⊕ ⊖ Computational ⊗ ⊘
Photography
Image Stabilization

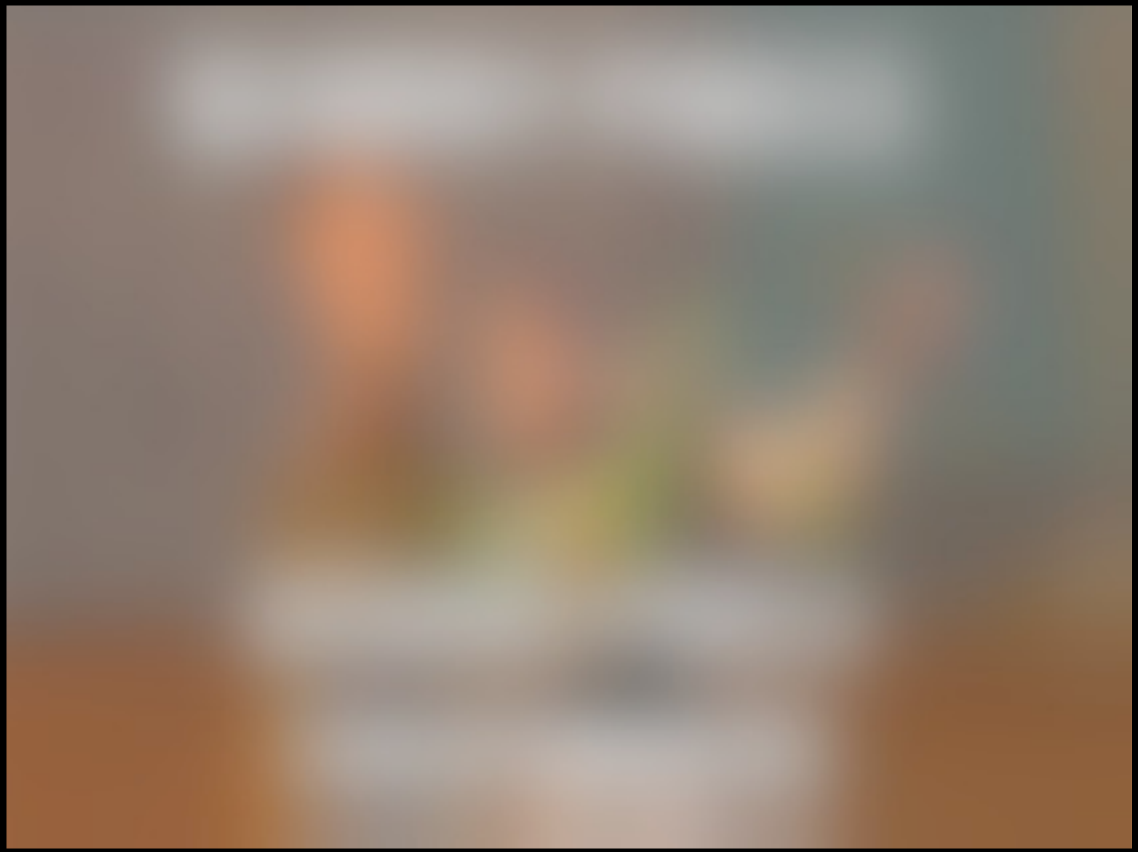
Jongmin Baek

CS 478 Lecture
Mar 7, 2012

Overview

- Optical Stabilization
  - Lens-Shift
  - Sensor-Shift
- Digital Stabilization
  - Image Priors
  - Non-Blind Deconvolution
  - Blind Deconvolution

