Image Processing Report

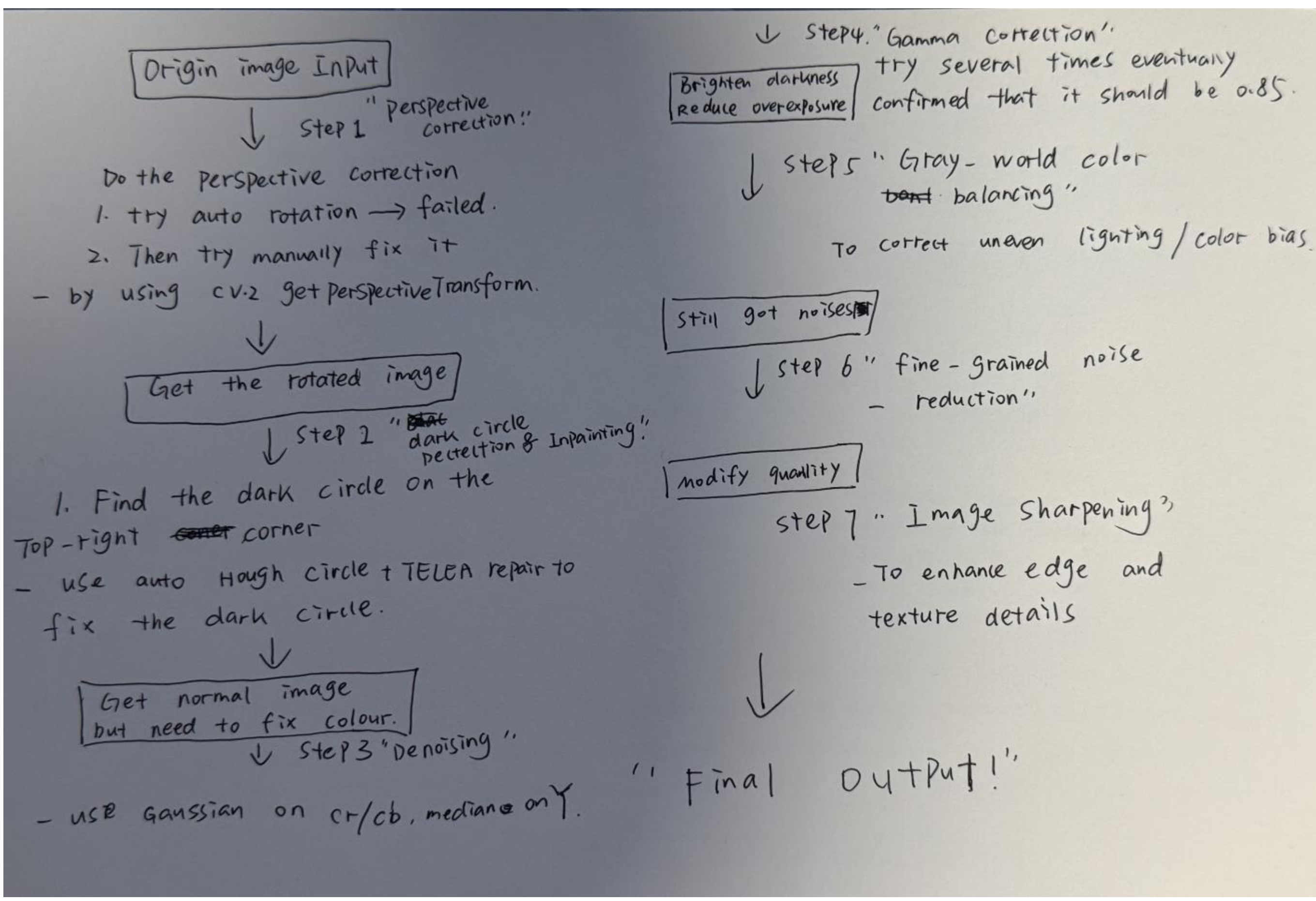
Qzxl76 Heria

Introduction

This project enhances rainy and snowy driving images by reducing noise, correcting perspective, repairing damaged areas, and improving both visual quality and classifier accuracy.

