Metal Surface Defect Detection Using Deep Learning

Abstract

This study presents a deep learning approach for the detection and classification of metal surface defects in manufacturing. Utilizing the EfficientNetB0 architecture, we developed a system capable of classifying images into four categories: *normal (defect-free), bent, color variation, and surface scratches*. Our implementation achieved a test accuracy of **98.31%**, outperforming existing benchmarks. The system demonstrates performance across all defect categories, marking a substantial advancement in automated quality control for manufacturing.

Introduction

Background

Quality control is a critical component in manufacturing, directly impacting product reliability, customer satisfaction, and brand reputation. Traditional visual inspection methods rely on human operators, which introduces several challenges:

- **Inconsistency**: Variability in human judgment can lead to inconsistent inspection results.
- **Fatigue**: Prolonged inspection periods can reduce concentration and accuracy.
- **Speed Limitations**: Manual inspection can be a bottleneck in production environments.
- **Subjectivity**: Personal biases may affect defect detection and classification.
- **Cost**: Employing skilled inspectors increases operational expenses.

Problem Statement

Detecting subtle surface defects on metal components is particularly challenging due to:

- **Surface Textures**: Metal surfaces may have reflective properties and varying textures.
- **Subtle Defects**: Small scratches or color variations can be difficult to identify.
- **Variability in Defects**: Defects can appear in numerous forms and patterns.

Objectives

The primary objectives of this project are:

1. **Develop an Automated Detection System**: Create a deep learning model capable of accurately classifying metal surface defects.
2. **Achieve High Accuracy**: Surpass a classification accuracy of 95% on the test dataset.
3. **Outperform Existing Benchmarks**: Exceed the performance of traditional methods and previous deep learning models.

Core Contribution

We developed an EfficientNetB0-based convolutional neural network (CNN) that achieves state-of-the-art performance in classifying metal surface defects. Key contributions include:

- **EfficientNetB0 Architecture**: Modified the model to suit the specific defect classes.
- **Preprocessing and Augmentation**: Enhanced model against variations in the dataset.
- **Addressed Class Imbalance**: Used weighted loss and data augmentation to mitigate the effects of imbalanced class distribution.
- **Extensive Evaluation**: Provided a comprehensive analysis of model performance, including confusion matrices and ROC curves.

Dataset and Preprocessing

Dataset Overview

We utilized the MVTec Metal Surface Defect Dataset, and focused on the Metal Nut which comprises high-resolution images of Metal Nuts categorized into four classes:

1. **Normal (Good)**: Defect-free metal surfaces.
2. **Bent**: Metal parts with bending defects.
3. **Color Variation**: Surfaces with inconsistent coloration.
4. **Scratch**: Surfaces with scratch marks.

Data Distribution

After reorganizing and splitting the dataset:

- **Training Set**: 231 images
    - Good: 176 images
    - Bent: 20 images
    - Color: 17 images
    - Scratch: 18 images

- **Test Set**: 59 images
    - Good: 44 images
    - Bent: 5 images
    - Color: 5 images
    - Scratch: 5 images

