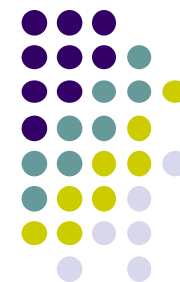


Image Processing and Computer Vision - Lab2



Roberta Macaluso
Politecnico di Torino

Dipartimento di Automatica e Informatica (DAUIN)

Torino - Italy

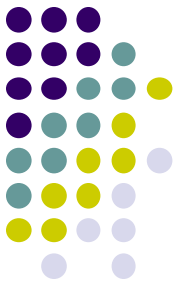
roberta.macaluso@polito.it



This work is licensed under the Creative Commons (CC BY-NC-SA)

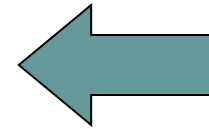


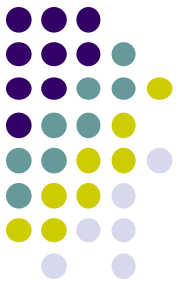
License. To view a copy of this license, visit
<http://creativecommons.org/licenses/by-nc-sa/4.0/>



The Plan

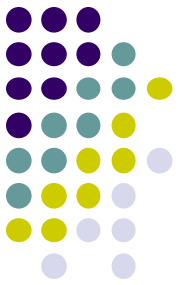
1. Intro to Image Processing
2. Intro to OpenCV
 - today (04/04) and next week (09/04)
3. Fourier Transform (and Friends)
4. Image Segmentation
5. CCD, CMOS, and Optical Systems
6. Car Lane Detection
7. Face Detection and Tracking
8. Neural Network Introduction





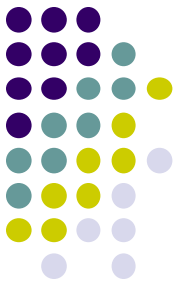
Intro to OpenCV

- Today and next week
 - 3 hours
- Text of the exercises/tasks
 - on the Teaching Portal
- You need a webcam
 - alternatively, you can use still images or a video
- Goal
 - Experiment with basic image processing operation with OpenCV



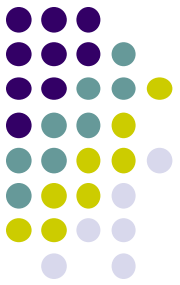
Intro to OpenCV

- Three (and a half) exercises
 - Warm Up: revise last lecture (code and slides)
 - 1: create a program for performing base operations with images
 - 2: add a logo (superimpose image) to each frame of the video shown in the application
 - 3: calculate and show the histogram of each video frame



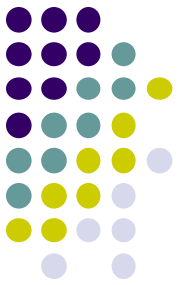
Add a logo

- ... also known as “adding two or more images”
- Three suggested options:
 - addition -> `cv2.add(img1, img2)`
 - linear blending -> `cv2.addWeighted(...)`
 - replacement of an image portion
- *Warning:* most (all) of these methods needs two images with the **same** width and height



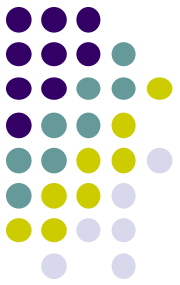
Histogram calculation

- **OpenCV** has a method to do that:
 - `cv2.calcHist()`
- It needs the source image (first parameter) as an array
 - e.g., `[img]`
- To compute the histogram of a RGB image, you have to execute the method three times, one for each channel



Histogram calculation

- **Matplotlib** has a method to do that:
 - `plt.hist()`
- It needs the source image (first parameter) as an array
 - e.g., `[img]`
- To compute the histogram of a RGB image, you have to execute the method three times, one for each channel

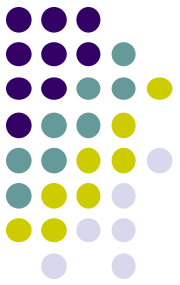


Draw a histogram

- You can use matplotlib...

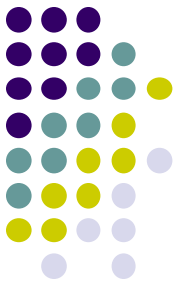
```
plt.plot(hist)
plt.xlim([0, 256])
plt.title("Histogram")
```

- If you want to show the histogram of all three RGB channels, you have to draw 3 histograms...



Histogram equalization

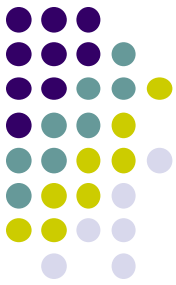
- `cv2.equalizeHist(img)`
- *Warning:* the method returns the equalized image, not its histogram
 - to show the histogram, you need to redo the previous steps (from the calculation onwards) but with the image from this method



Intro to OpenCV

- In the first session: you should be able to complete the Warm Up and the first exercise
- Hints, insights, links, etc. are in the text of the exercises
 - I am here for you...
 - ... please ask if you need any help or clarification

License






This work is licensed under the Creative Commons “Attribution-NonCommercial-ShareAlike International (CC BY-NC-SA 4.0)” License.

You are free to:

- **Share** - copy and redistribute the material in any medium or format
- **Adapt** - remix, transform, and build upon the material

for any purpose, even commercially.

Under the following terms:

-  **Attribution** - You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
-  **Noncommercial** - You may not use the material for commercial purposes.
-  **Share Alike** - If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.