Computer Vision I (CSE 40535/60535)

Practical 4: CNN-based object detection
Friday, November 15, 2019

Tasks to do at home **before class**

To work on practical 4, we are going to need Python3 (for instance 3.6.x) and a few libraries installed, in addition to the OpenCV library. **Please do not use Python 3.7.x for this practical**.

1. **If you installed OpenCV and Python3 (<3.7) to complete task 1 (color-based object detection), you can skip this step.** If you don't have Python3 and OpenCV, you can use your preferred way to install them, for instance install Anaconda Python (https://www.anaconda.com/download/), and then use its package manager to install OpenCV:

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

2. Now, all you need to do is to install `scikit-learn`, `tensorflow` and `keras`. This can be done with these commands:

   If you are using pip:

   ```
       $ pip install numpy scipy scikit-learn
       $ pip install h5py
       $ pip install tensorflow
       $ pip install keras
   ```

   If you are using anaconda:

   ```
       $ conda install -c anaconda numpy
       $ conda install -c anaconda scipy
       $ conda install -c anaconda scikit-learn
       $ conda install -c anaconda h5py
       $ conda install -c conda-forge tensorflow
       $ conda install -c conda-forge keras
   ```

   If you encountered any problems during these steps, these sites can help:
   http://scikit-learn.org/stable/install.html
   https://www.tensorflow.org/install/
   https://keras.io/#installation

3. Before using `keras`, we have to make sure it is set up to use `tensorflow` as the backend. Follow these instructions to make sure it is correctly configured: https://keras.io/backend/

4. Test the `keras` installation by calling Python and importing the `keras` module:

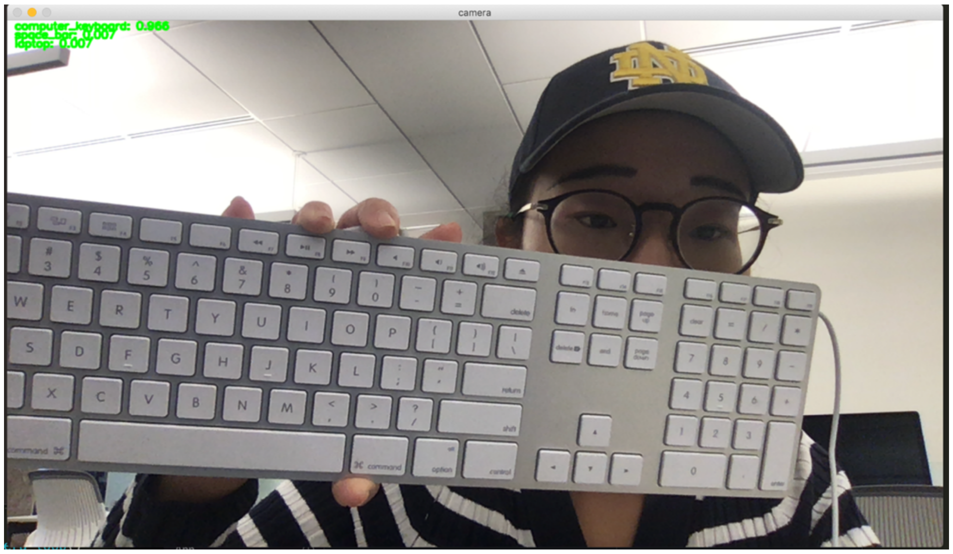
   ```
   $ python

   >>> import keras
   ```

   You should see a message saying that `keras` is using `tensorflow` backend.

5. Run the example code:

```
$ python main.py --classifier resnet
```

The first time the program runs, it will automatically download the trained weights for ResNet Convolutional Neural Network [https://arxiv.org/abs/1512.03385]. This program will use a stream of video from your webcam as the input and use ResNet to classify objects in the image.



```
computer_keyboard: 0.966
space_bar: 0.007
laptop: 0.007
```

You should be able to see the top 3 classes and the confidence associated with them in upper left corner. For this example, they are:

