Lecture 13 - Deconvolution

Lecture 13: Deconvolution

Light Microscopy for the Biological Sciences
BIOL 5984
Jeff Kuhn

So, naturalists observe, a flea
Hath smaller fleas that on him prey;
And these have smaller fleas to bite `em,
And so proceed *ad infinitum*.
– Jonathan Swift, 1733

Examples of Variable Substitution in Functions

Variable substitution in 1-D
$f(x) = \sin(x)$

Variable substitution: change x to u
$f(u) = \sin(u)$

Multiplication: multiply "u" by something
$f(2u) = \sin(2u)$
$f(-u) = \sin(-u)$

Addition: add some other value (e.g., "x")
$f(x - u) = \sin(x - u)$

Variable substitution in 2-D
$f(x,y) = \sin(x) + \cos(y)$

change x to u and y to v
$f(u,v) = \sin(u) + \cos(v)$

Other similar transformations …

2

Formal Definition of Convolution

Two functions $f(x)$ $h(x)$

$g(x) = f(x) * h(x)$ ← convolution (also $\otimes$)

$g(u) = \int_{-\infty}^{\infty} f(x)\, h(u-x)\, dx$

new x

new x

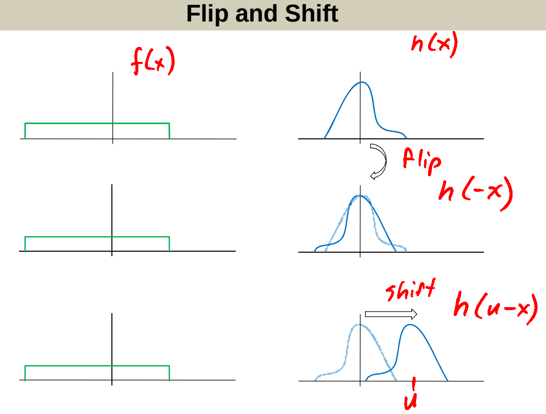
old x

$g(x_{new}) = \int_{-\infty}^{\infty} \underbrace{f(x_{old})\, h\underbrace{(x_{new} - x_{old})}_{\text{flip \& shift}}\, dx_{old}}_{\text{area under the curve}}$

multiply

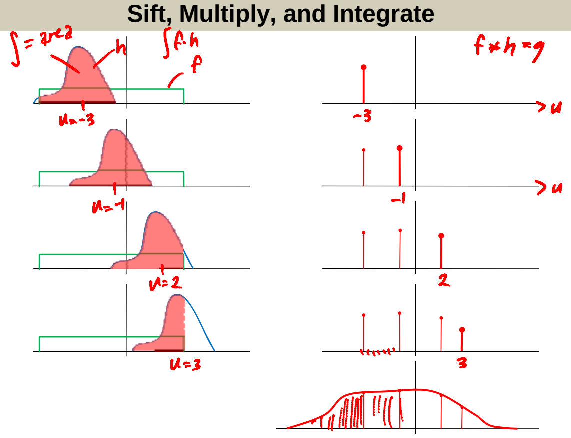
3

Flip and Shift



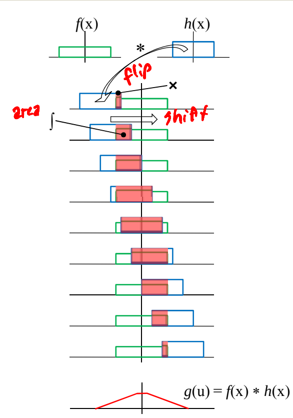
4

Sift, Multiply, and Integrate



5

Ex: Convolution of Two Square Waves

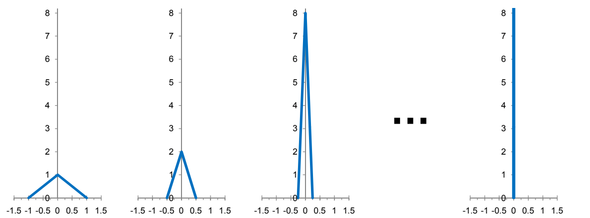


6

Lecture 13 - Deconvolution

Delta Function δ

$\delta(x)$



Triangle	Triangle	Triangle	Delta Function
Width = 2	Width = 1	Width = 0.25	Width = 1/∞ = 0
Height = 1	Height = 2	Height = 8	Height = ∞
Area = 1	Area = 1	Area = 1	Area = 1

