EECE\CS 253 Image Processing

Lecture Notes: Frequency Filtering

Richard Alan Peters II

Department of Electrical Engineering and Computer Science

Fall Semester 2007

Convolution Property of the Fourier Transform

Let functions $f(r,c)$ and $g(r,c)$ have Fourier Transforms $F(u,v)$ and $G(u,v)$.

$*$ = convolution
$\cdot$ = multiplication

Then,

$$\mathfrak{F}\{f * g\} = F \cdot G.$$

Moreover,

$$\mathfrak{F}\{f \cdot g\} = F * G.$$

The Fourier Transform of a convolution equals the product of the Fourier Transforms. Similarly, the Fourier Transform of a convolution is the product of the Fourier Transforms

Convolution via Fourier Transform

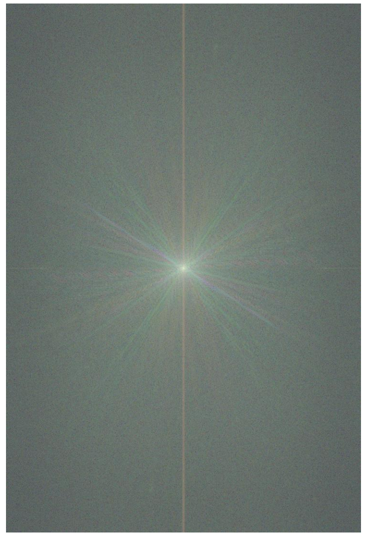
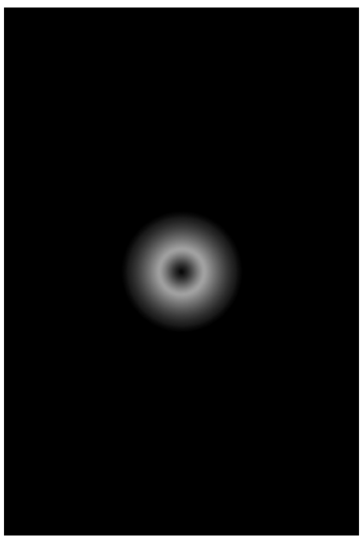
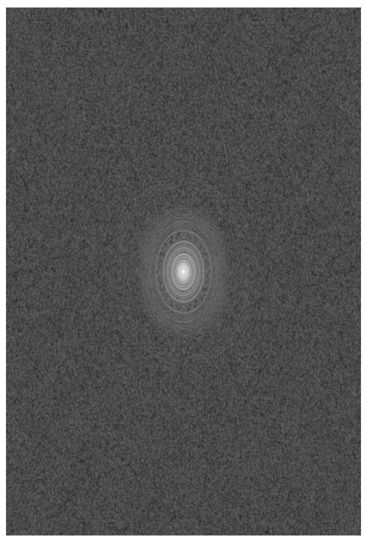
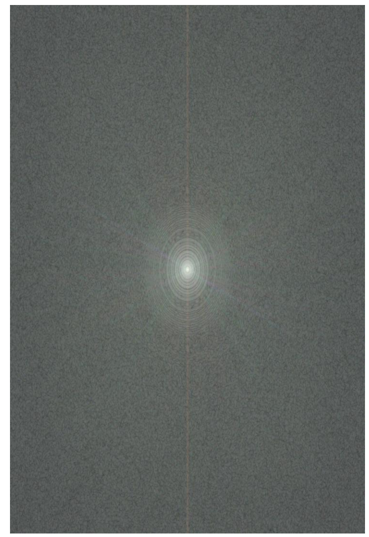



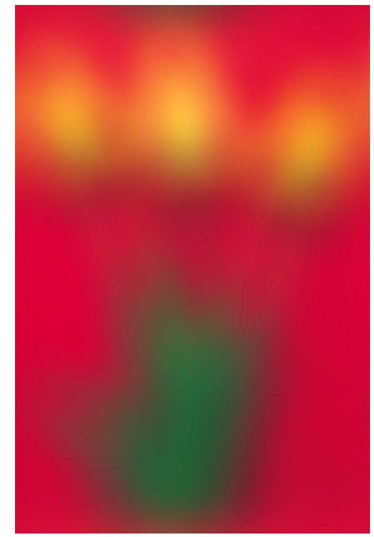
Image & Mask

Transforms



Pixel-wise Product



Inverse Transform

How to Convolve via FT in Matlab

1. Read the image from a file into a variable, say `I`.
2. Read in or create the convolution mask, `h`.
3. Compute the sum of the mask: `s = sum(sum(h));`
4. If `s == 0,` set `s = 1;`
5. Create: `H = zeros(size(I));`
6. Copy `h` into the middle of `H`.
7. Shift `H` into position: `H = ifftshift(H);`
8. Take the 2D FT of `I` and `H`: `FI=fft2(I); FH=fft2(H);`
9. Pointwise multiply the FTs: `FJ=FI.*FH;`
10. Compute the inverse FT: `J = real(ifft2(FJ));`
11. Normalize the result: `J = J/s;`

The mask is usually 1-band

For color images you may need to do each step for each band separately.

Coordinate Origin of the FFT

Center = $(\text{floor}(R/2)+1,\ \text{floor}(C/2)+1)$

Even

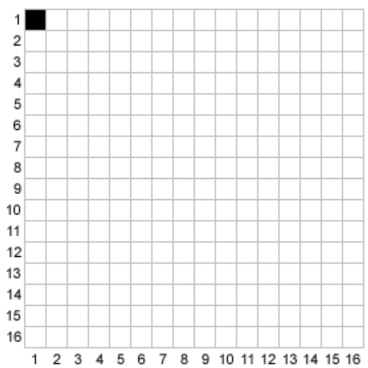
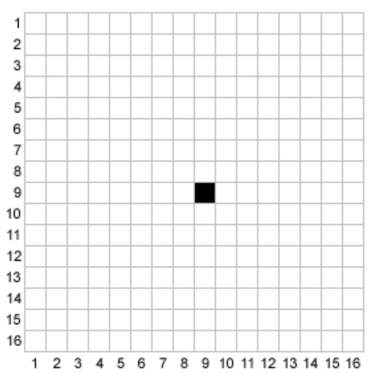


Image Origin



After FFT shift

Odd

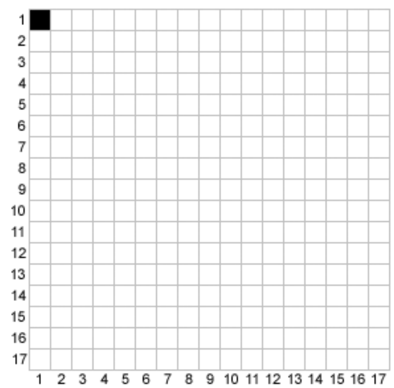
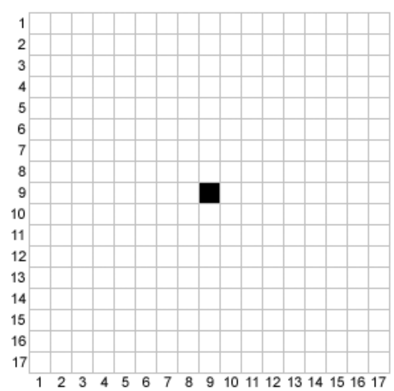
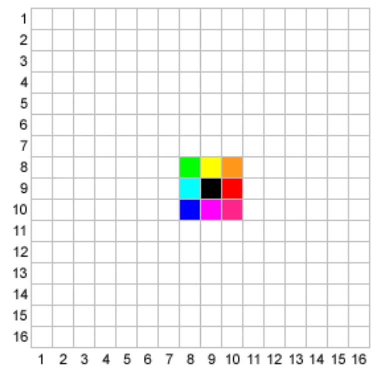


Image Origin

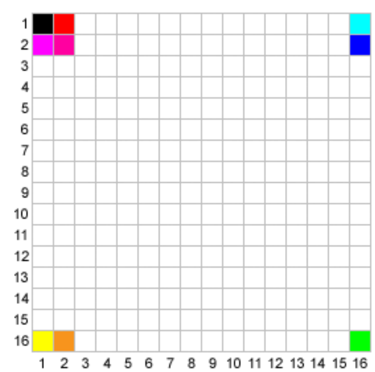


After FFT shift

Even

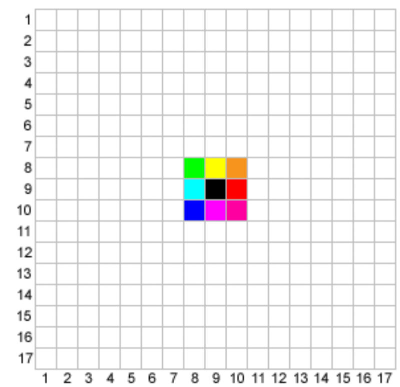


Weight Matrix Origin

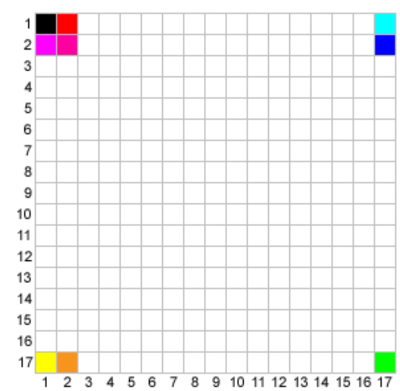


After IFFT shift

Odd



Weight Matrix Origin

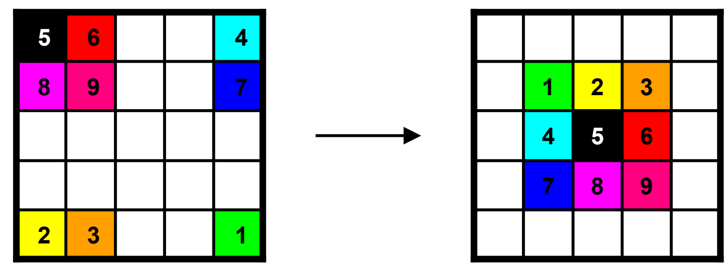


After IFFT shift

Matlab's `fftshift` and `ifftshift`

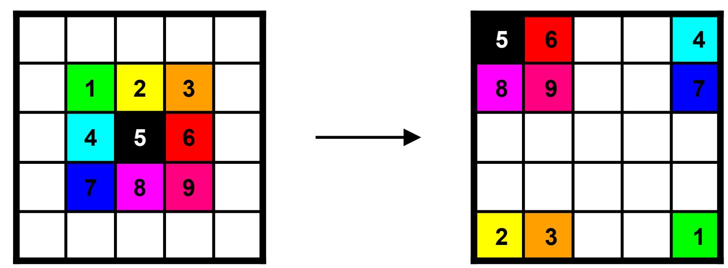
`J = fftshift(I):`

$$I(1,1) \rightarrow J(\lfloor R/2 \rfloor + 1, \lfloor C/2 \rfloor + 1)$$



`I = ifftshift(J):`

$$J(\lfloor R/2 \rfloor + 1, \lfloor C/2 \rfloor + 1) \rightarrow I(1,1)$$



where $\lfloor x \rfloor$ = `floor(x)` = the largest integer smaller than x.

Blurring: Averaging / Lowpass Filtering

Blurring results from:

- Pixel averaging in the spatial domain:
  - Each pixel in the output is a weighted average of its neighbors.
  - Is a convolution whose weight matrix sums to 1.
- Lowpass filtering in the frequency domain:
  - High frequencies are diminished or eliminated
  - Individual frequency components are multiplied by a nonincreasing function of ω such as $1/\omega = 1/\sqrt{(u^2+v^2)}$.

The values of the output image are all non-negative.

Sharpening: Differencing / Highpass Filtering

Sharpening results from adding to the image, a copy of itself that has been:

- Pixel-differenced in the spatial domain:
  - Each pixel in the output is a difference between itself and a weighted average of its neighbors.
  - Is a convolution whose weight matrix sums to 0.
- Highpass filtered in the frequency domain:
  - High frequencies are enhanced or amplified.
  - Individual frequency components are multiplied by an increasing function of ω such as $\alpha\omega = \alpha\sqrt{(u^2+v^2)}$, where α is a constant.

The values of the output image positive & negative.

Convolution Property of the Fourier Transform

Let functions $f(r,c)$ and $g(r,c)$ have Fourier Transforms $F(u,v)$ and $G(u,v)$.

Then,

$$\mathfrak{F}\{f * g\} = F \cdot G.$$

Moreover,

$$\mathfrak{F}\{f \cdot g\} = F * G.$$

Thus we can compute $f * g$ by

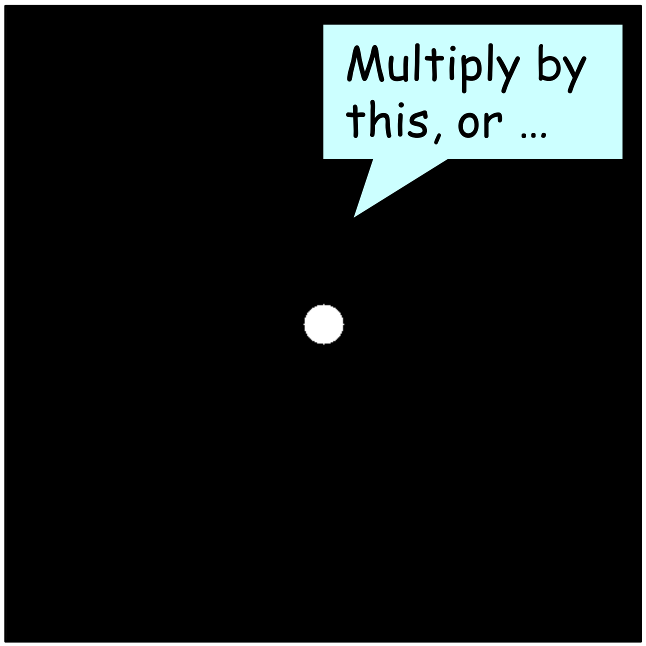
$$f * g = \mathfrak{F}^{-1}\{F \cdot G\}.$$

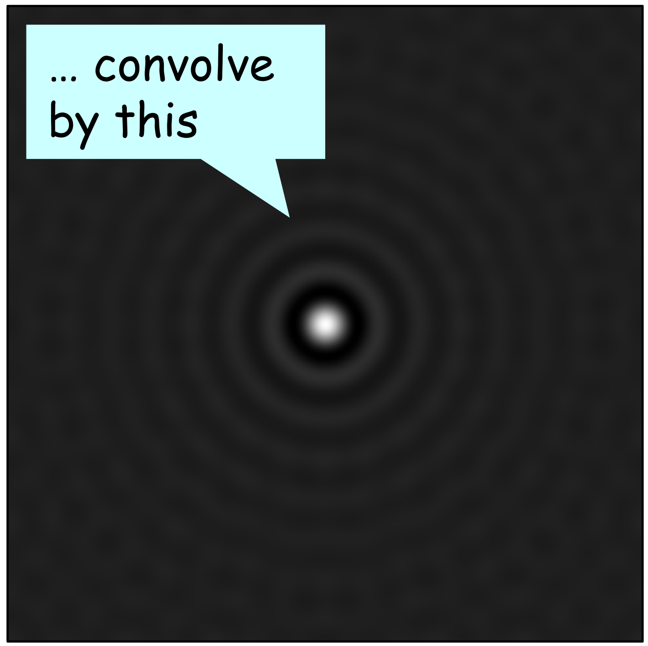
$*$ = convolution
$\cdot$ = multiplication

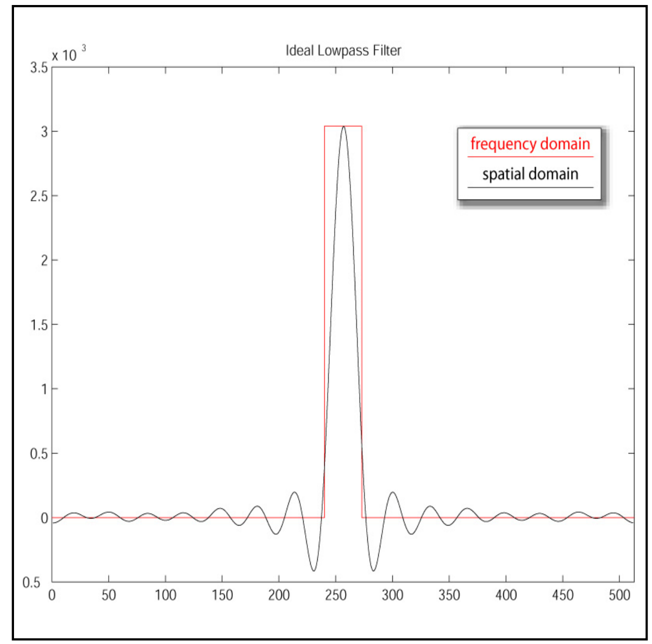
The Fourier Transform of a convolution equals the product of the Fourier Transforms. Similarly, the Fourier Transform of a convolution is the product of the Fourier Transforms

Ideal Lowpass Filter

Image size: 512x512
FD filter radius: 16







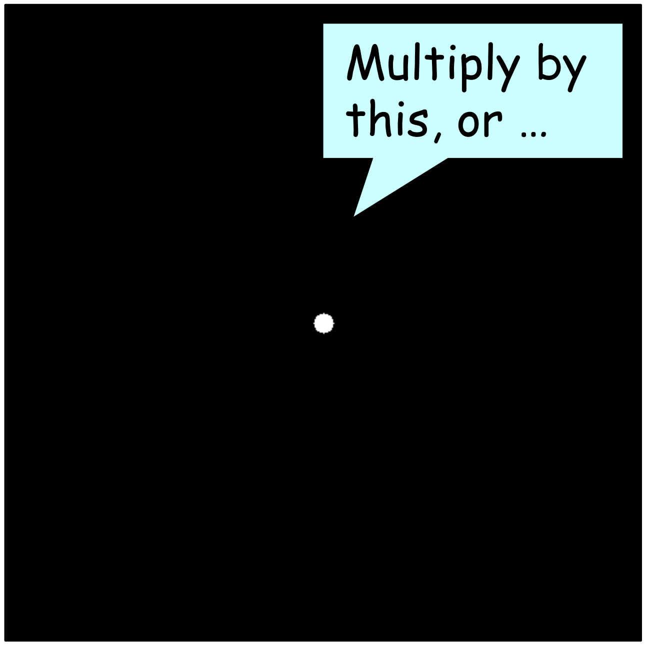
Fourier Domain Rep.

