Assignment 1

N Srivijay Ram Prathap 2019H1030559G

Problem statement :

To write a function which takes input of
- An image (either grey scale or RGB image)
- Number of filters (positive integer, m)
- Set of 2D matrices as filters
- Stride (positive integer, s) and

and impement the convoution operator on the image with the given filters. Appropriate consistency measures should be take care regarding the input arguments.

Background work:

Given an input of image it should be converted to an array of pixel densities, Considering the input of the function as name of the image example “dog.jpg” using “Image” class in “PIL” library the image is fetched and stored into an image object “image_obj”.

This image object is converted to an array using asarray method in the numpy library and stored into “data” variable. The number of fiters is stored in “m”. The set of 2D filters for an image should be equal to m*d where d is the depth of the image. So to check regarding the consistency of the input the number of 2D fiters be equal to m*d. Lets say number of input 2D filter be n then, n=m*d. Else return an error.

If these conditons satisfy then start the convolutiion operation. Consider the image has a shape of l*b*d, and filter has a shape of r*c. The output result array after the convolution should be of size [((l-r) / s) + 1 , ((b-c) / s) + 1)] where s is the stride length.

For a single filter the number of output 2D matrices will be equal to the depth of the image. In case of RGB the depth is 3. So for every filter there will be 3 2D matrices. These three matrices should be combined represnt again a image of depth 3.

The convolutions operation is implemented from the starting point of the image with a dimension of the fiter. This part of the image is element wise multiplied with the filter and sum of those values is store as result of first iteration. These operation continues by sliding the image input horizantally and vertically with the appropriate stride size till the end of the image.

All the calculated values are store into a array of the specified dimensions. These steps are repeated for each 2D filter. The final 3 outputs of three 2D filters are combine to form a 3D matrix with depth of 3.

RESULTS:

Example :

Input : An image of size 426 * 640 RGB , so the dimensions will be 426*640*3 and m=2.So the number of 2d matrices are 2*3 = 6. the set of filters is stored in a varibale fltr,

```
fltr=[[[1,0,1],
  [1,1,1],
  [1,0,1]],
```

[[0,0,1],
[1,1,1],
[1,0,1]],

[[1,0,0],
[1,1,1],
[1,0,1]],

[[1,0,1],
[0,1,0],
[1,0,1]],

[[1,0,1],
[1,0,1],
[1,0,1]],

[[1,0,1],
[0,0,0],
[1,0,1]]]
and a stride = 1
Output:
output matrix is a size of 2*424*638*3 which is two 3D matrices of size 424*638*3.

Conclusion:

The function myconv2D takes input as image and fiters and produce an convolved array.As expected this function gives the output of m 3D matrices. This can be used for grayscale images also.

EXECUTION ENVIRONMENT: JUPYTER NOTEBOOK OR COLAB
NEED “DOG.JPG” IMAGE IN THE EXECUTION FOLDER TO RUN THE EXAMPLE.