Blurs in Photography



Blurs in Photography

- Defocus Blur

1/60 sec, f/1.8, ISO 400

Blurs in Photography

- Handshake

2 sec, f/10, ISO 100

Blurs in Photography

- Motion Blur

1/60 sec, f/2.2, ISO 400

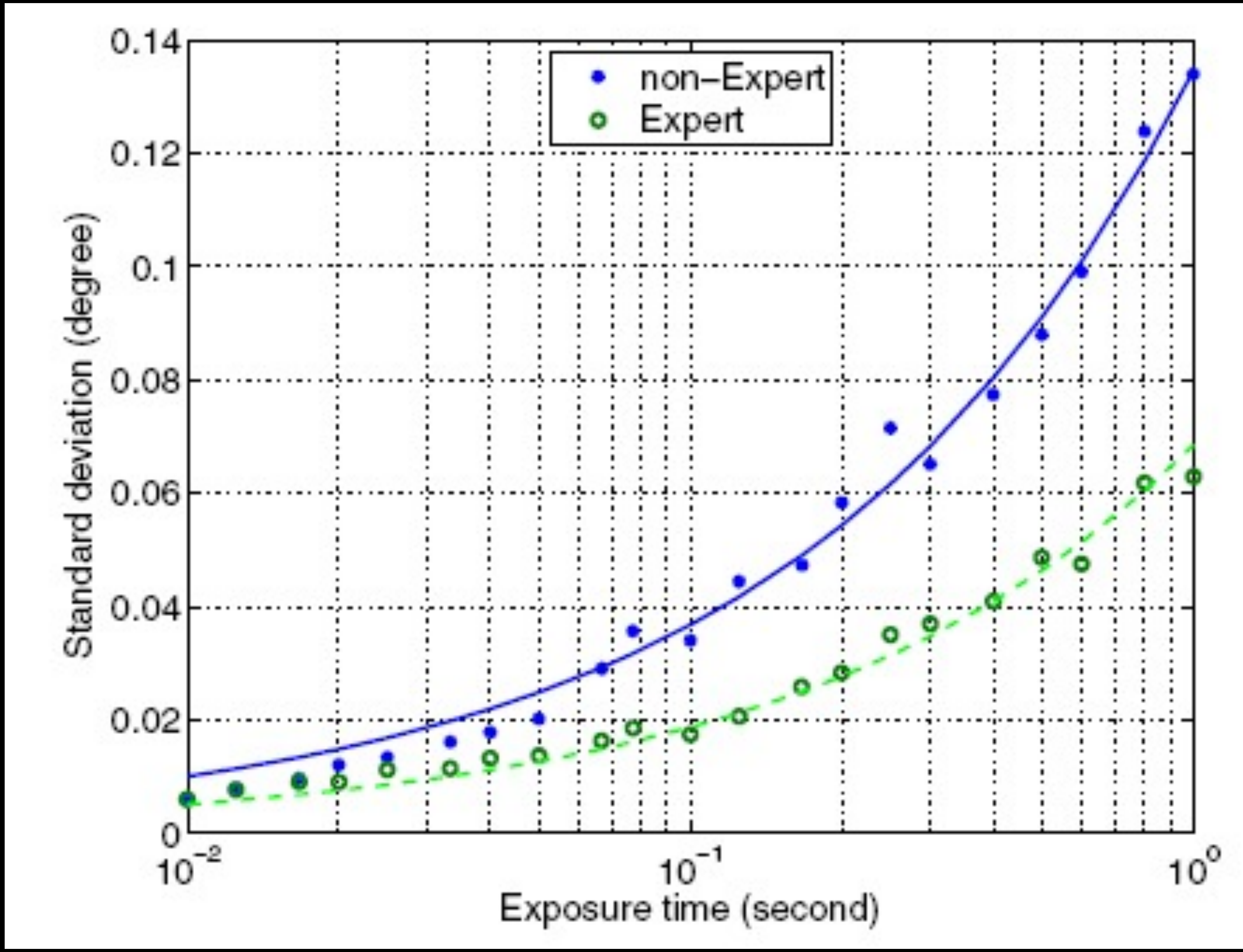
Blurs in Photography

- Some blurs are intentional.
  - Defocus blur: Direct viewer's attention. Convey scale.
  - Motion blur: Instill a sense of action.
  - Handshake: Advertise how unsteady your hand is.
    - Granted, jerky camera movement is sometimes used to convey a sense of hecticness in movies.

How to Combat Blur

- Don't let it happen in the first place.
  - Take shorter exposures.
  - Tranquilize your subject, or otherwise make it still.
  - Stop down.
- Sometimes you have to pick your poison.
  - Computational optics?

