**Task Progress Update Report**

**Name**: LIM SHI KAI (Sky)  
**Update Date**: 28-11-2024

# **Overview of Assigned Tasks**

**Task 1: Display Window for Interlace Result**

**Objective:** Enable users to tick a UI box to display interlace results.

**Current Status:** Completed  
**Details:**

* Implemented a display window for interlace results.
* Added a tick box in the UI, allowing users to toggle the display window.

**Task 2: DarkLine Conversion to Double 2D Pointer Method**

**Objective:** Convert the DarkLine implementation from std::vector to a double 2D pointer method for better performance.

**Current Status:** Completed  
**Details:**

* Converted DarkLine storage to a double 2D pointer (DarkLine\*\*).
* Implemented custom memory management using RAII principles.
* Enhanced data access with row-wise and column-wise optimizations.
* Updated related functions, such as removeDarkLinesSequential, detectLines, and helper methods (drawLineLabelWithCount).
* Integrated a new structure DarkLineArray to manage pointer-based lines.

**Task 3: Line Removal Methods and Updates**

**Objective:** Enable selective and sequential removal of detected lines using neighbour values or direct stitch methods.  
**Current Status:** Completed  
**Details:**

* Introduced a SelectedLines struct for pointer-based line handling.
* Updated removeDarkLinesSequential to process user-selected lines.
* Enhanced error handling and state preservation for Direct Stitch operations.
* Fixed issues in removeDarkLinesSequential and added workflows for neighbour-based and isolated line removals.

**Task 4: UI Enhancements for Line Detection and Removal**

**Objective:** Update the UI to better integrate pointer-based line detection and removal methods.

**Current Status:** Completed  
**Details:**

* Combined detection, removal, and reset functionalities into a unified control panel.
* Added buttons for resetting detected lines per method.
* Enhanced the line information box to display dynamic updates on detected and removed lines.
* Improved parameter bar responsiveness with dynamic size adjustments.

**Task 5: Edge Enhancement Functionality**

**Objective:** Provide users with adjustable edge enhancement tools using the Sobel operator.  
**Current Status:** Completed  
**Details:**

* Added controls to adjust gamma, sharpness, and contrast in a single button.
* Implemented regional adjustment options requiring pre-selection of the region.

**Task 6: Weighted Average for Merging Methods**

**Objective:** Dynamically adjust weights for merging methods.  
**Current Status:** Completed  
**Details:**

* Added dynamic weight calculation options: STATIC, INTENSITY\_BASED, GRADIENT\_BASED, and VARIANCE\_BASED.
* Updated UI to reflect selected methods and parameters.

**Task 7: Debugging and Testing**

**Objective:** Identify and fix bugs in calibration, interlacing, and merging processes.  
**Current Status:** In Progress  
**Details:**

* Debugged processYXaxis function but retained the original implementation due to minimal improvements.
* Investigated merging issues but found no effective solutions.
* Restructured and cleaned code for better readability and functionality.

1. **Roadblocks/Challenges**

* **Direct Stitch Issues:** Memory management inconsistencies were resolved by restructuring workflows and implementing robust error handling.
* **Y-Axis Calibration:** Detected issues with missing lines during Y-axis calibration; still under investigation.
* **Merging Process:** Loss of key details post-calibration and merging remains unresolved.
* **Pointer-Based Line Detection:** Frequent crashes due to pointer linkage; resolved with improved memory safety mechanisms.

### **Conclusion**

* Successfully transitioned DarkLine methods to double 2D pointers for improved performance.
* Implemented UI enhancements for better user interaction and feedback.
* Enhanced line detection and removal methods for versatility and accuracy.
* Debugged and tested various functionalities, including edge enhancement and dynamic weighted merging.
* Remaining tasks focus on resolving calibration and merging issues while ensuring robustness and performance in the final implementation.