Spatial Representation

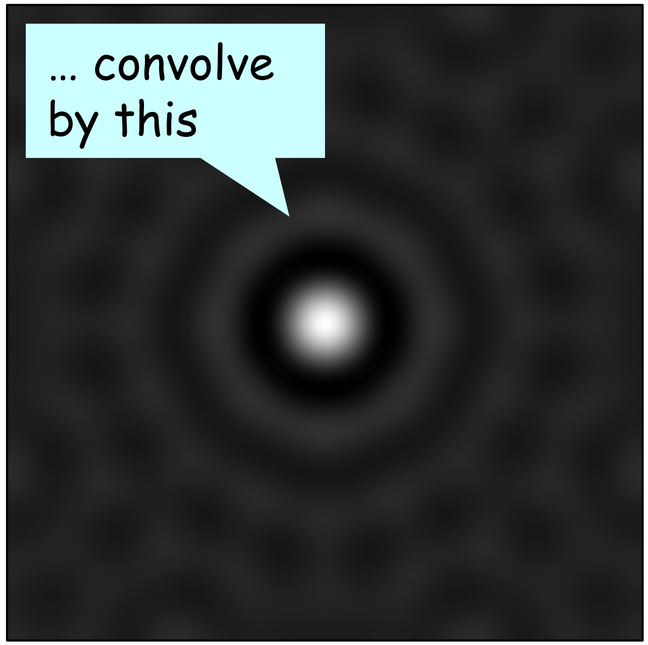
Central Profile

Ideal Lowpass Filter

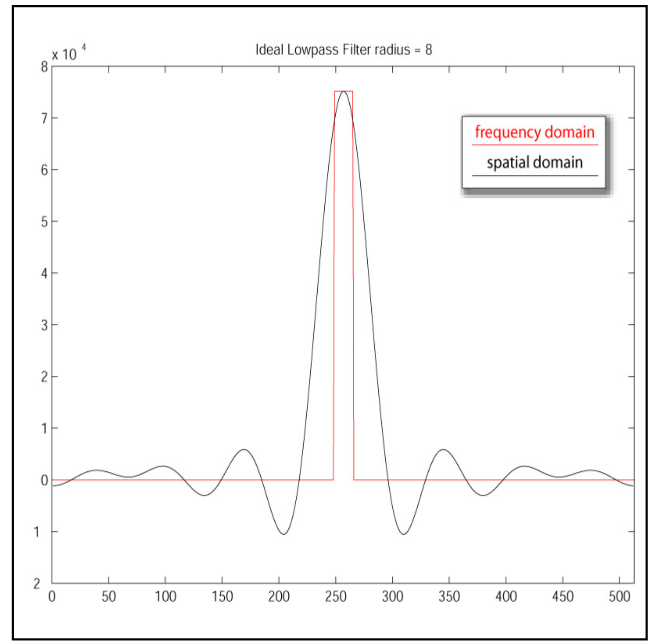
Image size: 512x512
FD filter radius: 8



Fourier Domain Rep.



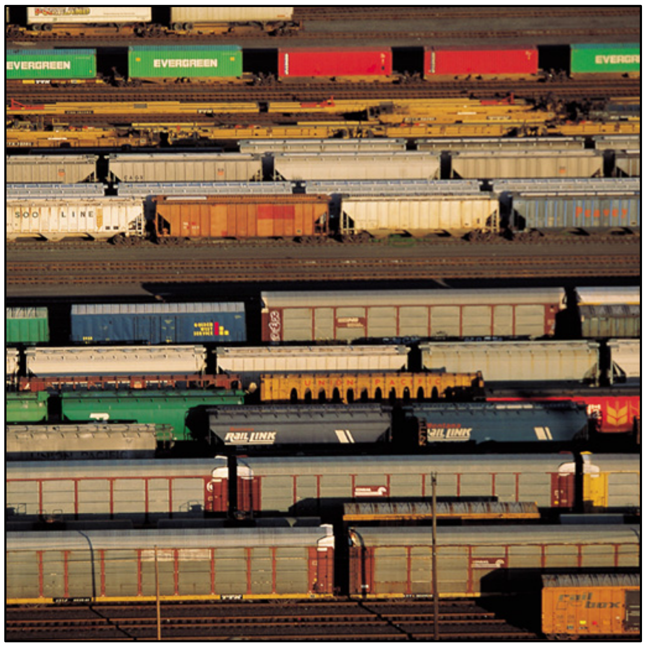
Spatial Representation



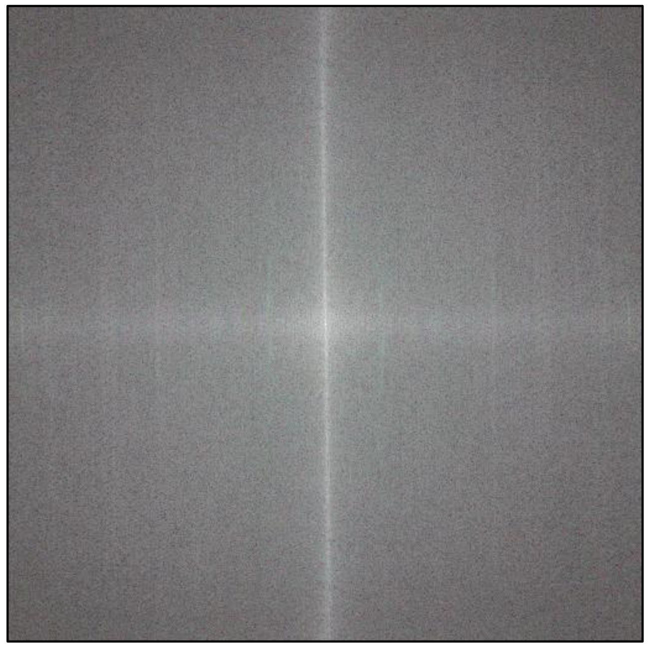
Central Profile

Consider the image below:

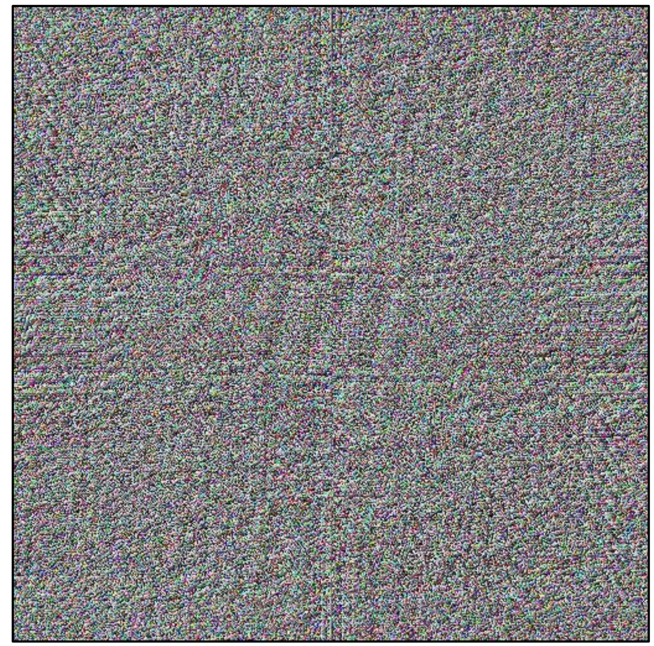
Power Spectrum and Phase of an Image



Original Image



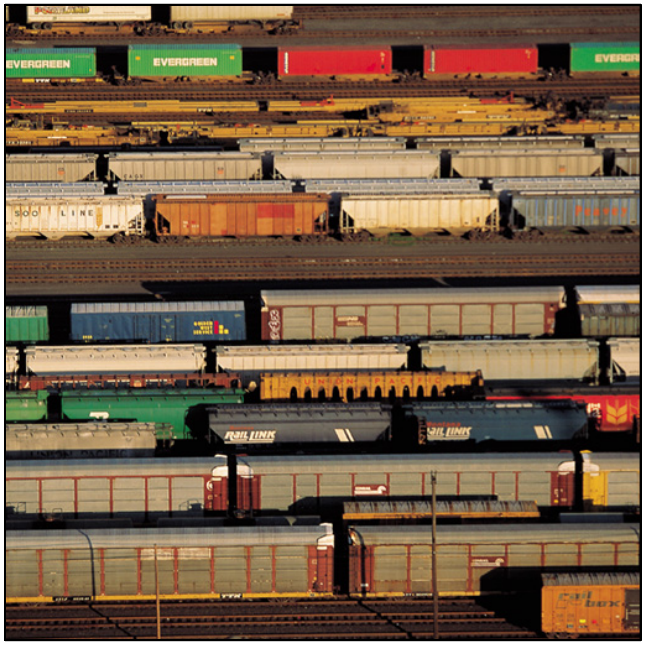
Power Spectrum



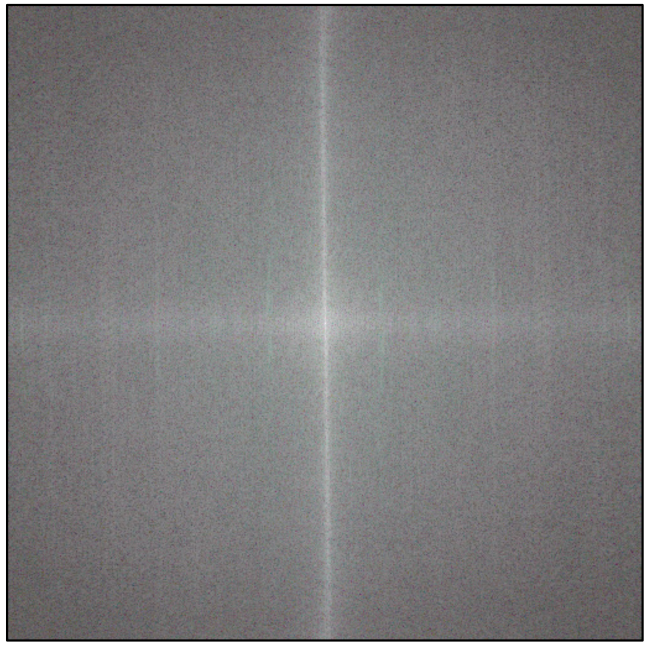
Phase

Ideal Lowpass Filter

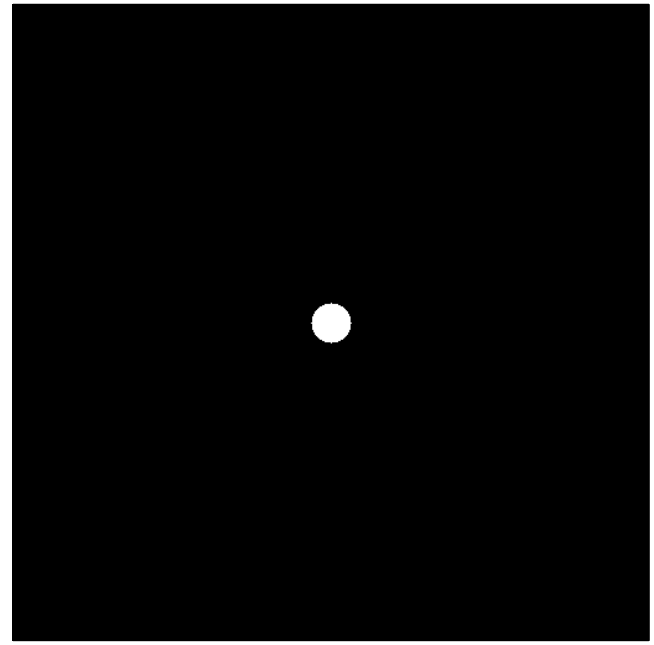
Image size: 512x512
FD filter radius: 16



Original Image



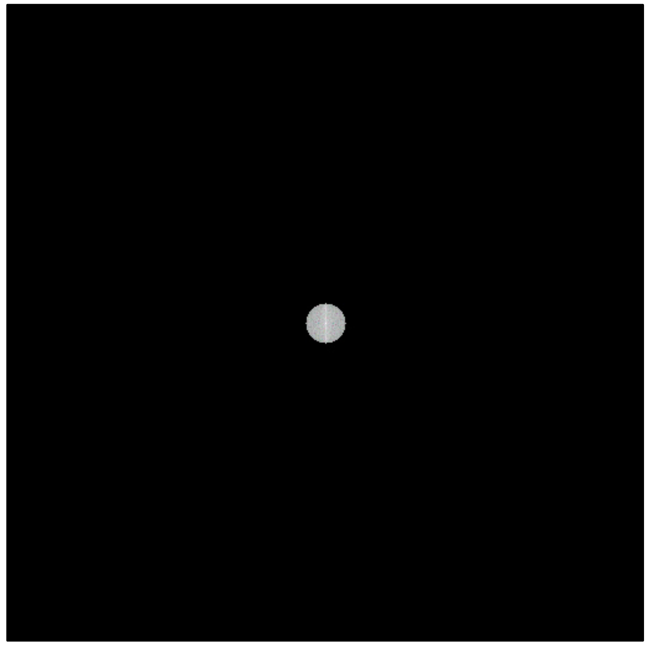
Power Spectrum

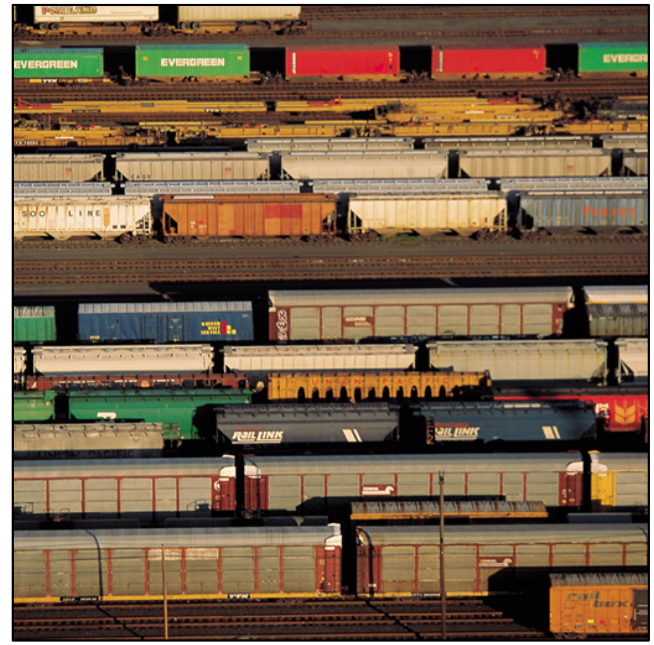


Ideal LPF in FD

Ideal Lowpass Filter

Image size: 512x512
FD filter radius: 16





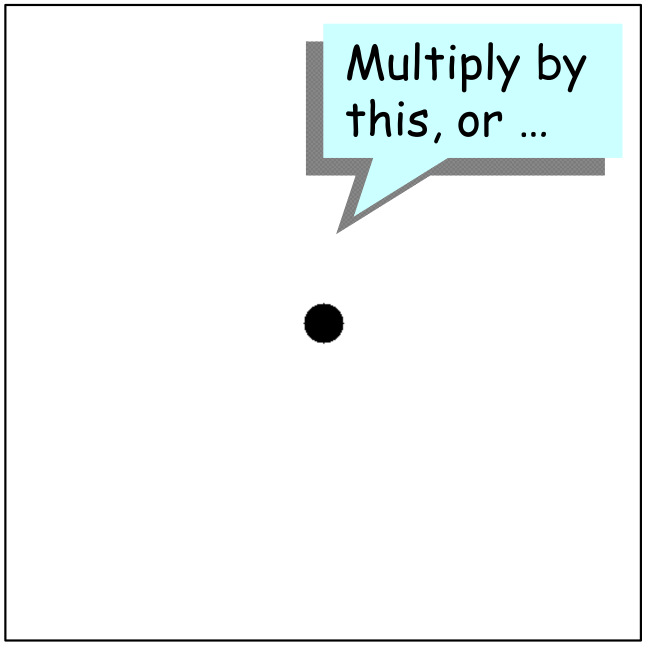
Filtered Image

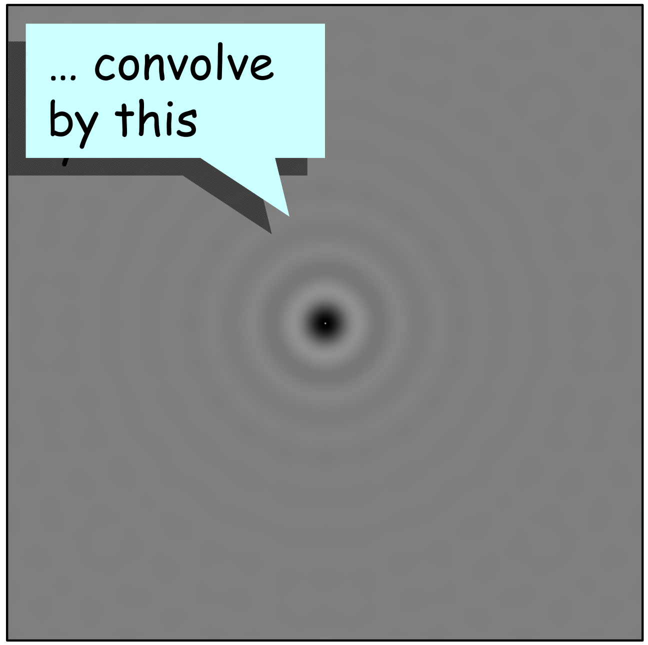
Filtered Power Spectrum

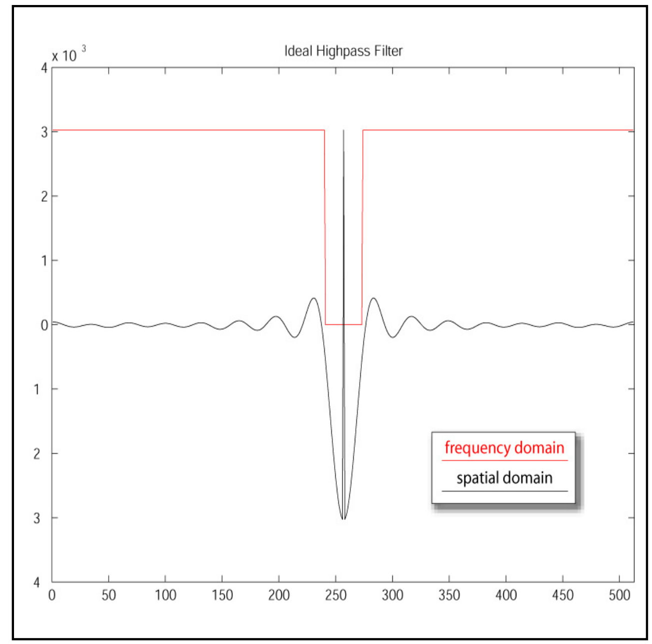
Original Image

Ideal Highpass Filter

Image size: 512x512
FD notch radius: 16







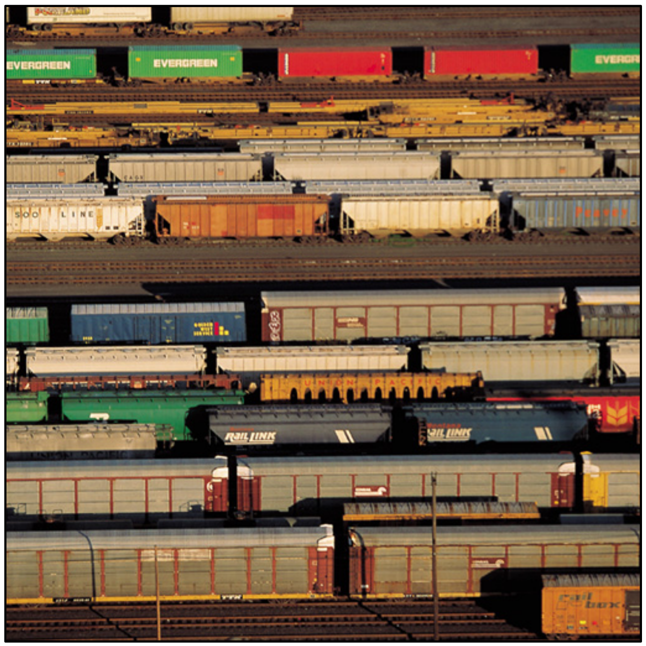
Fourier Domain Rep.

