1. Component Extraction

● A. Circle Detection (circle.png)

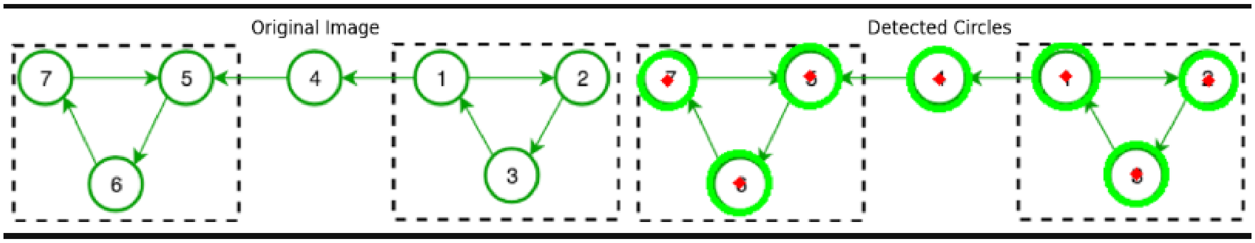
▪ Technique Used:

- **Gaussian Blur**: To reduce image noise.
- **Hough Circle Transform**: To detect circular shapes using edge information.

▪ Why this technique?

- Circles are defined by smooth curves. Hough Circle Transform detects these patterns efficiently after noise is reduced.

▪ Output:



▪ Discussion:

- **Pros**: Accurate for clear, well-lit circles.
- **Cons**: Fails with overlapping or low-contrast circles.
- **Quality**: Good with clear input.
- **Failure Reasons**: Noise, poor lighting, small circles.
- **Suggestions**: Try Canny edge detection before Hough or tune `param2`.

● B. Chart Component Detection (covid.png)

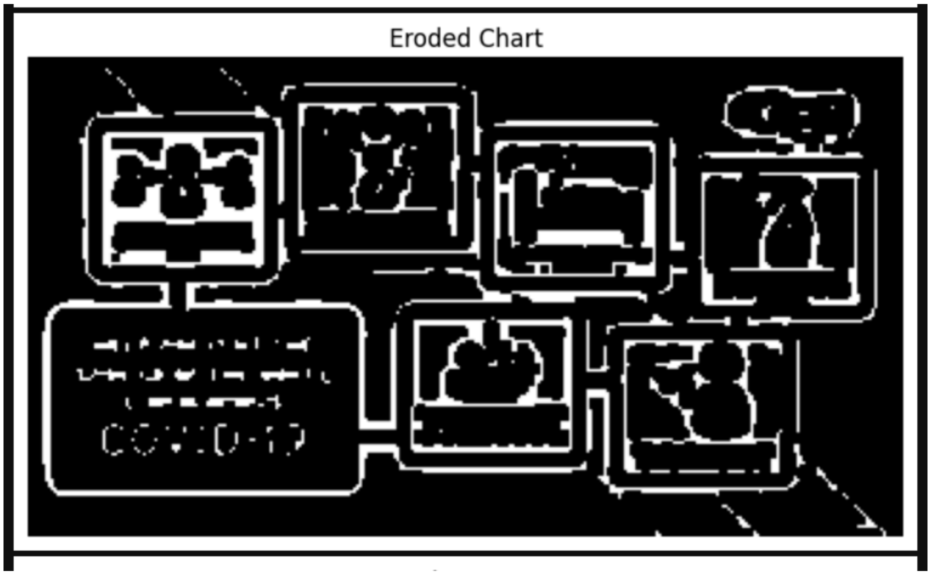
▪ Technique Used:

- **Grayscale → Adaptive Threshold → Morphology → Contours**

▪ **Why this technique?**

- The chart has complex, varying brightness. Adaptive thresholding handles that better than global thresholding.

▪ **Output:**



▪ **Discussion:**

- **Pros**: Works well on detailed line charts.
- **Cons**: May detect noise as a component.
- **Quality**: Medium to high, depending on contrast.
- **Failures**: Tiny text or overlapping shapes.
- **Suggestions**: Use dilation + erosion combo or remove small contours by area.