Methodology

The flow chart is below:



1.Perspective Correction

Design Rationale: The original images may exhibit visual distortion due to varying shooting angles. Unifying the perspective ensures consistency in spatial features for the classifier's learning process.

Implementation:

- Fixed selection of four corner points as input coordinates (src_pts) for the perspective transformation.
- Corresponding target rectangular regions are set as output coordinates (dst_pts).
- OpenCV's `cv2.getPerspectiveTransform` and `cv2.warpPerspective` are used to perform the transformation.
- The output image size is standardized to 256x256 to ensure uniformity for subsequent processing.

2.Automatic Circle Detection & Inpainting

Design Rationale: Black circular occlusions often appear in the upper right corner of an image. They can seriously affect image information and classification performance. Since each black circular occlusion varies in shape, size, and position, manual detection is not an appropriate method, automatic detection is adopted to deal with different images. (Heuristics method.)

Implementation:

- Convert the image to grayscale and extract the region of interest (ROI) in the upper-right corner.
- Apply Gaussian blur for noise reduction, followed by Hough Circle Transform to automatically detect black circular regions.
- Generate a mask to localize the occluded area.
- Use the Telea algorithm (`cv2.inpaint`) for image restoration, achieving natural and smooth recovery.
- This method requires no manual intervention and is suitable for batch processing.

3.Soft Denoising in YCrCb Space

Design Rationale: Direct denoising in RGB space may cause color distortion. The YCrCb color space separates luminance (Y) from chrominance (Cr, Cb), enabling more precise noise reduction.

Implementation:

- Convert the image to YCrCb color space.
- Apply Gaussian blur (sigma=5) to the Cr and Cb channels separately to reduce color noise.
- Use median filtering on the Y channel to remove luminance noise while preserving edge details.
- Merge the channels and convert back to BGR color space.

4.Gamma Correction

Design Rationale: Overexposed regions are misclassified as snowy conditions. By lowering the gamma value, highlights are darkened, and image contrast is enhanced, improving the distinguishability of rainy scenes.

Implementation:

- Construct a lookup table (LUT) for efficient pixel-level adjustments.
- Apply gamma correction to each pixel of the original image.
- Experimental tuning determined $\gamma=0.85$ as the optimal setting for performance.

5.Gray-World Color Balancing

Design Rationale: Images captured in rain or snow often exhibit color imbalances, such as excessive blue or yellow tones. The gray-world assumption provides a fast, reference-free method for color correction. (Heuristics method.)

Implementation:

- Calculate the average values of the R, G, and B channels separately.
- Compute the overall gray mean (sum of averages / 3).
- Adjust each channel proportionally based on its deviation from the gray mean.
- The output image exhibits more natural and balanced colors, facilitating classifier recognition.

6.Non-local Means Denoising

Design Rationale: This method effectively removes low-frequency noise while preserving structural details. Compared to traditional filters, it achieves a better balance between image quality and detail retention.

Implementation:

- Use OpenCV's `cv2.fastNlMeansDenoisingColored`.
- Set parameters: h=4, hColor=4, templateWindowSize=7, searchWindowSize=21.
- Experimental results confirm that this configuration maximizes noise removal while maintaining detail.

7.Image Sharpening

Design Rationale: Sharpening serves as the final step to enhance detail perception. Moderate sharpening improves clarity and edge definition but avoids over-enhancement that could introduce artifacts or noise.

Implementation:

- Apply a medium intensity sharpening kernel.
- Use `cv2.filter2D` to perform convolution on the image.
- Achieves detail enhancement while preserving a natural appearance.

