Multimedia Application Final Project

Ilaria Renna

May 15, 2018

TOPIC

The project is to develop a small GIMP-like image editor, with basic functions. The advantage of this project are:

- the modularity, it is easy to separate part of the code for group work
- using image manipulation
- adaptability, you can implement various functions and features depending on what interest you the most.

As usually in the approach by project, it will be done by group of 5–7 students.

Also it is mandatory that each group use a GitHub account, so tutors can check the student's code.

1 Project Time Table

This project will be realised during the last two weeks.

Tuesday group: 22 May and 29 May
Presentation: 5 June
Final report and Code: to be sent before 3/5/18 12 am.

Thursday group: 17 May and 31 May
Presentation: 7 June
Final report and Code to be sent before 5 June at 12 am.

You will be evaluated on your code, report and oral presentation.

The report has to explain the OpenCV functions you used and how you exploited them, the image processing technique you implemented. No code has to be in the report. At the oral exam you will present your work and show how the code works.

2 Function to use in the project

These are the functionalities you have to include in the project:

- dilatation / erosion
- resizing

- lighten / darken
- panorama / stitching
- Canny edge detection

2.1 Dilatation / erosion

These functions use the Open CV functions :
`cv::dilate`
`cv::erode`

The erosion size (respectively the dilatation size) should be a parameter the user can set within a given range. So there should be two parameters to this function :

- the input image
- the erosion size

Plus eventually a parameter to choose between erode and dilate (it is up to you to choose how you want to make that choice available for the user). The exact range of the sizes is not an issue.

2.2 Resizing

This functionnality will use the function:
`cv::resize`
It takes as input:

- an image
- either a factor or two dimension

2.3 Lighten / darken

This function doesn't require to use any specific function from OpenCV. It can be done with matrix manipulation. So there should be two parameters to this function:

- the input image
- the lightening / darkening factor

As for the erosion / dilatation, the user should be able to set the magnitude of the transformation in a given range.

2.4 Panorama / stitching

The purpose of this function is to take as parameters the name of several les, or a directory, and try to stitch them together, ie detecting the common part. You can use the cpp sample "stitching from OpenCV" to see how it works.

2.5 Canny edge detection

This functionnality will use the function:
`cv::Canny`
It takes as input:

- an image
- two thresholds
- a kernel

3 Interface

No specific interface is required. Command line is acceptable, especially because it won't work otherwise on ISEP computer.

4 Advanced functions

To conclude your project some more advanced functions could be added. Students are free to propose whatever interest them, here are some proposition:

- video manipulation
- face identification / recognition
- advanced gui