The 2D discrete fourier transform

continuous tune fourier transform:

$$X(w) = \int_{-\infty}^{\infty} x(t) e^{-jwt} dt \qquad w \in (-\infty, \infty)$$

discrete-time fourier transform:

$$X(w) = \sum_{n=-\infty}^{\infty} x[n] e^{-jwn} \qquad w \in [0, 2\pi]$$

discrete fourier transform:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi k}{N}n} \qquad k = 0, 1, \cdots N-1$$

2D discrete fourier transform

$$F[u,v] = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} F[x,y] e^{-(j\frac{2\pi u}{N}x + j\frac{2\pi v}{N}y)}$$

output im $M \times N$ input im $M \times N$

$u: 0 \sim N-1 \quad v: 0 \sim M-1$

$$F[x,y] = \frac{1}{MN} \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} F[u,v] e^{j\frac{2\pi u}{N}x + j\frac{2\pi v}{M}y}$$

if $M=N$ $1/N^2$, you may see $F[u,v] = \frac{1}{N}\sum\sum$

$$F[u,v] = \underbrace{\sum_{x=0}^{N-1} \Bigg(\underbrace{\sum_{y=0}^{M-1} f[x,y] e^{-j\frac{2\pi v}{M}y}}_{\text{1-D DFT along columns}} \Bigg) e^{-j\frac{2\pi u}{N}x}}_{\text{1-D DFT along rows}}$$

1-D DFT along rows 2D-DFT is separable

matlab: FFT2

As with 1-D DFT, 2D-DFT is like a decomposition of an image into complex exponentials (sine & cosine)

$$e^{-j\frac{2\pi ux}{N}} = \cos\frac{2\pi ux}{N} - j\sin\frac{2\pi ux}{N}$$

$F[0,0] = \sum\sum f[x,y]$ = sum of all pixel intesntiy

fourier transform property

Shift: $g(x,y) = F(x-a, y-b)$

$$G(u,v) = F(u,v)\underbrace{e^{-2\pi j\left(\frac{au}{N} + \frac{bv}{M}\right)}}_{\text{complex phase shift}}$$

$|G(u,v)| = |F(u,v)|$

scale/flip

g(x,y) = af(x,y) g(x,y) = f(ax,by)

G(u,v)= aF(u,v) G(u,v)= $\frac{1}{ab} F\left(\frac{u}{a}, \frac{v}{b}\right)$

if a or b = -1: flip in spatial domain => flip in frequency domain

rotation: if g(x,y) = f(x,y) rotated ccw by θ, then

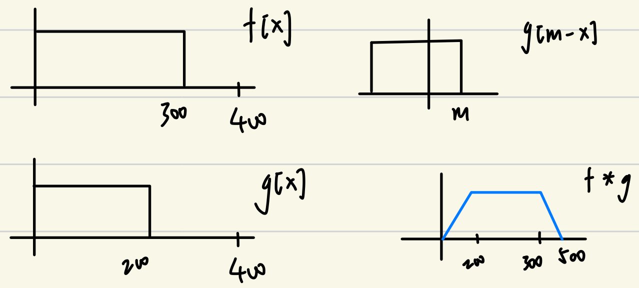
$G(u,v) = F(u,v)$ rotated ccw by θ

Convolution:

h(x,y) = f(x,y) * g(x,y) * = circular convolution

H(x,y) = F(u,v)G(u,v)

Regular convolution:



circular convolution => periodic

f*g is a periodic signal itself. this is not be what we want since copies may intrude into regions we don't expect

Solution: zero padding signals so that copies don't unexpectedly overlap

zero-padding to at least dimension:

(M1+M2-1)*(N1+N2-1) for circular convolution b/w M1*N1, M2*N2 images is sufficient

good news: filter2/imfilter in matlab does all this for u

note: if an image has strong image edges at θ, we see a strong contribution in the 2D-DFT at θ+90. Why?