Experiment no 1
To perform Handling file ,Camera ,GUIs
Date of Performance :17/07/2023
Date of Submission :24/07/2023

Aim:-

To perform Handling Files, Cameras and GUIs

Objective:-

To perform Basic I/O Scripts, Reading/Writing an Image File, Converting Between an Image and raw bytes, Accessing image data with numpy.array,Reading /writing a video file, Capturing camera, Displaying images in a window ,Displaying camera frames in a window

Theory:-

Basic I/O script:-

In python, input and output operations (I/O Operations) are performed using two built-in functions. Two built-in functions to perform output operations and input operations are:

1. **print( )** - Used for output operation.
2. **input( )** - Used for input operations.

Reading/Writing an Image File:-

Reading, displaying, and writing images are basic to image processing and computer vision.To perform any operation on image file, you'll need to first read in the images. So it's important to use these basic read operation.

OpenCV, provides the three built-in functions to read,display and write the image.They are:

1. ***imread()*** : helps us read an image

2. **imshow()** : displays an image in a window

3. **imwrite()** : writes an image into the file directory

Converting Between an Image and raw bytes:-

Conceptually, a byte is an integer ranging from 0 to 255. In all real-time graphic applications today, a pixel is typically represented by one byte per channel, though other representations are also possible.

An OpenCV image is a 2D or 3D array of the array type. An 8-bit grayscale image is a 20 array containing byte values A 24-bit BGR image is a 3D array, which also contains byte. values. We may access these values by using an expression, such a image[0, 0] or image[0, 0, 0]. The first index is the pixel's y coordinate or row, 0 being the top. The second index is the pixel's x coordinate or column, 0 being the leftmost. The third index (if applicable) represents a color channel.

For example, in an 8-bit grayscale image with a white pixel in the upper-left corner, image[0, 0] is 255. For a 24-bit BGR image with a blue pixel in the upper-left comer, image[0, 0] is [255,0,0]

Accessing image data with numpy. Array:-

The y array structure is well optimized for array operations, and it allows certain kinds of bulk manipulations that are not available in a plain Python list. These kinds of array type- specific operations come in handy for image manipulations in OpenCV. Let's explore image manipulations from the start and step by step though, with a basic example: say you want to manipulate a pixel at the coordinates, (0, 0), of a BGR image and turn it into a white pixel.

OpenCV provides the VideoCapture and Video Writer classes that support various video file formats. The supported formats vary by system but should always include an AVI. Vin its red method, a VideoCapture class may be polled for new frames until it reaches the end of its video file. Each frame is an image in a BGR format. Conversely, an image may be passed to the write() method of the VideoWriter class, which appends the image to a file in VideoWriter.

Capturing camera frames:-

A stream of camera frames is represented by the VideoCapture class too.However for a camera, we construct a VideoCapture class by passing the camera's device index instead of a video's filename.

Displaying images in a window :-

One of the most basic operations in OpenCV is displaying an image. This can be done with the imshow() function. If you come from any other GUI framework background, you would think it sufficient to call imshow() to display an image. This is only partially true: the image will be displayed, and will disappear immediately.

Displaying camera frames in a window :-

OpenCV allows named windows to be created, redrawn, and destroyed using the namedWindow(), imshow(), and destroyWindow() functions. Also, any window may capture keyboard input via the waitKey() function and mouse input via the setMouseCallback() function.

Conclusion:-

We have finally grasped how a computer vision application can be developed opencv consist of mat class which is used for pixel manipulation and it has a class relationship to application buffered Image class.