#ALGORITHM OF IMAGE FUSION

*BY SHUBHAM GARG*

1. Install the OpenCV (Open Source computer vision) Library and import it in the program.
2. Initialize an array to hold the images.

img\_fn = ["img1.JPG", "img2.JPG","img3.JPG", "img4.JPG"]

1. Read the images entered by user in the array.

img\_list = [cv2.imread(fn) for fn in img\_fn]

1. Calling (precisely declaring) the merge mertens function.

mergeMertens = cv2.createMergeMertens()

1. Using resolution fusion function (pre-defined for merge mertens algorithm as it will only fuse images if they are of same resolution) with merge mertens to apply fusion to the images read by img\_list function.

resFusion = mergeMertens.process(img\_list)

1. It is necessary to keep R, G, B subpixels in the range [0.0 .. 1.0]. To do so we multiply the resFusion function with 255 and pass that directly to imwrite. This takes care of clipping (i.e. does not produce resulting image negative or distorted colors).
2. Saving the resulting fused image as fusion.jpg in the working folder.   
   cv2.imwrite("fusion.jpg", resFusion\*255)

#IMAGE FUSION

*BY SHUBHAM GARG*

import cv2

img\_fn = ["img1.JPG", "img2.JPG", "img3.JPG", "img4.JPG"]

img\_list = [cv2.imread(fn) for fn in img\_fn]

mergeMertens = cv2.createMergeMertens()

resFusion = mergeMertens.process(img\_list)

cv2.imwrite("fusion.jpg", resFusion\*255)