Image processing using openCV

Subrabala Dash (AP21110010480)
Smaranika Bhattacherya (AP21110010507)
Piyush Shahapurkar (AP21110010505)

Introduction

Image processing is a technique used to manipulate an image through several sets of algorithms resulting in an output image or any feature abstracted from the image.

The manipulations in an image include conversion to grayscale, separating the layers of RGB pixels, blurring an image, edge detection and cat face detection.

What actually happens behind these photo editing applications?

An image is a collection of pixel values in the form of numpy arrays.

Before we jump into image processing, we need to first understand what exactly constitutes an image. An image is represented by its dimensions (height and width) based on the number of pixels. For example, if the dimensions of an image are 500 x 400 (width x height), the total number of pixels in the image is 200000.

- Grayscale - A pixel is an integer with a value between 0 to 255 (0 is completely black and 255 is completely white).
- RGB - A pixel is made up of 3 integers between 0 to 255 (the integers represent the intensity of red, green, and blue).
- RGBA - It is an extension of RGB with an added alpha field, which represents the opacity of the image.

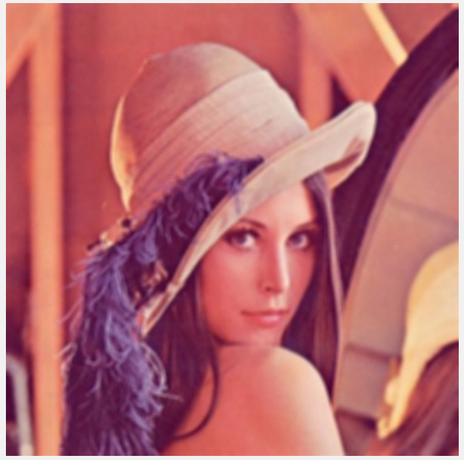
Our project mainly comprises of few basic filters applied on images using OpenCV and NumPy arrays.

It is a fun project which allows us to visualize how images are processed in a computer and the functioning of various inbuilt features in the above-mentioned libraries.

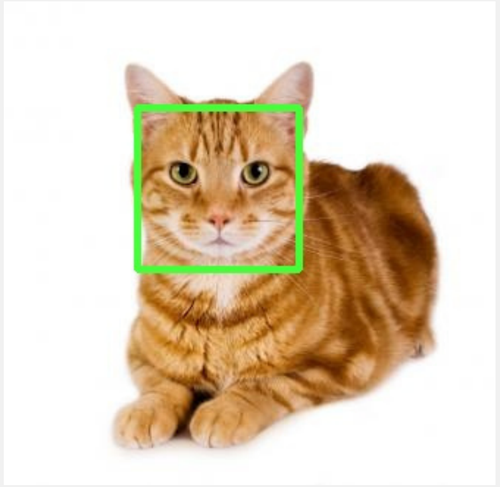
Features

- Grayscale
- Color channel separation
- Edge detection
- Circle detection
- Cat face detection
- Gaussian blur

FUNCTIONS



Negative	Blur effect	Grayscale
For negative transformation of an image then we need to invert the 3 channels (rgb). We do this by subtracting 255 from each of the original image's pixels.	The blur() function traverses through the numpy array of the image and takes the average value of the surrounding pixels and updates the value to create a blur image. This is also called average blur method.	A normal rgb image is a 3-d numpy array to convert it into a grayscale image, we need to remove two layers from the rgb panels. This can be done using in-built functions or manually.



Edge detection	Cat face detection	Circle detection
The canny filter is used to detect the edges in an image, we can adjust the frequence of these edges by giving required parameters.	An xml file is used as reference to predictthe location of cat’s face is present in the image. It is highlighted with a square according the dimension of the face.	Detection of elements in an image is done on gray-scale images. Using HoughCircle function and HoughGradient method, any circle present in the image gets detected and is highlighted .

Resources

- Github
- W3schools
- YouTube
- Research Papers

CONCLUSION
CONCLUSION

This task has challenged us in both qualitative and quantitative ways. We strive further to improve our code by adding new features as we explore the great depths of programming languages and keep expanding our horizons. As of now, I hope our efforts and hardwork are reflected through this projects.