Spatial Representation

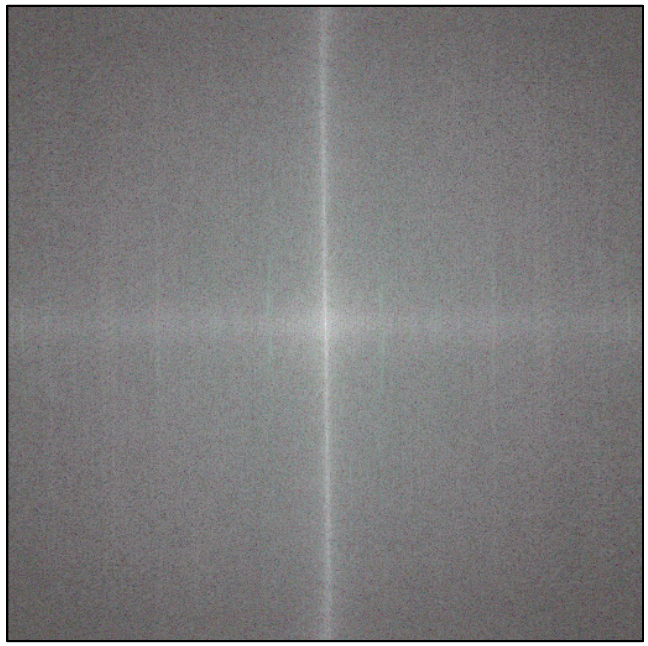
Central Profile

Ideal Highpass Filter

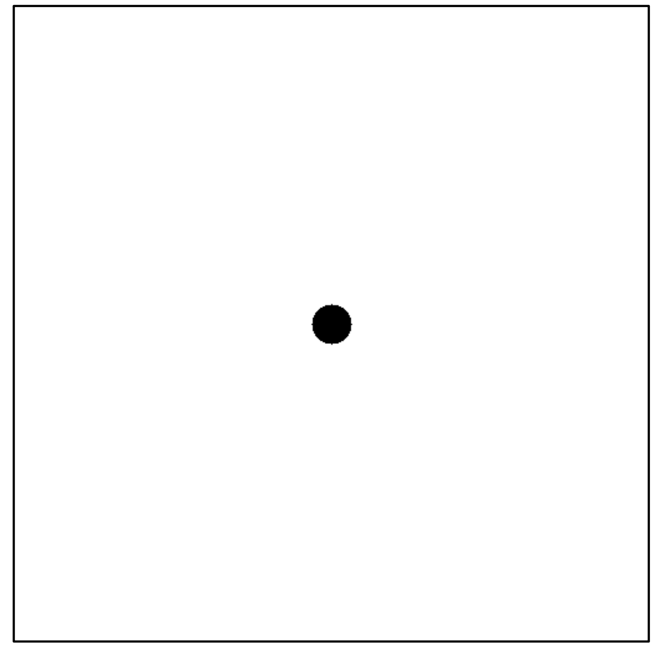
Image size: 512x512
FD notch radius: 16



Original Image



Power Spectrum

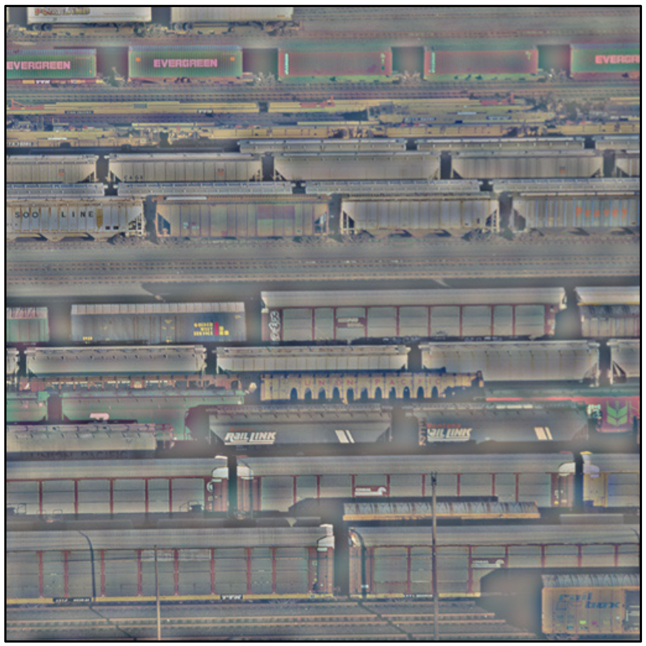


Ideal HPF in FD

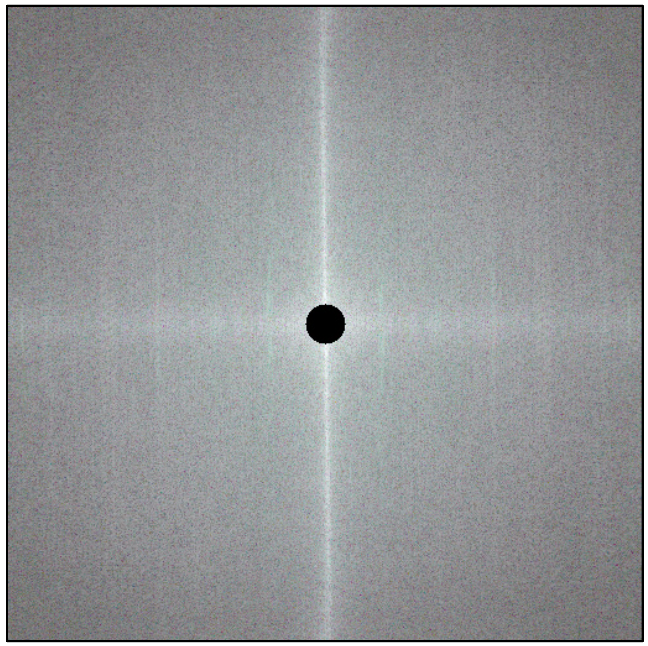
*signed image; 0 mapped to 128

Ideal Highpass Filter

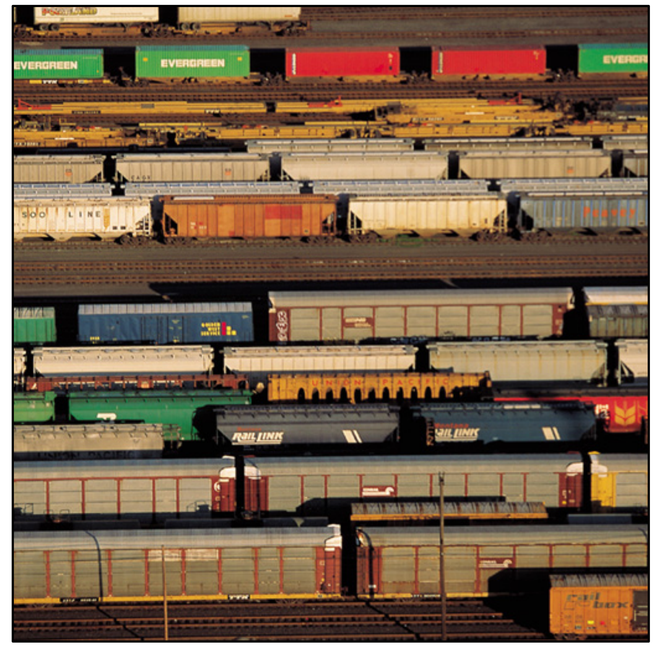
Image size: 512x512
FD notch radius: 16



Filtered Image*



Filtered Power Spectrum

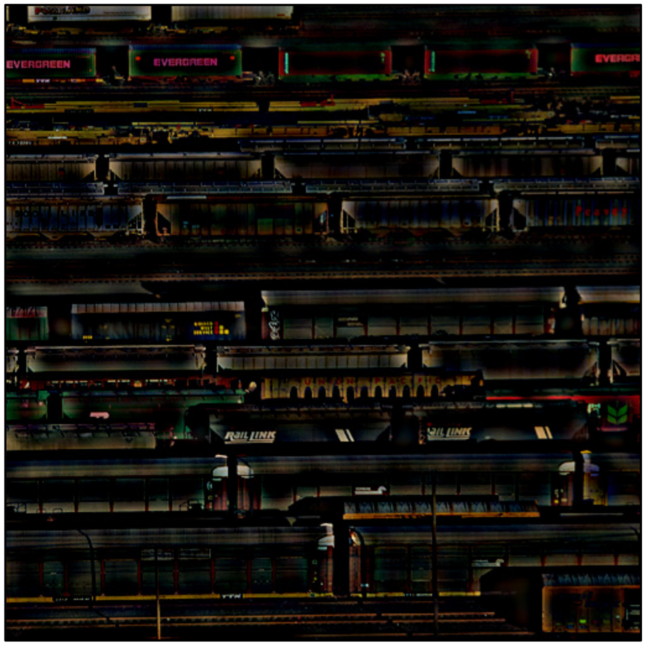


Original Image

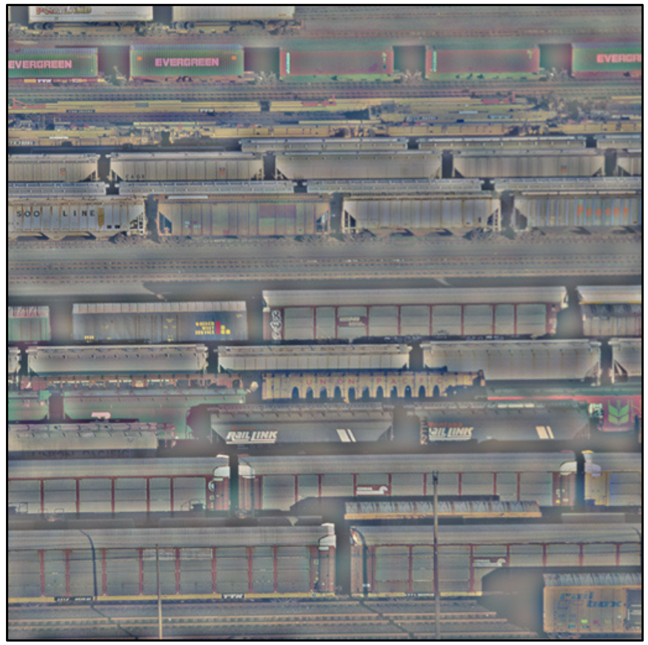
*signed image; 0 mapped to 128

Ideal Highpass Filter

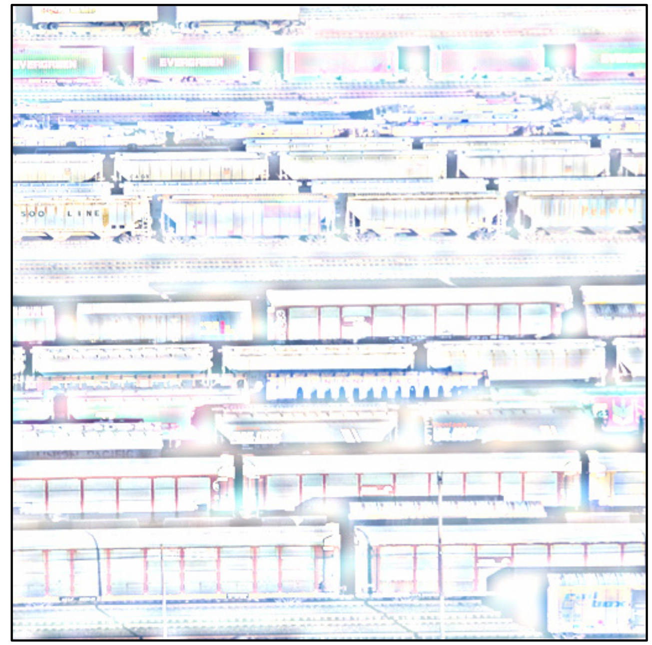
Image size: 512x512
FD notch radius: 16



Positive Pixels

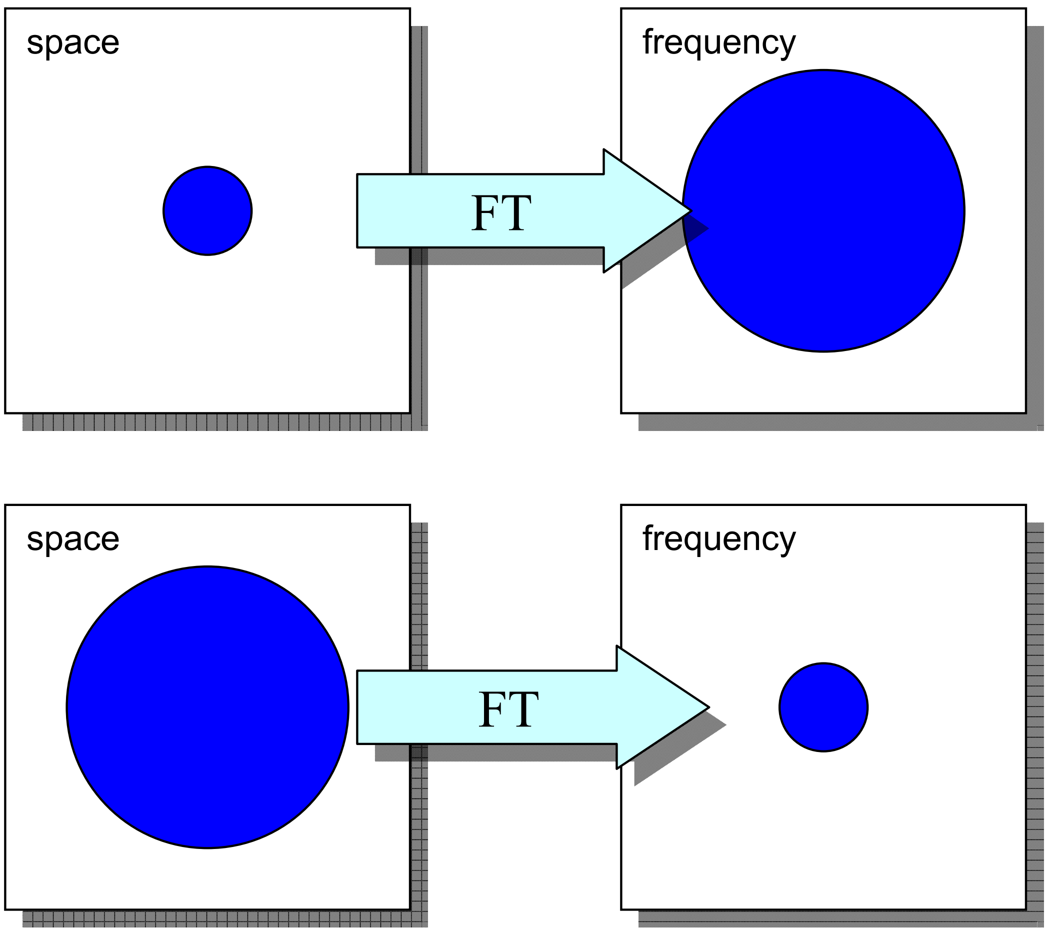


Filtered Image*



Negative Pixels

The Uncertainty Relation

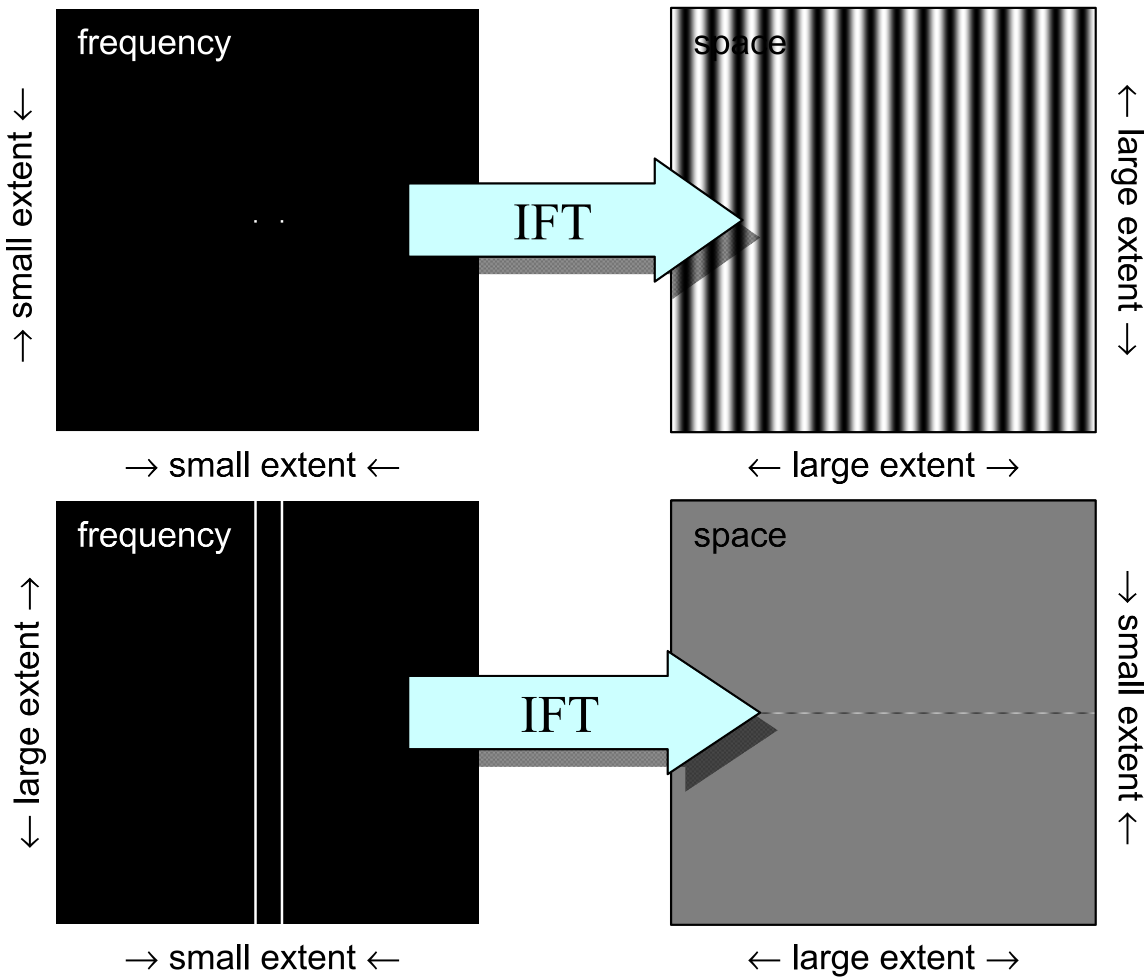


If $\Delta x\,\Delta y$ is the extent of the object in space and if $\Delta u\,\Delta v$ is its extent in frequency then,

$$\Delta x\,\Delta y \cdot \Delta u\,\Delta v \geq \frac{1}{16\pi^2}$$

A small object in space has a large frequency extent and vice-versa.

