Image Filtering Preliminaries

- Denote the image by F (to follow the book).
- Represent weights as a second image, H (the kernel).
- Pretend that images are padded to infinity with zeros (so sums don't need limits).
- To shift a function f (x,y) up and to the right by (a,b)
  - f (x-a, y-b)

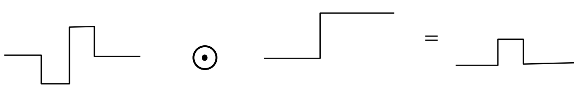
Correlation

- Denote by $\odot$
- Then the definition of discrete 2D correlation is:

$$R_{i,j} = \sum_{u,v} H_{u-i,\,v-j} \; F_{u,v}$$

Puts filter on (i,j)

Correlation example



(Extra slide, not done in class).

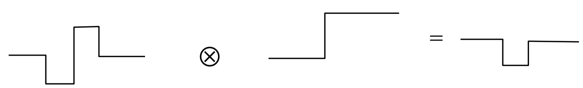
Convolution

- Denote by $\otimes$
  - Others symbols include * (for 1D) and ** (for 2D).
- The definition of discrete 2D convolution is:

$$R_{i,j} = \sum_{u,v} H_{i-u,\,j-v} \; F_{u,v}$$

- Notice weird order of indices (includes the flips)

Convolution example



(Extra slide, not done in class).

Properties of $R_{i,j} = \sum_{u,v} H_{i-u,j-v} \; F_{u,v}$

- Linear
- Commutative
- **Associative** (Can save CPU time!)

$$(A \otimes B) \otimes C = A \otimes (B \otimes C)$$

- Output is a s**hift-invariant** function of the input (i.e. shift the input image two pixels to the left, the output is shifted two pixels to the left)
- Converse of above is true: If a system is linear and shift invariant, then it is a convolution.

Shift invariant linear systems (§7.2)

- Shift invariant
  - Shift in the input means we simply shift the output
  - Example: Optical system response to a point of light
    - Light moves from center to edge, so does its image
- Linear shift invariant
  - Can compute the output due to complex input, based on the response to a single point input
    - Discrete version---function *box*(x,y) is zero everywhere except at (x',y') where is is 1.
    - Continuous version---delta function
- f(x,y) is a linear combination of shifted versions of *box*(x',y')

Rewrite f(i,j) as a sum over its natural basis

$$f(i,j) = \sum \sum box(i-u, j-v) f(u,v)$$

Box shifted by (u,v). Note subtraction!

Given that

$$\text{Response}(box(i,j)) = h(i,j)$$

Shift invariance means that

$$\text{Response}(box(i-u, j-v)) = h(i-u, j-v)$$

Linearity means we can bring the response inside the sum.

$$\text{Response(f(i, j))} = R_{ij} = \sum \sum \; h(i-u, j-v) f(u,v)$$

(Convolution by h)

Response as sum of basis functions (§7.2)

- The response is linear combination of shifted versions of the kernel
- The weights are the values of the function being convolved
- The shifted versions of the kernels form a basis over which the result image is constructed
- Thinking of an image as a weighted sum over a basis is a generally useful idea—e.g., Fourier transforms.

Convolution example (from MathWorks website)

For example, suppose the image is

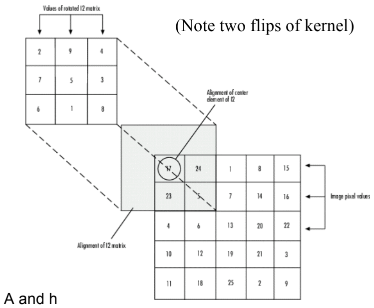
```
A = [17  24   1   8  15
     23   5   7  14  16
      4   6  13  20  22
     10  12  19  21   3
     11  18  25   2   9 ]
```

and the convolution kernel is

```
h = [8  1  6
     3  5  7
     4  9  2]
```

R(1,1) = 5*17+3*24+1*23+8*5

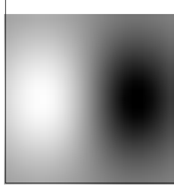
(Note two flips of kernel)

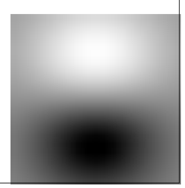


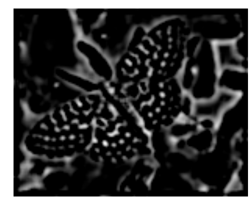
To do the complete convolution, set A and h as above in Matlab, and do conv2(A,h,'same'). Try also conv2(A,h) --- make sure you understand the difference!

Filters are templates

- Applying a filter at some **point** can be seen as taking a dot-product between the image and some vector
- Filtering the image yields a set of dot products
- Useful intuition
  - filters look like the effects they are intended to find
  - filters find effects that look like them

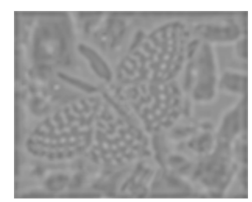






Positive responses

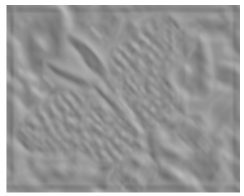
Zero mapped to gray



Scaled to have min 0, max 255

Positive responses

Zero mapped to gray



Scaled to have min 0, max 255

Normalized correlation

- Think of filters of a dot product
  - **problem**: brighter parts give bigger results even if the structure is same (often not what you want)
  - **normalized** correlation output is filter output, divided by root sum of squares of values over which filter lies

$$\frac{\mathbf{h} \bullet \mathbf{f}}{|\mathbf{f}|}$$ (**f** is limited to where h is non zero)

  - Can think in terms of angle between vectors. Recall

$$\cos(\theta) = \frac{\mathbf{h} \bullet \mathbf{f}}{|\mathbf{h}||\mathbf{f}|}$$ (|**h**| is not relevant to this problem)

Slide was skipped in lecture; included for reference.

Normalized correlation

- Some tricks of the trade
  - Consider template filters that have zero response to a constant region (helps reduce response to irrelevant background).
  - Consider subtracting average of image over filter area when computing the normalizing constant (can increase sensitivity).

Finding Edges

- Edges reveal much about images
- Edge representations can be seen as information compression (because boundary is fewer pixels than the inside)
- Edges are the result of many different things
  - simple material change (step edge, corners)
  - illumination change (often soft, but not always)
  - shading edges and bar edges in inside corners
- An edge is basically where the images changes---hence finding images is studying changes (differentiation)