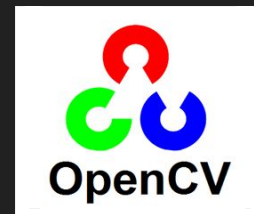# ECSE-415

Software for Computer vision

Introduction to OpenCV and Python

What is OpenCV?

- OpenCV is an open source library for developing computer vision applications
- Cross-platform:
  - C++, C, Python and Java interfaces
  - Supports Windows, Linux, Mac OS, iOS and Android
- Strong focus on real-time applications
  - Multi-core processing
  - Supports hardware acceleration: IPP, OpenCL, CUDA
- Under a BSD license, it can be freely used, distributed and adapted in both academic and commercial apps

OpenCV Python Interface

- Python is a high level general purpose programming language
  - simplicity and code readability
  - compared to C/C++, Python is slower
- It is possible to create a Python wrapper for a C/C++ code
  - The C/C++ code is running without any performance penalty
  - We can benefit from easy coding in Python
- This is how OpenCV-Python works, it is a Python wrapper around original C++ implementation
- We can also benefit from other highly optimized Python libraries
  - Numpy, SciPy, Matplotlib etc

Installing Python + Packages

Linux

- Most linux distributions has pre-installed python
- Type command 'python --version' in the terminal
- Will display current version of installed python

```
raghav@raghav-XPS-13-9360:~$ python --version
Python 2.7.15+
```

- Python3 may or may not be installed

```
raghav@raghav-XPS-13-9360:~$ sudo apt-get install python3
[sudo] password for raghav:
```

MacOS

- All MacOS comes with pre-installed python2.x

```
            -mbp:~                    $ python --version
Python 2.7.10
```

- Install python3

Windows

- Install Anaconda for windows. Follow [these](#) instructions. Download python3.7 installer from [here](#).

VirtualEnv for python2.x

- Linux

  ```
  raghav@raghav-XPS-13-9360:~$ sudo apt-get install virtualenv
  [sudo] password for raghav: █
  ```

- MacOS
  - brew install virtualenv

VirtualEnv for python3.x

- Linux or MacOS

```
raghav@raghav-XPS-13-9360:~$ python3 -m venv venv
```

- Windows
  - `conda create --name venv python=3.7`

Pip -- install python package

- Activate virtualenv "venv" in Linux or MacOS

    `raghav@raghav-XPS-13-9360:~/Downloads/ECSE_415$ source venv/bin/activate`

- Activate virtualenv "venv" in Windows (conda)
    - activate venv

- install 'python package'
    - Command: pip install 'python package'

`(venv) raghav@raghav-XPS-13-9360:~/Downloads/ECSE_415$ pip install numpy scipy matplotlib jupyter opencv-python==3.4.3.18`