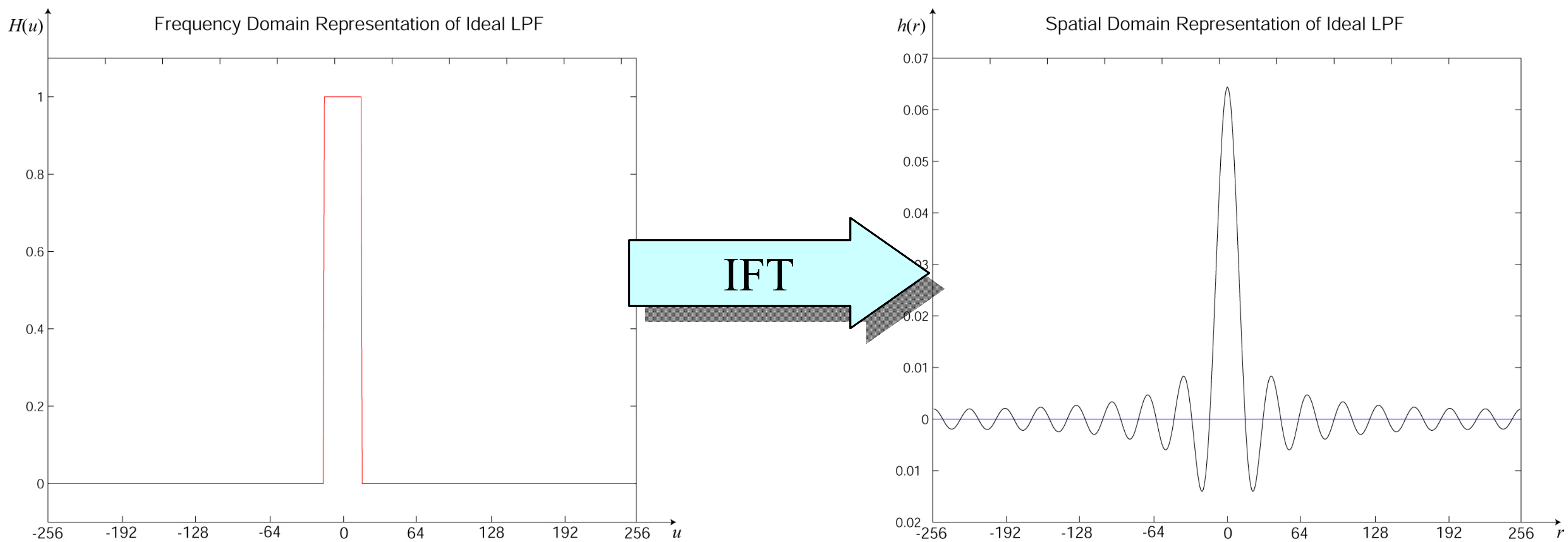
The Uncertainty Relation



Recall: a symmetric pair of impulses in the frequency domain becomes a sinusoid in the spatial domain.

A symmetric pair of lines in the frequency domain becomes a sinusoidal line in the spatial domain.

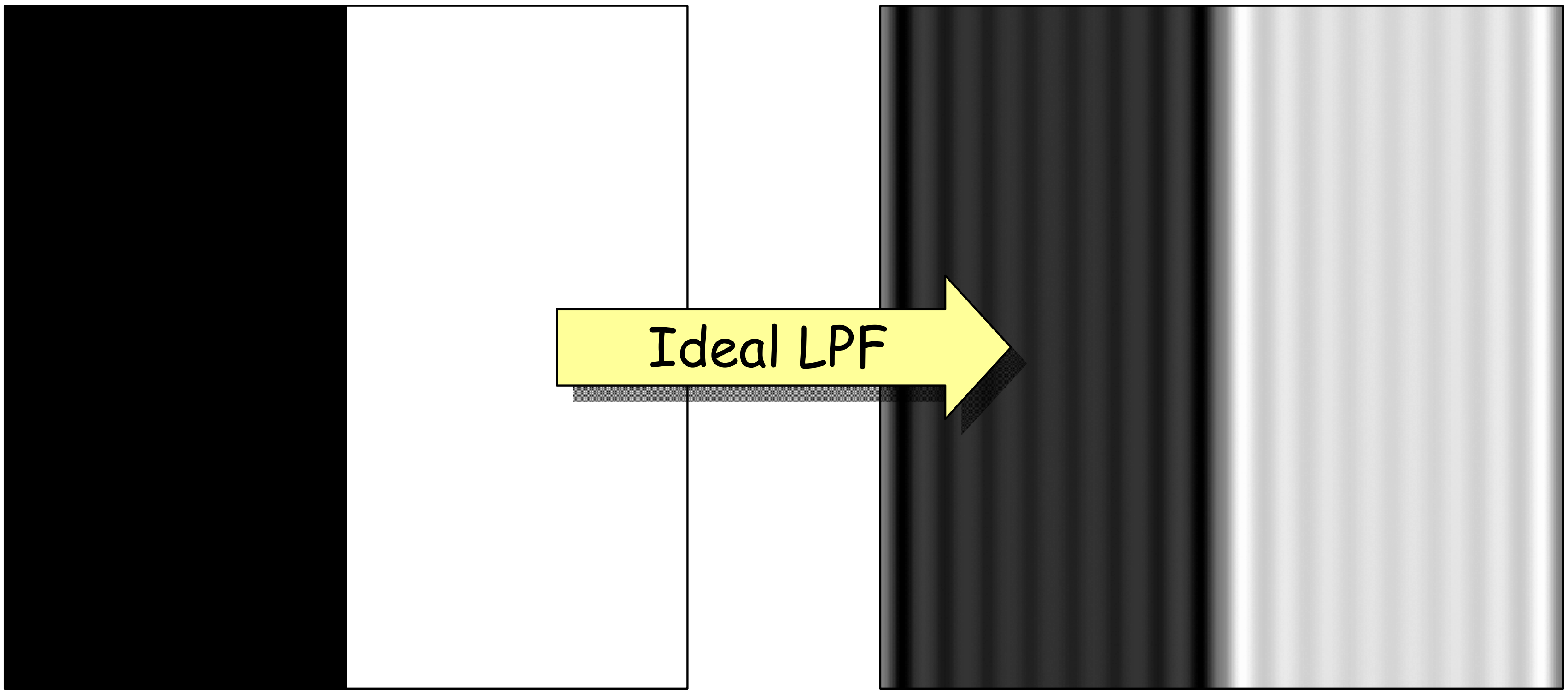
Ideal Filters Do Not Produce Ideal Results



A sharp cutoff in the frequency domain…

…causes ringing in the spatial domain.

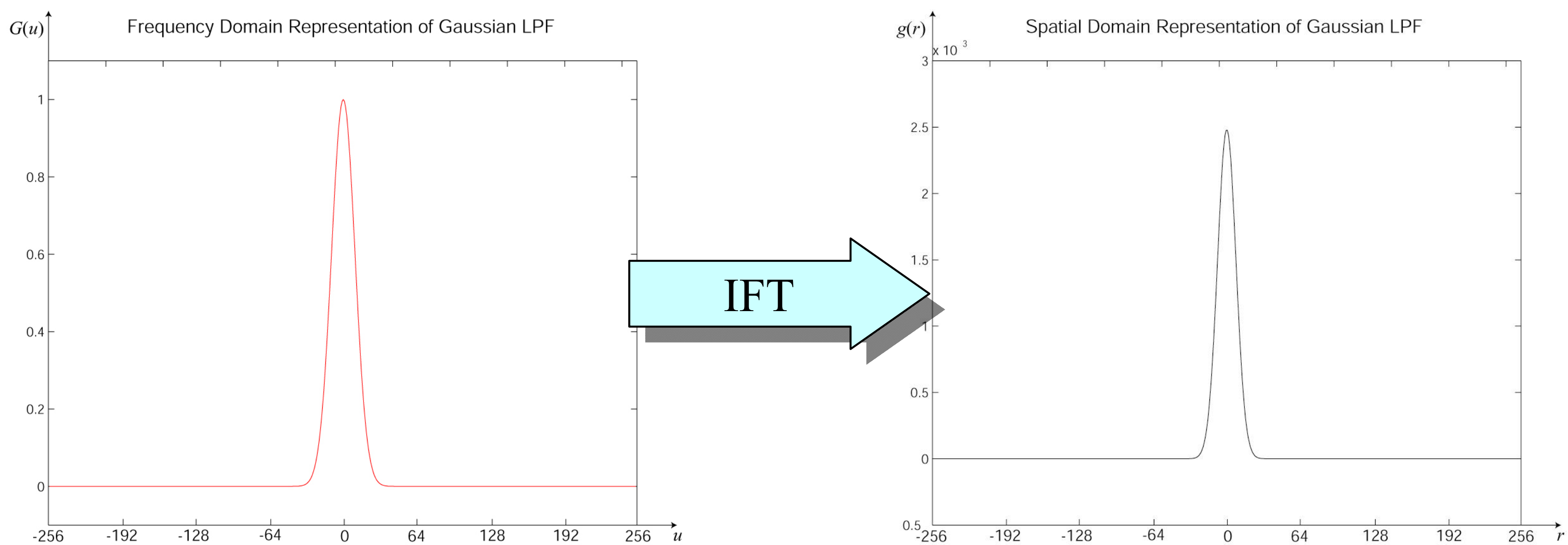
Ideal Filters Do Not Produce Ideal Results



Blurring the image above w/ an ideal lowpass filter…

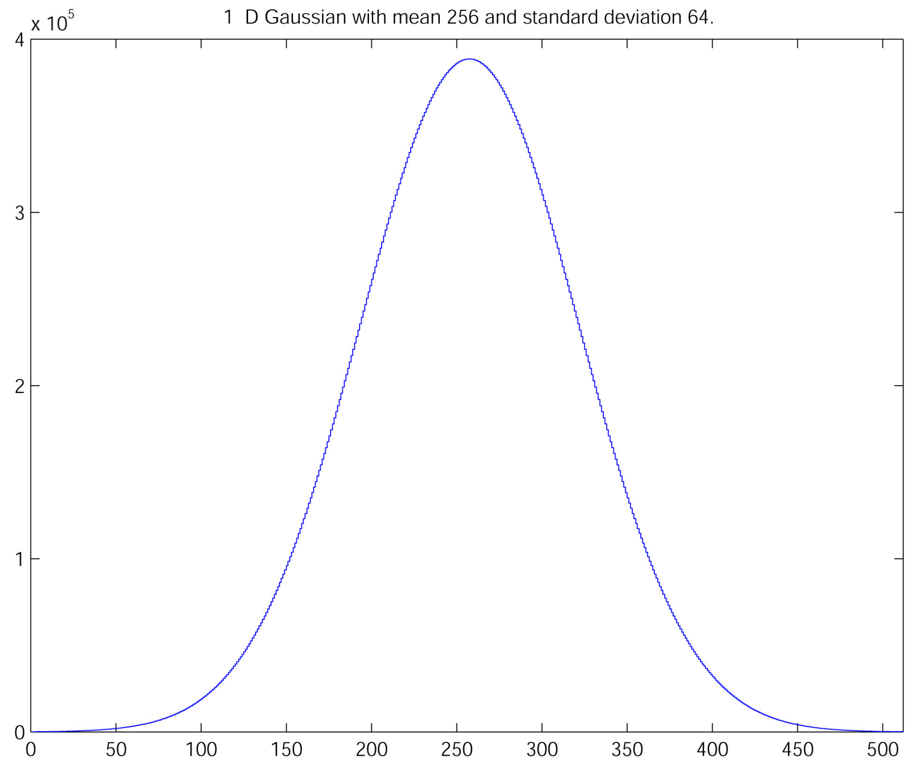
…distorts the results with ringing or ghosting.

Optimal Filter: The Gaussian



The Gaussian filter optimizes the uncertainty relation. It provides the sharpest cutoff with the least ringing.

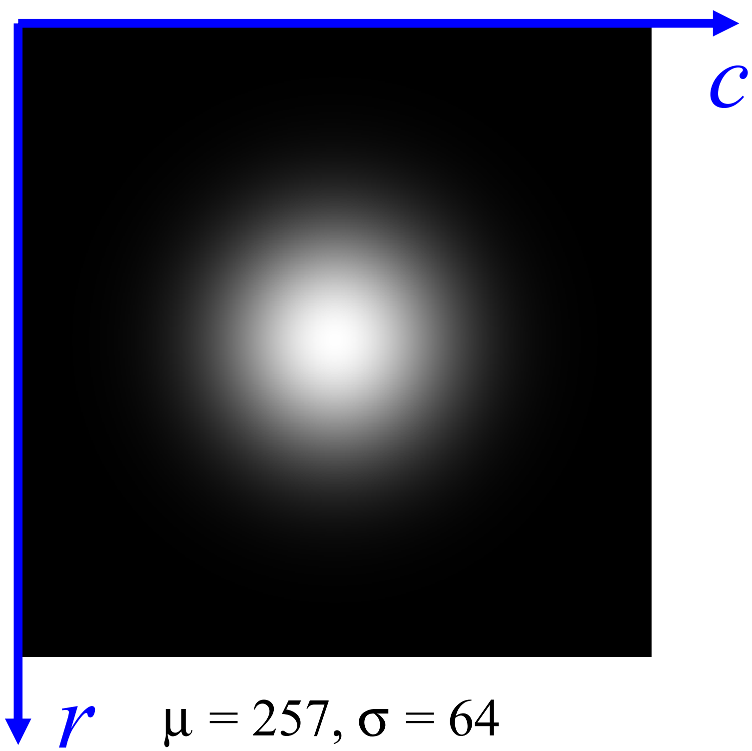
One-Dimensional Gaussian



$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

Two-Dimensional Gaussian

$R = 512,\ C = 512$



$\mu = 257,\ \sigma = 64$

If μ and σ are different for r & c…

$$g(r,c) = g(r)g(c)$$

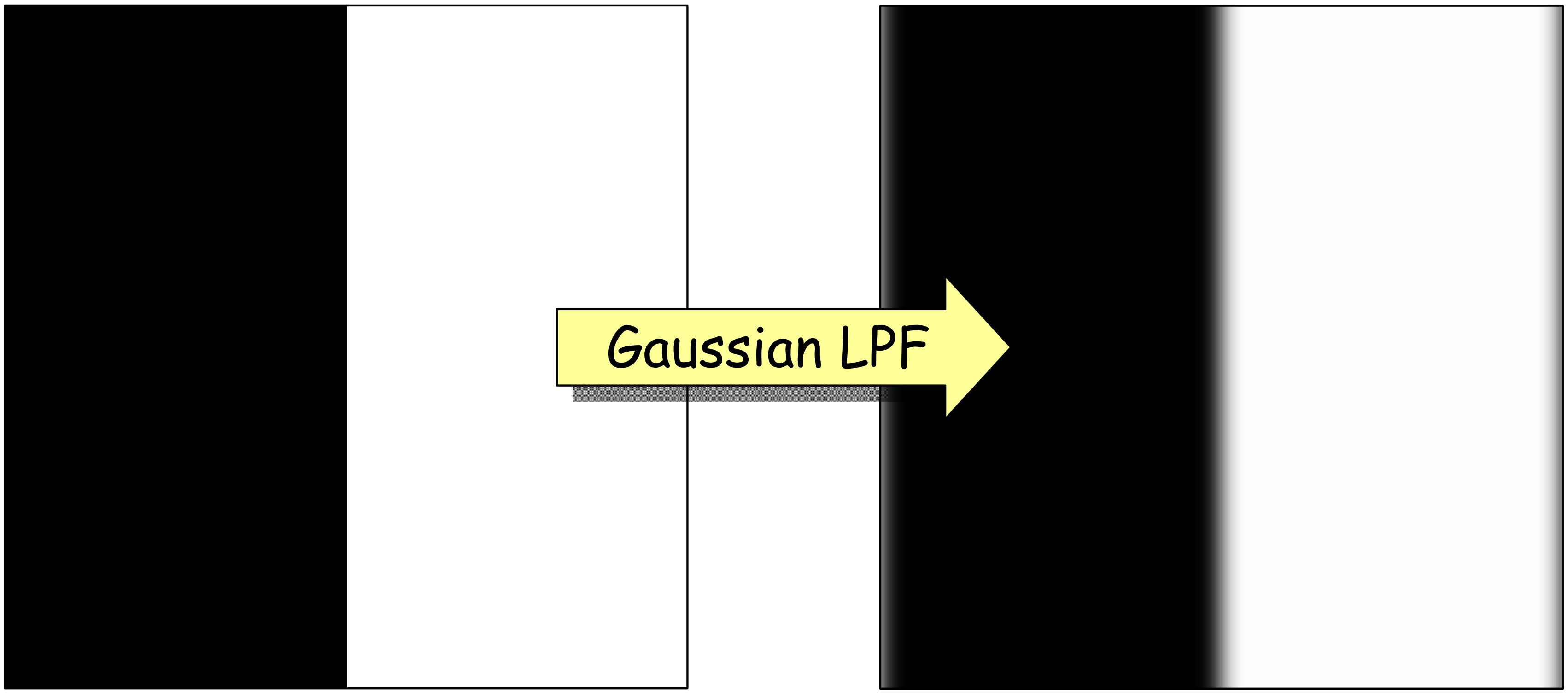
$$= \frac{1}{\sigma_r \sigma_c 2\pi} e^{-\frac{(r-\mu_r)^2}{2\sigma_r^2} - \frac{(y-\mu_c)^2}{2\sigma_c^2}}$$

$$= \frac{1}{\sigma_r \sigma_c 2\pi} e^{-\frac{\sigma_c^2 (x-\mu_r)^2 + \sigma_r^2 (y-\mu_c)^2}{2\sigma_r^2 \sigma_c^2}}$$

…or if μ and σ are the same for r & c.

$$g(r,c) = \frac{1}{\sigma^2 2\pi} e^{-\frac{(r-\mu)^2 + (c-\mu)^2}{2\sigma^2}}$$

Optimal Filter: The Gaussian

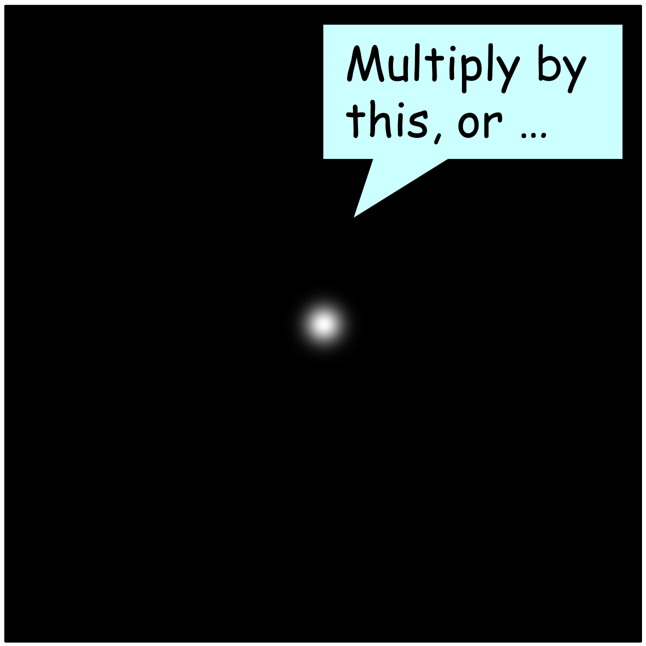


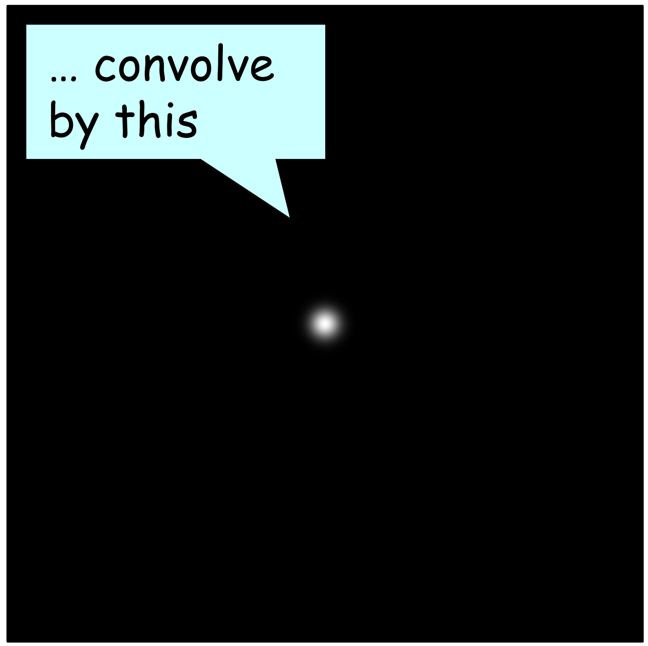
With a gaussian lowpass filter, the image above …

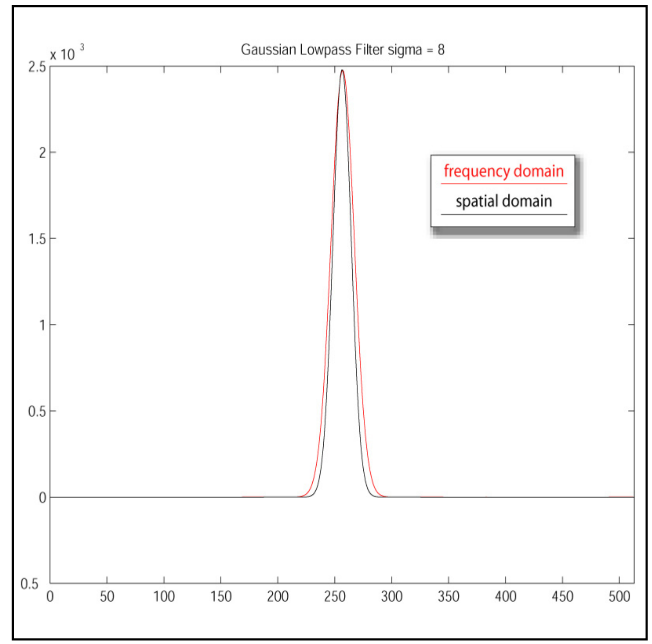
… is blurred without ringing or ghosting.

