Learning Face\_Recognition Using GoogleCoLab Python 3

In this worksheet,we will learn about

1. How to use Google CoLab

2.How to import Libraries

3.How to upgrade files to Google CoLab

4.Basics of Face Detection and Image Recognition

Lesson 1: Introduction to Google CoLab

Colaboratory is a free Jupyter notebook environment that requires no setup and runs entirely in the cloud.

With Colaboratory you can write and execute code, save and share your analyses, and access powerful computing resources, all for free from your browser.

First, you have to search for Google CoLab on Google.When you go in to the website,you will have to log in to your gmail. Log in and you are good to go.

After that, Click file and choose new python 3 notebook. That's gonna be our practice space.

We wanted to offer 5 tips for using it:

**1. TensorFlow is already pre-installed**

When you create a new notebook on [colab.research.google.com](http://colab.research.google.com/), TensorFlow is already pre-installed and optimized for the hardware being used. Just import tensorflow as tf, and start coding.

**2. Setup your libraries and data dependencies in code cells**

Creating a cell with !pip install or !apt-get works as you’d expect. It also makes it easy for others to reproduce your setup.

To get in your training data, you can follow these tutorials for popular data sources: [BigQuery](https://colab.research.google.com/notebooks/bigquery.ipynb), [Drive, Sheets, or Google Cloud Storage](https://colab.research.google.com/notebooks/io.ipynb). You also have access to the shell with !, so !wget, !pwd, etc. might also help.

**3. Use it with Github**

If you have a nice .ipynb on Github, it’s easy to create a one-click link for your readers to start playing with it. Just add your Github path to colab.research.google.com/github/ . For example, [colab.research.google.com/github/tensorflow/tensor2tensor/blob/master/tensor2tensor/notebooks/hello\_t2t.ipynb](https://colab.research.google.com/github/tensorflow/tensor2tensor/blob/master/tensor2tensor/notebooks/hello_t2t.ipynb) will load [this ipynb](https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/notebooks/hello_t2t.ipynb) stored on Github.You can also easily save a copy of your Colab notebook to Github by using File > Save a copy to Github…

**4. Share and edit collaboratively**

Colab notebooks are just like Google Docs and Sheets. They are stored in Google Drive and can be shared, edited, and commented on collaboratively. Just click the Share button in the top right of any notebook that you’ve created.

**5. Hardware acceleration**

By default, Colab notebooks run on CPU. You can switch your notebook to run with GPU by going to Runtime > Change runtime type, and then selecting GPU. You can also have a Colab notebook use your local machine’s hardware by following these [instructions](https://research.google.com/colaboratory/local-runtimes.html).

Importing a library that is not in Google CoLab

# **Importing a library that is not in Colaboratory**

To import a library that's not in Colaboratory by default, you can use !pip install or !apt-get install.

For example,

!pip install -q matplotlib-venn

!apt-get -qq install -y libfluidsynth 1

Upgrading TensorFlow

[TensorFlow](https://www.tensorflow.org/) is available by default but you can switch which version you're using.

# To determine which version you're using:

!pip show tensorflow

# For the current version:

!pip install --upgrade tensorflow

# For a specific version:

!pip install tensorflow==1.2

# For the latest nightly build:

!pip install tf-nightly

If you wish to learn about tensorflow, this is the place:

<https://www.youtube.com/playlist?list=PLQY2H8rRoyvwLbzbnKJ59NkZvQAW9wLbx>

# **Install 7zip reader** [**libarchive**](https://pypi.python.org/pypi/libarchive)

# https://pypi.python.org/pypi/libarchive

!apt-get -qq install -y libarchive-dev && pip install -q -U libarchive

import libarchive

# 

# 

Install GraphViz & [PyDot](https://pypi.python.org/pypi/pydot)

# https://pypi.python.org/pypi/pydot

!apt-get -qq install -y graphviz && pip install -q pydot

import pydot

Install [cartopy](http://scitools.org.uk/cartopy/docs/latest/)

!apt-get -qq install python-cartopy python3-cartopy

import cartopy

Lesson 2- Understanding Libraries for Face\_Recognition

**Imutils** - A series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization and displaying.

**Face Recognition** - Recognize and manipulate faces from Python or from the command line with the world’s simplest face recognition library.

**OpenCv** - Open Source Computer Vision and machine learning software library.Library has more than 2500 optimized algorithms.

* Detect / recognize faces, identify objects
* Classify human actions in videos
* Track camera movements
* Track moving objects
* Extract 3D model of objects
* Find similar image from image database
* Removed red eyes from image taken using flash
* Follow eye movements

Lesson 3- Introduction to Face\_recognition

Face\_recognition is technology to identify faces in pictures or video.Face recognition is a way to identify faces through technology. A face recognition system uses biometrics to map facial features from a photograph or video.

Lesson 4- Exercise

Let's Get Started.

Exercise 1: Face\_Detection

To find all the faces in the pictures and know the exact locations of faces.

First step :

If you don’t install libraries yet, you can install by writing this code.

Example :

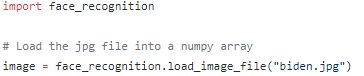


If you are using jupyter notebook, type this command in anaconda prompt.



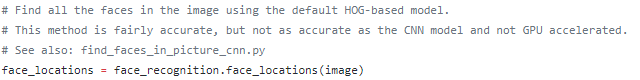
Second step :

You have to import module and load the image.



Third step :

Call the function face\_locations(image) to find the faces in the image.



Histogram of Oriented Gradients, or **HOG** for short, are descriptors mainly used in computer vision and machine learning for object detection.

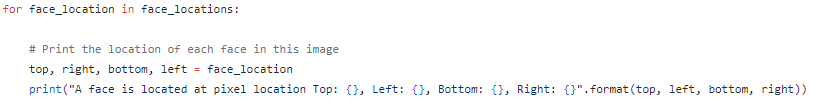
Fourth step :

To print out the number of faces in the image..

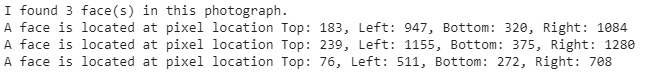


Fifth step :

Write a loop to print the locations of the faces.



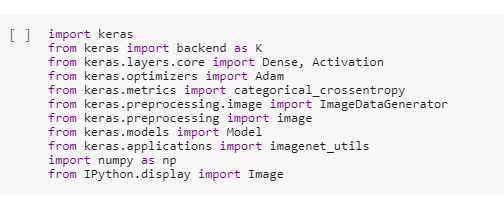
Output :

Example,

--------------------------------------------------------------------------------------------------------------------------------

Exercise - 2 (TensorFlow workshop)

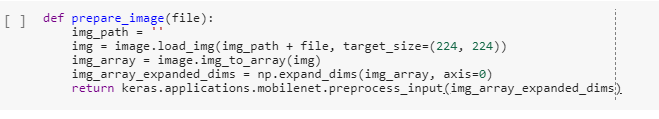
Import Libraries



Import MobileNet - the image recognition model

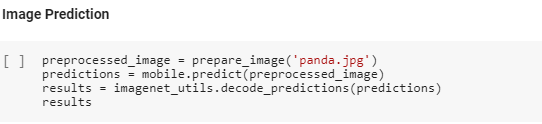


Function to Prepare Images



Preview image





Output

