate its superior capability and feasibility for real-time defect detection. The application of transfer learning techniques proved highly beneficial, enabling the model to be successfully fine-tuned for a specific task using a limited dataset, consistent with findings in other studies in the field.

The present study focused exclusively on the "scratch" class, as this defect represented a critical and priority demand for quality control at the industrial unit during the research period. This scope delimitation enabled a more in-depth analysis of the model's performance for this specific type of defect. However, the adopted architecture natively supports multiclass detection, allowing the inclusion of additional categories of surface defects.

The implications for the manufacturing industry are extensive, beginning with improved quality control, as the system reduces human error and enhances consistency. Consequently, efficiency increases, as real-time detection allows for inspection of 100% of products without creating bottlenecks, directly leading to cost reduction. Additionally, the model demonstrates adaptability for retraining on new defect types and generates valuable data for continuous process improvement. The incorporation of new classes would require the collection and annotation of representative samples for each type of defect, followed by retraining of the model. This architectural flexibility enables the incremental evolution of the system toward a more comprehensive defect detector in future work.

References

1 Wang J, Ma Y, Zhang L, Gao RX, Wu D. Deep learning for smart manufacturing: methods and applications. Journal of Manufacturing Systems. 2018;48:144-156.

2 Nunes JBL, Cordeiro NH. Aplicação de técnicas de visão computacional para inspeção de defeitos na superfície de chapas laminadas de aço no processo produtivo industrial [monografia]. Votuporanga: Instituto Federal de Educação, Ciência e Tecnologia de São Paulo; 2022.

3 Tudeschini RB, Soares AMS. Automatic inspection system of adhesive on vehicle windshield using computational vision. Journal of the Brazilian Society of Mechanical Sciences and Engineering. 2023;45:124.

4 Boff VC. Sistema de visão computacional para medição sem contato das dimensões de capacitores [monografia]. Caxias do Sul: Universidade de Caxias do Sul; 2020.

5 Tao X, Zhang D, Ma W, Liu X, Xu D. Automatic metallic surface defect detection and recognition with convolutional neural networks. Applied Sciences. 2018;8(9):1575.

6 Sakuma DT. Implementação de um framework de visão computacional para inspeção de qualidade no assoalho de automóveis [dissertação]. São Paulo: Universidade Federal de São Paulo; 2023.

7 Fernandes RR. Inspeção visual em linha de produção utilizando visão computacional [monografia]. São Paulo: Universidade de São Paulo; 2024.

8 Souza KS. Implementação de um sistema de visão com deep learning para otimizar o processo de inspeção de emendas das bobinas na fabricação do cinturão de segurança [dissertação]. Manaus: Instituto de Tecnologia e Educação Galileo da Amazônia; 2024.

9 Pan SJ, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering. 2010;22(10):1345-1359.

10 Xu L, Xu EL, Li LX. Industry 4.0: state of the art and future trends. International Journal of Production Research. 2018;56(8):2941-2962.

11 Loshchilov I, Hutter F. Decoupled weight decay regularization. arXiv preprint. 2017;1711.05101.

12 Jocher G, Chaurasia A, Qiu J. Ultralytics YOLOv8 [Internet]. 2023 [cited 2025 Oct 9]. Available at: https://github.com/ultralytics/ultralytics

13 Terven J, Córdova-Esparza DM. A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Mach Learn Knowl Extr. 2023;5(4):1680-1716.

14 Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft COCO: Common objects in context. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T, editors. Computer Vision – ECCV 2014. Cham: Springer; 2014. p. 740-755.

15 He Y, Song K, Meng Q, Yan Y. An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE Transactions on Instrumentation and Measurement. 2020;69(4):1493-1504.

16 Luo Q, Fang X, Liu L, Yang C, Sun Y. Automated visual defect detection for flat steel surface: A survey. IEEE Transactions on Instrumentation and Measurement. 2020;69(3):626-644.

17 Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017;39(6):1137-1149.

18 Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: Single shot multibox detector. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016. Cham: Springer; 2016. p. 21-37.

Received: 9 Oct. 2025
Accepted: 18 Feb. 2026

Editor-in-charge:
André Luiz Vasconcellos da Costa e Silva