PROJECT ABSTRACT

Steel Surface Defect Detection using CNN

A convolutional neural network classifying steel surface defects into the six NEU-DET families, with real GLCM texture analysis, a live inspection demo, training notebook and viva kit. Delivered built-to-order with report, PPT and Q&A.

---

01. Title

Steel Surface Defect Detection using CNN

02. Introduction / Background

Hot-rolled steel strip is inspected at line speed for surface defects — crazing, patches, inclusions, pitted areas, rolled-in scale and scratches — because a missed defect becomes a rejected coil or a failed part downstream. Human inspection is slow and inconsistent, which is why automated visual inspection with CNNs is a staple of industrial machine vision. The NEU surface-defect database (Northeastern University, China) is the standard public benchmark: 1,800 grayscale images, 300 per defect family, captured under controlled lighting. This project trains a 6-class CNN on NEU-DET end to end.

03. Problem Statement

Student defect-detection projects often train on ad-hoc web-scraped images with no defect taxonomy and no classical baseline, leaving them unable to explain what the network sees or how the industry actually inspects steel. This project addresses that by using the standard NEU taxonomy, pairing the CNN with a genuine GLCM texture front-end (the classical inspection method) in a live demo, and evaluating with per-class metrics so every confusion between defect families is quantified.

04. Objectives

- Train a 6-class CNN (Conv 32/64, pooling, dense 256, softmax) on the NEU-DET dataset.
- Implement real GLCM texture analysis (contrast, homogeneity, energy, correlation) on uploaded photos.
- Build a live defect-scanner demo with a transparent heuristic pre-screen.
- Evaluate with per-class precision/recall/F1 and the confusion matrix.
- Deliver the complete student kit: notebook, trained model, demo, report, PPT and viva Q&A.

05. Proposed Methodology

The pipeline has four stages. (1) Data: NEU-DET images (200×200 grayscale, 300 per class) are normalized and augmented with documented transforms (rotation, flip, brightness jitter). (2) Modelling: the CNN is trained with class-balanced sampling and early stopping. (3) Evaluation: the held-out split is scored once for accuracy, per-class metrics and the confusion matrix. (4) Demo: an uploaded photo is converted to grayscale, a real GLCM is computed (16 gray levels, four orientations) with the four texture statistics, and a documented heuristic scores each NEU defect family for the on-screen pre-screen verdict.

06. System Architecture / Working

The training notebook produces a serialized CNN plus preprocessing parameters. The single-file web demo has three views: the Defect Scanner (photo upload, live GLCM statistics, pre-screen verdict and defect-family likelihoods), the Texture Lab (illustrative GLCM fingerprints per defect family with the texture signatures), and the System Report (CNN architecture diagram, NEU dataset facts and design-target metrics).

07. Hardware / Software Requirements

- Python 3, TensorFlow/Keras, NumPy, OpenCV, scikit-learn
-

Jupyter Notebook (training & evaluation)

- NEU surface defect database
- HTML5 canvas + JavaScript (live GLCM demo)
- Any modern laptop; no GPU required for inference

08. Expected Outcomes

- A trained 6-class CNN with design-target accuracy ≥97% and per-class F1 ≥0.94.
- A live GLCM defect scanner computing real texture features from uploaded photos.
- Confusion matrix and per-class analysis for the report.
- A complete report, PPT and viva Q&A on inspection vision and CNNs.

09. Applications

- Teaching CNNs for industrial machine vision.
- Demonstrating classical texture analysis (GLCM) alongside deep learning.
- Prototype front-end for steel-mill inspection stations.
- Benchmark study of defect-family confusions.

10. Future Scope

- Defect localization (detection) instead of whole-image classification.
- Domain adaptation to real mill lighting and scale texture.
- Lightweight models for line-speed edge deployment.
- Anomaly detection for defect types absent from NEU-DET.