Image Processing in Python:

Images define the world, each image has its own story, it contains a lot of crucial information that can be useful in many ways. This information can be obtained with the help of the technique known as **Image Processing**.

It is the core part of computer vision which plays a crucial role in many real-world examples like robotics, self-driving cars, and object detection. Image processing allows us to transform and manipulate thousands of images at a time and extract useful insights from them. It has a wide range of applications in almost every field.

Python is one of the widely used programming languages for this purpose. Its amazing libraries and tools help in achieving the task of image processing very efficiently.

Through this article, you will learn about classical algorithms, techniques, and tools to process the image and get the desired output.

Let’s get into it!

What is image processing?

As the name says, image processing means processing the image and this may include many different techniques until we reach our goal.

The final output can be either in the form of an image or a corresponding feature of that image. This can be used for further analysis and decision making.

## Image processing tools

### **1. OpenCV**

It stands for Open Source Computer Vision Library. This library consists of around 2000+ optimised algorithms that are useful for computer vision and machine learning. There are several ways you can use opencv in image processing, a few are listed below:

* Converting images from one color space to another i.e. like between BGR and HSV, BGR and gray etc.
* Performing thresholding on images, like, simple thresholding, adaptive thresholding etc.
* Smoothing of images, like, applying custom filters to images and blurring of images.
* Performing morphological operations on images.
* Building image pyramids.
* Extracting foreground from images using GrabCut algorithm.
* Image segmentation using watershed algorithm.

where x and y are spatial coordinates. The amplitude of F at a particular value of x,y is known as the intensity of an image at that point. If x,y, and the amplitude value is finite then we call it a digital image. It is an array of pixels arranged in columns and rows. Pixels are the elements of an image that contain information about intensity and colour. An image can also be represented in 3D where x,y, and z become spatial coordinates. Pixels are arranged in the form of a matrix. This is known as an **RGB image**.

### **2. NumPy**

With this library you can also perform simple image techniques, such as flipping images, extracting features, and analyzing them.

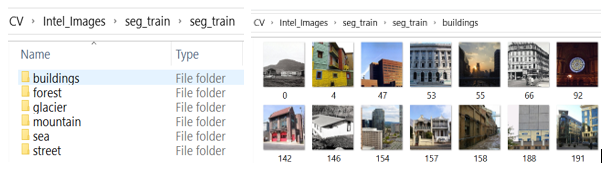
Images can be represented by numpy multi-dimensional arrays and so their type is **NdArrays**. A color image is a numpy array with 3 dimensions. By slicing the multi-dimensional array the RGB channels can be separated.

Below are some of the operations that can be performed using NumPy on the image (image is loaded in a variable named **test\_img** using imread).

**Typical steps for loading custom dataset for Deep Learning Models**

1. **Open the image file**. The format of the file can be JPEG, PNG, BMP, etc.
2. **Resize the image to match the input size for the Input layer of the Deep Learning model**.
3. **Convert the image pixels to float datatype.**
4. **Normalize the image** to have pixel values scaled down between 0 and 1 from 0 to 255.
5. **Image data for Deep Learning models should be either a numpy array or a tensor object.**

**The folder structure of the custom image data**



RGB image: It contains three layers of 2D image, these layers are Red, Green, and Blue channels.

* Grayscale image: These images contain shades of black and white and contain only a single channel.



### **Edge Detection in image processing**

Edge detection is an image processing technique for finding the boundaries of objects within images. It works by detecting discontinuities in brightness.

This could be very beneficial in extracting useful information from the image because most of the shape information is enclosed in the edges. Classic edge detection methods work by detecting discontinuities in the brightness.

It can rapidly react if some noise is detected in the image while detecting the variations of grey levels. Edges are defined as the local maxima of the gradient.

The most common edge detection algorithm is **sobel edge detection algorithm**. Sobel detection operator is made up of 3\*3 convolutional kernels. A simple kernel Gx and a 90 degree rotated kernel Gy. Separate measurements are made by applying both the kernel separately to the image.

 \* denotes the 2D signal processing convolution operation.



# **Use the Unsharp Mask filter**

The Unsharp Mask filter reproduces a traditional film technique used to sharpen edges in an image. The Unsharp Mask filter corrects blur introduced during photographing, scanning, resampling, or printing. It is useful for images intended for both print and online viewing.

Unsharp Mask locates pixels that differ from surrounding pixels by the threshold you specify and increases the pixels’ contrast by the amount you specify. For neighboring pixels within the specified radius, the lighter pixels get lighter, and the darker pixels get darker.

The effects of the Unsharp Mask filter are far more pronounced on‑screen than in high-resolution printed output. If your final destination is printed output, experiment to determine what settings work best for your image.



**Y**ou have seen a few of the features of a good introductory image processing program. There are many more complex modifications you can make to the images. For example, you can apply a variety of filters to the image. The filters use mathematical algorithms to modify the image. Some filters are easy to use, while others require a great deal of technical knowledge. The software also will calculate the ra, dec, and magnitude of all objects in the field if you have a star catalog such as the Hubble Guide Star Catalog (although this feature requires the purchase of an additional CD-ROM). The standard tricolor images produced by the SDSS are very good images. If you are looking for something specific, you can frequently make a picture that brings out other details. The "best" picture is a very relative term. A picture that is processed to show faint asteroids may be useless to study the bright core of a galaxy in the same field. Research Challenges