Objectives

- Optimise the file loading process and implement a mechanism to display loading or processing times.
- Transition CLAHE functions to a double 2D pointer method and resolve related processing issues.
- Enhance the UI and functionality for image loading and saving, with support for multiple formats.
- Debug and refine threshold CLAHE and GPU-based CLAHE processes.

Activities

- **Progress Bar and Processing Time Display:**
  - Attempted to add a progress bar for file loading but removed it due to crashes.
  - Implemented processing time display in the parameters box.
- **File Loading Optimisations:**
  - Reduced file loading time from 12–14 seconds to 1–3 seconds by:
    - Buffered reading with a 64KB buffer.
    - Pre-allocating memory for vectors.
    - Efficient parsing and minimised string operations.
    - Binary mode reading for faster performance.
  - Enhanced 16-bit value handling:
    - Clamped values to [0, 65535].
    - Robust error handling for invalid data.
    - Improved handling of file format variations and memory safety.
- **CLAHE Function Refactoring:**
  - Transitioned CLAHE methods to use a double 2D pointer structure.
  - Updated `image_processor.h` and UI for compatibility with the new method.
  - Normalised input data to [0,1], applied CLAHE processing, and rescaled output to resolve black image issues.
- **GPU CLAHE Debugging:**
  - Resolved `GpuMat::copyTo` issues by addressing:
    - Mask depth mismatches (converted masks to CV_8U).
    - Binary mask requirements (thresholded masks to 0/255).
    - Mask size mismatches (ensured dimensions matched).
  - Improved validation for GPU operations.

- **Threshold CLAHE Enhancements:**
    - Added new features for threshold-based CLAHE:
        - `DarkPixelInfo` structure for storing dark pixel data.
        - Multithreaded CPU implementation for improved performance.
        - Memory management handled directly in `CLAHEProcessor`.
    - Updated `control_panel.cpp` to integrate CLAHE functions and error handling.
    - Debugged and resolved black output issues by addressing input normalization and buffer management.
- **Image Loading and Saving Enhancements:**
    - Added new functions:
        - `loadImage`: Unified function for loading both text files and image formats (JPG, PNG, TIFF, BMP).
        - `isImageFile`: Helper function to verify supported image file formats.
    - Enhanced UI for loading and saving files:
        - Added file type filters in the dialog.
        - Supported formats for saving: PNG, JPEG, TIFF, BMP.

Achievements

- Reduced file loading time significantly through optimisations.
- Successfully transitioned CLAHE functions to a double 2D pointer structure with improved accuracy.
- Resolved issues with GPU CLAHE processing and threshold CLAHE outputs.
- Enhanced UI for robust support of multiple image file formats in loading and saving.
- Improved debugging and error handling for CLAHE functions and file handling.

Problem & Solution

- **Problem:** Progress bar caused crashes during file loading. **Solution:** Replaced it with a processing time display in the parameters box.
- **Problem:** Black output in CLAHE GPU processing. **Solution:** Normalised input data, fixed mask depth issues, and ensured binary masks.

- **Problem:** Slow file loading time.
  Solution: Implemented buffered reading, memory pre-allocation, and binary mode processing.
- **Problem:** Black output in threshold CLAHE.
  Solution: Debugged input normalization, buffer management, and applied fixes to output scaling.
- **Problem:** Limited file format support.
  Solution: Unified file handling with added support for multiple formats (JPG, PNG, TIFF, BMP).