Computer Vision I (CSE 40535/60535)

Practical 1: Detection of color objects in a live stream
Monday, September 16, 2019

Tasks to do at home **before class**

This "warming up" practical will allow us to use a few basic functions from the amazing OpenCV package – an open source suite that implements various useful computer vision algorithms. We can download OpenCV source codes, compile them and then link them into our C++ programs (and drink a lot of coffees or teas during this process). However, we will make our lives easier and use precompiled OpenCV libraries for Python.

To work on our practicals, we are going to need Python3 (for instance 3.6.x), and a few libraries installed in addition to the OpenCV package. Choosing a correct version of Python and correct versions of libraries we need can be a chore. Thus, we suggest not to use Python 3.7.x for practicals to avoid some conflicts later in the semester, when we will play with deep learning. Environment management is strongly recommended (for those, who want to sandbox their existing Python projects), but not necessary to complete practicals.

Part 1: Install Python and OpenCV

Instructions for MacOS

1. If you don't have Python3 and OpenCV3, you can use your preferred way to install them, for instance install Anaconda3 (https://www.anaconda.com/download/). It comes with python 3.7 but please indicate the python version when you create the new environment for the practicals. An example is as follows:

   ```
   $ conda create --name name_your_env python==3.6.5
   ```

   You are free to use any Python version < 3.7 as long as they can be compatible with the packages we need. You can use the following command to check all the python versions that are available:

   ```
   $ conda search python
   ```

   And then you can activate the new environment and use Anaconda package manager to install OpenCV:

   ```
   $ conda install -c conda-forge opencv
   ```

2. In this practical we will need two other Python libraries, numpy and matplotlib:

```
$ conda install -c anaconda numpy
$ conda install -c conda-forge matplotlib
```

3. Test the OpenCV is correctly installed:

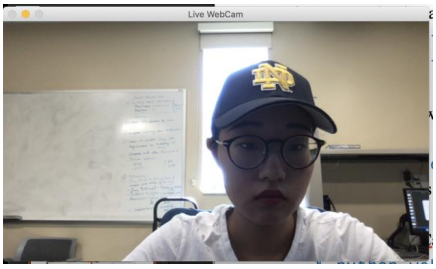
```
$ python
>>> import cv2
```

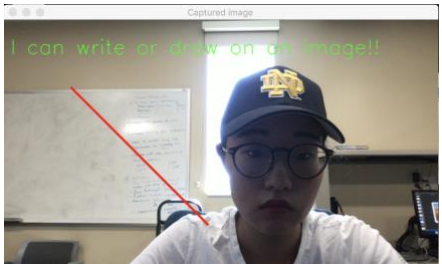
If nothing shows up after this line, the initial test for the OpenCV is successful.

4. Run the "webCamExample.py" using the following command:

```
$ python webCamExample.py
```

It tests if you can access your webcam from Python, and if you can use OpenCV to write and draw on your captured image. Press "c" to capture an image and Esc to stop the loop if it works well.



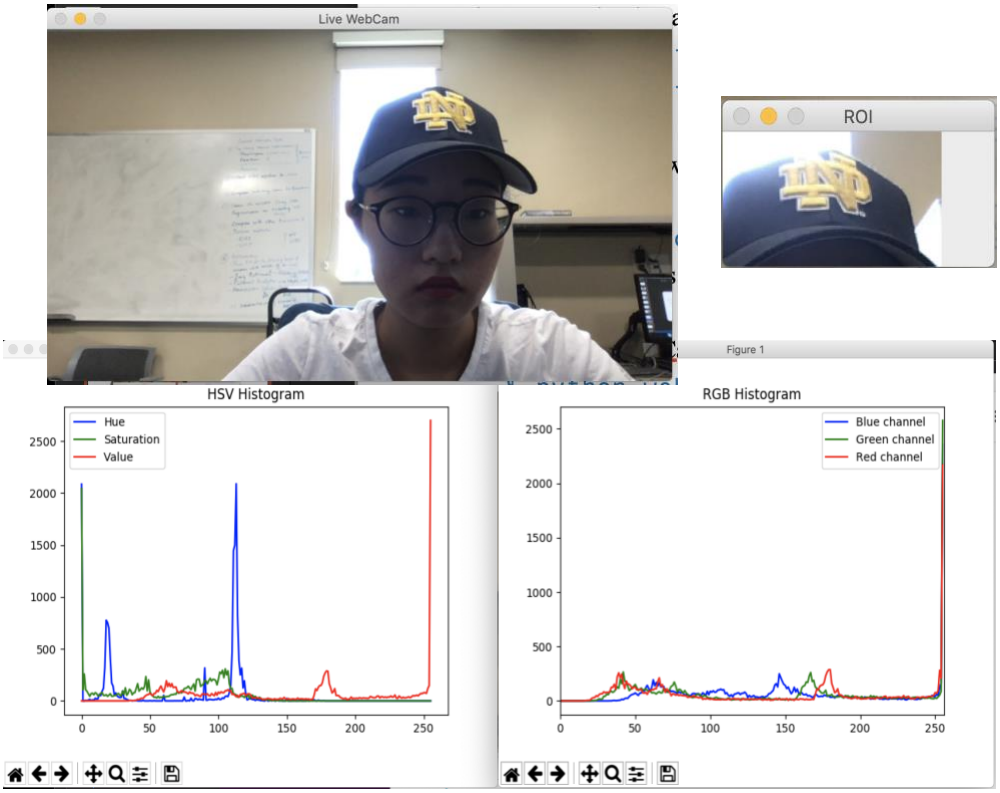


5. Run the "hsvSelection.py" using the following command:

```
$ python hsvSelection.py
```

We will use this script in class to select ranges of RGB or HSV values. It's also a good test if your matplotlib package works.

Following the steps and comments given in the codes, you should be able to produce the following results.



Troubleshooting

- If you see a conflict in the packages from anaconda and could not install the OpevCV, please check your python version first and confirm you are not using a python 3.7.x.
- If you find out you are not able to obtain the HSV and RGB plot, make sure you have installed the right version of Matplotlib. More details can be found here: https://matplotlib.org
- If the script crash and report an error telling you there is "unrecognized selector sent to instance" when you try to press "SPACE" or "ENTER", this is an error due to the backend of MacOS UI and the Matplotlib. You may check here for a sample solution: https://github.com/MTG/sms-tools/issues/36. And make sure you are using "TkAgg" as the backend. You may also add following lines at the start of your script to solve the problem:

```
from sys import platform as sys_pf
if sys_pf == 'darwin':
    import matplotlib
    matplotlib.use("TKAgg")
import matplotlib.pyplot as plt
print(matplotlib.get_backend())
```

Instructions for Windows

1. If you do not have Python installed, go to https://www.python.org/downloads/ and install Python 3.6.x (the first practical was tested with Python 3.6.5). Double check if "pip" option is selected (we will need pip).

2. Install OpenCV using pip:

   ```
   > pip install opencv-python
   > pip install opencv-contrib-python
   ```

3. We will need numpy and matplotlib in this first practical:

   ```
   > pip install numpy
   > pip install matplotlib
   ```

4. Follow points 3-5 of the instructions for MacOS (above).

Part 2: Follow these tutorials allowing you to start the adventure with OpenCV

Basic Operations on Images
link: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_core/py_basic_ops/py_basic_ops.html

Changing Color Spaces
link: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_colorspaces/py_colorspaces.html#converting-colorspaces

Image Thresholding
link: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_thresholding/py_thresholding.html