Task Progress Update Report

Name: LIM SHI KAI (Sky)

Update Date: 03-01-2025

1. Overview of Tasks

Task 1 : CLAHE Enhancements and Optimization (Memory related)

Objective: Enhance CLAHE processing for improved performance and usability.

Status: Done

Details:

- Migrated CLAHE functions to the `malloc2D` method for efficient memory allocation.
- Simplified and optimized CLAHE functionality, ensuring accurate normalization and scaling for 16-bit to 8-bit image data.

Task 2 : UI Functionality Improvements

Objective: Enhance usability by addressing functionality and state management issues in the application UI.

Status: Done

Details:

- Solved cursor issues by simplifying state management and removing redundant variables.
- Resolved bugs related to zoom functionality and updated warning logic for better user guidance.
- Improved the "Clear All" and "Reset All" functions to fully reset image data and UI elements, albeit still debugging crashes caused by mixed memory management methods.

Task 3 : Pre-processing and Auto-calibration Features

Objective: Enhance automation in pre-processing steps for user convenience.

Status: Done

Details:

- Added a "Process Image" button to integrate calibration, interlacing, and merging in a single step.
- Introduced an auto-calibration tracker with user options to reapply calibration as needed.
- Updated interlace functionality to support fixed energy and row modes with clear calibration parameters.

Task 4 : Memory Optimization

Objective: Enhance automation in pre-processing steps for user convenience.

Status: Done

Details:

- Replaced `new/delete` with `malloc2D` for consistent memory handling.
- Optimized memory copying operations using `memcpy`, improving performance and reducing overhead during CLAHE and other processing tasks.

Task 5 : Memory Optimization through Single Loop Implementation

Objective: Replace nested loops with single-loop operations for efficient memory usage.

Status: Done

Details:

- Optimized data copying by replacing double loops with single-loop `memcpy` operations.
- Applied the same optimization principles across CLAHE-related functions to improve performance and reduce computational overhead.
- Enhanced the `applyCombinedCLAHE` function to leverage GPU acceleration and improve cache utilization.

Task 6 : Development of CLAHE_uint32

Objective: Implement CLAHE processing for unsigned 32-bit integer data.

Status: In Progress

Details:

- Created a new class, `CLAHE_uint32`, to handle 32-bit integer data.
- Established workflows for converting 32-bit data to 8-bit for processing and back to the original range.

- Separated CLAHE implementations for double 2D pointers and 32-bit integer pointers to ensure compatibility and prevent crashes.

2. Roadblocks/Challenges

- Debugging CLAHE_uint32 for seamless integration and ensuring accurate scaling of data.
- Refining CLAHE normalization to avoid black image outputs in edge cases.
- Further testing required for optimized single-loop implementations across all functions

3. Conclusion

- Significant progress in CLAHE optimization and implementation across different data types.
- Improved UI functionality with bug fixes and enhanced state management.
- Memory usage optimized through single-loop implementations, improving processing efficiency.
- Progress made in developing `CLAHE_uint32` for unsigned 32-bit integer data integration.
- Pre-processing and auto-calibration features updated for better user experience.
- Continued development of internship report, Gantt chart, and presentation slides.
- Focus on resolving crashes and ensuring accurate data scaling in CLAHE functions.