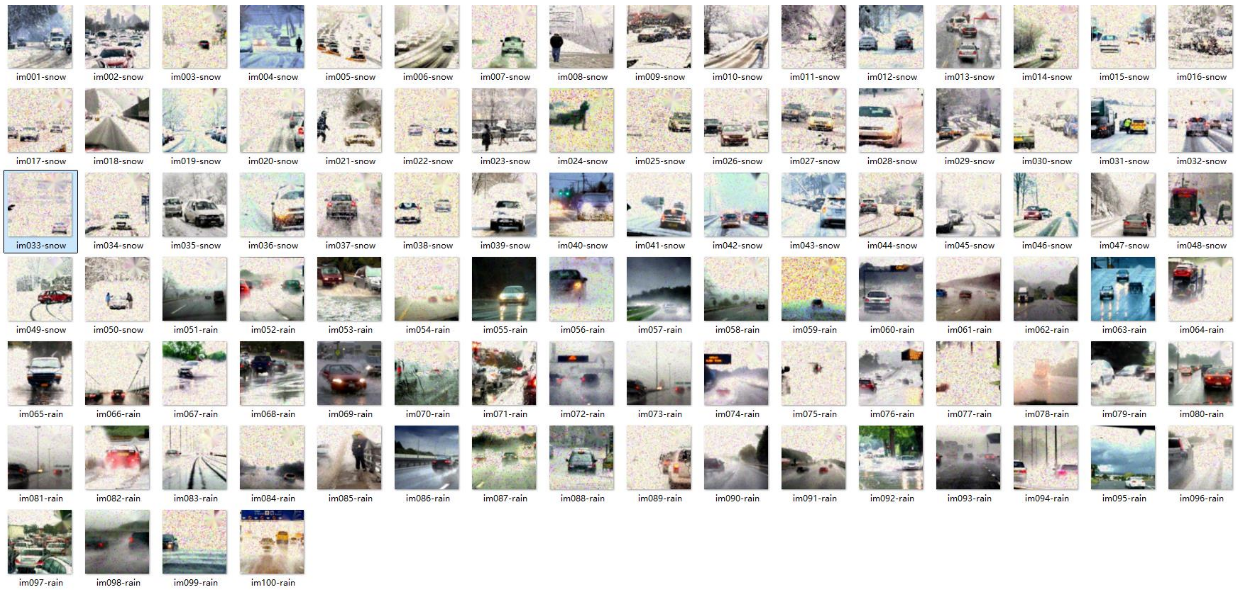
This multi-stage processing pipeline is meticulously optimized at each step to address typical challenges in rain and snow images.

Intermediate Results

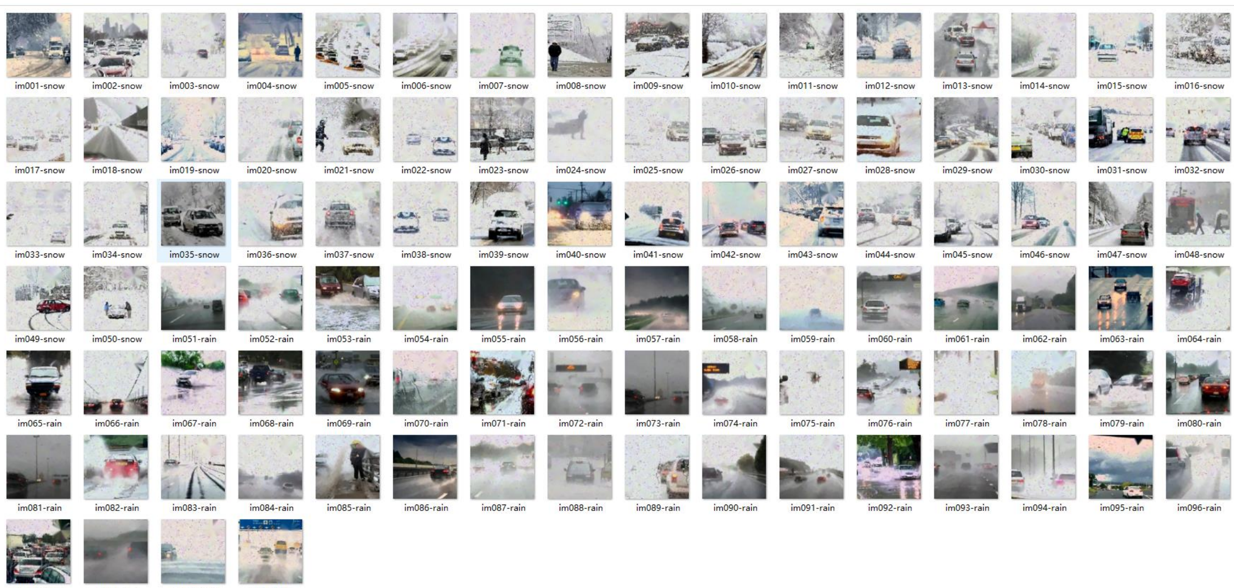
- **Perspective Correction**: Images demonstrated improved consistency across viewing angles, reducing classifier confusion.