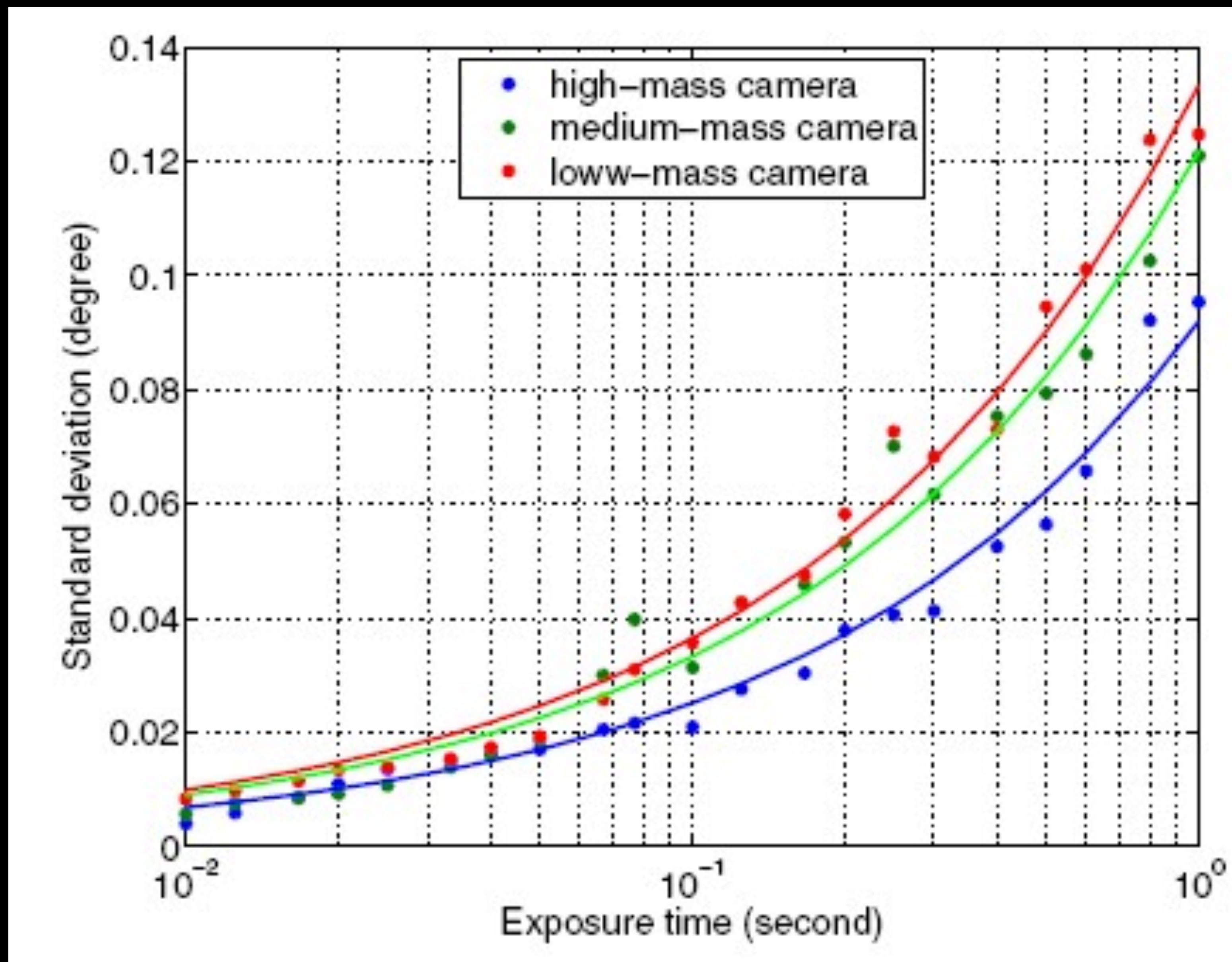
How to Combat Handshake



You can train yourself to be steady.

figures stolen from
Sung Hee Park

How to Combat Handshake



Use a heavier camera.

figures stolen from Sung Hee Park

Optical Image Stabilization

- Fight handshake.
  - Lens-Shift Image Stabilization
    - Vary the optical path to the sensor.
  - Sensor-Shift Image Stabilization
    - Move the sensor to counteract motion.