$\int \delta(x)\,dx = 1$

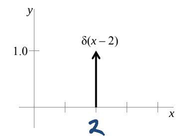
Integrating with Delta Functions

Definition of a delta function

$\int_{-\infty}^{\infty} \delta(x)\,dx = 1$ total area = 1

$\delta(x) = \infty$ when $x = 0$
$\delta(x) = 0$ when $x \neq 0$

Representing delta functions



Delta functions "select" single points from other functions

$$\int_{-\infty}^{\infty} f(x)\,\delta(x)\,dx = f(0)$$

$$\int_{-\infty}^{\infty} f(x)\,\delta(x-2)\,dx = f(2)$$

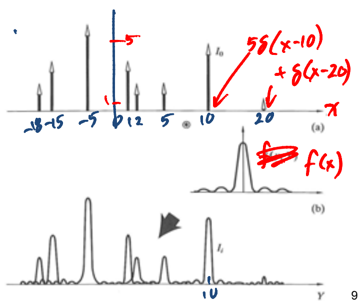
Convolutions with Delta Convolutions

$f(x) * \delta(x) = f(u)$

$$\int_{-\infty}^{\infty} f(x)\,\delta(u-x)\,dx = f(u)$$

δ functions can shift $f(x)$

$f(x) * \delta(x-2)$
$= f(u-2)$



Example PSF Convolutions

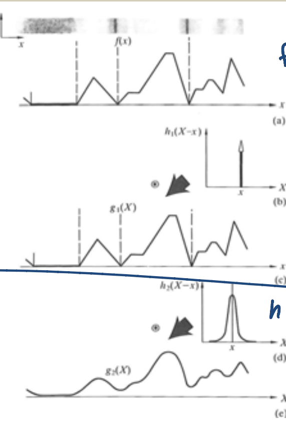
specimen

$f(x)$

$h(x) = \delta(x)$

image 1

$f(x) * h(x)$



$h(x) = \text{gaussian}$

image 2

$g(u) = f(x) * h(x)$

Fourier Transform turns Convolution into Multiply

$F(s) = \mathcal{F}\{f(x)\}$ fourier transform

frequency

$H(s) = \mathcal{F}\{h(x)\}$ $G(s) = \mathcal{F}\{g(x)\}$

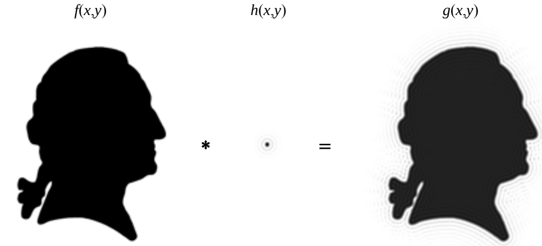
$g(x) = f(x) * h(x)$ convolution in "real" space

$G(s) = F(s)\,H(s)$ multiplication in "frequency" space

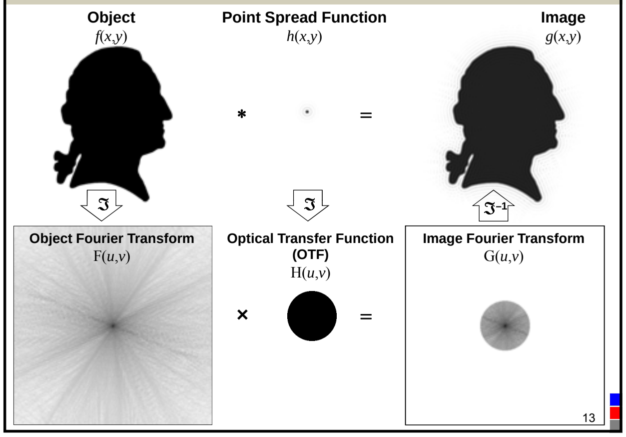
$g(x) = \mathcal{F}^{-1}\{G(s)\}$ inverse fourier transform

