



Gabriele Tenucci

Assignment 6



Code

```
#slide over every output cell to calculate its value
for currentOutRow in range(outRows):
    for currentOutCol in range(outCols):
        #slide over every kernel cell, calculate the current outputcell value
        for currentKerRow in range(kerRows):
            for currentKerCol in range(kerCols):
                rowDiff = currentOutRow - currentKerRow #----sliding bottom to top
                colDiff = currentOutCol - currentKerCol #----sliding right to left
                if (rowDiff >= 0) and (rowDiff < imgRows):
                    if (colDiff >= 0) and (colDiff < imgCols):
                        oldOut = out[currentOutRow, currentOutCol]
                        currentKer = kernel[currentKerRow, currentKerCol]
                        currentImg = image[rowDiff, colDiff]
                        out[currentOutRow, currentOutCol] =
                            oldOut + (currentKer * currentImg)

return out
```

```
for image in images:
    print("Elaborating " + image + ".bmp")
    img = cv2.cvtColor(cv2.imread("./" + folder + image + ".bmp"),
                      cv2.COLOR_BGR2GRAY)

    #apply filters and save images
    filterX, filterY = getKernel("sobel")
    Gx, Gy, magnitude = computeImage(img, filterX, filterY)
    saveImage("out", image, "sobel", Gx, Gy, magnitude)







    filterX, filterY = getKernel("prewitt")
    Gx, Gy, magnitude = computeImage(img, filterX, filterY)
    saveImage("out", image, "prewitt", Gx, Gy, magnitude)

    filterX, filterY = getKernel("roberts")
    Gx, Gy, magnitude = computeImage(img, filterX, filterY)
    saveImage("out", image, "roberts", Gx, Gy, magnitude)
```

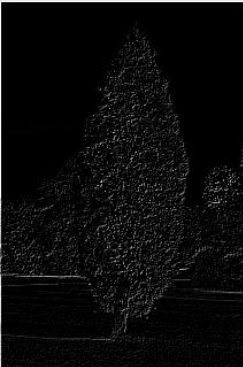
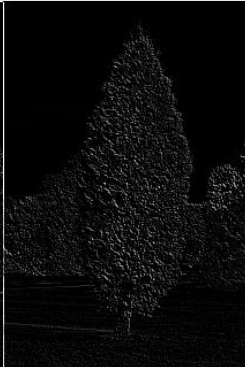
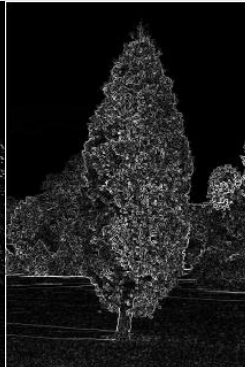
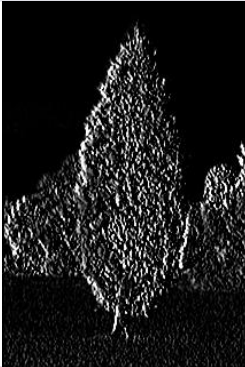
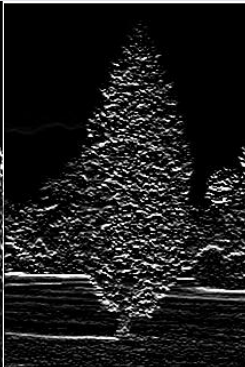
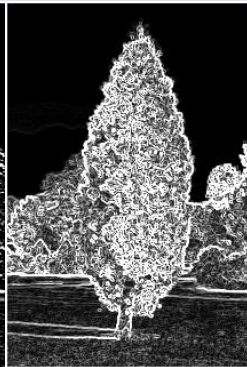
Libraries:

- OpenCV - RGB to grayscale conversion, loading and saving images to disk
- Numpy - array and matrices manipulations

Applying the filters - face

	Gx	Gy	Magnitude
Prewitt			
Roberts			

Applying the filters - tree

	Gx	Gy	Magnitude		Gx	Gy	Magnitude
Roberts				Sobel			

More kernels



$$\begin{bmatrix} 0.0625 & 0.125 & 0.0625 \\ 0.125 & 0.25 & 0.125 \\ 0.0625 & 0.125 & 0.0625 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Improvements

- Smaller kernels result in darker images
 - Possible solution: output scaling
- Implementing RGB convolutions
- Improving efficiency
 - Matrix multiplication