Lens-Shift Image Stabilization

- Lots of different names
  - Image Stabilization (Canon)
  - Vibration Reduction (Nikon)
  - Optical Stabilization (Sigma)
  - Vibration Compensation (Tamron)
  - Mega OIS (Panasonic, Leika)

content stolen from
Sung Hee Park

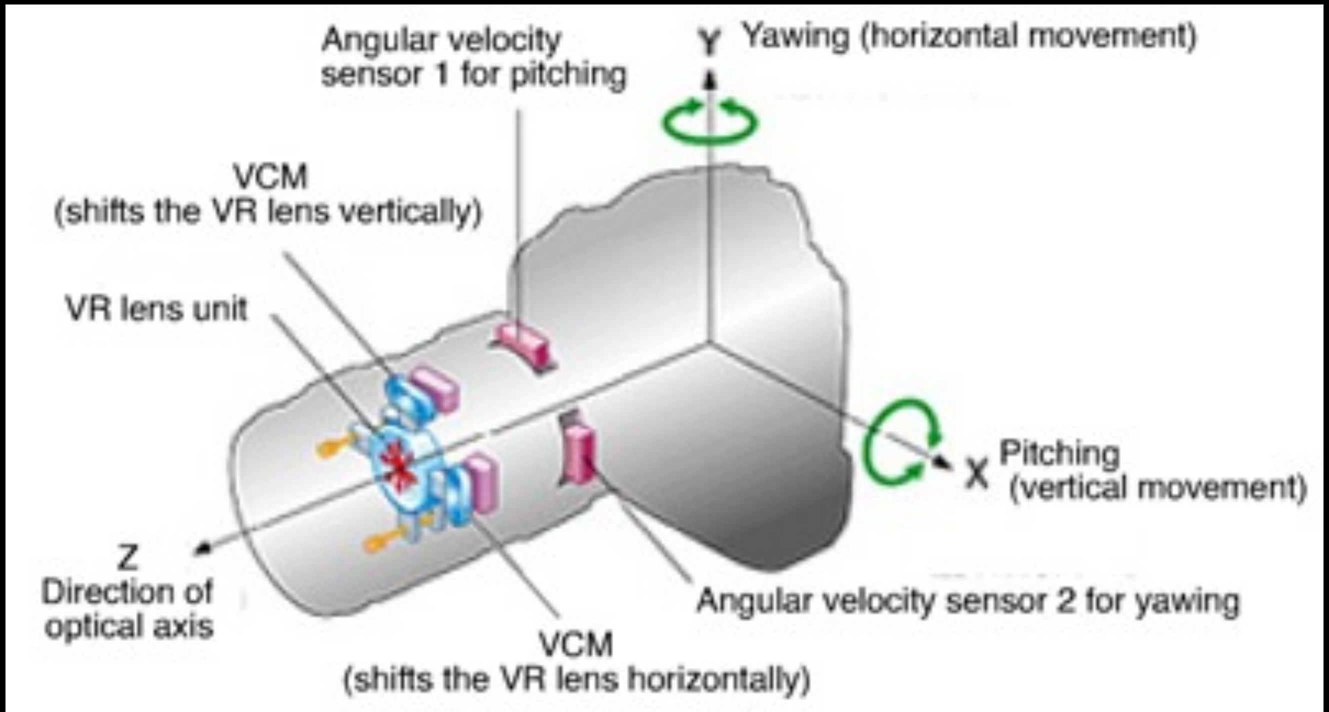
History of Image Stabilization

Canon IS

Year	Lens	Stability	Characteristic
1995	75-300mm f/4-5.6 IS USM	2 stop	The first IS lens
1997	300mm f/4L IS USM	2 stops	New IS mode
1999	300mm f/2.8L IS USM	2 stops	Tripod detection
2001	70-200mm f/2.8L IS USM	3 stops	
2006	70-200mm f/4L IS USM	4 stops	
2008	200mm f/2L IS USM	5 stops	