2D Convolution in Microscopy

Object $f(x,y)$ **Point Spread Function** $h(x,y)$ **Image** $g(x,y)$

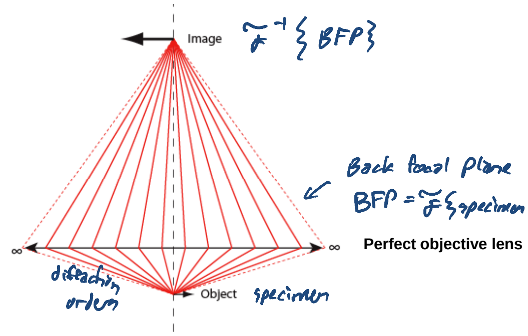


Convolution in Fourier Space



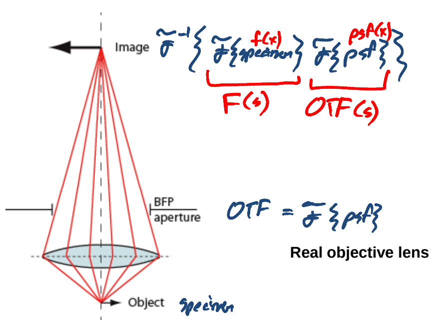
13

OTF is the Fourier Transform of the PSF



14

OTF is the Fourier Transform of the PSF



$\mathcal{F}^{-1}\{\underbrace{\mathcal{F}\{specimen\}}_{F(s)}\underbrace{\mathcal{F}\{psf\}}_{OTF(s)}\}$ f(x) psf(x)

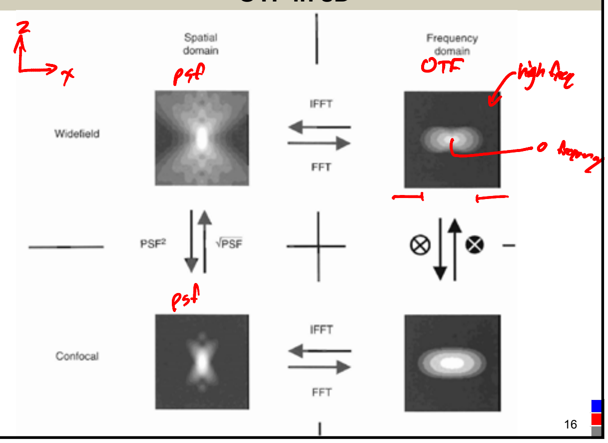
$OTF = \mathcal{F}\{psf\}$

objective lenses convolve

image $g(x) = f(x) * psf(x)$

15

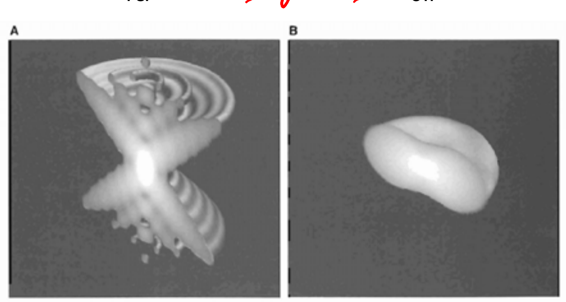
OTF in 3D



16

OTF in 3D

PSF → $\mathcal{F}$ → OTF



17

Basic Principle of Deconvolution Microscopy

$\mathcal{F}\{image\}$ $\mathcal{F}\{specimen\}$ $\mathcal{F}\{psf\} = OTF$

$$G(u,v) = F(u,v) \cdot H(u,v)$$

frequency domain

capture want measure

$$F = \frac{G}{H}$$

deconvolution

$$f(x,y) = \mathcal{F}^{-1}\{F(u,v)\}$$

18

Lecture 13 - Deconvolution

Real Deconvolution

image = specimen * psf

$g(x) = f(x) * h(x) + \text{noise}$

$G = FH \pm N$ ← $\mathcal{F}\{\text{noise}\}$

$F = \frac{G \pm N}{H}$ ← noise amplified when H is small

$F \to \infty$ when $H \to 0$ at some points

Deconvolution is an "inverse" problem.
i.e. How do we compute the "inverse" (1/H) of the OTF?

