

## CS 423/523 Computer Vision

Fall 2014

### Project 1

**Due: October 17<sup>th</sup>, Friday 2014, 23:55**

## Linear Filters

**Description:** In this project you will develop an OpenCV program that takes input arguments from the console, processes some images (or optionally videos) and saves them after processing.

### Details:

- 1) Create a new cpp (C++ source code) file whose name uses the template PRJ1-**STUDENTID**.cpp (insert your own student id in the bold part)
  - a. You may use any operating system you want (Windows, MacOS).
  - b. The output executable file must be in the format PRJ1-**STUDENTID**.exe. \*.exe suffix is only required for Windows compiles.
  - c. You **must** use CMake for producing the makefile/solutionfile required to compile your project. Hence, you can use any compiler you want (MSVC, GCC, Clang, ...)
  - d. You **must** use OpenCV (2.4.\*) as the assistant library.
  - e. No MATLAB allowed in the project.
- 2) Write the code for loading an image using OpenCV's imread function. The image will be stored in a **cv::Mat**.
- 3) Write the code for reading a CSV (Comma separated value) file which defines an N by N float matrix of the linear filter to be applied. The filter matrix will be stored in a **cv::Mat**. Check for the filter matrix so that it is a square matrix. You can assume the dimension N as the first row's number of elements.
- 4) Write the code for applying a linear filter to the image. You must write your own code. Don't use an OpenCV function here. Your function's signature will be exactly as stated below. It will create a new **cv::Mat** which contains the image after the filter is applied:  

```
cv::Mat applyFilter(cv::Mat image, cv::Mat filter);
```
- 5) Create CSV files for the filters: Sobel Vertical, Sobel Horizontal, Roberts (any one of the two), Prewitt Vertical, Prewitt Horizontal, 5x5 Gaussian (any variance will do). You can write the CSV files by hand using a text editor.
- 6) Your executable is going to be used as below:  
PRJ1-**STUDENTID**.exe [inputfilename] [filterfilename.csv] [outputfilename]

It will read the image into memory, read the CSV file storing the filter to the memory, apply the filter to the image, and finally write the result to the output file.

- 7) 10% BONUS: Instead of input and output files to be images, use them as video files. For instance, get a video file, apply the linear filter, and write a video file.

Your code should be clean and easy to read by possessing the following properties;

- *Clean structure:* The overall code should be neatly organized, where the related statements are grouped together with enough spacing among them.
- *Consistent indentation:* Statements should be consistently indented according to the nesting.
- *Appropriate use of comments:* There should be comments explaining what the program, and different groups of statements are supposed to do. Don't overdo it.
- *Meaningful and consistent variable naming:* The names of variables should be meaningful with respect to the purpose and usage of these variables.

### IMPORTANT:

**Submission:** By uploading your whole project file to the LMS as a single ZIP archive. No other methods (e.g., by e-mail) accepted. (You may resubmit as many times as you want until the deadline).

**Warning:** This homework is an individual assignment. DO NOT SHARE YOUR CODE WITH OTHERS. Your programs are checked and compared against each other using automated tools. Any act of cheating will be punished **severely**. The code that does not compile **will receive 0 points**.

Also:

- Name your archive file uploaded exactly as requested. Your archive file must be named as **PRJ1-STUDENTID.zip**
- Make sure that your program runs and gives the **expected output**
- The first lines of your code must include your name, surname, student number, and department as a **comment**. An example comment is as follows:  
*/\* John Smith S0001 Department of Industrial Engineering \*/*
- **Don't include** debug files such as \*.obj, \*.pch, \*.ilk, **or your image and video files** in your archive. The single file that can be greater than 100KB in your archive must be the executable file.

Good luck ☺