Check if everything is install correctly

```
(venv) raghav@raghav-XPS-13-9360:~/Downloads/ECSE_415/Tutorial_1$ python
Python 3.6.8 (default, Jan 14 2019, 11:02:34)
[GCC 8.0.1 20180414 (experimental) [trunk revision 259383]] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy as np
>>> import scipy
>>> import matplotlib.pyplot as plt
>>> import cv2
>>> print(cv2.__version__)
3.4.3
>>>
```
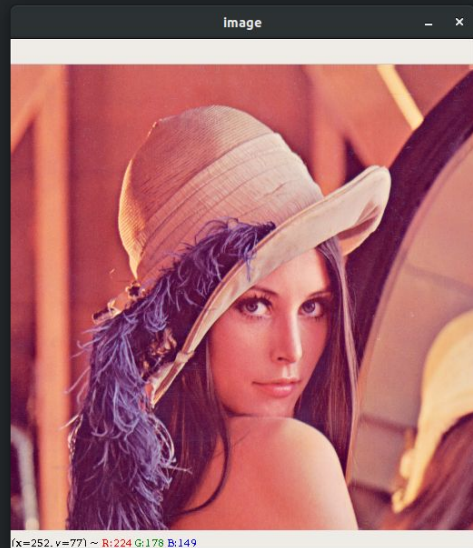
Read an image and display

```
## import required python package

import numpy as np
import cv2

## read the image
img = cv2.imread('Lenna_image.png')

## check shape of image
img.shape

## display image
cv2.imshow('image',img)
## infinitely wait for a user keypress
cv2.waitKey(0)
## close all windows
cv2.destroyAllWindows()
```
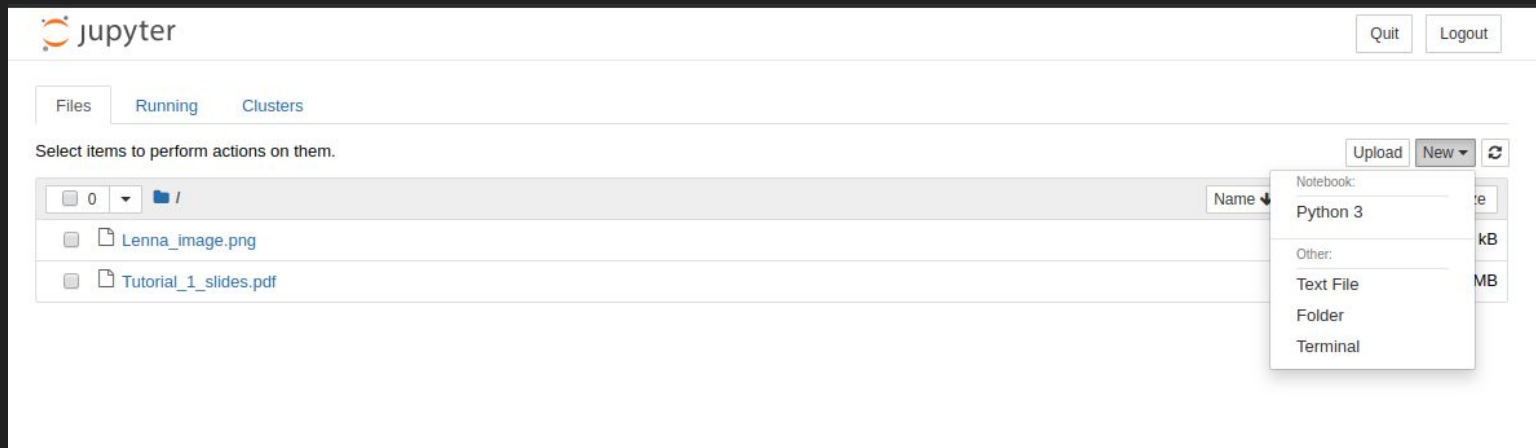
Jupyter notebook

- The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more

-
```
(venv) raghav@raghav-XPS-13-9360:~/Downloads/ECSE_415/Tutorial_1$ ls
Lenna_image.png   Tutorial_1_slides.pdf
(venv) raghav@raghav-XPS-13-9360:~/Downloads/ECSE_415/Tutorial_1$ jupyter notebook
```