content stolen from
Sung Hee Park

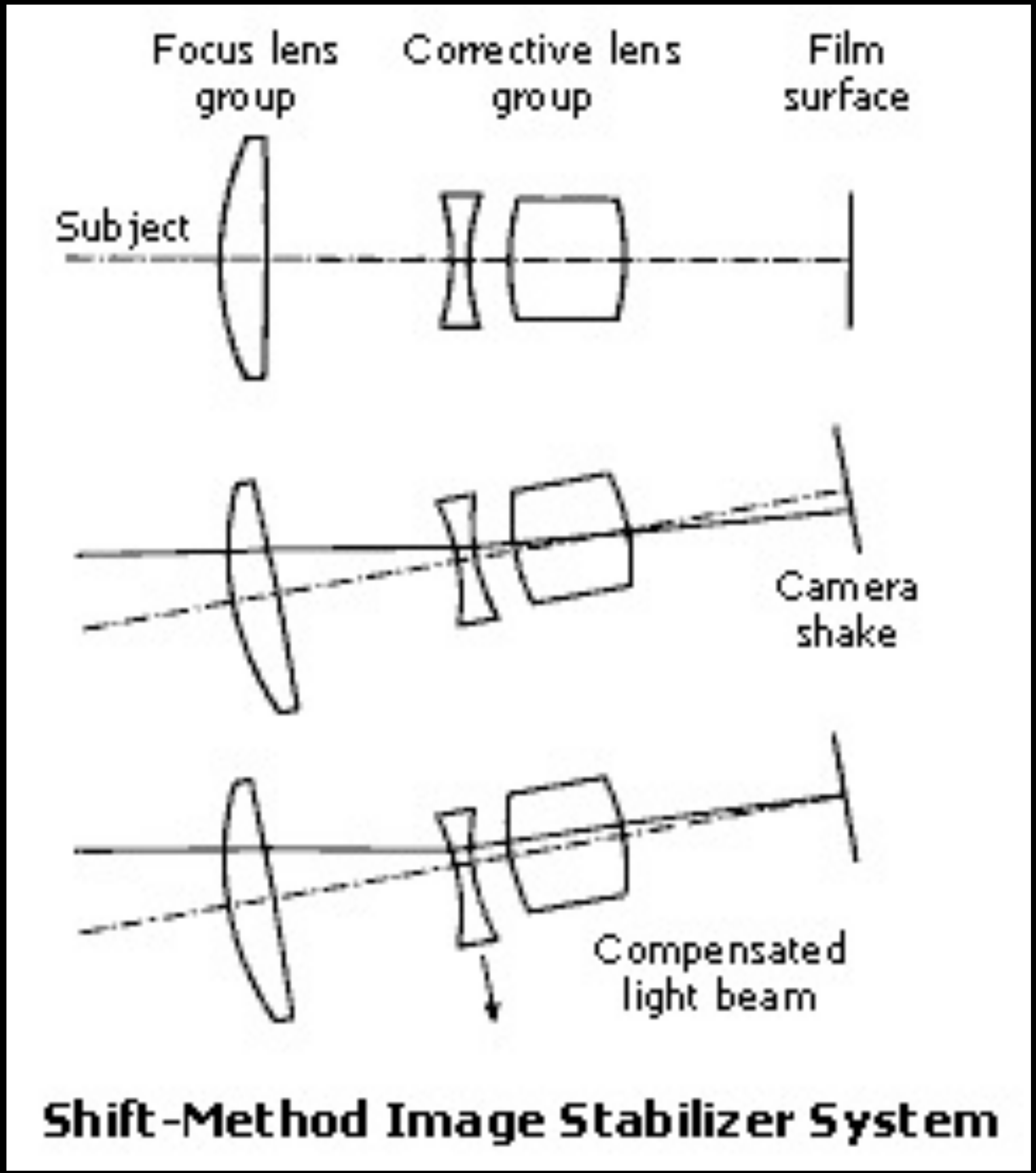
Lens-Shift Image Stabilization



- A floating lens element moves orthogonally to the optical axis, using electromagnets.
  - Vibration is detected by two gyroscopes.
  - Pitch and yaw movements are compensated.
  - Roll and linear movement are not.