Gaussian Lowpass Filter

Image size: 512x512
SD filter sigma = 8







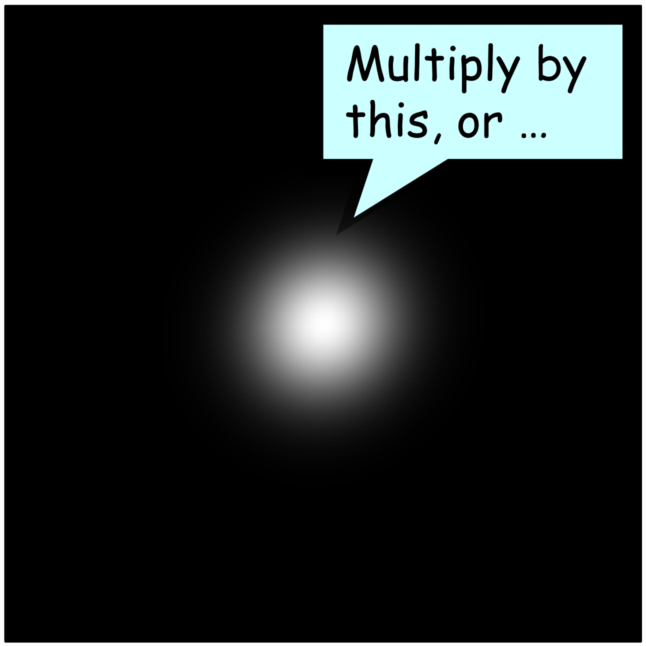
Fourier Domain Rep.

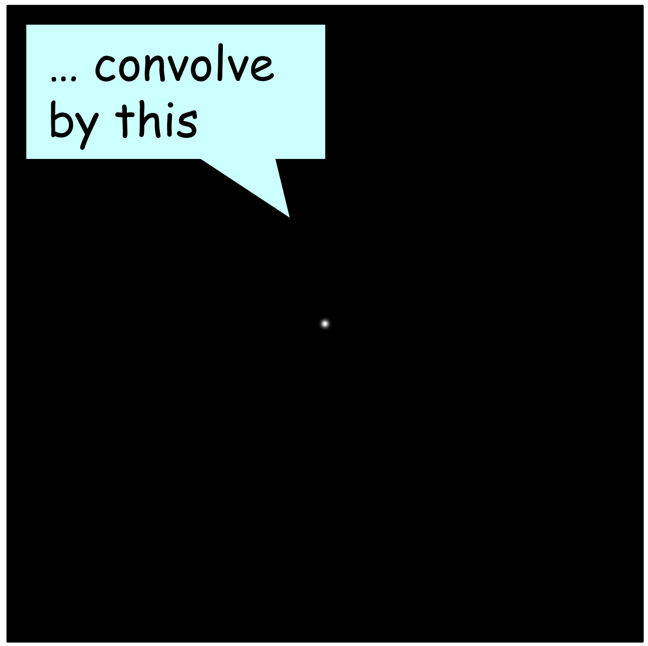
Spatial Representation

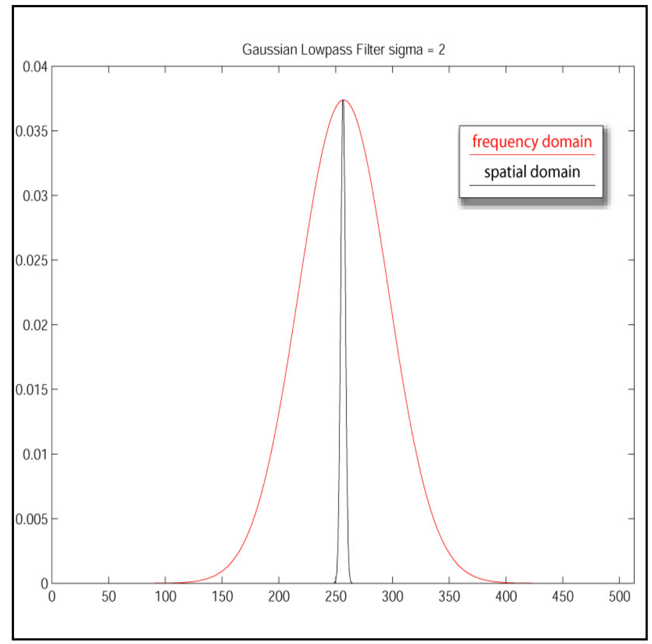
Central Profile

Gaussian Lowpass Filter

Image size: 512x512
SD filter sigma = 2







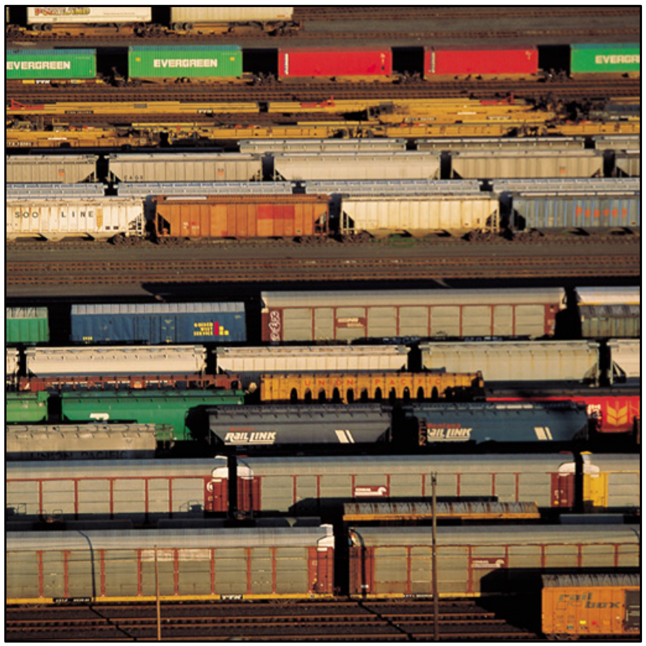
Fourier Domain Rep.

Spatial Representation

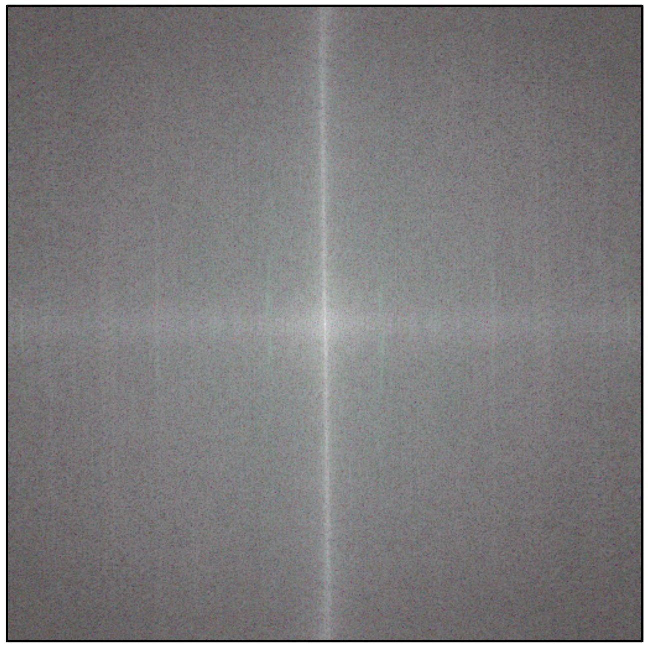
Central Profile

Gaussian Lowpass Filter

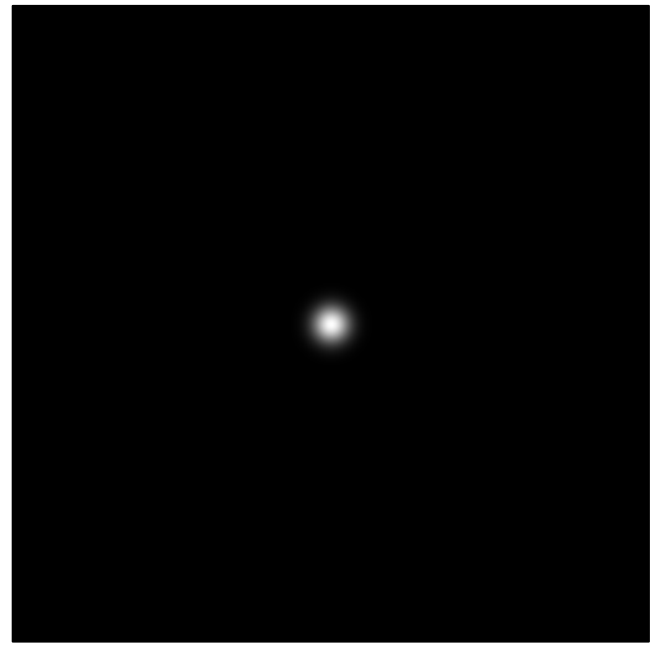
Image size: 512x512
SD filter sigma = 8



Original Image



Power Spectrum

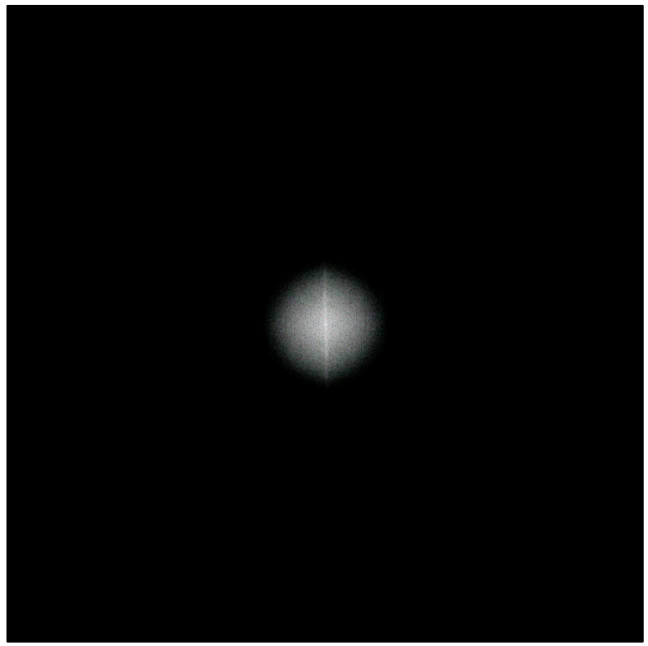


Gaussian LPF in FD

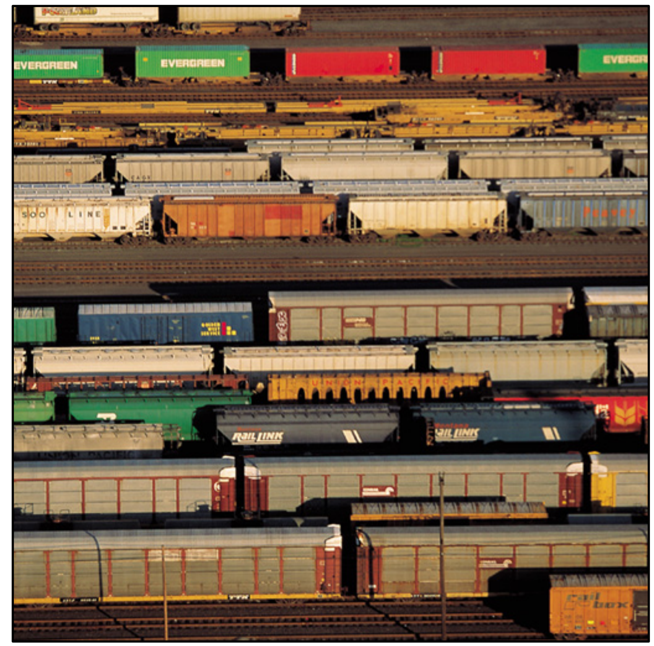
Gaussian Lowpass Filter

Image size: 512x512
SD filter sigma = 8

Filtered Image



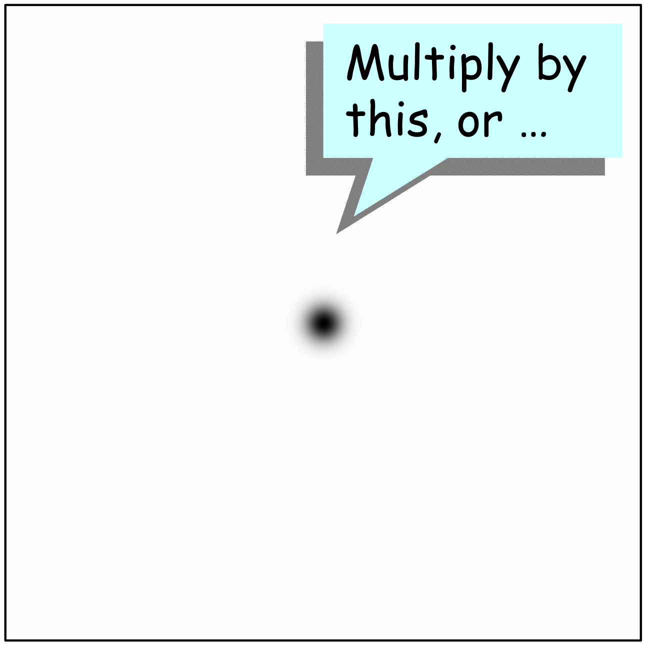
Filtered Power Spectrum

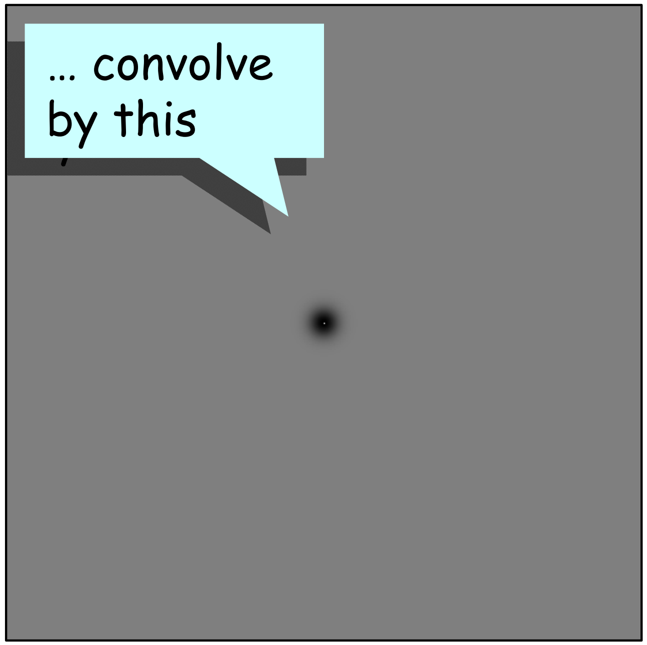


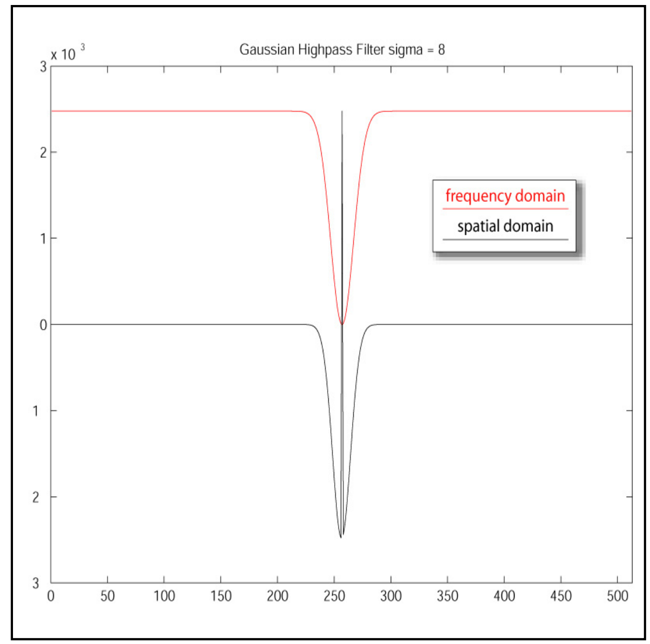
Original Image

Gaussian Highpass Filter

Image size: 512x512
FD notch sigma = 8







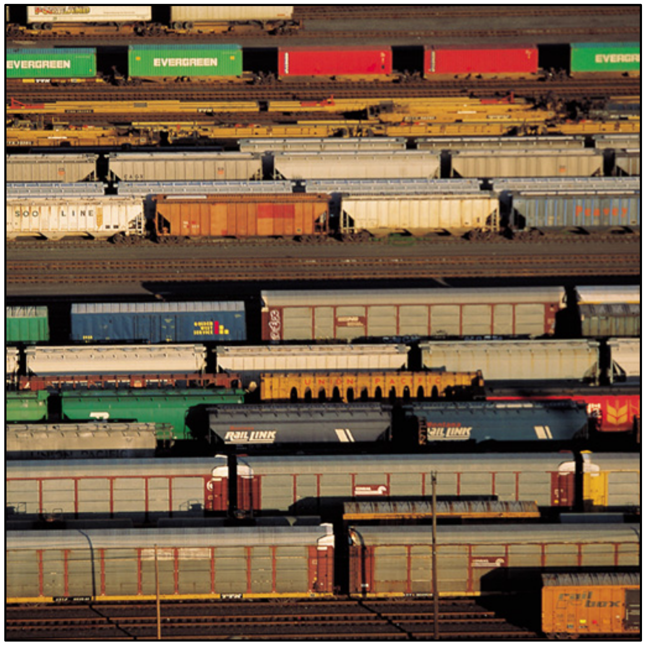
Fourier Domain Rep.

