1. Objectives:

- Resolve issues with the histogram graph display and enhance user interaction features.
- Address state management and logic issues with CLAHE to allow sequential applications of normal and threshold CLAHE.
- Modularize the CLAHE code into separate files for better structure.
- Implement functionality for detecting and removing dark lines in bright regions.
- Identify and address issues in the dark line detection and removal process.

2. Activities:

- **Histogram Graph Enhancements:**
    - Resolved the histogram display issue using a "Toggle Histogram" button.
    - Added tooltips to display pixel and frequency values on the enlarged histogram graph.
    - Implemented a zoom function for the enlarged histogram graph, allowing zoom with mouse scroll and reset to the original scale with a button.
- **CLAHE Sequential Application Issue:**
    - **Issue 1: State Management** – When normal CLAHE was applied, finalImage was modified with no mechanism to preserve original pixel values.
    - **Issue 2: Logic** – After normal CLAHE, pixels were normalized, meaning dark areas might no longer meet the threshold condition for further processing.
    - **Solution Status:**
        - **Status 1:** Added state tracking to store the image state before any CLAHE operation. Added a flag to track CLAHE applications.
        - **Status 2:** Adjusted the logic to use the preserved original values when applying threshold CLAHE after normal CLAHE.
        - **Status 3:** Ensured normal CLAHE stores the pre-CLAHE state, while threshold CLAHE checks the flag to apply based on the preserved image.
        - **Status 4:** Used CLAHE-processed pixel values for threshold-based decisions.
        - **Status 5:** Enhanced adaptive threshold calculations and boosted CLAHE, sharpening kernel, and blended values by 50% for a more pronounced effect.
        - **Status 6:** Achieved a clear output with successful application of threshold CLAHE after normal CLAHE.

- **CLAHE Code Modularity:**
    - Split the CLAHE code into CLAHE.h and CLAHE.cpp files.
    - **Issue:** Threshold CLAHE could not apply after normal CLAHE when using the modularized code.
    - **Temporary Solution:** Reverted to a functioning codebase until the modularized version is debugged.
- **Dark Line Detection and Removal Functionality:**
    - **Detection of Dark Lines:**
        - **Bright Region Detection:** Uses isInBrightRegion() to verify if a pixel lies within a bright area, examining a 5x5 window and requiring at least 70% bright pixels.
        - **Dark Line Search:** Implements Depth-First Search (DFS) for tracking connected dark pixels, focusing on bright regions and tracking line thickness.
        - **Line Refinement:** Merges close line segments, calculates line properties (start, end, thickness).
    - **Removing Dark Lines:**
        - Calculates line vectors (dx, dy) and processes points along the line at small intervals.
        - For each point, considers pixels perpendicular to the line, interpolating to fill in dark line pixels based on thickness.
    - **Optimization:** Bright region detection uses multiple threads, enhancing performance by distributing image portions across threads.
    - **Visualization:** Detected lines are visualized in red, numbered for reference, and detailed in a message box.
    - **Issues Identified in Dark Line Detection and Removal:**
        - **Issue 1:** Thick dark lines are not detected.
        - **Issue 2:** Dark lines that don't start at the x-coordinate zero may be skipped.
        - **Issue 3:** Minimum detected line length needs to be 1.
        - **Issue 4:** Parameters are not currently stored in the parameters header.
        - **Issue 5:** Information on detected lines needs clearer output in the info label.

3. Achievements:

- Successfully implemented a toggle feature for the histogram graph with additional tooltip and zoom functionality for enhanced user interaction.
- Resolved the sequential CLAHE application issue, enabling threshold CLAHE to apply effectively after normal CLAHE.
- Completed initial implementation of dark line detection and removal functions, with multi-threading for efficiency.
- Visualized detected lines for easy reference, enhancing usability and debugging.

4. Problems & Solutions:

- **Problem 1:** CLAHE threshold failed to apply after normal CLAHE due to state management and logic issues.
    - **Solution:** Implemented state tracking and logic adjustments to preserve original pixel values and verify conditions, allowing proper sequential application.
- **Problem 2:** Thick dark lines in bright regions were not detected.
    - **Solution:** Refine the detection algorithm to handle varying thicknesses, potentially adjusting parameters for detecting connected pixel regions.
- **Problem 3:** Some dark lines were skipped due to non-zero x-coordinate origins.
    - **Solution:** Modify the DFS or line search algorithm to account for starting coordinates and ensure comprehensive line detection.
- **Problem 4:** CLAHE code modularization caused issues with threshold CLAHE application.
    - **Solution:** Investigate dependencies and function calls in the separated files to resolve application issues before re-integrating the modularized code.