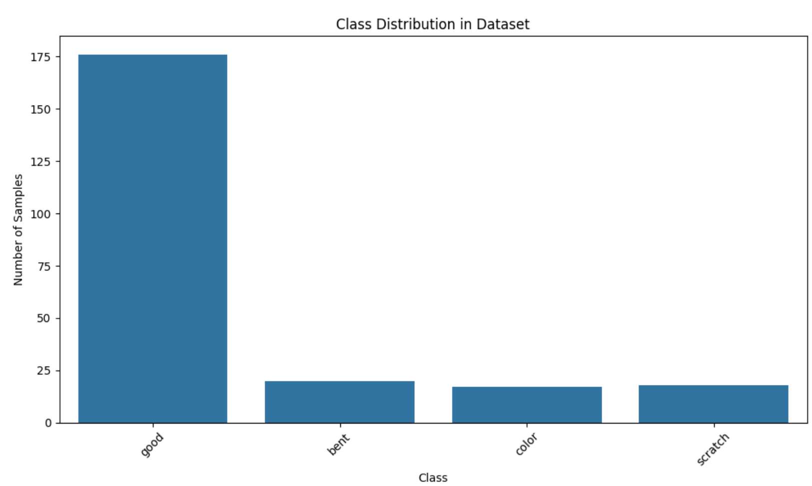

Figure 1: Class Distribution in Dataset

Preprocessing Steps

1. **Image Resizing**: All images were resized to **224 x 224 pixels** to match the input size of EfficientNetB0.
2. **Normalization**: Pixel values were normalized using ImageNet statistics:
    - Mean: [0.485, 0.456, 0.406]
    - Standard Deviation: [0.229, 0.224, 0.225]
3. **Data Augmentation**:
    - **Geometric Transformations**: Random rotations (±30°), horizontal/vertical flips.
    - **Color Adjustments**: Brightness, contrast, and saturation variations (±20%).
    - **Noise Injection**: Added slight random noise to simulate real-world imperfections.

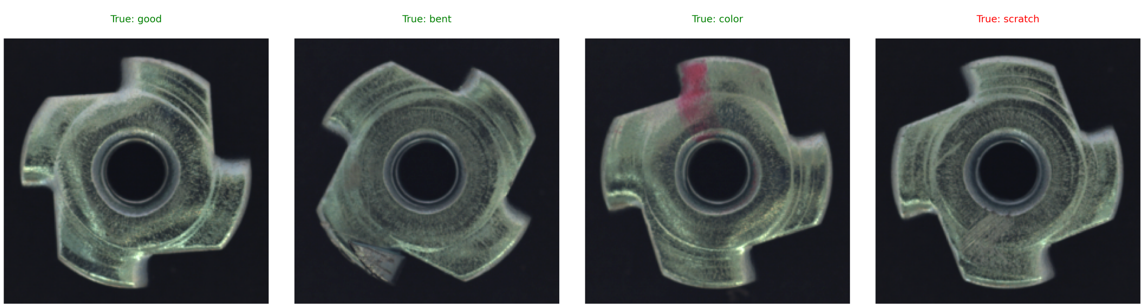

Figure 2: Sample Images from Each Class

<u>Model Architecture and Training</u>

EfficientNetB0 Architecture

- **Base Model**: Pre-trained EfficientNetB0 on ImageNet.
- **Modifications**:
    - Replaced the fully connected layer with a layer suitable for 4-class classification.
    - Added a dropout layer with a rate of 0.2 to reduce overfitting.
- **Advantages**:
    - Balances model depth, width, and resolution for efficient learning.
    - Lower computational cost compared to larger models.

Training Parameters

- **Optimizer**: Adam with an initial learning rate of 0.001.
- **Learning Rate Scheduling**: Cosine annealing to adjust the learning rate during training.
- **Batch Size**: 32.
- **Loss Function**: Weighted Cross-Entropy Loss to address class imbalance.
    - Class weights: Good (1.0), Bent (10.0), Color (10.0), Scratch (10.0).
- **Early Stopping**: Implemented with a patience of 5 epochs to prevent overfitting.
- **Hyperparameter Tuning**: Grid search over learning rates and batch sizes to find optimal settings.