Jupyter Notebook

- Open new notebook

Why Jupyter notebook

- Easy to visualize output
- Better organization of code ("code") and written observations ("markdown") about output
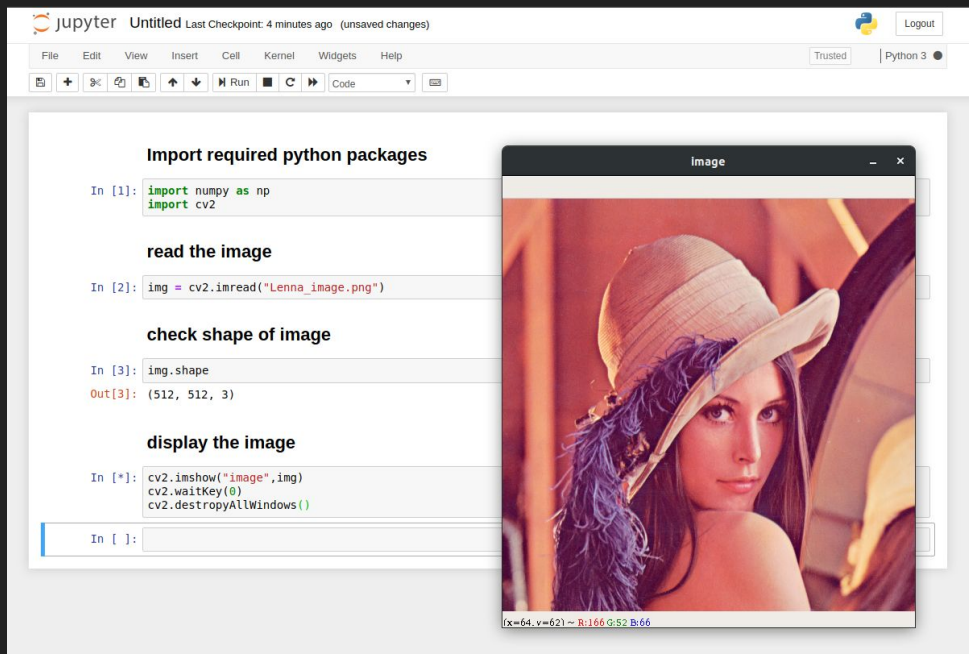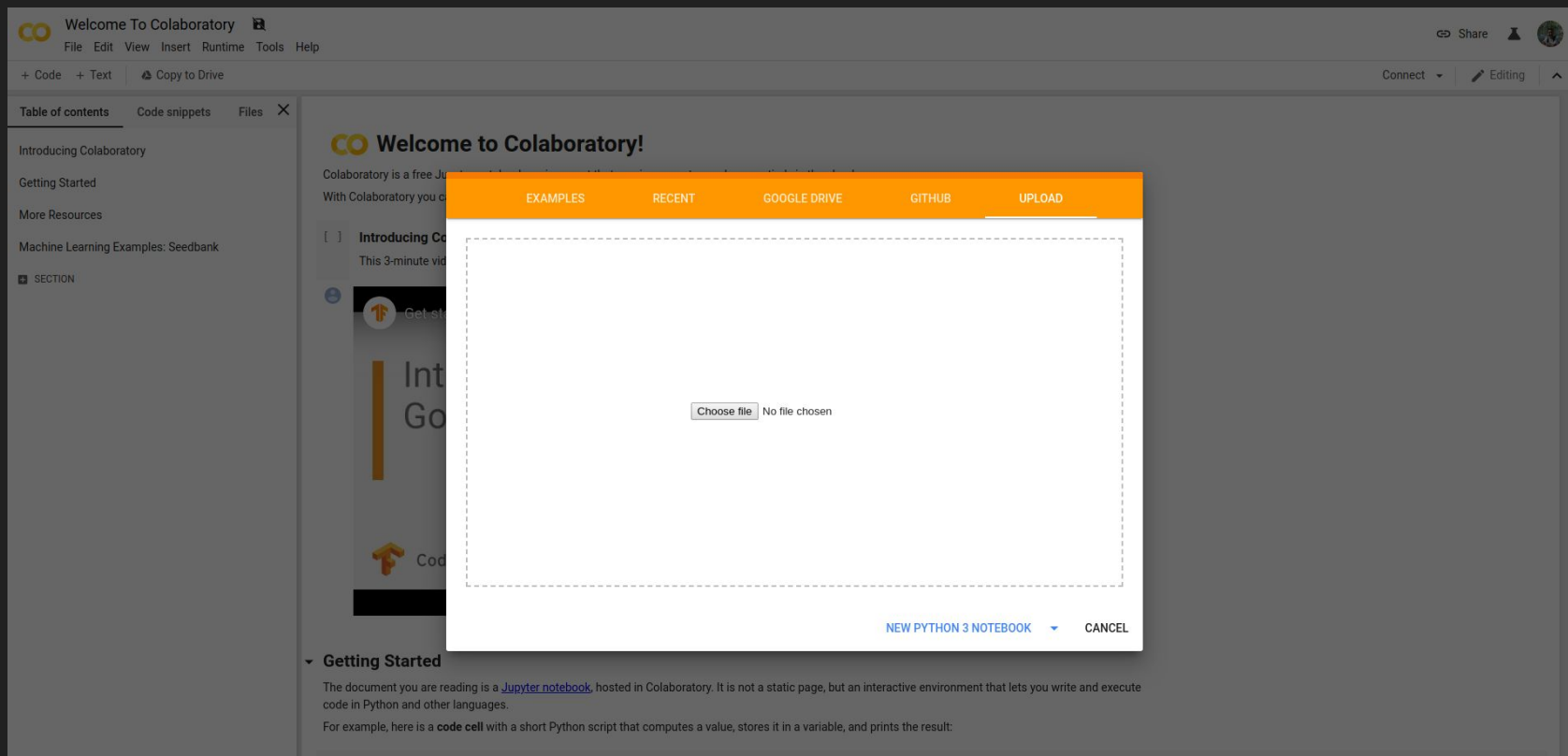
Google Colab (colab.research.google.com)

