

README

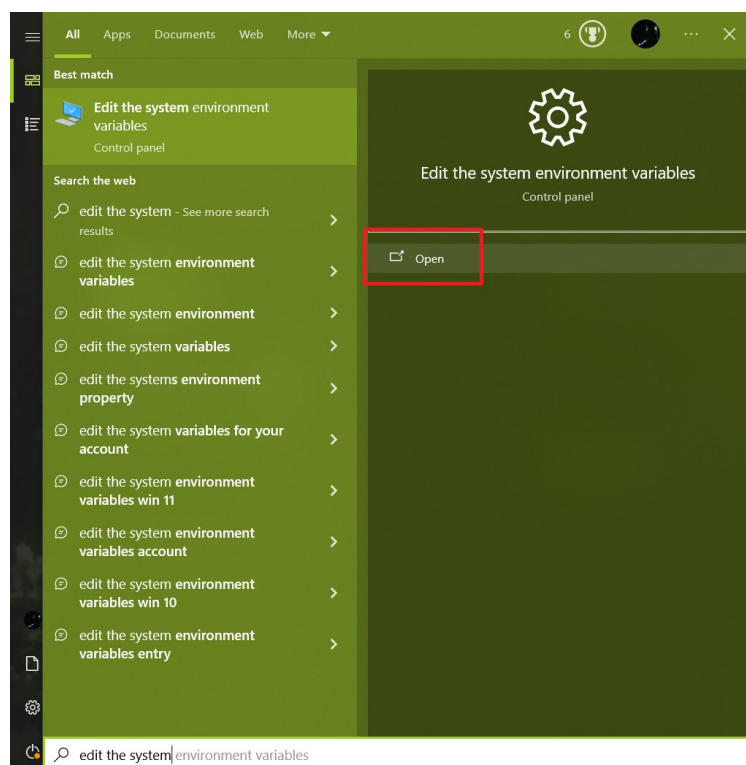
This is the server module of the Image Processing Application. This application communicates with the Client module to receive an image and a filter to apply on it, and returns the image with the filter applied.

Requirements

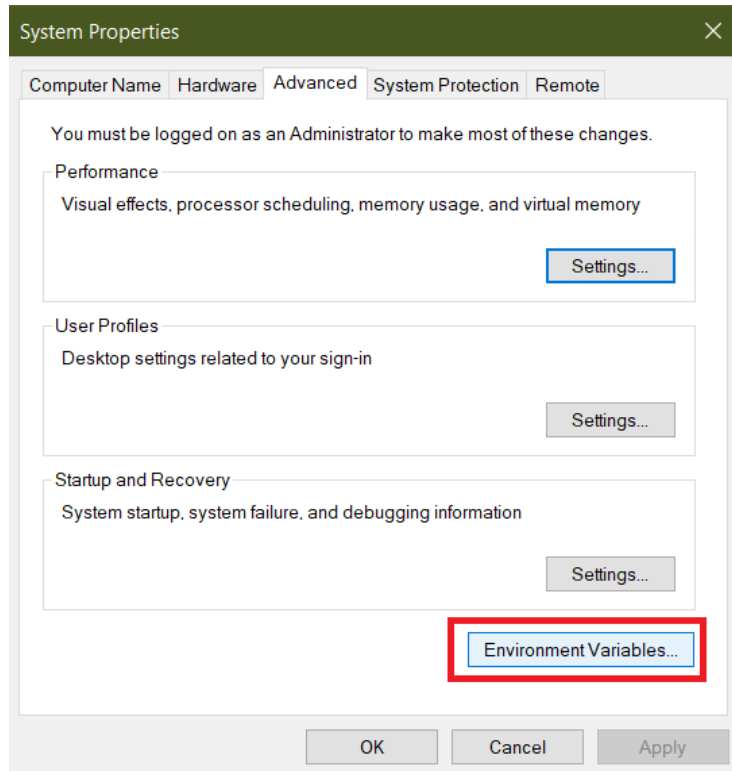
- 64-bit Windows
- OpenCV 4.8.0
- Client companion module
- Visual Studio (2022 preferred)