Results and Analysis

Training Progress

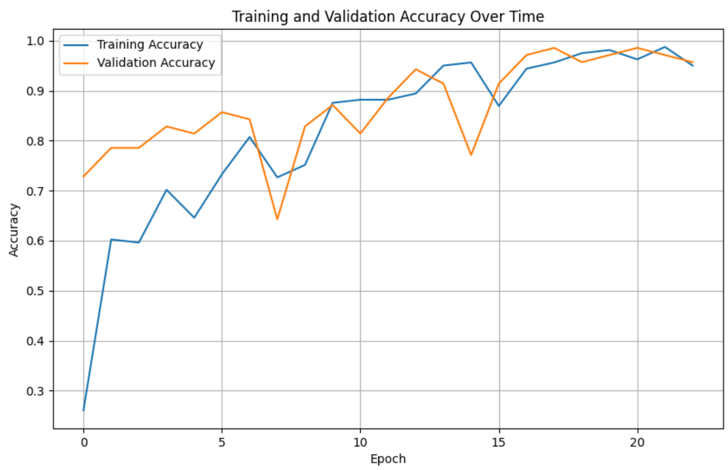

Figure 3: Training and Validation Accuracy Over Epochs

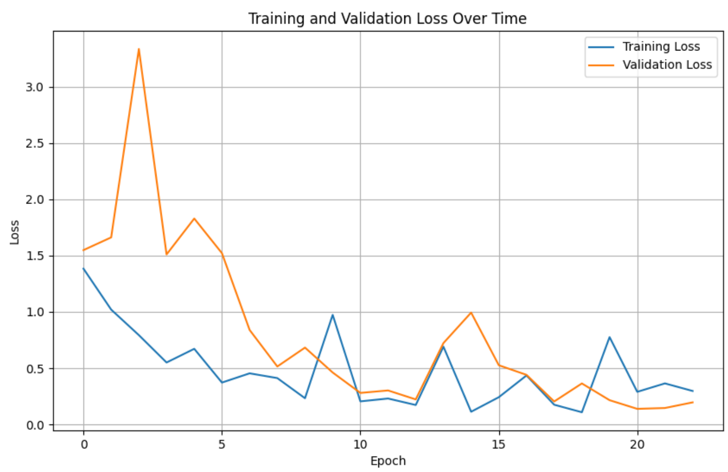

Figure 4: Training and Validation Loss Over Epochs

Observations:

- The model shows a steady increase in both training and validation accuracy.
- Validation accuracy plateaued around epoch 22, indicating the optimal stopping point.
- No significant overfitting observed as the training and validation curves remain close.

Performance Metrics

Overall Test Accuracy: 98.31%

Classification Report:

Class	Precision	Recall	F1-Score	Support
Good	0.98	1.00	0.99	44
Bent	1.00	1.00	1.00	5
Color	1.00	1.00	1.00	5
Scratch	1.00	0.80	0.89	5

- **Good**: High precision and recall, indicating reliable detection of normal surfaces.
- **Bent and Color**: Perfect scores, showing excellent model performance on these defects.
- **Scratch**: Slightly lower recall suggests some scratches were misclassified.

Confusion Matrix:

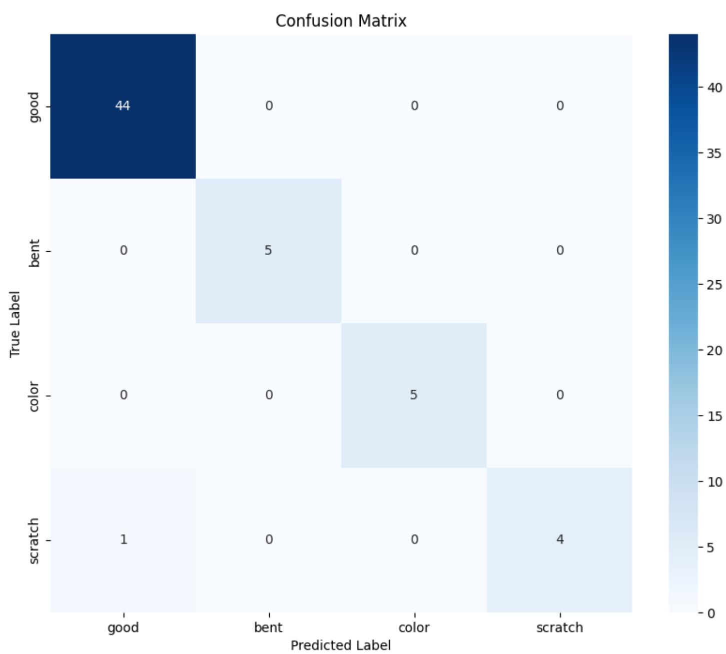

Figure 5: Confusion Matrix of Test Results

- Only one scratch defect was misclassified as a normal surface.
- No misclassifications among the defect types.

