

CS195: Computer Vision

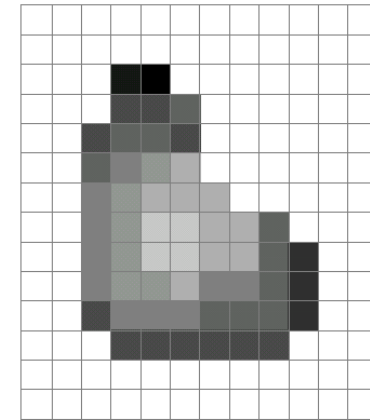
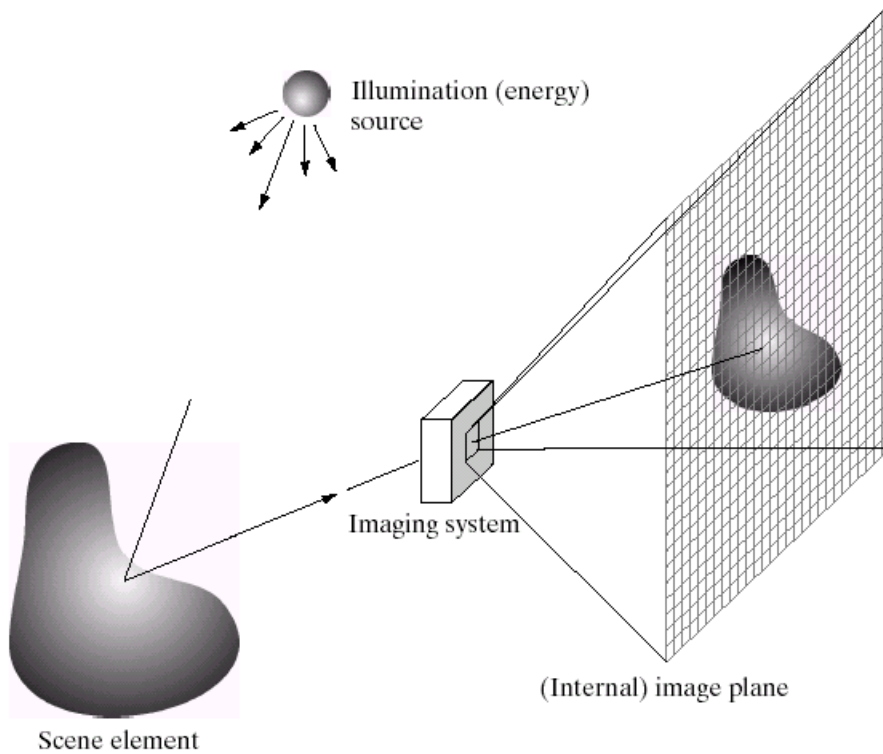
Image Filtering
Cross-correlation

August 28, 2024

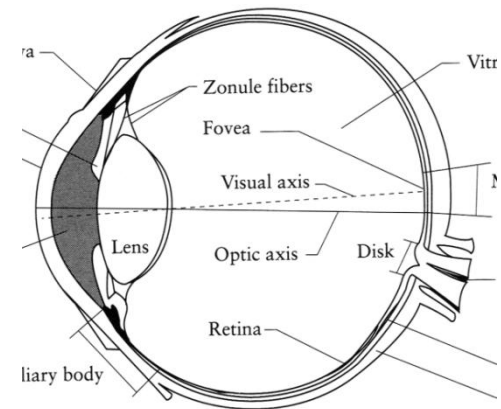


Md Alimoor Reza
Assistant Professor of Computer Science

What is an image?



Digital Camera



The Eye Source: A. Efros

Coding activity: simple image manipulation tasks using PIL library

Coding activity: cross-correlation

Cross-correlation

The mean filter is just a specific case of a *cross-correlation*, a very general operation

Let F be an image, H be a kernel (of size $2k+1 \times 2k+1$), then the cross-correlation of F with H is:

$$G = H \otimes F$$

H			F		
1/9	1/9	1/9	0	0	0
0	0	0	0	0	0
0	0	0	90	90	90

$$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v] F[i + u, j + v]$$

$0+0+0+0+0+(1/9*90)+(1/9*90)+(1/9*90)$
 $= 30$

```
print('')
...

# compute cross-correlation
F = img_pil
new_img = img_pil

for y in range(kernel_size, rows-kernel_size):
    for x in range(kernel_size, cols-kernel_size):

        # compute cross-correlation centered at pixel location (x, y)
        old_pixel_value = F.getpixel((x, y)) # we don't need it anymore

        new_pixel_value = 0.0
        for v_row in range(-k, k+1):
            for u_col in range(-k, k+1):
                cur_kernel_value = H[v_row + k, u_col + k] # small trick to adjust the indexing
                cur_pixel_value = F.getpixel((x + u_col, y + v_row))
                # MODIFICATION 1: calculate the updated value of pixel
                # your code ...

        # update the value at location (x,y) with newly computed pixel value
        # MODIFICATION 1: change the pixel value in the 'new_img' with the calculate new value
        # your code ...
```

More activities

- In class activities: Implement linear filtering with the following Kernels.

.33	0	0
.33	0	0
.33	0	0

Kernel 1

0	0	.33
0	.33	0
.33	0	0

Kernel 2

.33	.33	.33
0	0	0
0	0	0

Kernel 3

.33	0	0
0	.33	0
0	0	.33

Kernel 4