Objectives

1. Finalize the Threshold CLAHE process by ensuring its compatibility with the normal CLAHE output.
2. Resolve limitations with Split & Merge and calibration data for better functionality and clearer output.
3. Advance the dark line detection by adding functionalities to show and manage detected line data and selectively remove lines.
4. Reorganize the UI flow for better access to pre-processing and core functions.
5. Refine the information display for line detection, including auto-sizing, scrollable content, and selective line removal.

Activities

1. **Threshold CLAHE Process Optimization:**
    - Solved the issue where Threshold CLAHE couldn't apply correctly after normal CLAHE. Updates include:
        - Reused the hasCLAHEBeenApplied flag to store the state of normal CLAHE application.
        - Introduced a preProcessedImage variable to save raw data before normal CLAHE processing.
        - Updated normal and Threshold CLAHE functions:
            - Normal CLAHE now uses preProcessedImage for consistent data.
            - Threshold CLAHE references the hasCLAHEBeenApplied flag to determine whether to use original or processed data.
        - Adjusted clip limit for Threshold CLAHE by doubling it when normal CLAHE is applied, ensuring that non-processed areas retain the user-defined values.
2. **Split & Merge Functionality Fix:**
    - Resolved an issue with Split & Merge not using processed calibration data:
        - Replaced oriImage with finalImage during the split process to ensure accurate data usage.
3. **Dark Line Detection Enhancements:**
    - Created a demo using a newly provided raw file to showcase the dark line detection.

- Began implementing additional features:
  - Added functionality to display the count of detected and removed lines.
  - Set up a line information display box that auto-resizes up to a max height, activating a scrollbar if content exceeds space.
  - Integrated an option to selectively delete lines (Delete All Lines, Delete In Object Lines, or Delete Isolated Lines) using removeDarkLinesSelective. Currently configuring the in-object line removal function for complete flexibility.

4. **UI Flow Update for Image Processing:**
   - Reorganized buttons into a Pre-Processing Operations section for better workflow, containing Calibration and Split & Merge options.
   - Moved interlacing operations (e.g., Rotation, Zoom, and Crop) into Basic Operations to streamline user navigation.
5. **Line Information Box Update:**
   - Developed an auto-fitting information box for line detection details, expanding up to a set maximum height and adding scroll functionality if content overflows.
   - Configured the box to appear only when dark line detection or removal functions are triggered, making it context-sensitive and uncluttered.

Achievements

1. Resolved Threshold CLAHE application logic, ensuring consistent output when combined with normal CLAHE and improved control over clip limits.
2. Improved Split & Merge function accuracy by correctly using processed image data.
3. Enhanced dark line detection with better information output, selective deletion options, and an adaptable information display box.
4. Simplified the UI flow by grouping pre-processing tasks, improving user experience and accessibility.
5. Created a responsive line information display, which enhances clarity and usability during dark line detection and removal.

Problems and Solutions

Problem 1: Threshold CLAHE failed to apply correctly if normal CLAHE was previously used.
Solution 1: Introduced a flag and a preProcessedImage variable to track and manage the CLAHE states, including adjustments to clip limits for clarity in non-processed areas.

Problem 2: Split & Merge relied on uncalibrated image data, leading to inconsistencies.
Solution 2: Updated the function to use finalImage, ensuring that it references processed calibration data.
Problem 3: Information overload in line detection display due to inconsistent sizing and content overflow.
Solution 3: Developed an auto-sizing, scrollable line information box that appears only when detection functions are activated, providing a clean and functional UI.
Problem 4: Incomplete selective dark line removal, with in-object lines unable to be deleted.
Solution 4: Began creating an individual function to handle selective line removal, improving control over which lines are retained or removed.