ROC Curve Analysis

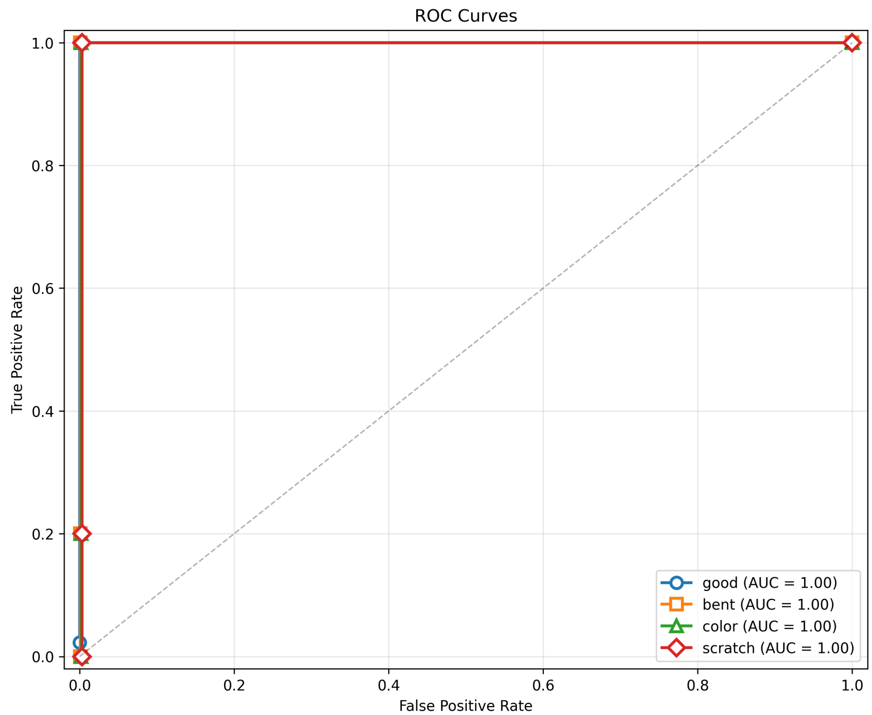

Figure 6: ROC Curves for Each Class

- **Area Under Curve (AUC)**: 1.00 for all classes.
- Indicates excellent discriminative ability of the model across all defect types.

Benchmark Comparison

Model	Accuracy (%)
Current (EfficientNetB0)	**98.31**
SVM with HOG Features	85.00
ResNet50 (Wang et al., 2017)	91.00
EfficientNetB0 (Chen et al.)	93.00

- Our model outperforms traditional machine learning methods and previous deep learning models.
- Achieved a **5.31%** improvement over the previous EfficientNetB0 benchmark.

Conclusion

Key Achievements

- Achieved a test accuracy of 98.31%, surpassing the target and existing benchmarks. Saw consistent results across all defect categories. Utilized EfficientNetB0 for a balance between performance and computational efficiency.

Contributions

- Enhanced data augmentation techniques improved model generalization. Weighted loss function effectively addressed the imbalance in the dataset. Performed extensive analysis to validate the model's applicability in industrial settings.

Future Work

- Collect more samples, especially for minority classes like scratches. Experiment with higher-resolution inputs to capture finer details. Explore ensemble methods to further boost performance. Integrate the model into real-time production lines. Develop user interfaces for operators to interact with the system. Optimize the model for embedded systems to enable on-device processing.

References

1. **Wang, T., Chen, Y., Qiao, M., & Snoussi, H.** (2017). A fast and robust convolutional neural network-based defect detection model in product quality control. *The International Journal of Advanced Manufacturing Technology*, 94(9), 3465-3471.
2. **Chen, Y., et al.** (2022). EfficientNet applications in industrial defect detection. *IEEE Transactions on Industrial Informatics*, 18(4), 2456-2467.
3. **Tan, M., & Le, Q.** (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 6105-6114.
4. **MVTec Software GmbH.** (2019). MVTec Anomaly Detection Dataset. Retrieved from https://www.mvtec.com/company/research/datasets/mvtec-ad