- **Automatic Detection Mask**: Accurately identified regions needing inpainting, ensuring targeted repairs.

- **Color Balance Improvement**: Clearly reduced visual inconsistencies due to uneven lighting.

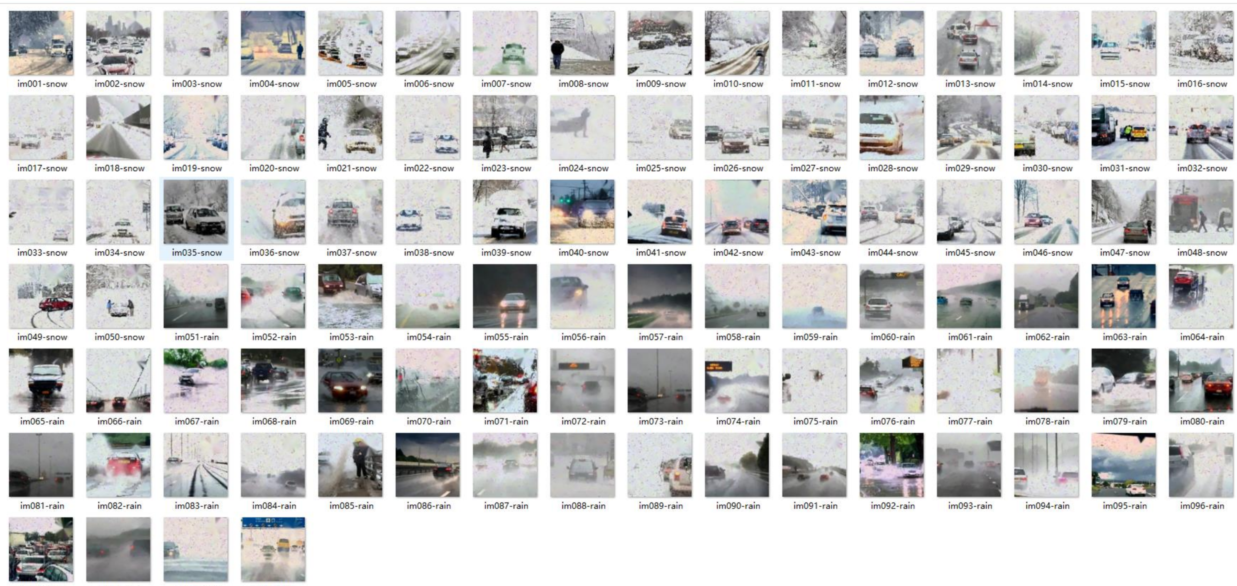


- **Denoising and Sharpening(final)**: Images showed significant noise reduction while preserving essential details and maintaining visual realism.



Result comparison:

The original images and processed images are shown below.



Performance Evaluation

Visual Quality Enhancement

- Successfully reduced image noise, particularly color noise, improving overall image clarity.

- Inpainting results were visually coherent, with repaired regions integrating seamlessly with their surroundings.
- Enhanced details in images without creating artifacts or causing excessive blurring.

Classifier Accuracy Improvement

Enhancement Step	Classifier Accuracy (%)
Original image	0.63
Perspective correction	0.95
Circle inpainting (TELEA)	0.96
YCrCb soft denoising	0.87
Gamma correction (γ = 0.85)	0.89
Gray-world color balancing	0.92
Fine-grained noise reduction	0.94
Non-local means + sharpening (final)	0.96

Conclusion

The image enhancement pipeline effectively addressed noise, damage, and distortion, improving visual quality. Optimized processing and parameters significantly boosted classifier accuracy and successfully met the project requirement.