Deconvolution Strategies

Strategy 1. Use only a few neighboring image planes
- **"Nearest Neighbor" algorithm**

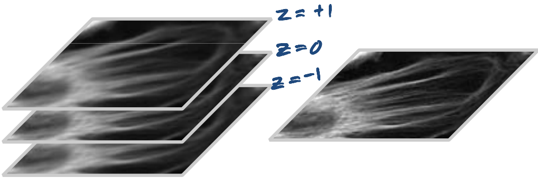
Strategy 2. Use a few "tricks" and do division in Fourier Space
- **"Regularization" tricks**
  1. **Cut-off 1/H at high frequencies**
  2. **Smoothly cut-off 1/H**

Strategy 3. Use an "iterative" solution:
1. Guess what the object looks like (F_{guess})
2. Multiply guess by the OTF ($H \times F_{guess}$)
3. Compare to actual Image (G)
4. Update guess and go back to 1

1. Nearest Neighbor Deconvolution – Simple

- **Simple Algorithm**
  1. Take stack of 3 images
  2. Blur upper and lower image
  3. Subtract blurred upper/lower images from center image

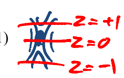
$$g'(x,y,z) = g(x,y,z) - \alpha \times g_{blurred}(x,y,z-1) - \alpha \times g_{blurred}(x,y,z+1)$$



Advantage: Very fast and does not require much memory
Disadvantage: Not a "true" inverse.

1. Nearest Neighbor Deconvolution – Complex

- **More Complex Algorithm**
  1. Measure or calculate out of focus PSF: $h(x,y,z+1)$
  2. Take stack of 3 images
  3. Convolve upper/lower images with out-of-focus PSF
  4. Subtract from central image
  5. Subtract blurred upper/lower images from center image

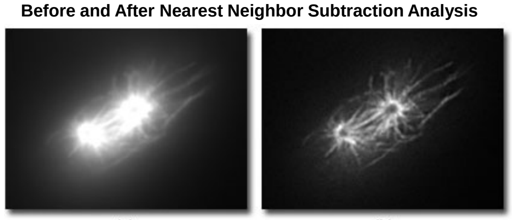


$$G'(u,v,z) = G(u,v,z) - \alpha\, H(u,v,z+1)\, [\, G(u,v,z-1) - G(u,v,z+1)\,]$$

- **Requires 2D Fourier Transforms (to compute convolution)**
- **Slower than simple nearest neighbor**
- **Can use more planes above and below for better "deblurring"**
- **Not a true 3D deconvolution method**

Example of Nearest Neighbor Subtraction

Before and After Nearest Neighbor Subtraction Analysis



(a) (b)
Figure 1

Regularized 3D deconvolution

Truncated inverse filter **F = G/H**
- Ignore 1/H(s) for small values of H

New function

$$J(s) = \begin{cases} \frac{1}{H(s)} & \text{if } |H(s)| > \varepsilon \\ 0 & \text{otherwise} \end{cases}$$

← small number

$F = GJ$

Regularized 3D deconvolution

Apodized inverse filter (Tikhonov Regularization) **F = G/H**

- Gently roll-off 1/H(s) for small values of H
- Avoids overly large pixel values

complex conjugate

$$J(s) = \frac{H^*(s)}{|H(s) + \alpha|^2}$$

small Number

$$F = G \cdot J$$

complex number

$A = a + ib$

real imaginary

$i = \sqrt{-1}$ imaginary number

conjugate

$A^* = a - ib$

26

Regularization Filters

Advantages	Disadvantages
• Fast	• Some negative pixels in final
• Little to moderate memory requirements	• Cannot restore "missing cone"
	• Noise is a problem

