Industrial Surface Crack Classification

Candidate: Yuvraj Singh

Assignment: IndustrialAI Crack Classification – DeltaX

1. Introduction

The objective of this assignment is to develop a deep learning-based classification system capable of distinguishing cracks from non-defective regions on stamped industrial metal surfaces.

The problem is formulated as a **binary image classification task**:

- **Defect (Crack)**
- **No Defect (Non-Crack)**

Cracks occurring during stamping processes are often thin, irregular, and visually similar to natural bent regions of metal panels. This structural similarity introduces ambiguity and makes accurate classification challenging.

A transfer learning-based approach using EfficientNet-B0 was implemented to address this problem.

2. Problem Understanding

In industrial metal stamping:

- Real cracks appear as thin discontinuities in the material surface.
- Bent or curved panel structures may visually resemble cracks.
- Lighting variations and surface reflections further complicate classification.

The main challenge lies in distinguishing:

- True micro-cracks
- Stamping-induced bend lines
- Surface texture patterns

This is fundamentally a **texture-sensitive classification problem**, where discriminative information may occupy only a small portion of the image.

3. Dataset Organization

3.1 Training Dataset

The training dataset did not follow a standard class-folder structure. Labels were defined using filename numeric ranges as specified in the assignment instructions.

A custom PyTorch dataset pipeline was implemented to:

- Parse filenames
- Assign correct labels
- Load images
- Apply transformations

This ensured consistent and reproducible label mapping during training.

3.2 Validation Dataset

The validation dataset followed a structured directory format:

```
val/
  defect/
  no_defect/
```

This was loaded using torchvision.datasets.ImageFolder.

Validation was performed at the end of every epoch, and the best-performing model was saved.

4. Data Preprocessing

Training Transformations

To improve generalization and robustness, the following preprocessing pipeline was used:

- Resize to 256 × 256
- Random resized crop (224 × 224)
- Random horizontal flip
- Random rotation (small angle)
- Conversion to tensor
- Normalization using ImageNet mean and standard deviation

These augmentations simulate variations in orientation and surface conditions during manufacturing.

Validation Transformations

- Resize to 224 × 224
- Conversion to tensor
- ImageNet normalization

No augmentation was applied during validation to ensure fair evaluation.

5. Model Architecture

Base Model: EfficientNet-B0 (ImageNet Pretrained)

EfficientNet-B0 was selected due to:

- Strong performance on texture-rich classification tasks
- Efficient parameter scaling
- Good balance between accuracy and computational cost

Modifications:

- Final classifier layer replaced
- Output dimension changed to 2 classes

All backbone layers were unfrozen to allow full fine-tuning.

6. Training Configuration

- Input size: 224 × 224
- Loss function: CrossEntropyLoss
- Optimizer: Adam
- Learning rate: 1e-5
- Batch size: 32
- Epochs: 5–15 (multiple experiments)
- Device: NVIDIA P100 GPU

Training loop consisted of:

1. Forward propagation

2. Loss computation
3. Backpropagation
4. Optimizer update
5. Validation evaluation

The model with highest validation accuracy was saved.

7. Experiments and Results

Multiple experiments were conducted to explore performance improvements.

Experiment	Configuration	Validation Accuracy
Baseline Full Fine-Tuning	EfficientNet-B0 (lr=1e-5)	~65%
Augmentation Variations	Increased transformations	~72%
Optimizer Adjustment	AdamW (lr=1e-4)	**76.34%**

Best Validation Accuracy Achieved:

76.34%

8. Confusion Matrix Analysis

Analysis of the best-performing model revealed:

- False positives occur when bent metal structures resemble crack patterns.
- False negatives occur for thin, low-contrast cracks.
- Global CNN pooling sometimes emphasizes background texture over localized crack features.

These findings highlight the structural ambiguity present in the dataset.

9. Error Analysis

Manual inspection of misclassified samples showed:

1. Curved stamping lines frequently predicted cracks.
2. Fine cracks with low contrast are occasionally missed.
3. Lighting reflections influencing feature extraction.

This indicates that crack classification is highly sensitive to local texture variations and that global image-level classification may not fully capture fine crack structures.

10. Technical Observations

Key insights from experimentation:

- Transfer learning significantly accelerates convergence.
- Domain shift between open-source crack data and industrial metal data affects validation performance.
- Crack regions occupy small spatial areas relative to full image.
- Texture similarity between defect and non-defect classes is the primary challenge.

11. Potential Improvements

To move toward the 99% validation target, the following strategies could be explored:

- Hard negative mining
- Edge-enhanced preprocessing (Sobel, CLAHE)
- Patch-based training instead of full-image classification
- Attention mechanisms to focus on localized crack regions
- Focal loss for difficult samples
- Segmentation-based crack localization

These approaches may help improve sensitivity to thin crack structures.

12. Conclusion

A deep learning-based crack classification system was implemented using EfficientNet-B0 with transfer learning.

The final model achieved **76.34% validation accuracy** under current experimental settings. The results demonstrate the feasibility of applying transfer learning to industrial surface defect detection while also highlighting the complexity of distinguishing subtle crack patterns from visually similar bent structures.

The provided codebase is fully reproducible, modular, and extensible for further optimization toward higher accuracy targets.