---

2. Blurry Image Enhancement

● A. Buildings

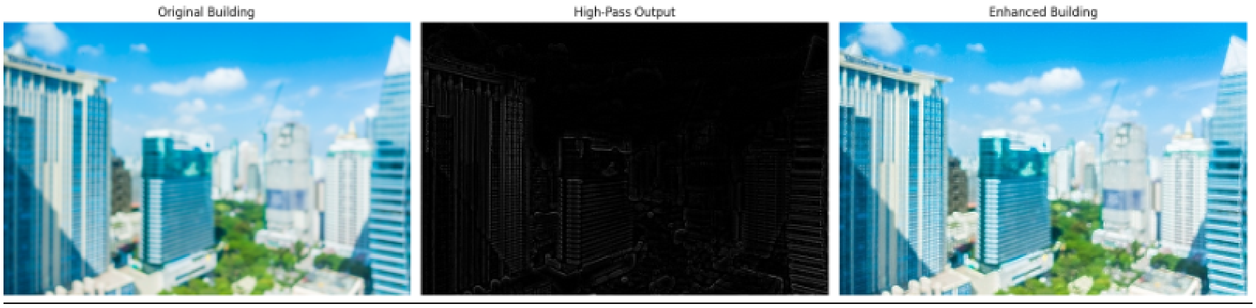
▪ Technique Used:

- **High-Pass Filter**: Edge detection via convolution kernel.

▪ Why this technique?

- Highlights edges by subtracting low-frequency components.

▪ Output:



▪ Discussion:

- **Pros**: Sharpens details and edges effectively.
- **Cons**: Amplifies noise if present.
- **Quality**: Medium to high.
- **Failures**: Blurry + noisy image worsens.
- **Suggestions**: Combine with denoising before sharpening.

● B. Dog

▪ Technique Used:

- **Sharpening Kernel + Image Blending**: Smooth but enhanced.

▪ Why this technique?

- Balanced sharpening while preserving natural texture.

▪ **Output:**

Original Dog

Sharpened Output

Blended Enhancement

▪ **Discussion:**

- **Pros**: Clean result, balanced sharpness.
- **Cons**: Subtle enhancement may not be visible.
- **Quality**: Good for soft features like fur.
- **Suggestions**: Tune weights or apply CLAHE first.

---

3. Noise Removal

● Images: text.png, rocket.png, wind.png

▪ **Techniques Used:**

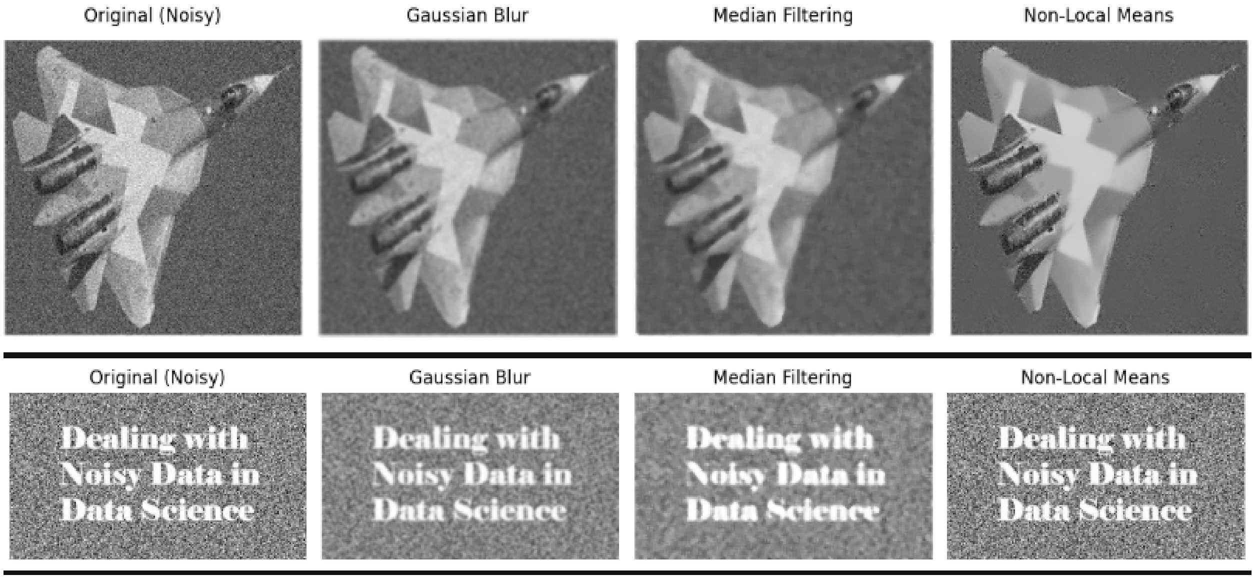
1. **Gaussian Blur**
2. **Median Blur**
3. **Non-Local Means (NLM)**

▪ **Why?**

- Each method suits different types of noise:
  - **Gaussian**: Reduces overall noise.

