Objectives

- Rebuild the OpenCV library into debug mode to suit the received library and header files.
- Resolve compatibility and configuration issues with Qt, OpenCV, and CUDA environments.
- Address and fix errors in the histogram.cpp file under the `cudaimgproc` library.
- Successfully test and validate the rebuilt library using sample programs.
- Document the process for future reference.

Activities

- **Rebuild OpenCV in Debug Mode:**
  - Configured CMake for building OpenCV with debug options, including support for CUDA, CuDNN, and Qt.
  - Installed OpenCV version 4.10.0 and corresponding `opencv-contrib` modules.
  - Adjusted configurations in CMake to optimise for debugging.
- **Qt Compatibility Fixes:**
  - Identified missing `Qt6Core5Compat` module in Qt 6.8.0.
  - Uninstalled Qt 6.8.0 and installed Qt 6.8.1 with MinGW and MSVC 17 2022 kits, which included the required module.
- **Library Build and Configuration:**
  - Encountered errors while building `opencv_world` with CUDA. Resolved issues by disabling unnecessary modules and making adjustments in CMake and OpenCV configurations.
  - Built and installed the debug version successfully in MSVC 17 2022.
- **Histogram.cpp Bug Fixes:**
  - Addressed issues related to:
    - Function pointer mismatches.
    - Template definitions and compile-time constants.
    - Stream context handling for CUDA streams.
    - Buffer size type inconsistencies.
    - Initialization and memory management.
  - Implemented fixes using proper type definitions, template improvements, and backward compatibility measures.
- **Testing the Debug Build:**

- Ran test programs to validate functionality:
  - **Program 1:** `printShortCudaDeviceInfo()` – *Successful*.
  - **Program 2:** `getCudaEnabledDeviceCount()` – *Successful*.
  - **Program 3:** Test image operations (Gaussian filter, Sobel filter, grayscale) – *Successful*.

- **Documentation:**

  Summarised all steps, configurations, and fixes into a detailed document for future use.

Achievements

- Successfully rebuilt the OpenCV library in debug mode with full compatibility for CUDA, CuDNN, and Qt environments.
- Fixed critical bugs in the `cudaimgproc` library, ensuring functionality of the histogram module.
- Validated the debug build with test programs, confirming the stability and usability of the library.
- Documented the entire process for streamlined future implementations.

Problem & Solution

- **Problem:** Missing `Qt6Core5Compat` module in Qt 6.8.0.
  Solution: Installed Qt 6.8.1, which included the required module.
- **Problem:** Build errors in `opencv_world` with CUDA support.
  Solution: Disabled unnecessary modules, adjusted CMake configurations, and ensured compatibility with CUDA environments.
- **Problem:** Errors in `histogram.cpp` within the `cudaimgproc` library.
  Solution: Implemented fixes for function pointers, templates, stream contexts, buffer size management, and initialization issues.
- **Problem:** Debug mode integration of libraries caused discrepancies with older NPP versions.

  Solution: Maintained backward compatibility using preprocessor directives and type-safe implementations.
- **Problem:** Complex debugging and error handling during build.
  Solution: Refactored code, improved structure, added proper error checking, and simplified function calls.