

# CMSC 317: The Computational Image

## Assignment 3: Filtering

### Part One

As part of a Processing sketch, you should implement the following filters (each written as a function that takes a `PImage` and returns a new `PImage`):

- Blur (using the box or gaussian blur)

$$\frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

- Vertical Sobel (the resulting values may be negative, either add an offset or use the absolute value)

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

- Horizontal Sobel (the resulting values may be negative, either add an offset or use the absolute value)

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

- Sobel (as the magnitude of the gradient from the horizontal and vertical sobel operators)

### Part Two

Reimplement your blur filter as two one-dimensional filters. Compare the computation time spent with the standard  $k^2$  filter and the  $2k$  implementation that exploits the linear separability of the filter.

$$\frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \frac{1}{25} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} [1 \ 1 \ 1 \ 1 \ 1] \quad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} [-1 \ 0 \ 1]$$

### Part Three

Use your filters to display the webcam's images in a creative manner. Some possible ideas:

- Use a sequence of filters in some creative way.
- Use different filters in different parts of the image.
- Use a sequence of filters overlayed using an alpha layer on the original image.