SPATIAL FILTERING FOR EDGE DETECTION AND NOISE REDUCTION

Rafay Adeel

1. Introduction

In digital image processing, the extraction and enhancement of meaningful details from images are crucial for tasks such as medical diagnostics, satellite image interpretation, and defect detection in manufacturing. However, raw images often suffer from issues such as noise, poor contrast, or blurred edges, which reduce their interpretability.

Spatial filtering techniques are fundamental tools used to address these challenges. They work directly on the pixel neighborhood of an image to either suppress unwanted details (smoothing filters) or enhance critical details such as edges and fine structures (sharpening filters).

In this study, we focus on the application of **edge detection and image sharpening** to improve the visibility of fine details while reducing noise. We compare first-order and second-order derivative filters for edge detection and evaluate multiple smoothing filters for noise suppression. The effectiveness of combining smoothing and sharpening is also analyzed.

---

2. Dataset Description

For experimentation, we selected the **NEU Surface Defect Database**, a real-world dataset widely used for defect detection in industrial manufacturing. The dataset contains images of metal surfaces with different types of surface defects such as scratches, rolled-in scale, and patches.

- **Total images**: 1,800 (300 for each of 6 defect classes).
- **Image size**: 200 × 200 grayscale.
- **Characteristics**:
    - High-texture backgrounds, making defects difficult to isolate.
    - Images often contain subtle defects that require detail enhancement.

This dataset is appropriate because **edge detection and sharpening can highlight subtle defects** that might otherwise remain hidden, which is crucial in manufacturing quality control.

---

3. Methodology & Justification

We applied both **smoothing filters** (for noise reduction) and **sharpening filters** (for detail enhancement).

3.1 Smoothing Filters

- **Mean (Averaging) Filter**
  - Kernel:

$$h(x,y) = \frac{1}{n^2} \sum_{i=-k}^{k} \sum_{j=-k}^{k} f(x+i, y+j)$$

  - Purpose: Reduces Gaussian noise by averaging neighborhood pixels.
  - Limitation: Blurs edges along with noise.
- **Median Filter**
  - Operation: Replaces each pixel with the median of its neighborhood.
  - Purpose: Excellent for removing **salt-and-pepper noise** while preserving edges.
- **Mode Filter**
  - Operation: Assigns each pixel the most frequently occurring intensity in its neighborhood.
  - Purpose: Effective for repetitive patterns or backgrounds where a majority gray level dominates.

3.2 Sharpening Filters

- **Sobel Filter (First Derivative)**
  - Kernels:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

  - Purpose: Detects edges by computing gradients in horizontal and vertical directions.
  - Justification: Useful for highlighting directional edges of defects.
- **Laplacian Filter (Second Derivative)**
  - Kernel (example):

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

  - Purpose: Sensitive to intensity changes in all directions.
  - Justification: Enhances fine details but amplifies noise.
- **Combined Sobel + Laplacian**
  - Purpose: Applies Sobel for directional edge detection, followed by Laplacian for stronger edge enhancement.
  - Justification: Produces sharper results when images contain subtle variations.

3.3 Combined Pipeline

- **Smoothing before sharpening**: Prevents noise amplification in derivative filters.
- **Hypothesis**: Median filter + Sobel/Laplacian will yield the most interpretable defect edges.

---

4. Results & Analysis

4.1 Visual Results

- **Mean filter**: Successfully reduced Gaussian noise but blurred defect edges.
- **Median filter**: Removed salt-and-pepper noise while preserving sharp defect boundaries.
- **Mode filter**: Preserved background uniformity but less effective for small isolated defects.
- **Sobel filter**: Produced clear directional edges but missed fine details.
- **Laplacian filter**: Detected small scratches but also highlighted unwanted texture noise.
- **Sobel + Laplacian**: Best overall, producing sharp boundaries around defects.

4.2 Comparative Discussion

- **First-order vs Second-order**: Sobel was more stable under noise, while Laplacian captured finer details but exaggerated noise.
- **Smoothing before sharpening**: Median + Sobel combination gave the cleanest defect outlines.
- **Noise-specific effectiveness**:
  - Gaussian noise → Mean filter worked best.
  - Salt-and-pepper noise → Median filter clearly superior.
  - Mode filter → Useful for repetitive patterns.

---

5. Conclusion

This study demonstrated the importance of spatial filtering in enhancing manufacturing defect images:

- **Sobel filter** effectively captured directional defect edges.
- **Laplacian filter** amplified subtle features but also noise.
- **Median filter** was the most robust smoothing technique, especially against salt-and-pepper noise.
- The **combined pipeline (Median + Sobel + Laplacian)** yielded the most interpretable results, balancing noise reduction with edge enhancement.

Such an approach can support **automated defect detection systems** in quality assurance pipelines, ensuring that manufacturing faults are detected early and reliably.

---

6. Appendix: Code Implementation

A well-documented **Google Colab notebook** was implemented, integrating:

- Image upload and preprocessing.
- Noise injection (Gaussian and salt-and-pepper).
- Smoothing filters (Mean, Median, Mode).
- Sharpening filters (Sobel, Laplacian, Sobel+Laplacian).
- Combined smoothing-sharpening pipeline.
- Interactive dashboard with widgets for filter selection.

The code allows dynamic experimentation with different filters and parameters, and visualizes results alongside histograms for detailed analysis.

*Full implementation code is provided in the attached Colab notebook Also can find on **github**.*