figures stolen from Sung Hee Park

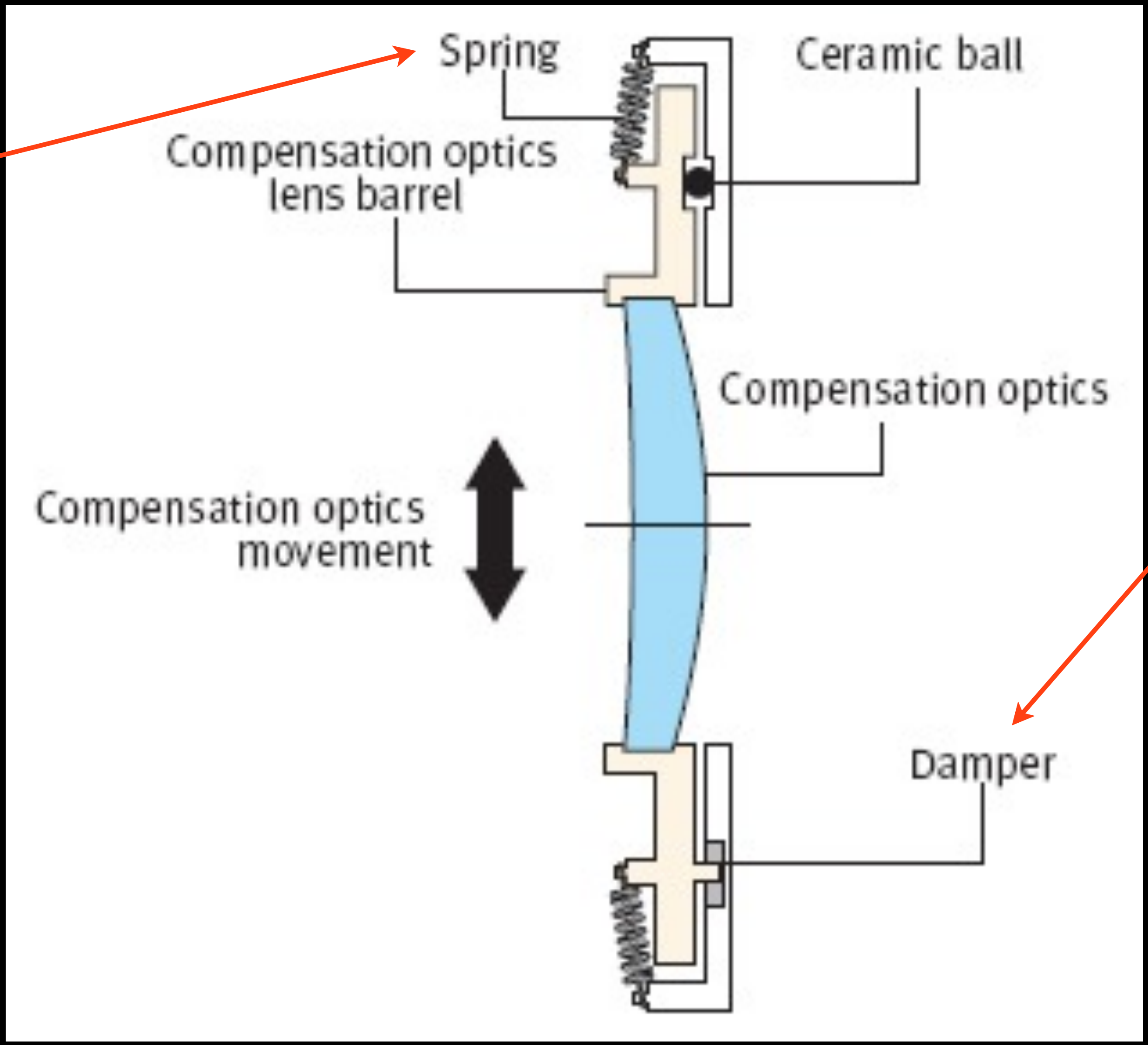
Lens-Shift Image Stabilization



figures stolen from Sung Hee Park

Lens-Shift Image Stabilization

Springs suspends the compensation optics assembly.