- Online version of Jupyter notebook by google
- Use of google server for computational and store requirement
- Connected with your google drive
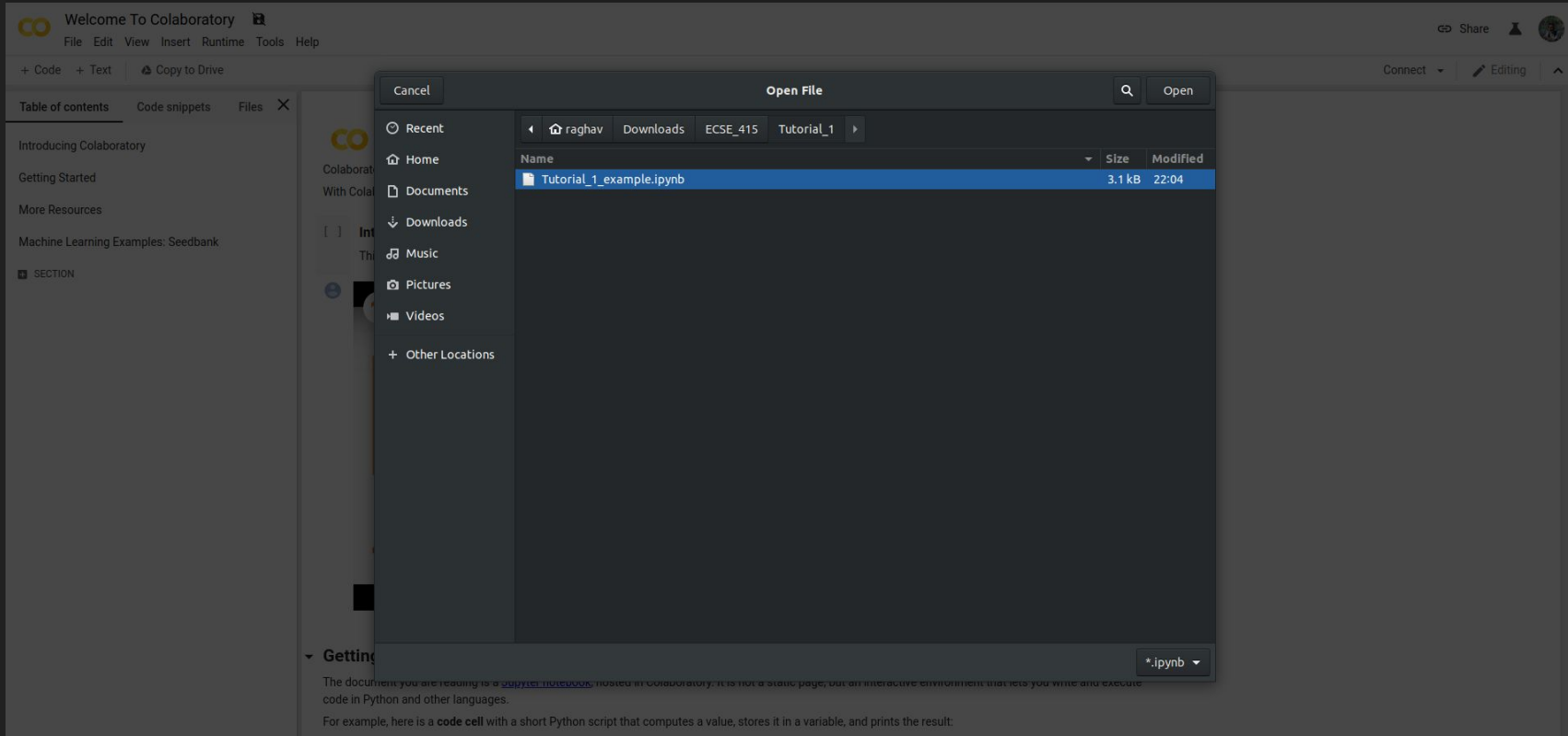- Easily accessible from any computer with internet
- No need to install anything locally

Uploading local Jupyter Notebook to Google colab

Uploading local Jupyter Notebook to Google colab

Mount your google drive

```
[1]   from google.colab import drive
      drive.mount('/content/gdrive')

      Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.googleusercontent.com&redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.0%3Aoob&scope=ema

      Enter your authorization code:
      ..........
      Mounted at /content/gdrive
```

```
[2]   !ls gdrive/'My Drive'/'ECSE_415'/'Tutorial_1'/

      Lenna_image.png  Tutorial_1_example.ipynb  Tutorial_1_slides.pdf
```

Google colab

Import required python packages

```python
import numpy as np
import cv2
```

read the image

```python
datapath = 'gdrive/My Drive/ECSE_415/Tutorial_1/'
```

```python
img = cv2.imread(datapath+"Lenna_image.png")
```

check shape of image

```python
img.shape
```

```
(512, 512, 3)
```

Shortcut keys for Jupyter and Colab

- **Jupyter Notebook**
  - Tab: Autocomplete function name or suggest function name
  - Shift + Tab: Display expected arguments for a function

- **Google Colab**
  - Ctrl + space: Autocomplete function name or suggest function name
  - Tab: Display expected arguments for a function