**Objectives**

1. Complete the dark line removal feature by adding isolated and in-object line removal options, incorporating new stitching and neighbor methods.
2. Refine the line information display to enhance the visibility of line details, removal progress, and current image dimensions.
3. Implement and test the interlace processing method with flexible start points and energy-based segmentation.
4. Update the UI to provide intuitive options for dark line removal, interlace processing, and merging interlaced results.

---

**Activities**

1. **Dark Line Removal Enhancements:**
    - **In-Object Line Removal:** Created a dedicated function allowing lines in objects to be removed either by:
        - **Neighbor Method:** This method removes line sections in increments (40 pixels per pass), applying surrounding values to fill gaps.
        - **Stitch Method:** This method connects pixels across the removed line to "stitch" the image together, either moving pixels up (for horizontal lines) or inward (for vertical lines).
        - **Sequential Removal:** Resolved issues of removing multiple lines in a single pass by implementing removeDarkLinesSequential() for the stitch method removing method, enabling the removal of individual lines as the user selects.
    - **Isolated Line Removal:** Addressed interference from removeDarkLinesSequential() by using removeDarkLineSelective() for isolated lines, providing clearer line separation.
2. **Line Information Display Update:**
    - Enhanced the line information box to show the coordinates of detected lines, distinguishing between remaining and removed lines.
    - Updated the box format to display:
        - Line coordinates and statuses (detected, removed, or remaining).
        - Real-time feedback as lines are removed, with the box highlighting only the remaining lines.

- Improved the update mechanism by eliminating resetDetectedLines() after line removal and directly modifying m_detectedLines.

3. **Control Panel Improvements for Dark Line Processing:**
   - Added a pop-up for selecting the line removal method (neighbour or stitch) when selecting in-object line removal.
   - Configured single-selection mode to avoid multi-line processing, enhancing control over line management.
   - Added the "Image Size:" field to display the current pixel dimensions, helping users assess the effect of removal actions on the image dimensions.
4. **Interlace Processing Implementation:**
   - **Initial Setup:** Created an energySection Enum to define low- and high-energy regions for interlacing and set an InterlaceStartPoint variable.
   - **UI Integration:** Added an "Interlace" button in the Pre-Processing Operations group with options for starting points and stretch factors, making configuration easy.
   - **Function Design:**
     - Low energy sections start from LeftLeft, alternating between LeftLeft and LeftRight for rows, per the configured interlace pattern.
     - Designed the interlace processing to show results on the control panel (with left and right energy segregation) and in a separate display_window, including save and zoom functionalities.
   - **Control Panel Updates:** Users can select a start point and adjust the stretch factor, with a descriptive note provided for guidance.
5. **Interlace and Merge Testing:**
   - Tested current Split & Merge functions on interlaced results, successfully merging both parts using the weighted average method.
   - Preparing to create a dedicated "Merge" button to improve functionality separation from the existing "Split & Merge" button.

---

**Achievements**

1. Successfully completed in-object and isolated dark line removal with flexible method selection, enhancing user control.
2. Improved line information visibility by updating the display format and refresh mechanisms, ensuring accuracy and ease of navigation.

3. Integrated interlace processing with a user-friendly interface for energy-based segmentation, adding flexibility to the image processing workflow.
4. Verified the compatibility of interlaced results with the current Split & Merge method, paving the way for further refinement in image merging.

---

**Problems and Solutions**

**Problem 1:** In-object lines with large widths weren't removed in one step using the neighbour method.

- **Solution 1:** Adjusted to remove 40 pixels at a time and provided a stitch method as an alternative, allowing continuous filling across larger sections.

**Problem 2:** Line information box didn't display real-time updates for removed lines, leading to confusion.

- **Solution 2:** Removed unnecessary resets and updated m_detectedLines directly, showing only the remaining lines after each removal.

**Problem 3:** The stitching method for dark line removal occasionally causes unnatural visual artifacts.

- **Solution 3:** Added removeDarkLinesSequential() to process lines individually based on user selection, avoiding excessive distortion.

**Problem 4:** UI control layout became overly complex with new line processing and interlacing features.

- **Solution 4:** It was streamlined with separate pop-ups and tooltips, added clear groupings and notes for each function, and improved user experience.