

## Experiment 3

# Spatial Filtering

---

*Spatial filtering* is a technique for modifying or enhancing an image. For example, you can filter an image to emphasize certain features or remove other features. Image processing operations implemented with filtering include smoothing, sharpening, and edge enhancement. Filtering is a *neighborhood operation*, in which the value of any given pixel in the output image is determined by applying some algorithm to the values of the pixels in the neighborhood of the corresponding input pixel. A pixel's neighborhood is some set of pixels, defined by their locations relative to that pixel.

---

### Problem Objective

Write C/C++ modular functions/subroutines to design spatial filters - **mean, median, gradient, Laplacian, Sobel kernels (horizontal, vertical, diagonal)** on a stack of grayscale images (say, 15 images per stack)

Use OpenCV (or) ImageJ for image reading, writing and GUI development only. Use the OpenCV **tracker (slider)** functionality to show outputs for varying sizes of neighborhoods. You may have different sliders to select (i) Image (ii) Filter (iii) Neighborhood size

(a) **Input:** Path to the stack of images. Input stack **should** contain the (provided) noisy images, and may also contain the normal test images, e.g. jetplane.jpg, lake.jpg, livingroom.jpg, mandril\_gray.jpg, pirate.jpg, walkbridge.jpg

(b) **Output:** Filtered stack of images should be shown beside input stack in the same pane of GUI with a slider to vary filter/kernel size/change image.

### Note

1. Do not hardcode the filenames and/or image size into the code.
2. Use proper code commenting and documentation.
3. Use self explanatory identifiers for variables/functions etc.

### References

1. Gonzalez, Woods "Digital image processing" 3/e, Chapter 3, Prentice Hall.