Spatial Representation

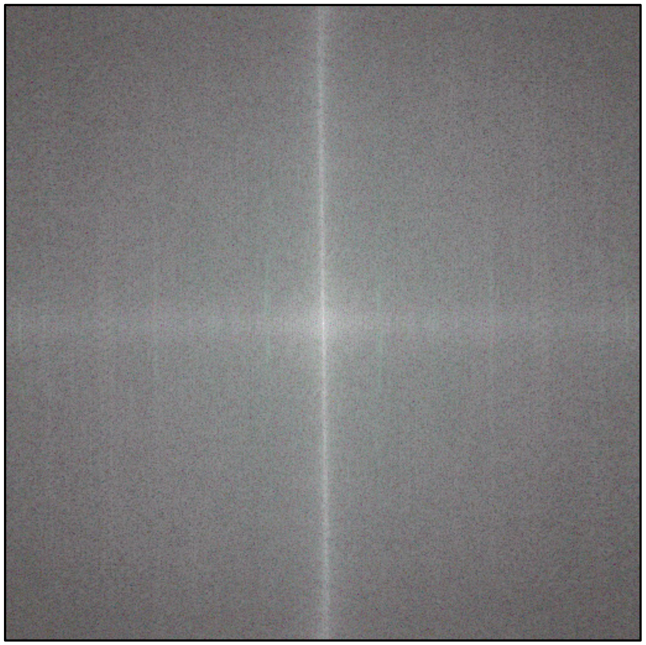
Central Profile

Gaussian Highpass Filter

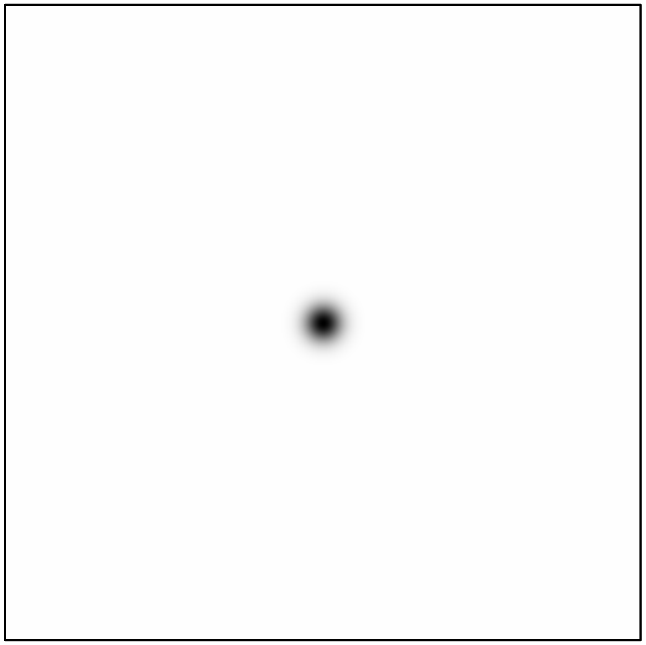
Image size: 512x512
FD notch sigma = 8



Original Image



Power Spectrum

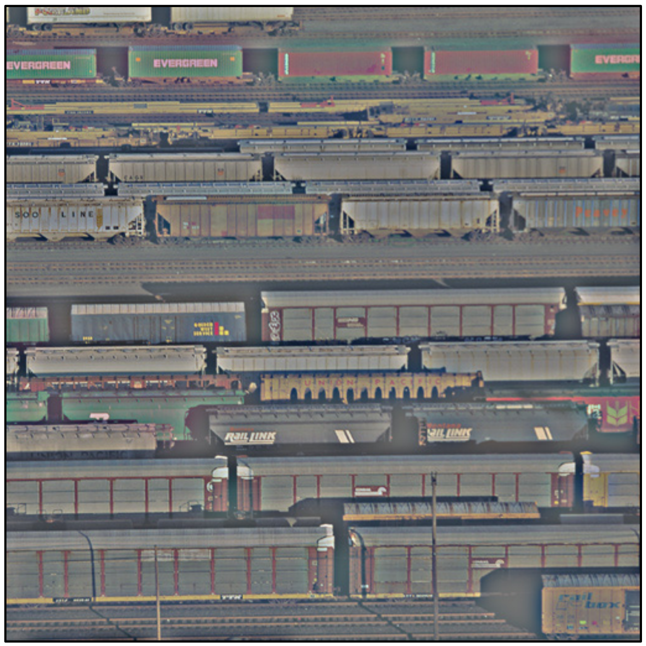


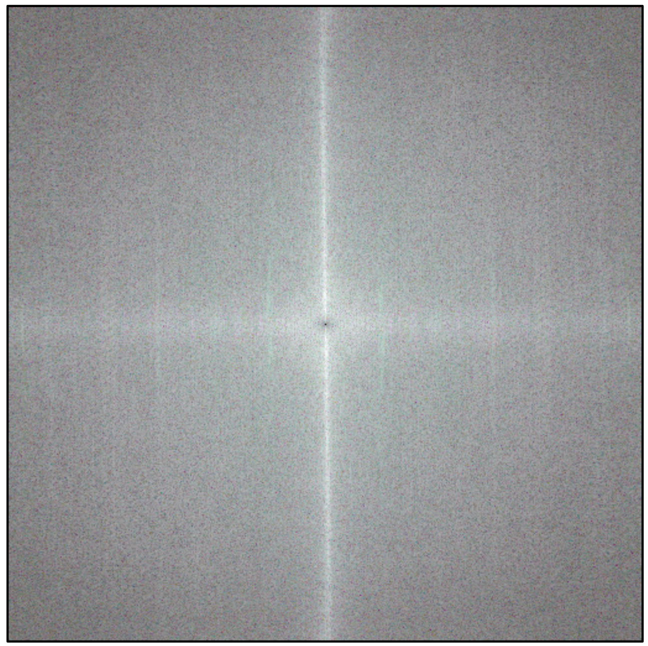
Gaussian HPF in FD

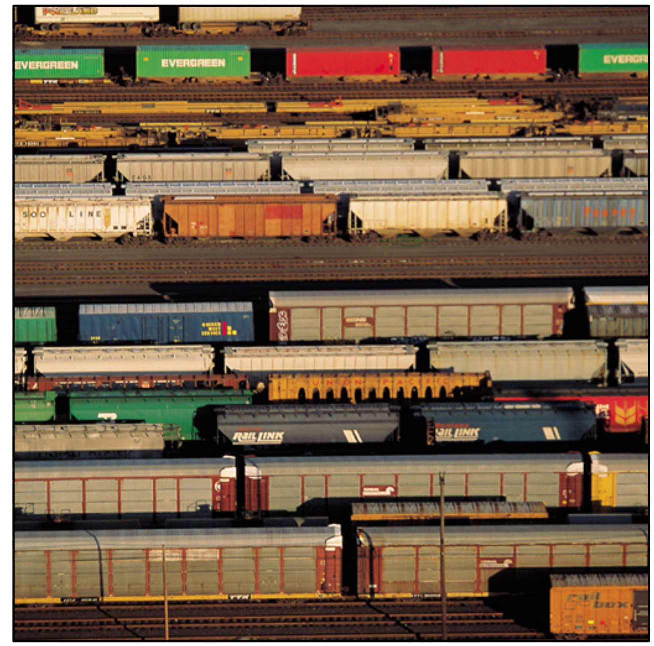
*signed image; 0 mapped to 128

Gaussian Highpass Filter

Image size: 512x512
FD notch sigma = 8







Filtered Image*

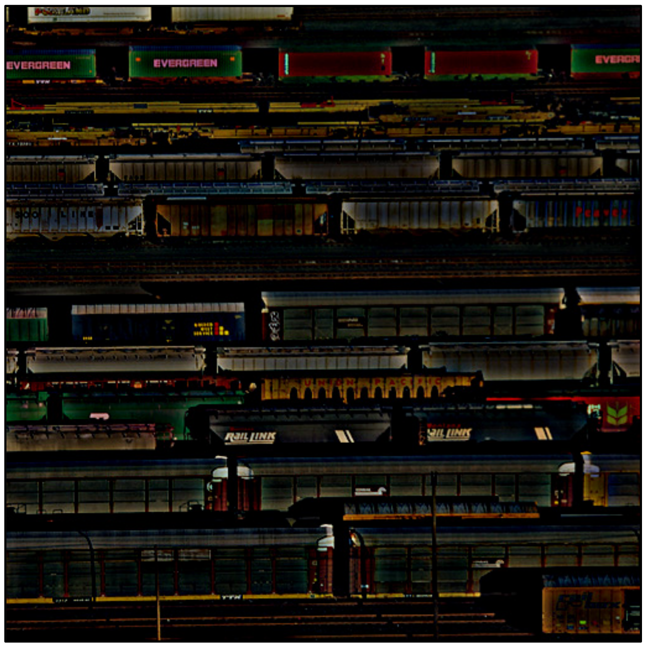
Filtered Power Spectrum

Original Image

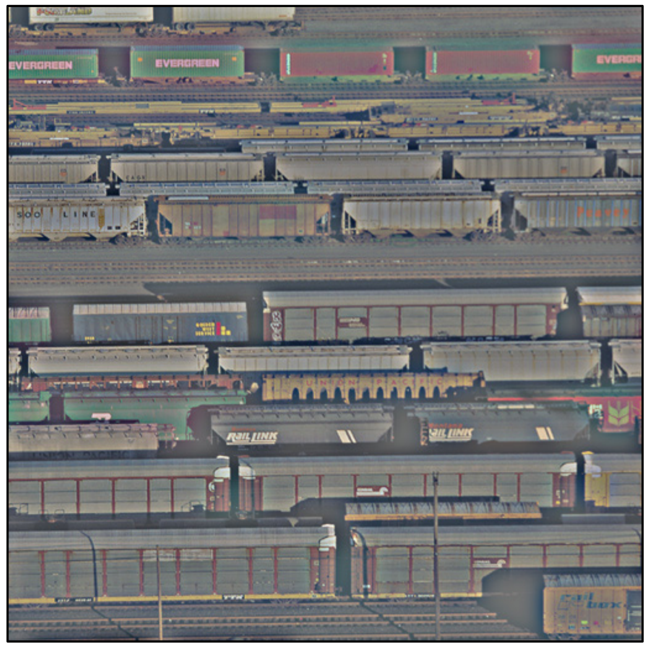
*signed image; 0 mapped to 128

Gaussian Highpass Filter

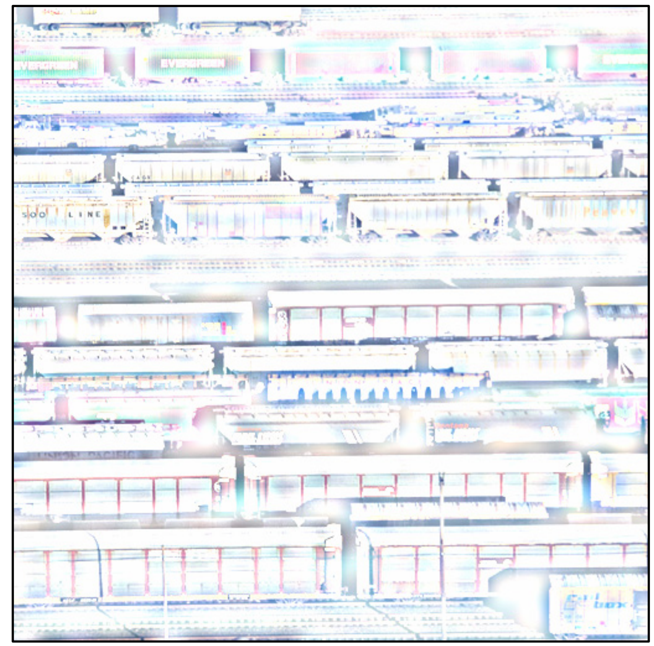
Image size: 512x512
FD notch sigma = 8



Positive Pixels



Filtered Image*

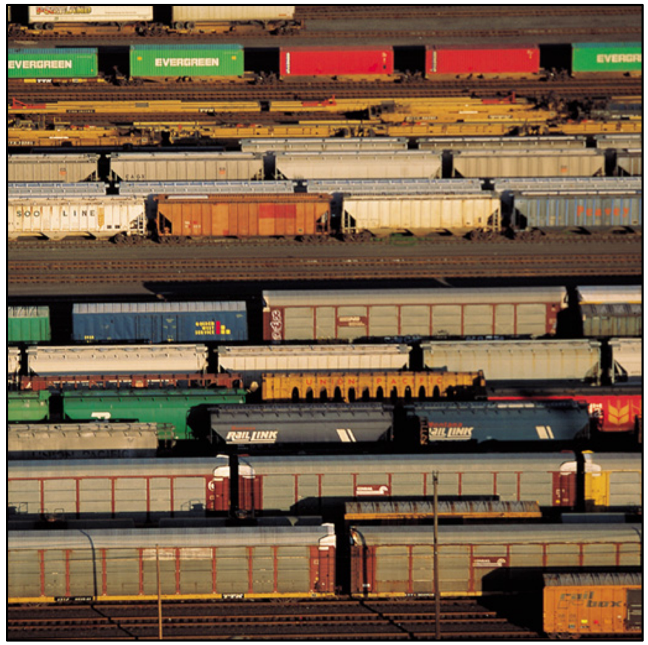


Negative Pixels

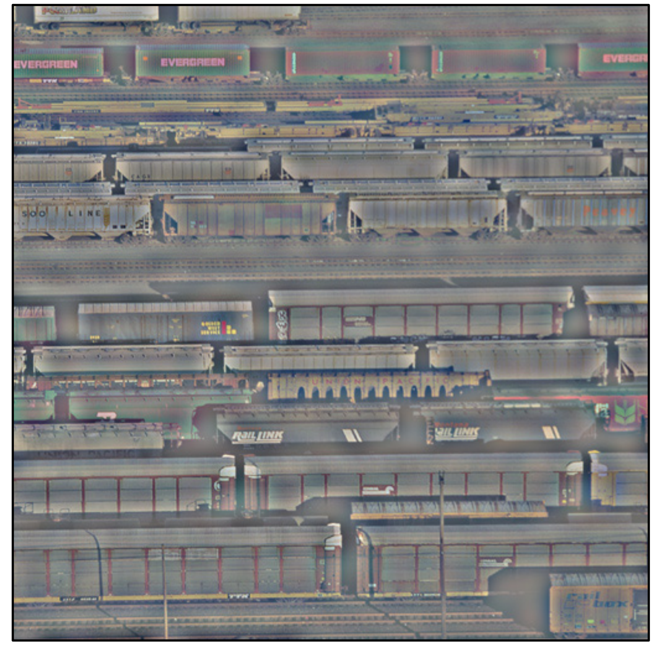
*signed image; 0 mapped to 128

Comparison of Ideal and Gaussian Filters

Ideal LPF



Original Image

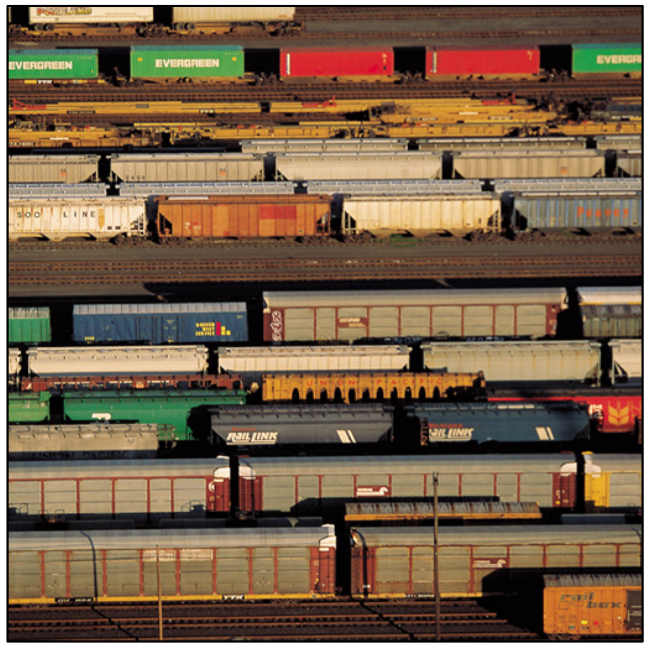