Configuring OpenCV 4.8.0

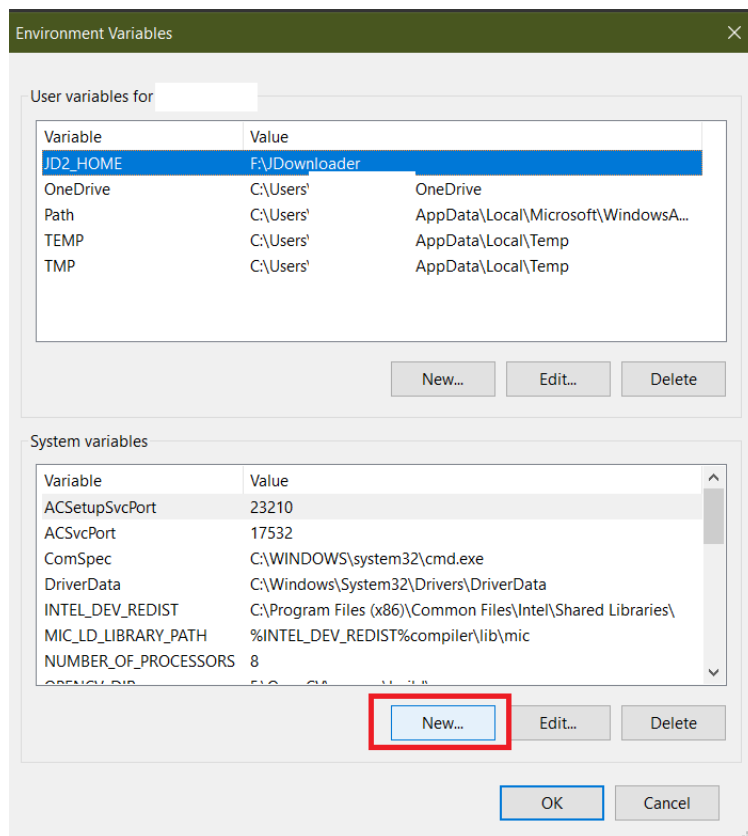
1. Download 4.8.0 from <https://opencv.org/releases/>.
2. Install OpenCV to the desired location.
3. Add the OpenCV directory environment variable. Below are the steps to do this on Windows:
 - Open Start menu, search for 'Edit the system environment variables' and click on 'Open' as shown:



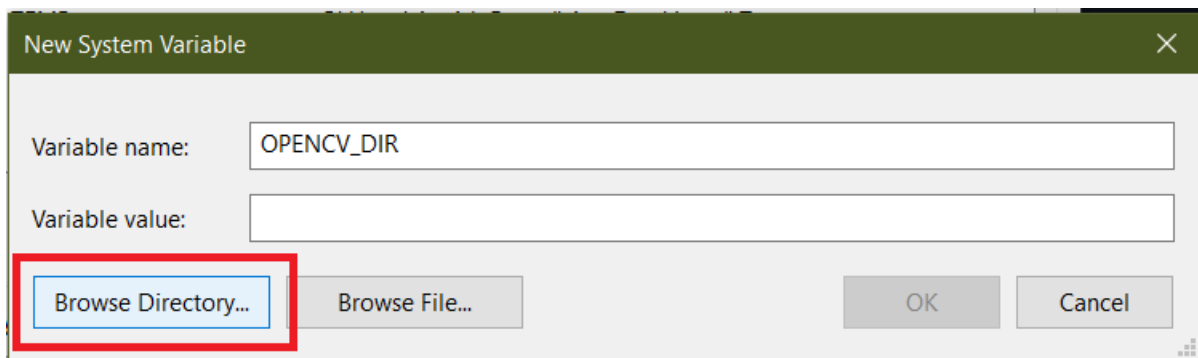
- In the 'System Properties' dialog box, click the 'Environment Variables...' button:



