# **CVI620/DPS920- Lab 4**

# **Creating images and capturing video**

| Total Mark: | 10 marks (2.5% of the total course grade)   * 5 out of 10: Blackboard submission (Due: End of week 4) * 5 out of 10: Lab demo (Due: During lab of week 5) |
| --- | --- |
| Submission file(s): | * Lab04\_01.cpp * Lab04\_02.cpp * Lab04\_20W.docx |

Please work in **groups** to complete this lab. This lab is worth 2.5% of the total course grade and will be evaluated through your written submission, as well as the lab demo. During the lab demo, group members are *randomly* selected to explain the submitted solution. Group members not present during the lab demo will lose the demo mark.

Please submit the submission file(s) through Blackboard. Only one person must submit for the group.

1. Start a new OpenCV project and name it Lab04. Add the debug property sheet as in Lab 2.

## **Part I: Image Arithmetic**

1. Create a program (save as Lab04\_1.cpp). Include code to select option A or B:
2. If option A is selected, include code to:
   1. Open a color image and display.
   2. Increase the brightness by adding a constant (e.g. 100) to the all color channels of the image. Display in a separate window.
   3. Increase the contrast by multiplying the image by a constant (e.g. 1.5). Display in a separate window.
   4. Output and compare the values of the 3 channels of above 3 images at a certain pixel (e.g. row 100 and column 200). Are the values as expected? What happens to values falling out of valid range?
3. If option B is selected, include code to:
   1. Open two color images and display them.
   2. Ask the user for a number (alpha) between 0 and 1.
   3. Implement a linear blend of the two images and display.
   4. Use *calcHist* with 8 bins to calculate the histogram of only the BLUE channel of the blended image. Output the histogram values.
   5. BONUS: save the above values in a .csv file and open in Microsoft Excel to plot the histogram.

## **Part II: A drawing application**

1. Download example\_09-02.cpp from <https://github.com/oreillymedia/Learning-OpenCV-3_examples> and add it to the project as Lab04\_2.cpp.
2. Run the program without any arguments. You should be able to draw red rectangles using the mouse.
3. Change the code:
   1. Increase the size of the image to 480 (rows) x 640 (columns).
   2. Move the top left corner of the window to location (10, 30).
   3. As in Lab 2-2, allow the user to choose the color for the drawing by typing one letter.
   4. If the shift key is held while drawing with the mouse, draw a circle (instead of a rectangle) enclosed within the specified box.
   5. If the ctrl key is held while drawing with the mouse, draw a vertical or horizontal ellipse (instead of a rectangle) enclosed within the specified box.
   6. Increase the thickness of the drawings (any shape) to 2.
   7. When ‘ESC’ is pressed, close the drawing window. Then ask if the user wants to save the image before exiting and save accordingly.
   8. On the top center, 40 pixels lower than the top edge of the drawing window, write “My Drawing Application”. Use FONT\_HERSHEY\_SCRIPT\_SIMPLEX, fontScale=1, and thickness =1; Draw a red line showing the base line. Hint: Use the example provided in the documentation of getTextSize():
4. Add this declaration to your file:

We, Jason, Matteo, Muqing, declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. We have not copied any part of this assignment, manually or electronically, from any other source including web sites, unless specified as references. We have not distributed our work to other students.

1. Specify what each member has done towards the completion of this work:

|  | Name | Task(s) |
| --- | --- | --- |
| 1 | Jason | Code |
| 2 | Matteo | Code |
| 3 | Muqing | Code |