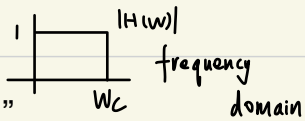
spatial filtering

1-D signal processing

$y = x * h \qquad y[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$



In 2-D, the analogue to "time domain" is "spatial domain"

[-1 2 -1] 1-D

h(-1,-1)	h(-1,0)	h(-1,1)
h(0,-1)	h(0,0)	h(0,1)
h(1,-1)	h(1,0)	h(1,1)

2-D

0	-1	0
-1	5	-1
0	-1	0

J(x,y) = 5I(x,y) - I(x+1,y) -I(x-1,y)-I(x,y+1) -I(x,y-1)

$J(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} h(s,t) I(x+s, y+t)$ looks almost like convolution, actually more like correlation

smoothing filter (aka low pass filter)
moving average filter.
idea: replace each pixel by a weighted average of its neighbours
pro: remove/reduce noise
con: blur the images, remove detail.

1/9

1	1	1
1	1	1
1	1	1

two ways of filter in matlab:
filter2(h, im)
- get actual, floating points result
- not scaled to [0,255]

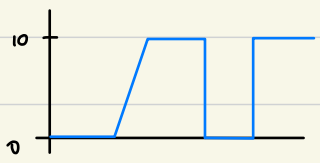
imfilter(im, h)
- unit8 -> unit8
- work with colour
- may lose important values which <0 or >255

low-pass filter: ones(n)/n^2

1	2	1
2	4	2
1	2	1

on Gaussian

sharpening filters:
image difference (return to differentiation) result in sharpening



F(x) = 0 0 2 4 6 8 10 10 5 0 0 10
F'(x) = 0 2 2 2 2 2 0 -5 -5 0 10 0
F''(x) = 2 0 0 0 0 -2 -5 0 5 10 -10 0
sign change

the first step in sharpening: find a filter that reacts strongly to edge

1-D [-1 1]

$g(x) = F(x+1) - F(x) \approx \frac{\partial F}{\partial x}$

[1 -2 1]

$g(x) = F(x+1) + F(x-1) - 2F(x) = \frac{\partial^2 F}{\partial x^2}$

$\left|\frac{\partial F}{\partial x}\right|$ is large when there is an edge

$\frac{\partial^2 F}{\partial x^2}$ shows sign changes near edges but is 0 elsewhere

2-D version: find edges in both x and y direction
we use an approximation of the Laplacian:

$\nabla^2 F = \frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial y^2} \rightarrow F(x,y+1) + F(x,y-1) - 2F(x,y)$

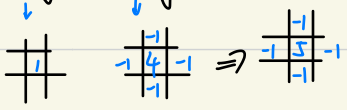
F(x+1,y) + F(x-1,y) -2F(x,y)

F(x+1,y)+F(x-1,y)+F(x,y+1)+F(x,y-1)-4F(x,y)

0	1	0
1	-4	1
0	1	0

how to enhance/sharpen images?
idea: strengthen edges of original image by adding a multiple of the edge map to it

$J(x,y) = I(x,y) - \sigma^2 I(x,y)$



noise is also sharpened, which is bad.
we can just add a fraction of the edge back in, for a more subtle effect.
Unsharp masking:

orginal im : F
low-pass im : $\bar{F}$
high-pass im : $F - \bar{F} = F_{hp}$

Output = $F + k \cdot F_{hp}$
k: tuning parameter

filters need not be symmetric

-1	-2	-1
0	0	0
1	2	1

sobel horizontal edge detection

-1	0	1
-2	0	2
-1	0	1

vertical edge detection

not all image filter are linear - median filter
J(x,y) = median{ 3*3 pixel neighborhoods around I(x,y)}

50	48	46	42
52	0	50	48
46	47	255	40
51	48	46	42

salt and pepper noise

{0 46 46 47 48 50 50 52 255}
median

gaussian noise -> low pass filter
salt and pepper / impulsive noise -> median filter