    - **Median**: Best for salt-and-pepper noise.
    - **NLM**: High-quality denoising, keeps edges.

▪ **Output:**


Original (Noisy)
Gaussian Blur
Median Filtering
Non-Local Means
Original (Noisy)
Gaussian Blur
Median Filtering
Non-Local Means
Dealing with Noisy Data in Data Science
Dealing with Noisy Data in Data Science
Dealing with Noisy Data in Data Science
Dealing with Noisy Data in Data Science

▪ **Discussion:**

- **NLM**: Best quality, edge-preserving.
- **Gaussian**: Blurs image.
- **Median**: Simple and fast for salt noise.
- **NLM Slow** on large images.
- **Suggestions**: Auto-select method based on noise type.

4. Visual Enhancement

● A. Newspaper

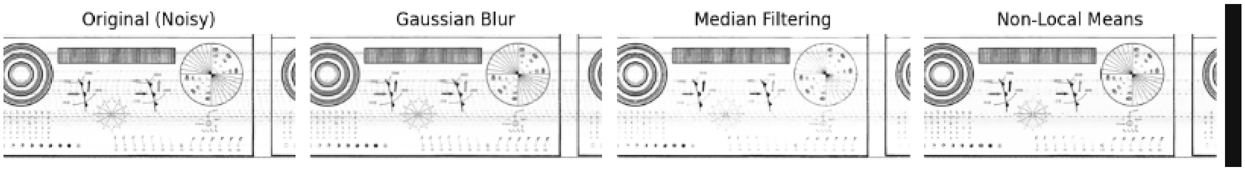
▪ **Technique Used:**

- **Blur + Sharpening Combo**

▪ **Why?**

- Blur removes background noise; sharpening restores text edges.

▪ **Output:**



Original (Noisy)

Gaussian Blur

Median Filtering

Non-Local Means

▪ **Discussion:**

- **Pros**: Text becomes readable, noise reduced.
- **Cons**: May still blur tiny fonts.
- **Suggestions**: Use bilateral filter or histogram equalization.

● B. Name Plate

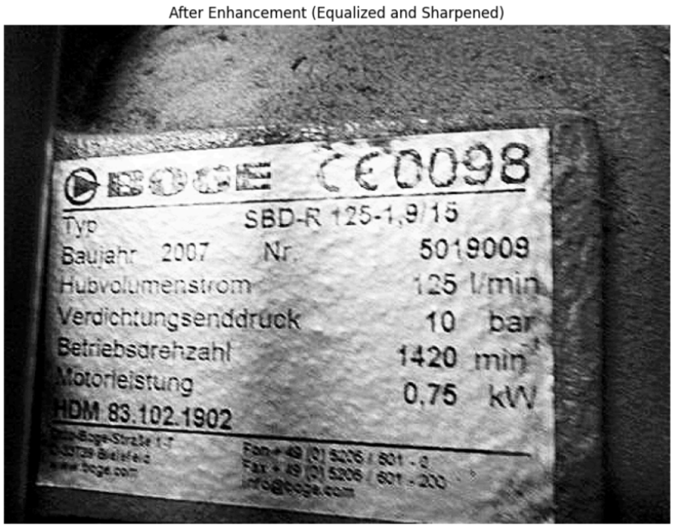
▪ **Technique Used:**

- **Histogram Equalization + Sharpening**

▪ **Why?**

- Enhances contrast of faded letters, then sharpens them.

▪ **Output:**



After Enhancement (Equalized and Sharpened)

▪ **Discussion:**

- **Pros**: Letters pop out.
- **Cons**: Can overexposed bright areas or it’s still not clear enough.
- **Suggestions**: Use CLAHE (local histogram equalization).

Conclusion

This project explored different enhancement and cleaning techniques using OpenCV. Key takeaways:

- Use the right method for the type of blur or noise.
- Combining techniques (e.g., denoise + sharpen) often gives best results.

Future Suggestions

- Automate choosing techniques using image statistics.
- Try deep learning models for image restoration (like DnCNN).
- Explore CLAHE and bilateral filters for better contrast.