

Review Python

Background

This lab provides basic knowledge that you will need in order to complete future Python lab assignments. Note that we will not focus learning Python in general, rather we will focus on utilizing Python capabilities for various image processing tasks.

The most important and central Python package for doing scientific computing is 'NumPy'. This package provides convenient and fast arbitrary-dimensional array manipulation routines. Another very important package is 'PIL' which is used for performing various image processing tasks.

For starter, you would need Python 2.7 or higher. Some packages like, Numpy, PIL, os, glob, etc. You may install further Python packages when required.

Task #1: Reading and writing images with Python Imaging Library (PIL)

The most important class in the Python Imaging Library is the **Image** class, defined in the module with the same name. You can create instances of this class in several ways; either by loading images from files, processing other images, or creating images from scratch.

To load an image from a file, use the **open** function in the **Image** module.

```
>>> import PIL
>>> from PIL import Image
>>> im = Image.open("lena.ppm")
```

If successful, this function returns an **Image** object. You can now use instance attributes to examine the file contents.

```
>>> print im.format, im.size, im.mode
PPM (512, 512) RGB
```

Once you have an instance of the **Image** class, you can use the methods defined by this class to process and manipulate the image. For example, let's display the image we just loaded:

```
>>> im.show()
```

The Python Imaging Library supports a wide variety of image file formats. To read files from disk, use the **open** function in the **Image** module. You don't have to know the file format to open a file. The library automatically determines the format based on the contents of the file.

To save a file, use the **save** method of the **Image** class. When saving files, the name becomes important. Unless you specify the format, the library uses the filename extension to discover which file storage format to use.

Convert files to JPEG

```
import os, sys
import Image

for infile in sys.argv[1:]:
    f, e = os.path.splitext(infile)
    outfile = f + ".jpg"
    if infile != outfile:
        try:
            Image.open(infile).save(outfile)
        except IOError:
            print "cannot convert", infile
```

A second argument can be supplied to the **save** method which explicitly specifies a file format.

Important: For further details and playing around with different functions available in PIL, follow the link: <http://effbot.org/imagingbook/introduction.htm>

Task #2: Please follow the tutorial available at <http://www.python-course.eu/numpy.php> for a quick review of numpy arrays.

Task #3: Play around with Python (PIL, numpy, etc.)
--

- | |
|--|
| <ul style="list-style-type: none">a- Convert RGB to Greyscaleb- Apply Smoothing filter (of your choice)c- Apply Sharpening filter (of your choice)d- Convert an RGB image to Greyscale (without using any Python package) |
|--|

<p>Task for home: Draw a histogram of intensity values from any Greyscale image (without using the histogram function from packages)</p>
--

Skim Python Tutorial

Bob Dowling has written a very nice introduction to Python for absolute beginners at: <http://www.ucs.cam.ac.uk/docs/course-notes/unix-courses/PythonAB/files/handout.pdf>

Note:

Visit the following link to download pip.

<https://bootstrap.pypa.io/get-pip.py>

all the whl files are available here :-

<https://www.lfd.uci.edu/~gohlke/pythonlibs/>

Hand in

Hand in the source code from this lab at the appropriate location on the blackboard system at LMS. You should hand in a single compressed/archived file named Lab_1_<your reg. No. ABC without angle brackets>.zip that contains the following.

1. All resulting image files representing the work accomplished for this lab.
2. A plain text file named OUTPUT.txt that includes a) author information at the beginning, b) a brief explanation of the lab, c) any comments, or suggestions, d) your response to the task #3.

To Receive Credit

1. By showing up on time for lab, working on the lab solution, and staying to the end of the class period, only then you can receive full credit for the lab assignment.
2. Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.