Computer Vision Course — A.A. 2021/2022

Lab 1:
OpenCV Intro

Nicola Garau
nicola.garau@unitn.it

Recommendations

- Feel free to interrupt and ask questions 🙋‍♂️🙋‍♀️

- If you have any doubt, you can:

  - ask me before/after lectures

  - drop an e-mail: <u>nicola.garau@unitn.it</u> 📧

Feedback

- First time the Lab is run by me 🧪

- Any feedback is welcome, **especially** negative ones 📉

- Anonymous feedback form at the end of the course 🎭
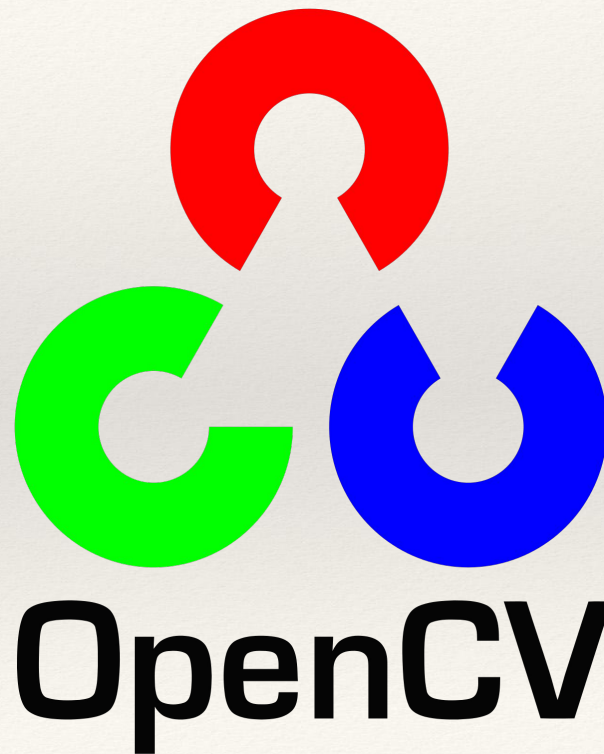
Any questions so far?

What's up today?

- What is OpenCV? 🤔

- The Virtual Machine 💾

- How to initialise a project in OpenCV 📄

- How to open and display images 🖼️

- How to open and display videos 🎥

What is OpenCV?

- Computer Vision library

- Open source ♻️

- Website: opencv.org

- Docs: docs.opencv.org

What is OpenCV?

OpenCV contains a collection of C/C++, **Python** and Java implementations of some of the popular algorithms of image processing and computer vision, which cover:

- 2D/3D feature toolkit

- Works for **images** and **videos**

- Face/gesture recognition

- Segmentation and recognition

- Tracking

- Image/video load, save, display

- Many more...

Some numbers: OpenCV

- Fully supported and widely used (**50k+** forks on GitHub)

- Open Source (**31k+** commits on GitHub)

- Huge number (**2500+**) of algorithms ready to use

- Recognised as the reference library by the research community (**60k+** stars on GitHub)

- Has a good interface also for newbies 👨🏽‍💻

The virtual machine

⚠️Using the virtual machine is not recommended, but it is useful if you don't want to set up the environment on your machine

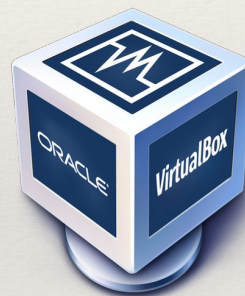Prerequisites:

- Virtual Box: www.virtualbox.org

- Virtual Box Guest Additions

- Recommended: Virtual Box Extension Pack

Characteristics:

- OS: Ubuntu 18.04 LTS

- User: mmlab

- Password: mmlab

OpenCV version: 4.1.2-pre
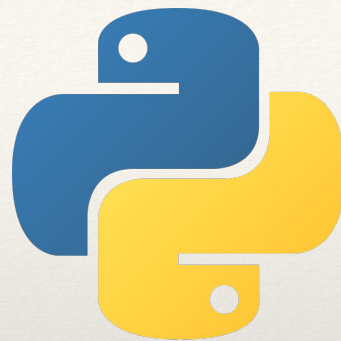
What if I hate VMs?

You are not alone! 🥳

Prerequisites:

- [Visual Studio Code](#) (aka VSCode)
- [Python](#) (at least version 3 should be enough)
- All the dependencies installed (see [GitHub](#) page)

Some questions for you!

How to initialise a project: Python

- Create a new **folder** in your working directory

- Open it in **VSCode**

- Create a new **Python script** (e.g. `main.py`)

- Open and edit script

- Open a terminal in VS Code

- Run the script (e.g. `python main.py`)

How to open and display images: Python

```
import cv2

image = cv2.imread("../material/OpenCV.png")
cv2.imshow('Window title', image)
cv2.waitKey(0)
```

...simple, isn't it? 🎉

How to open and display videos: Python

```python
import cv2 as cv

cap = cv.VideoCapture(0)

while(True):
    # Capture frame-by-frame
    ret, frame = cap.read()

    # Display the resulting frame
    cv.imshow("Frame name",frame)
    cv.waitKey(1)

# When everything done, release the capture
cap.release()
```

Meme time! Questions?

We are still newbies, but soon we'll see how to use OpenCV for harder tasks!

The next slides are for C++ users
(you don't need them if you use Python)

How to initialise a project: C++

- Start Visual Studio Code

- Create a new folder under C++ path

- Create a new file e.g. main.cpp

- Create a new file CMakeLists.txt

```
cmake_minimum_required(VERSION 2.8)
project( ProjectName )
find_package( OpenCV REQUIRED )
include_directories( ${OpenCV_INCLUDE_DIRS} )
include_directories( ${PROJECT_SOURCE_DIR} )
add_executable( ProjectName main.cpp )
target_link_libraries( ProjectName ${OpenCV_LIBS} )
```

- Now you have linked the OpenCV libraries with your project

- Open the project folder in Terminal

- In Terminal: 'cmake .' to compile the project in the current folder

- In Terminal: 'make' to build an executable of your project

- In Terminal: './ProjectName' to run your executable

How to open and display images: C++

```cpp
#include <opencv2/opencv.hpp>
#include <opencv2/highgui.hpp>

using namespace cv;

int main( int argc, char** argv )
{
  Mat image;
  image = imread("Google.jpg", 1);

  namedWindow("Window",1);
  imshow("Window", image);
  waitKey(0);

  return 0;
}
```

How to open and display videos: C++

```cpp
#include <opencv2/opencv.hpp>
#include <opencv2/highgui.hpp>
using namespace cv;

int main( int argc, char** argv )
{
  Mat image;
  VideoCapture cap;
  cap.open("Video.mp4");
  if(!cap.isOpened())
        return 0;

  namedWindow("Window",1);

  for(;;){
        cap >>image;
        imshow("Window", image);

        if(waitKey(10) >= 0) break;
  }
  return 0;
}
```