Prelab 7

Prelab 7 - OpenCV Intro

Summary

In this prelab, you get familiarized with OpenCV library. Our task for the lab is to construct a video based KCF tracker, so for this prelab we want you to warm up and have a quick glance at the power of the OpenCV library.

Downloads

- Test Image: logo.png

Video Overview

We explored Image Processing in Lab 6, where each image is a 2-D array, containing pixel values of a planar world. Video, on the other hand, uses images as its basic element: frames. For a video clip, each frame is just a single image. These images are put together in a certain order, to construct a video clip. You can think of video as providing extra information about how an image changes through the new axis of 'time'.

When the video is played, the frames are displayed to us one by one in time order. With the help of human visual perception, we are able to see a fluent transition between frames, when the frames are displayed fast enough. This is called the frames per second (fps), which is similar to audio sampling frequency. Typically, fps ranges from 24fps for early films to 60fps for High Definition Video.

Install OpenCV python

From command line (or python terminal) run the following command to install the latest OpenCV library with contrib modules included.

COPY

```
pip install opencv_contrib_python
```

Refer to lab 7 page for more information about OpenCV.

Part 1 - Image Manipulation with OpenCV

To warm up, lets try some simple OpenCV flavor image photoshoping. The following lines will load the test image using OpenCV interface.

COPY

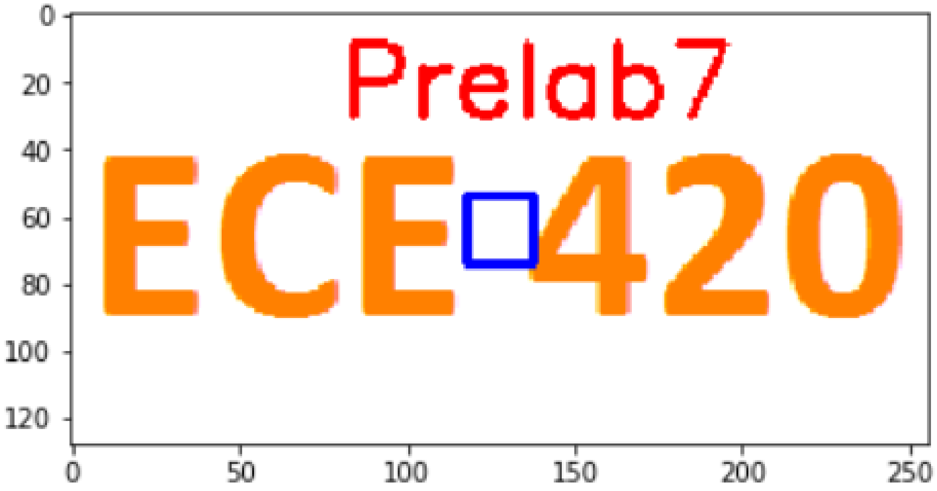
```
import cv2
import matplotlib.pyplot as plt

img = cv2.imread('logo.png')

plt.figure()
# OpenCV image channel is BGR so we flip the channels to RGB
plt.imshow(img[:,:,::-1])
plt.show()
```

> **Note**
> OpenCV have its own `cv2.imshow()` , however it is not compatible with Jupyter Notebook, so we still uses the `plt` module. Note that OpenCV loaded image is still just a Numpy array.

Draw a square with `size 20` and `color blue` on to the center of the image. Add the text `"Prelab7"` with `font FONT_HERSHEY_SIMPLEX` and `color red` on to the center top of the image. A sample image is shown below.



> **Assignment 1**
> Draw the rectangle and add the text using OpenCV functions, in particular, `cv2.rectangle()` and `cv2.putText()`.

Part 2 - Think about the Assigned Lab and the Final Project

In a few weeks, you will be doing your own course project. Please think about what you want to do for the project. Which topic do you choose? Why is it interesting or important? What are the potential challenges you could foreseen?

Refer to the course project pages for more information.

> **Note**
> You do not need to include this exercise as part of your solution in your Jupyter Notebook. We intentionally made this prelab short, so you could take some time to go through some papers and think carefully about your plan for the second half of the course. Please be aware that the final project proposal is due soon.

Grading

Prelab 7 will be graded as follows:

- Assignment 1 –> 2 point

  > *Draw rectangle on to the image (1 point)*
  >
  > *Add text on to the image (1 point)*