**Lab1**

Part1: Introduction to Image Processing in Python

**Objective:**

* Understanding the basics of Python, Jupyter, Skimage.
* Reading an image and then plot it.
* Indexing Numpy matrices
* HSV colormap.
* Function with positional or keyword arguments

**Basics about Python + Skimage + Numpy:**

* Images are treated as Matrices where every pixel is a matrix element.
* All operation defined on matrices works with images (e.g. \*, /, +, -, sin, cos ...etc.)
* Can read different image formats (BMP, GIF, HDF, JPEG, PCS, PNG, TIFF, XWD).
* Image matrices are
  + 2D matrix: {0, 1} in Binary Images.
  + 2D matrix: double[0, 1] or uint8[0, 255] in Intensity Images (Gray Scale).
  + 3D matrix (MxNx3).

**Requirements:**

* Read an image and save it in a variable and copy half of the image to another variable (Hint: Use io.imread).
* Print the shape of an image (Hint: Use: shape property of image object)
* Display image (Hint: Use io.imshow).
* Convert RGB image to Gray Scale image (Hint: Use rgb2gray).
* Write a function that takes a path of the file where it gets the gray scale of the image and display original image and the gray scale one side by side (use show\_images function).
* For the given images, show the RGB image and the 3 channels of HSV image separated:
  + Use rgb2hsv (to get the hsv representation of the image).
  + To separately get the Hue, Saturation and Value channels, use hsvImg[:,:,X], where hsvImg is the hsv representation of the image. Hue is the first channel, Saturation is the second and value is the last channel.
  + Test for the three images. And comment on the results.

Part2: Noise

**Objective(s):**

* Understand Noise effect on images and how to produce it.

**Requirement(s):**

1. For an image of your choice (the effect of noise must be obvious):
   * + Read the image.
     + Convert it to grayscale.

* Apply salt & pepper noise with amount=0.05, 0.5 and 0.9

***Hint***

*Use* ***random\_noise(image, mode='s&p',amount)***

\*\* 2. From the other images. Recommend one image that won’t be greatly affected by the noise and state why.

Part3: Histogram

**Objective(s):**

* Understand Histogram.
* Get histogram for different images and understand the difference.

**Requirement(s):**

1. For the given images:
   * + Read the image.

* Apply histogram and show it.

***Hint***

*Use* histogram***(image)***

\*\* 2- Draw a grey-scale image that has uniform histogram (same number of pixels for all intensity levels) using code only. Let the size of the image be 256x256.

Useful Functions and Attributes

|  |  |  |
| --- | --- | --- |
| Name | Attribute or Function | Usage |
| shape | Attribute for Numpy array | Gets the shape of the matrix (Array). |
| io.imread | Function | Reads an image into a Numpy matrix |
| io.imshow | Function | Shows an image to a plot |
| plt.figure | Function | Generates a new figure |
| rgb2gray | Function | Converts RGB image to Gray |
| rgb2hsv | Function | Converts RGB image to HSV |
| random\_noise | Function | Adds noise to image |
| histogram | Function | Gets histogram of an image |

Custom Functions:

|  |  |  |
| --- | --- | --- |
| Name | Attributes or functions | Usage |
| show\_images | Function | Takes two arrays one for images’ matrices and the second for images’ titles and draws images accordingly for example  show\_images([img1,img2],[‘Title1’,’Title2’] |
| Show\_hist | Function | Reads an image into a Numpy matrix |