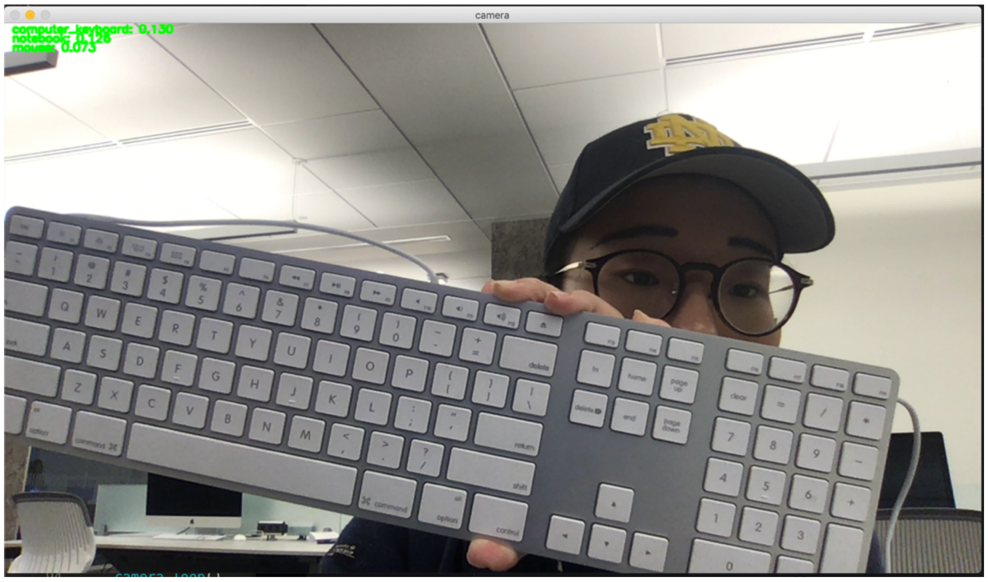
```
Computer keyboard: 0.966

Spade bar: 0.007

Laptop: 0.007
```

Here you can check the names and images of 1000 classes which the ResNet and VGG were trained for:

http://image-net.org/challenges/LSVRC/2015/browse-synsets

Present to the camera objects that belong to the classes used in training. For instance "American alligator" ☺

6. If you run the program using `--classifier vgg`, the VGG Convolutional Neural Network [https://arxiv.org/abs/1409.1556] will be downloaded and used:



```
computer_keyboard: 0.130
notebook: 0.128
mouse: 0.073
```

You should be able to see the top-3 classes and the probabilities for them on upper left corner. For this example, they are:

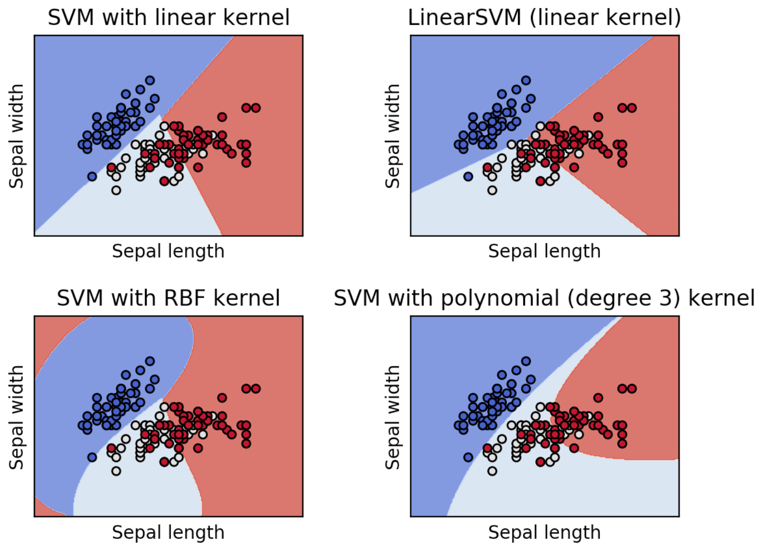
```
Computer keyboard: 0.130

Notebook: 0.128

Mouse: 0.073
```

7. Alternatively, you can run the program without any parameters, and press ‘r’ or ‘v’ to capture a frame and classify it using respectively ResNet or VGG.

8. Just to double check that the `scikit-learn` and matplotlib are up and running, run this example code that illustrates the SVM-based multiclass classification on a very simple dataset of flower features:

```
$ python svm_plot_iris.py
```

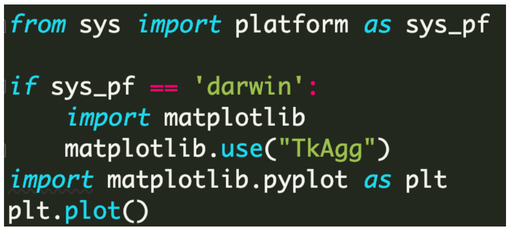
If everything works well, you should see the following picture:



Trouble shooting:

If you are using MacOS and encounter problems with matplotlib backend when running this example, you can find the code at this URL: https://scikit-learn.org/0.18/auto_examples/svm/plot_iris.html

You may copy and paste the code to a separate python script, and add following lines at the beginning of the script to set the backend:



```python
from sys import platform as sys_pf

if sys_pf == 'darwin':
    import matplotlib
    matplotlib.use("TkAgg")
import matplotlib.pyplot as plt
plt.plot()
```