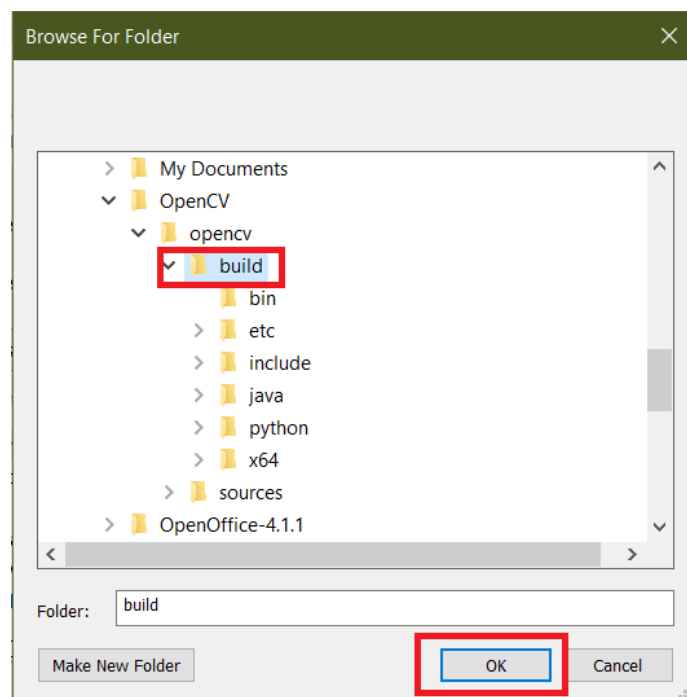
- Click 'New' at the bottom:



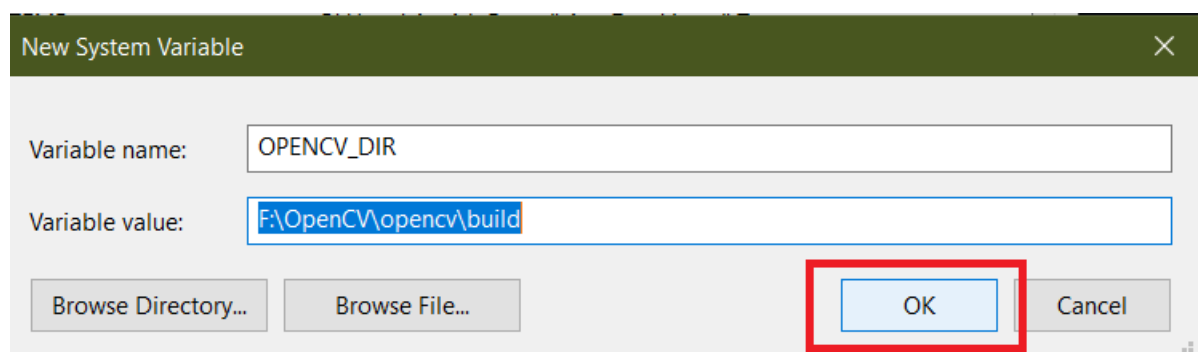
- Type in 'OPENCV_DIR' (without the quotes) in the 'Variable name' field as shown, and click on the 'Browse Directory' button:



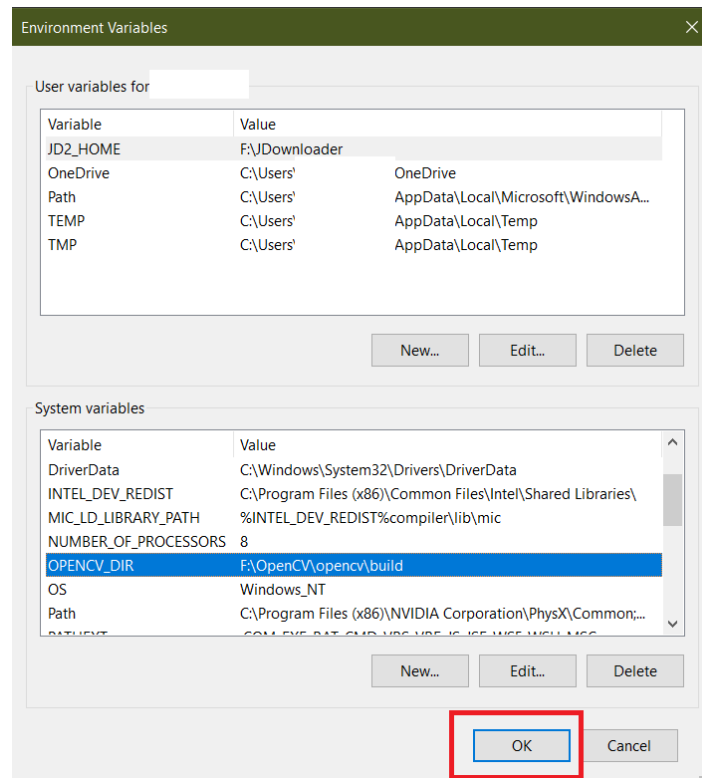
- Browse to the 'build' folder of your OpenCV installation, as shown below. For example, if you installed OpenCV in the 'Documents' folder, browse to 'C:\Users\abc\Documents\opencv\build'. Select the 'build' folder and click 'OK':