Ideal HPF*

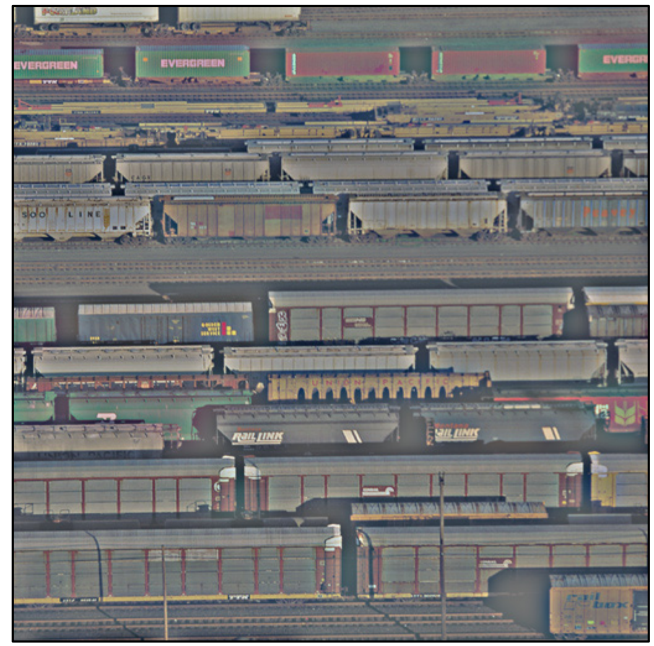
*signed image; 0 mapped to 128

Comparison of Ideal and Gaussian Filters

Gaussian LPF



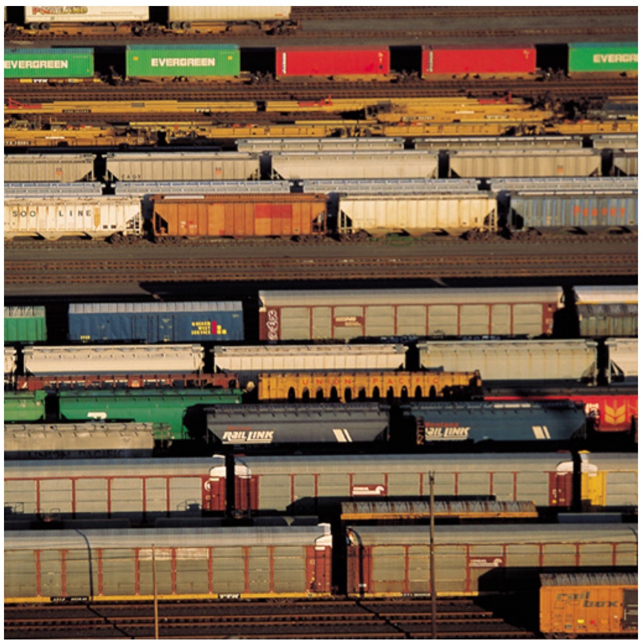
Original Image



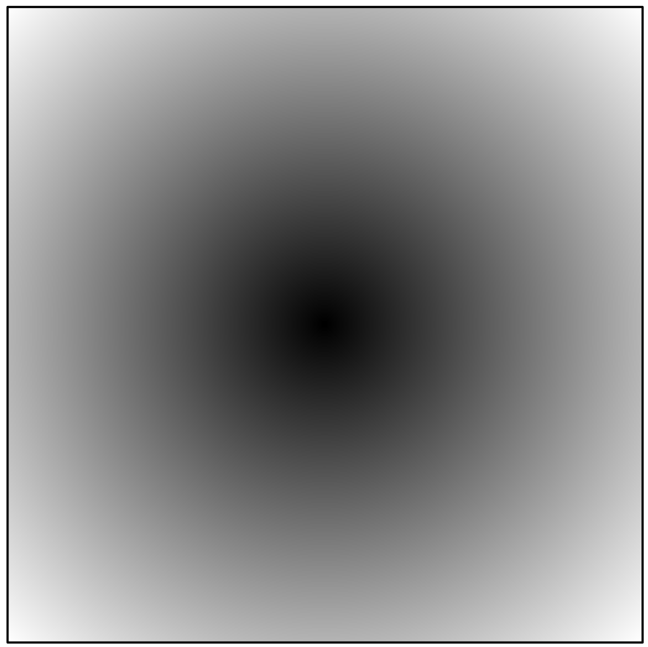
Gaussian HPF*

*signed image; 0 mapped to 128

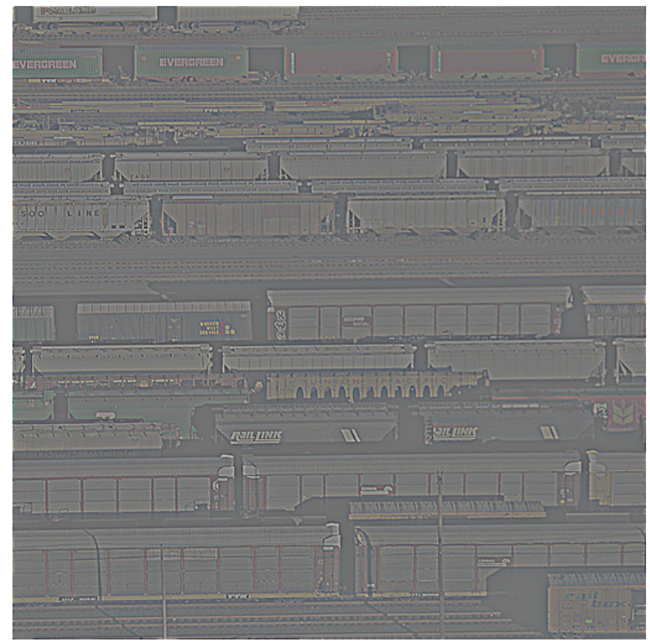
Another Highpass Filter



original image

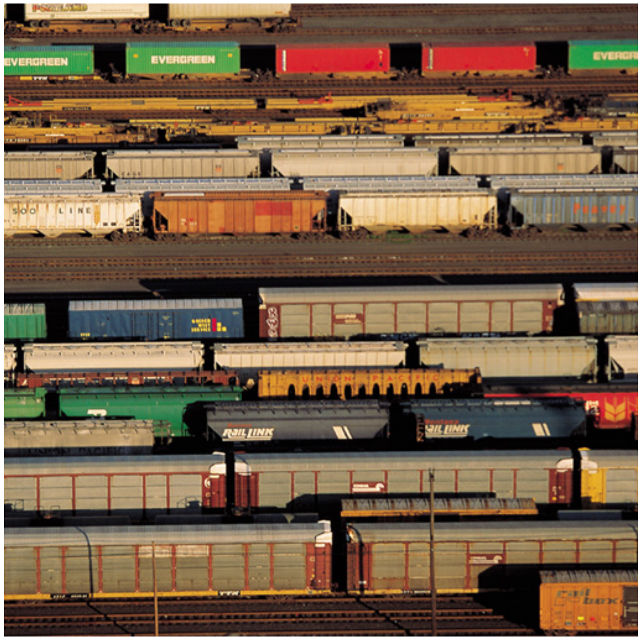


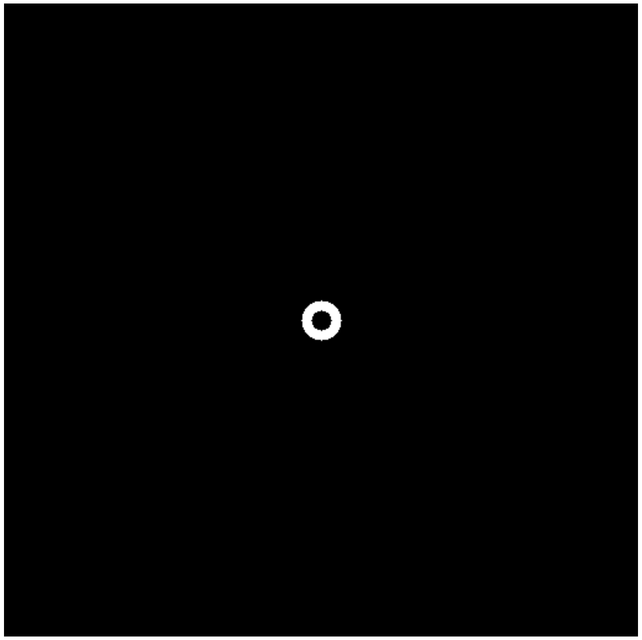
filter power spectrum



filtered image*

Ideal Bandpass Filter





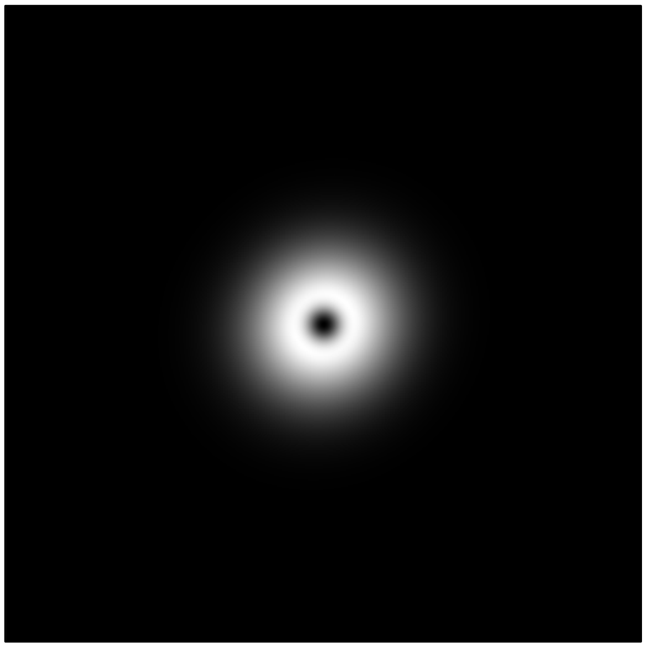
original image

filter power spectrum

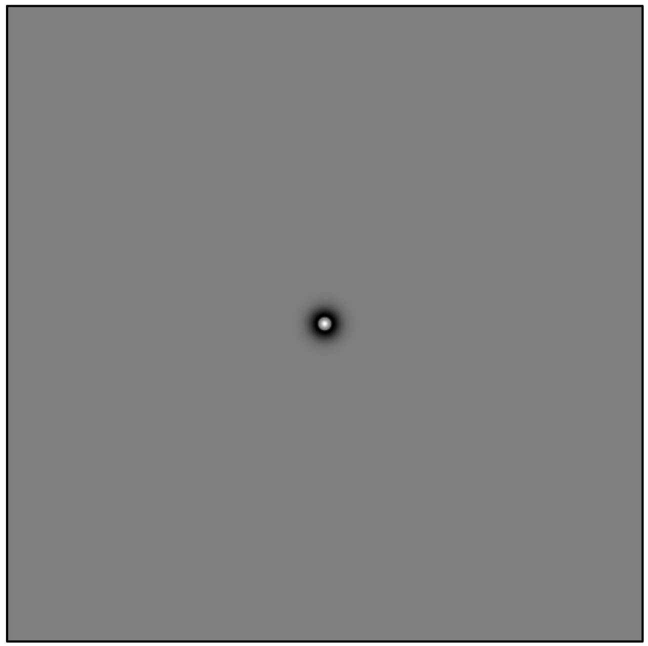
filtered image*

Gaussian Bandpass Filter

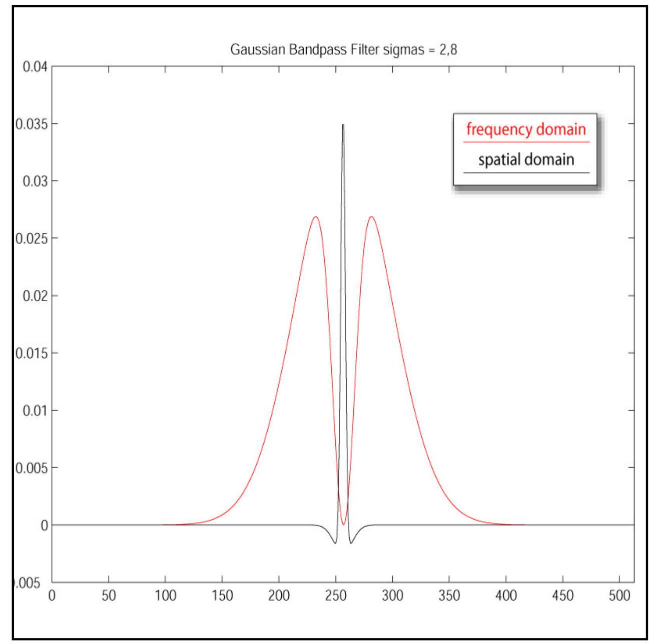
Image size: 512x512
sigma = 2 - sigma = 8



Fourier Domain Rep.



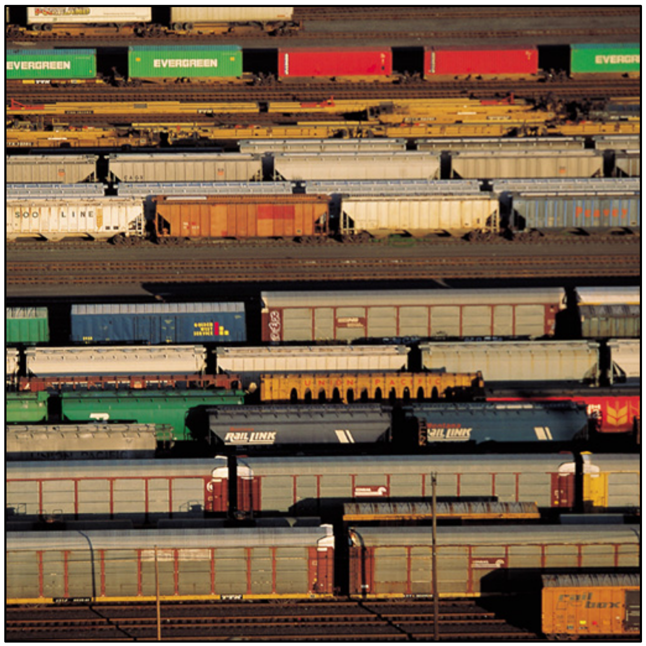
Spatial Representation



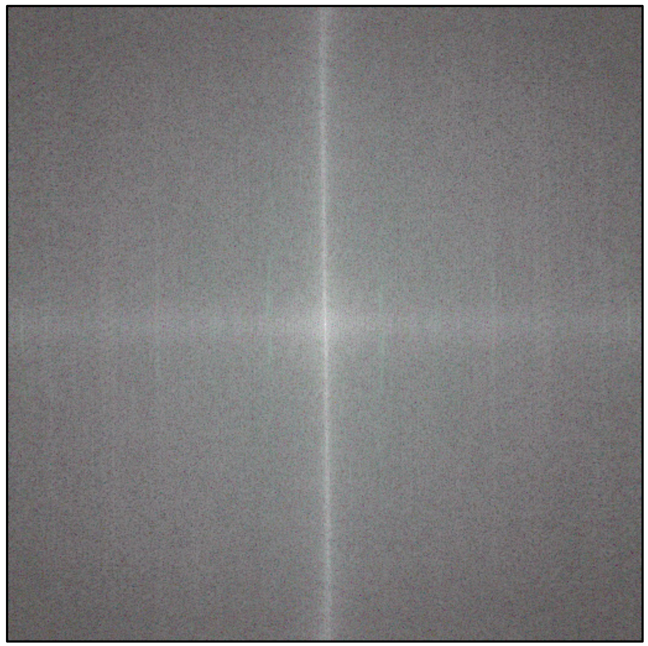
Central Profile

Gaussian Bandpass Filter

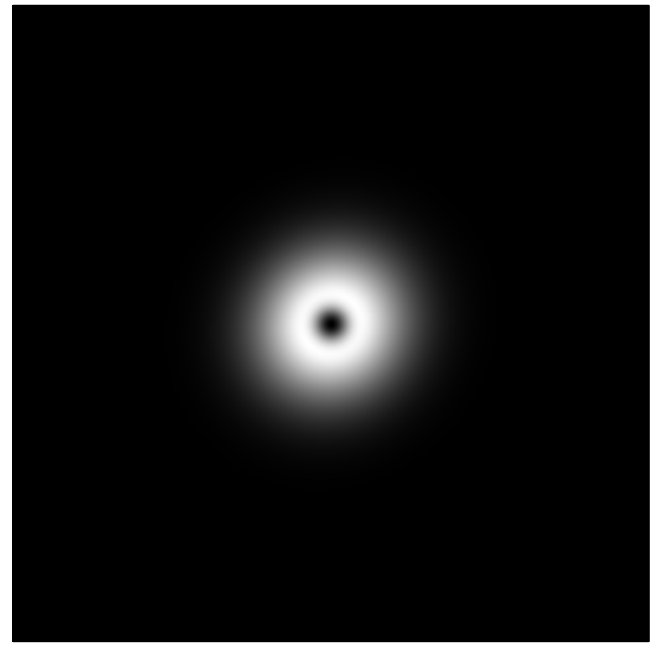
Image size: 512x512
sigma = 2 - sigma = 8



