Basics of Python and Its Application for Image Processing through OpenCV

Authors: Amirhossein Khalilian
Last edit: January 20, 2018

1   Basic Information

In this course we will be using Python as our main programming language and it is necessary for students to gain some experience in programming using Python to do the assignments. The aim of this assignment is to familiarize students with basic Python packages that will be used in the future assignments.

It is not necessary to install the latest releases of the Python for the purposes of this course and almost all of the tasks required can be done both in `2.7.x` and `3.6.x` versions. Most of the mac OS X and Linux systems have Python builds by default. Following links can be used for downloading latest versions of Python.

https://www.python.org/downloads/windows/
https://www.python.org/downloads/mac-osx/

Additionally, ANACONDA is a Python platform that has a collection of open source packages for Python and can be used on Windows, mac OS X and Linux. Almost all of the packages used in the course and many other packages can be installed and used with the ANACONDA platform. In order to download and install ANACONDA please visit following links.

https://www.anaconda.com/download/
https://docs.anaconda.com/anaconda/install/

2   Useful Packages in Python

In this section some of the useful packages in Python are introduced. These packages add more functionality to Python and allow users to perform many tasks using built-in functions. In the future assignments we will be using these packages and their functions to perform many image and video processing tasks. A brief introduction and some useful links to the documentations of these packages are presented here for students reference.

2.1   NumPy

NumPy is a package for scientific computing using Python. In NumPy the basic data type is a multidimensional array. Operations on these arrays in all dimensions including 2D are element-wise operations. However, there is a special matrix type for doing linear algebra. Operations on matrix-class arrays are linear algebra operations. Please see following link for a quick and useful tutorial on NumPy:

https://docs.scipy.org/doc/numpy-dev/user/quickstart.html

Additionally, following link can be very helpful to proficient MATLAB users to learn similarities and differences between NumPy and MATLAB.

In the provided sample codes, `Example1.py` shows some simple array and linear algebra operations using NumPy.

2.2 SciPy

SciPy is a Python-based ecosystem of open-source software for mathematics, science, and engineering. The SciPy library is one of the core packages that make up the SciPy stack. It provides many user-friendly and efficient numerical routines such as routines for numerical integration, optimization and signal processing. For useful examples and tutorials please refer to the following link:

Signal processing (`scipy.signal`) and Linear algebra (`scipy.linalg`) submodules of SciPy can be useful for the future assignments in this course. `Example2.py` shows a short example of calculating norms of a vector using `scipy.linalg` library.

2.3 matplotlib

Matplotlib is a Python plotting library and can be used to generate plots, histograms, power spectra, bar-charts, error-charts, scatter-plots, etc. For simple plotting the `pyplot` module provides a MATLAB-like interface and users can have control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users. Matplotlib is included in ANACONDA packages. For documentation and tutorials please refer to the following links:

As an example, in `Example3.py` file in the sample codes, this library is used for plotting some arbitrary generated data. The aim of this example is to show plotting of 2D data with different styles and colors, adding legend to the plot and displaying and saving the result as shown in Fig.1.
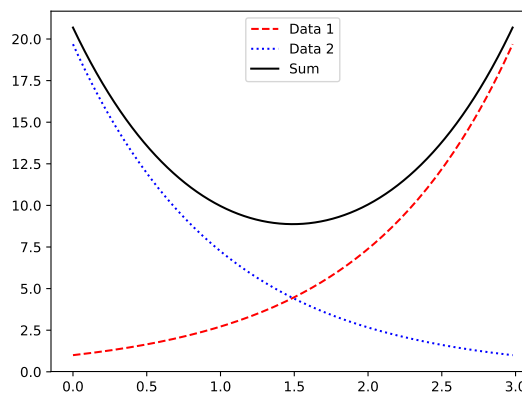

Figure 1: Result of Example3.py in sample codes

2.4 OpenCV

OpenCV is an open source computer vision library and many useful functions can be found in this library for the purposes of this course. For tutorial examples of OpenCV using Python please refer to the following link:

http://opencv-python-tutroals.readthedocs.io/en/latest/

`Example4.py` shows how to load and display an image using OpenCV in Python.

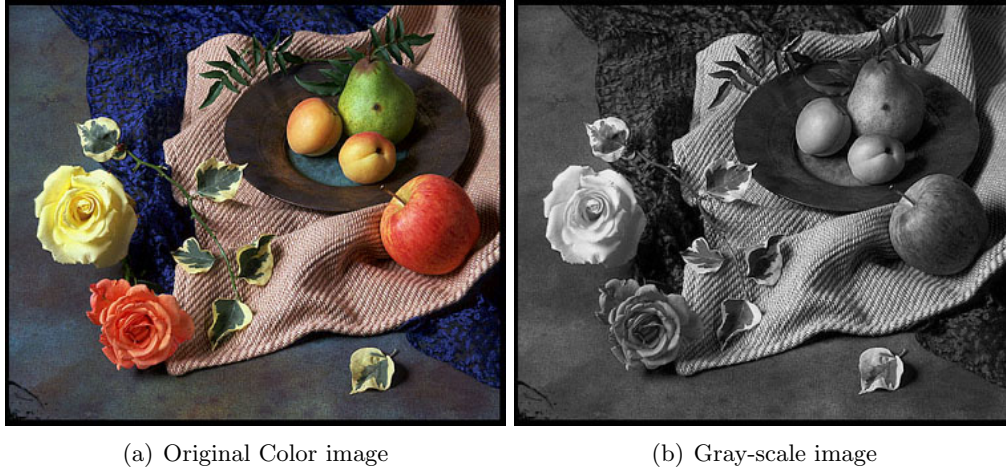(a) Original Color image   (b) Gray-scale image

Figure 2: Result of Example4.py in sample codes

`Example5.py` shows how to apply a simple filter on an image using OpenCV. The kernel used in `Example5.py` is a simple moving average and has a blurring effect.

(a) Original image   (b) Blurred image

Figure 3: Result of Example5.py in sample codes