27

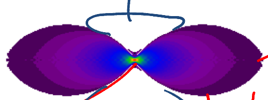
The Missing Cone Problem in Widefield

z x

Widefield OTF in x and z

OTF

missing frequencies



high freq in x

x resolution

missing z information

high freq in z direction

z-resolution

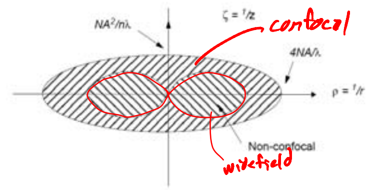
zero frequency (order) background

28

Confocal Starts with More Z-information

OTF

Confocal frequency response



confocal

widefield

Twice the band width in both directions
(lateral and axial)

No missing cone
(Missing cone is very detrimental to 3-D images

29

Constrained Iterative (CI) Deconvolution

Forces pixels to be non-negative

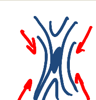
Algorithm

1. **Guess the 3D specimen distribution** $f_1(x,y,z)$ first guess
2. **Convolve guess with PSF** $g_1(x,y,z) = f_1(x,y,z) * h(x,y,z)$
3. **Compare to recorded image** $d_1 = g - g_1$ ← guess; difference; recorded image
4. **Weight to enforce non-negative pixels** $\gamma(x)$ varies across image
5. **Update guess (based on difference)** $f_2(x,y,z) = f_1(x,y,z) + \gamma d_1(x,y,z)$
6. **Repeat** 1–5
7. **Stop when difference is small or large # of iterations**

30

Maximum Likelihood Estimation (MLE)

- **Very Similar to Constrained Iterative**
- **Assumes noise follows "Poisson" distribution**
- **Differs in how guess is updated after each step** (math too complex to discuss here)
- **More accurate than Constrained Iterative**



z-info compressed

restores missing cone

- **Example guess update:**

image

$$r_1(x,y,z) = \mathcal{F}^{-1}\left\{ \mathcal{F}\left\{ \frac{g}{f_1 * psf} \right\} \cdot OTF^* \right\}$$

ratio

guess

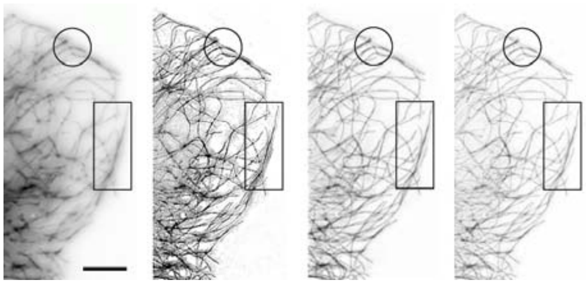
$$f_2(x,y,z) = f_1(x,y,z) \cdot r_1(x,y,z)$$

next guess, current guess

31

Comparison of Iterative Restoration (microtubules)

Raw Image | Constrained Iterative | MLE Method 1 | MLE Method 2



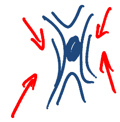
Images inverted for clarity

32

MLE Deconvolution

Advantages

- Image always non-negative
- Can fill missing cone



Disadvantages

- Very slow
  Minutes to hours per stack
- Requires a lot of memory
- Must measure PSF

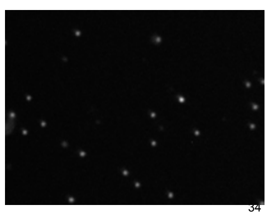
33

Measuring the PSF

1. Use sub-resolution fluorescent beads (~200 nm diameter)
   Tetraspeck PSF beads from Invitrogen

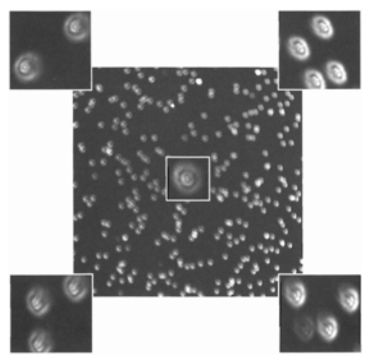
   1a Sonicate to disperse
2. Dilute beads in solution
3. Spread drop of bead solution on slide
4. Wait for water to dry
5. Add drop of glycerol, water, immersion oil, or other mounting media – then coverslip
6. Take image stacks of several beads
7. Align and average bead images