Original Image



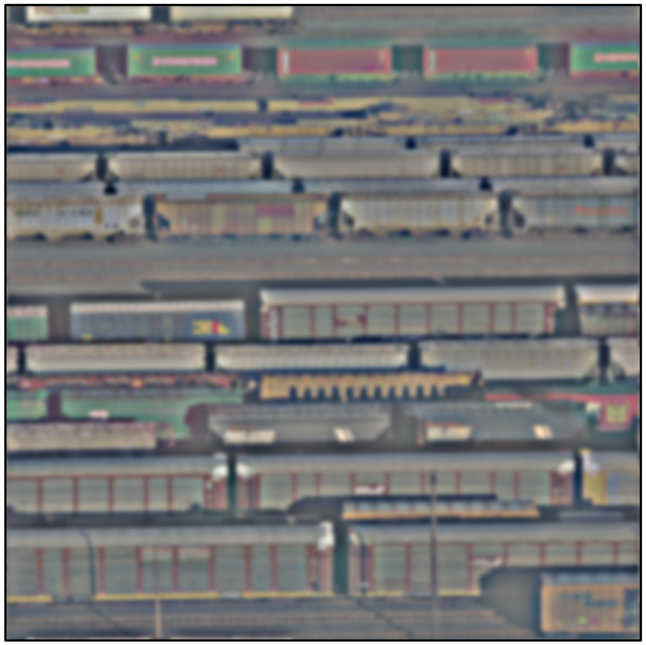
Power Spectrum

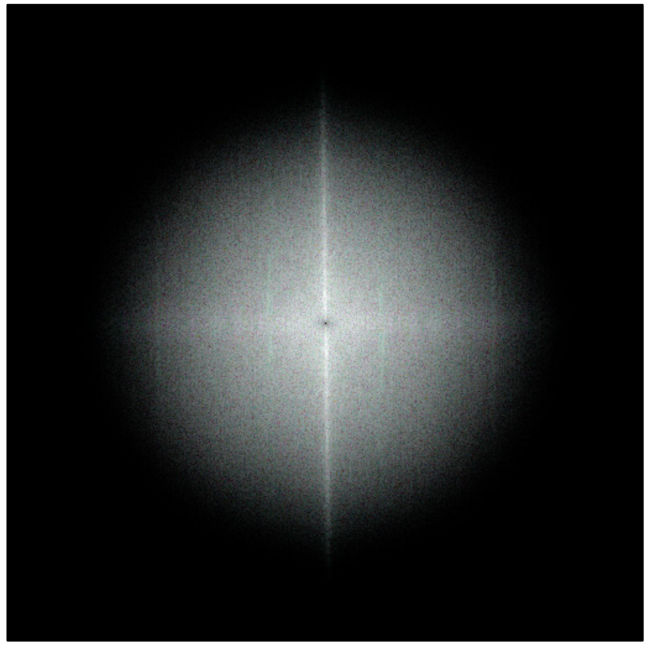


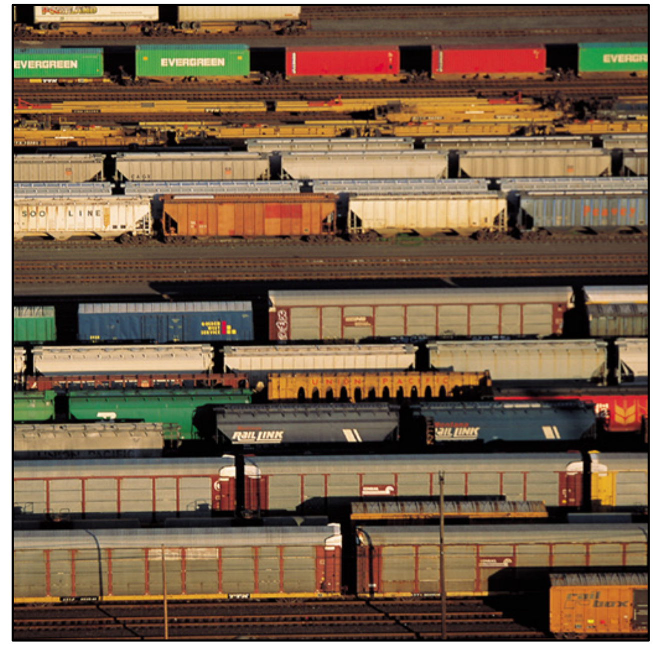
Gaussian BPF in FD

Gaussian Bandpass Filter

Image size: 512x512
sigma = 2 - sigma = 8







Filtered Image*

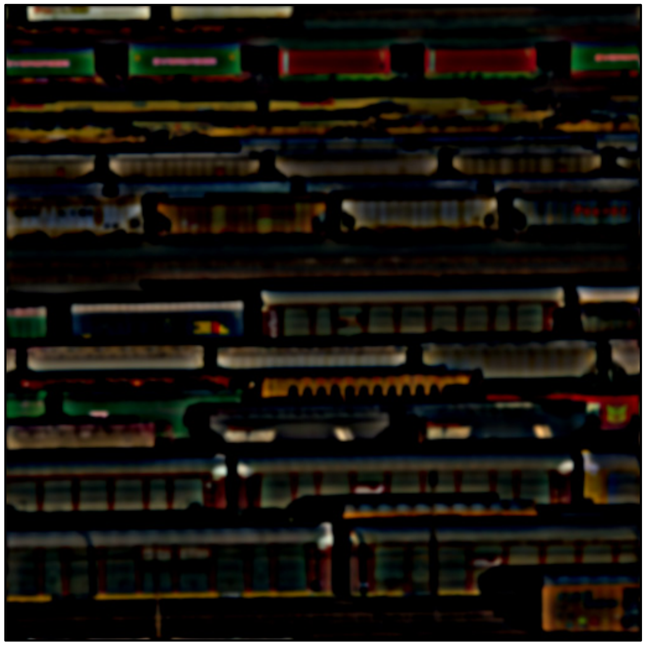
Filtered Power Spectrum

Original Image

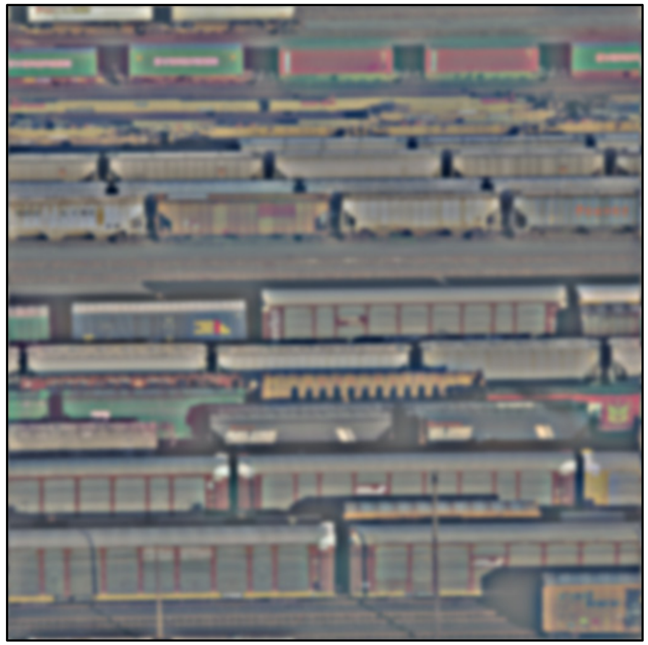
*signed image; 0 mapped to 128

Gaussian Bandpass Filter

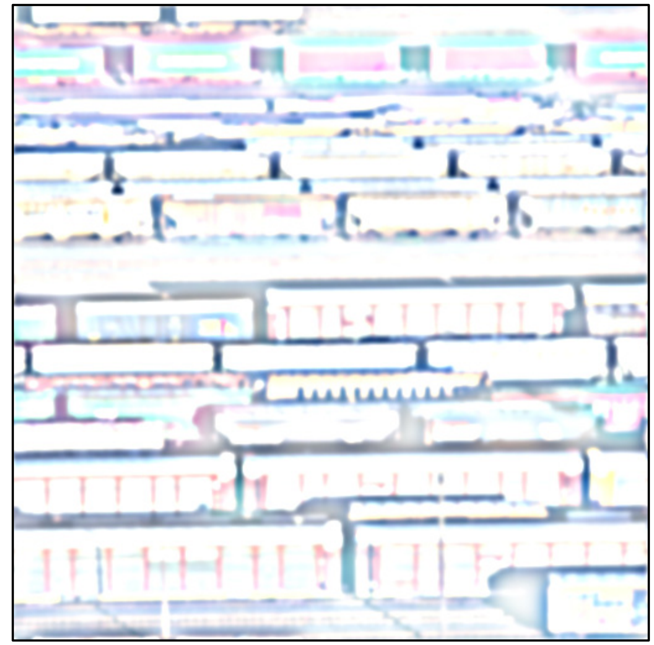
Image size: 512x512
sigma = 2 - sigma = 8



Positive Pixels



Filtered Image*

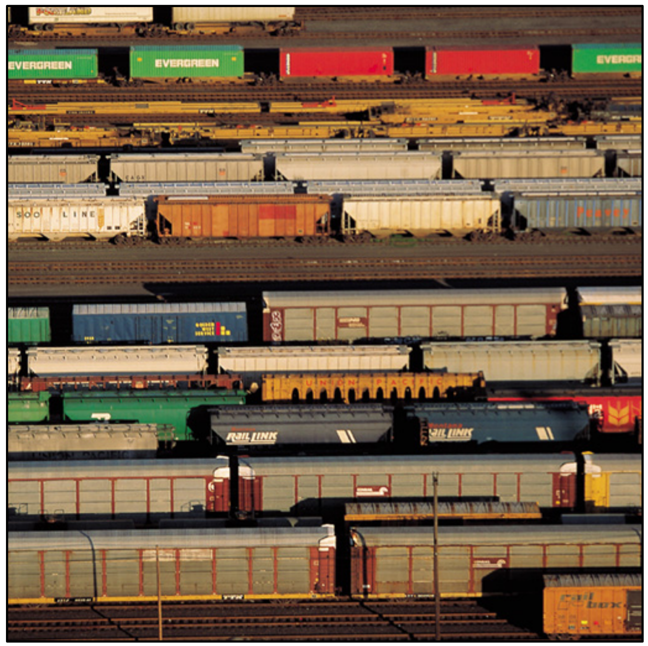


Negative Pixels

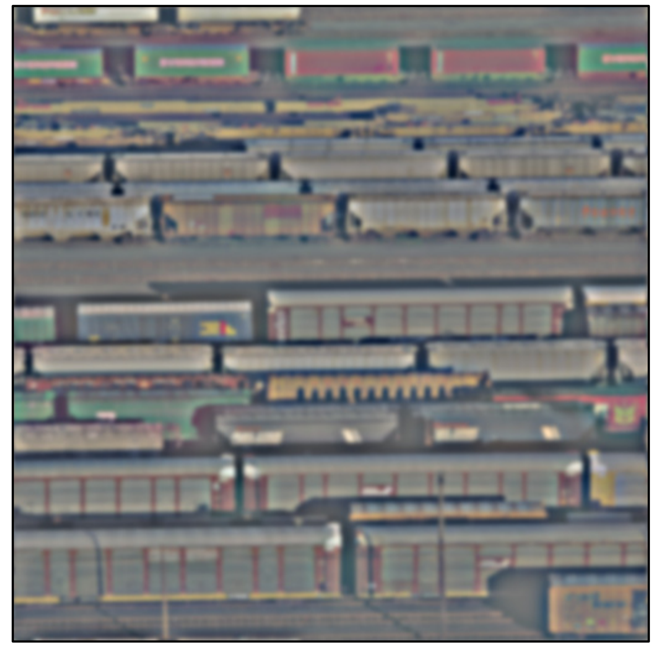
*signed image; 0 mapped to 128

Comparison of Ideal and Gaussian Filters

Ideal BPF*

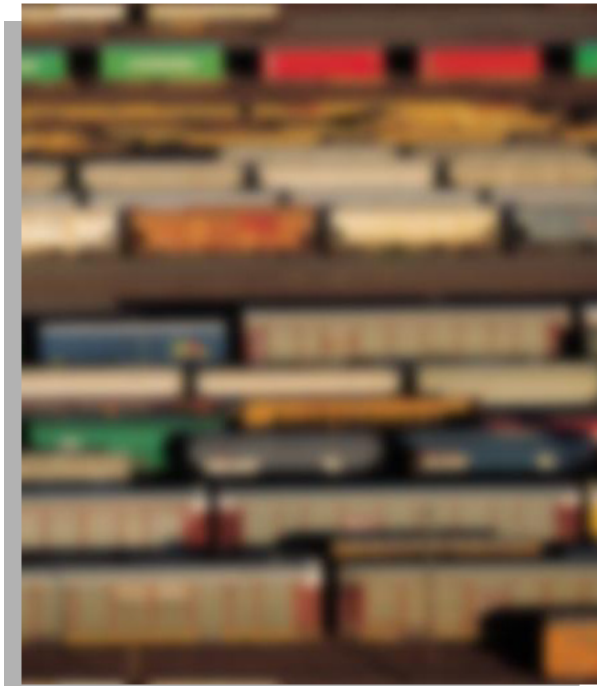


Original Image

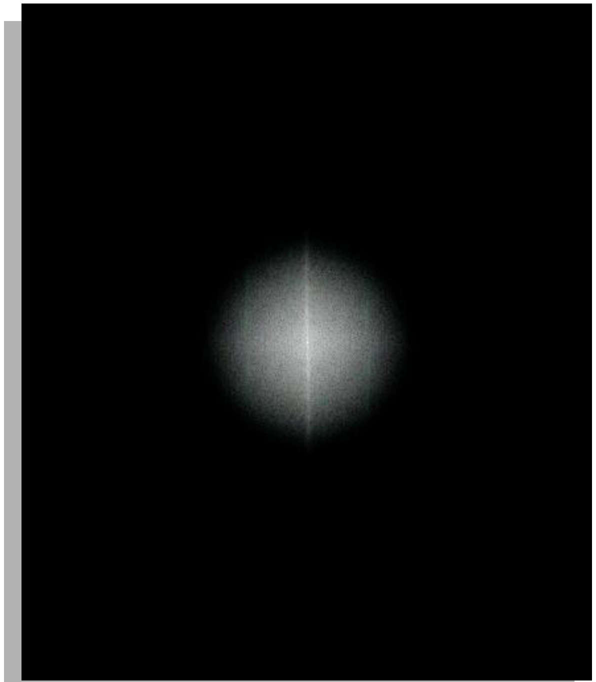


Gaussian BPF*

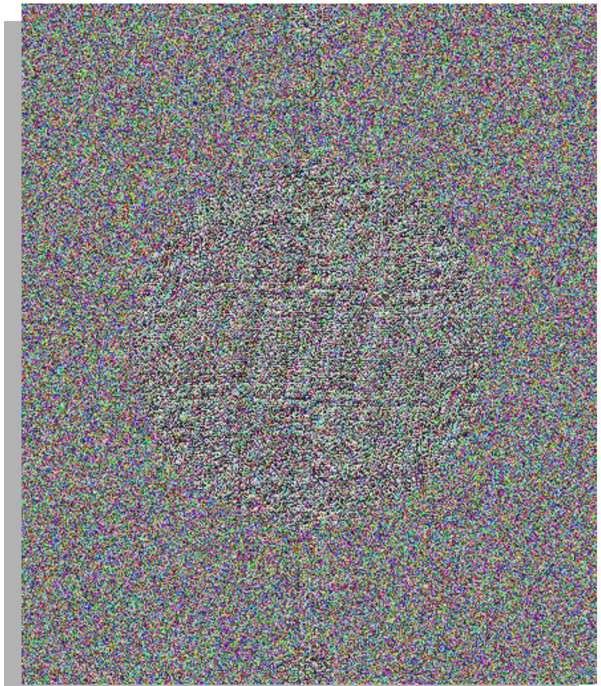
Power Spectrum and Phase of a Blurred Image



blurred image

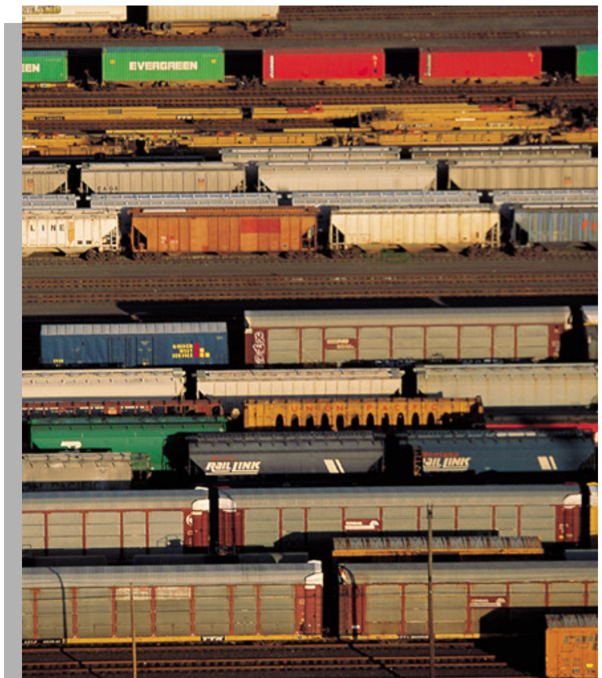


power spectrum

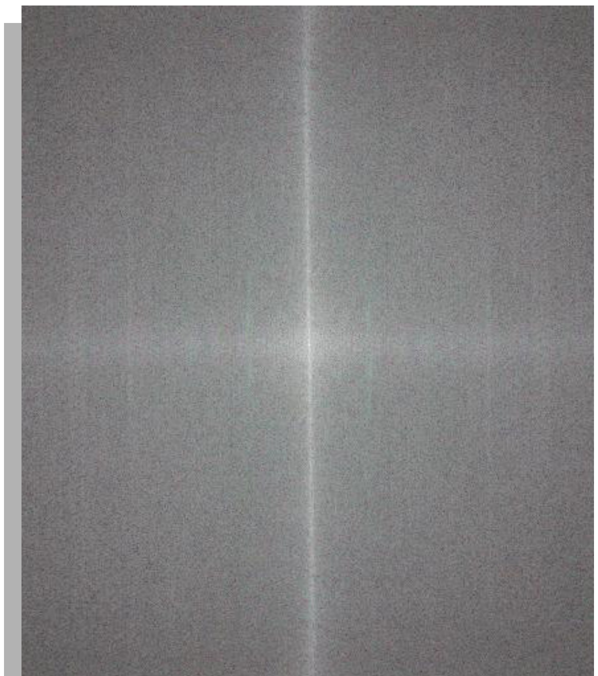


phase

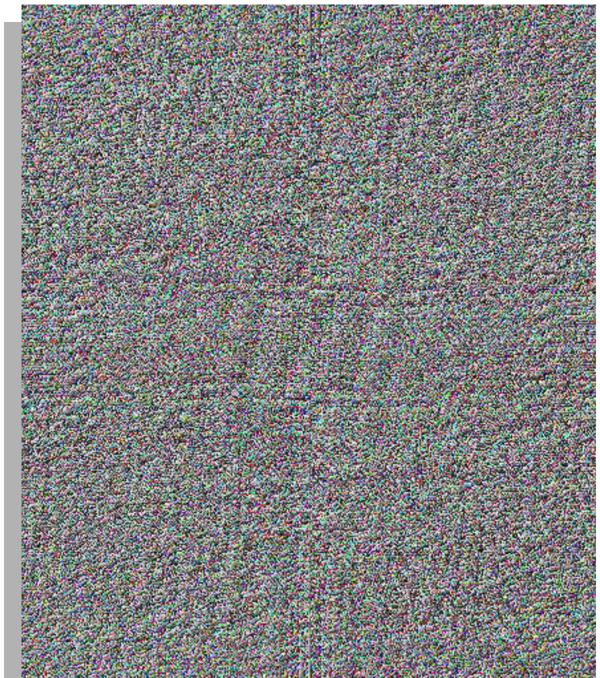
Power Spectrum and Phase of an Image



original image

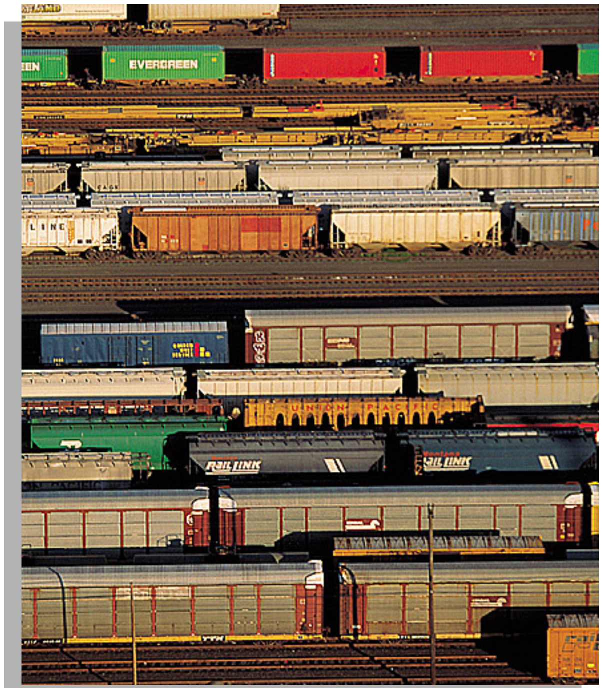


power spectrum

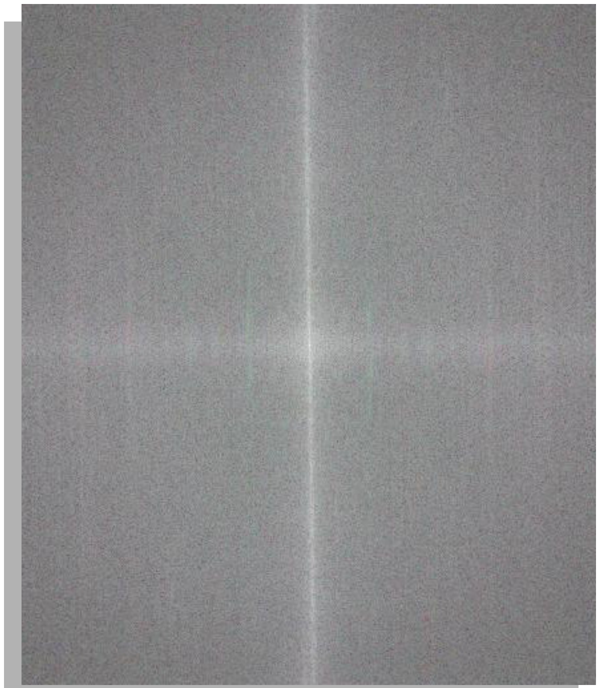


phase

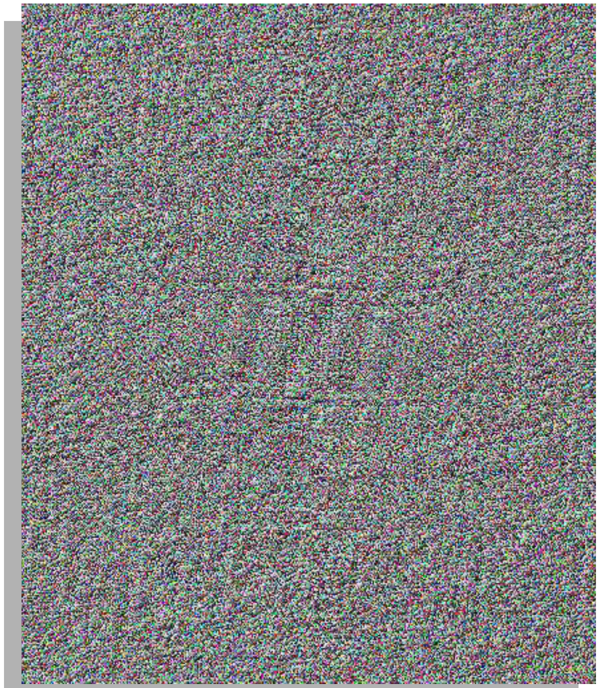
Power Spectrum and Phase of a Sharpened Image



sharpened image



power spectrum



phase