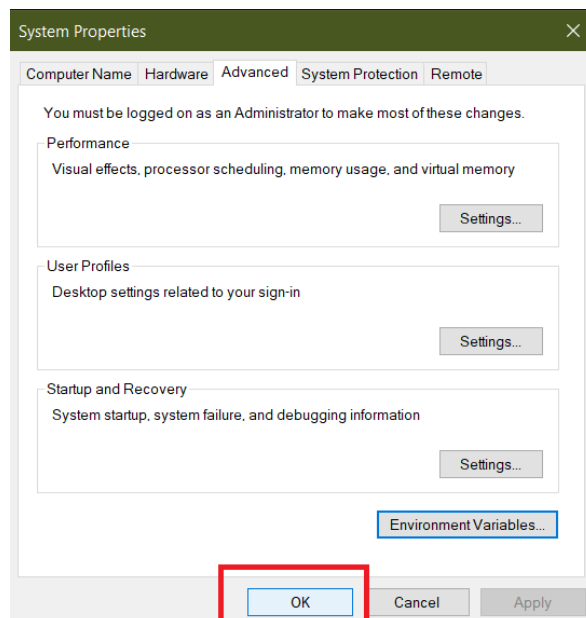
- The 'Variable value' field in the 'New System Value' dialog box should now contain the address of the OpenCV build folder, as shown below. Click OK:



- The OPENCV_DIR variable will now be added to the 'System Variables' list in the 'Environment Variables' dialog box. Verify the value and click 'OK':



- Finally, click OK in the 'System Properties' dialog box:



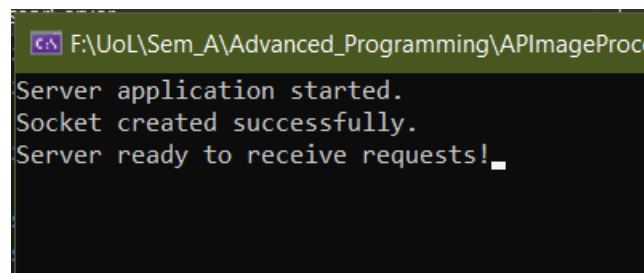
- More information on how to set up environment variable for OpenCV can be found here (<https://www.opencv-srf.com/2017/11/install-opencv-with-visual-studio.html>).

4. Build the application. Press Ctrl + B in Visual Studio for this.

5. The program is ready to run now. Execution starts in APIImageProcessorServer.cpp. Press Ctrl +

F5 in Visual Studio to run the program.

6. On successful execution, a console window like the one shown below should appear:

A screenshot of a console window with a dark background and light green text. The title bar at the top shows the file path 'F:\UoL\Sem_A\Advanced_Programming\APImageProc'. The console output consists of three lines: 'Server application started.', 'Socket created successfully.', and 'Server ready to receive requests!' followed by a cursor. The text is displayed in a monospaced font.

```
F:\UoL\Sem_A\Advanced_Programming\APImageProc
Server application started.
Socket created successfully.
Server ready to receive requests!_
```

7. By default, the server runs on port 8080 on the local machine. Support to change this setting is not available currently.