- Take bead stack with SAME IMAGING PARAMETERS as image stack (same x-,y-,z-spacing)
- Keep bead density low (prevents PSF overlap)
- Stay near center of field of view



34

Field Curvature Can Effect PSF

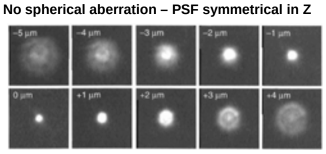


- PSF might vary at edges of field
- Use "PSF" objective – corrects
  or
- Always use center of field
- Most deconvolution algorithms CANNOT correct field curvature
- Some blind deconvolution algorithms can correct!

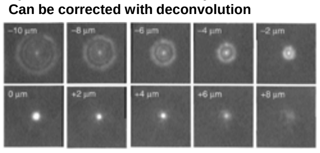
35

Spherical Aberration

No spherical aberration – PSF symmetrical in Z

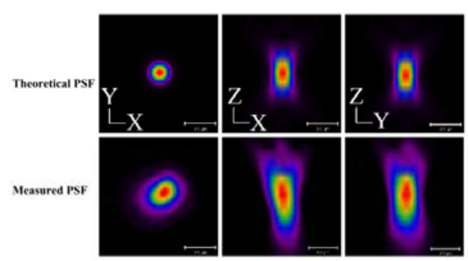


Spherical aberration – PSF asymmetrical in Z
Can be corrected with deconvolution



36

Measured PSF can differ from Theory



37

Blind Deconvolution

What if we did not measure the PSF?

- **Blind Deconvolution: We do not know the PSF beforehand**
- **Or difficult to measure PSF within specimen**

- **Calculate an initial "guess" of the PSF based on microscope optics**
- **Calculate an initial "guess" of the specimen**
- **Use an iterative method as before, but**
  - **Update specimen guess each iteration**
  - **Update PSF guess each iteration**

$g(x)^{1,2} = f(x)^{1,2} * psf(x)^{1,2}$

guess guess

- **"Blind" PSF can vary over image plane**
  Corrects for field curvature and coma

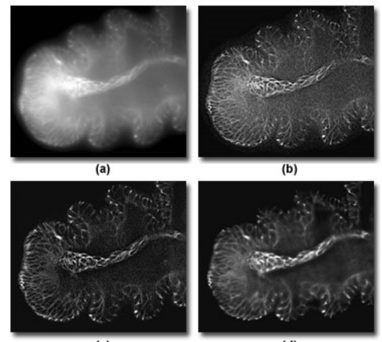
38

Comparison of Techniques

Fruit fly embryo leg

Raw image

Nearest Neighbor Subtraction



(a) (b)

Simple Inverse Filter

Blind Deconvolution

(c) (d)

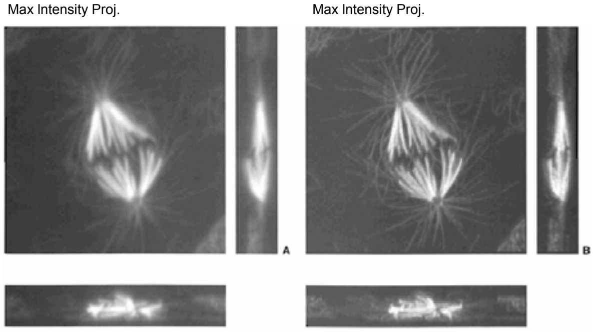
Figure 3

39

Deconvolution of Confocal Stacks

Confocal Stack
Max Intensity Proj.

Deconvolved Stack MLE
Max Intensity Proj.



40