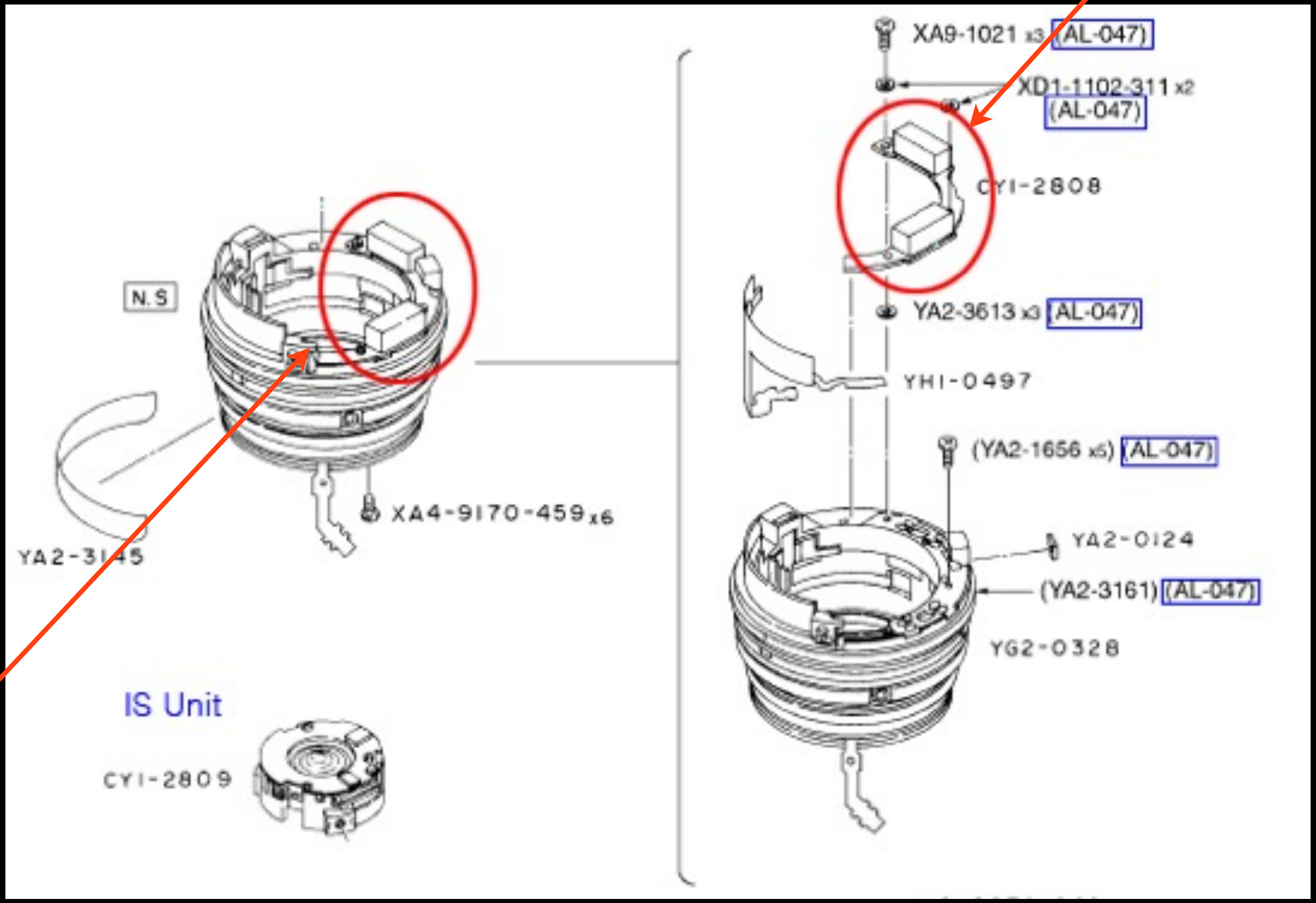
Resin damper dampens strong vibration

Canon EF-S 18-55mm IS

figures stolen from Sung Hee Park

Lens-Shift
Image Stabilization

Sensing rate: 100-150 Hz
Handshake: 10-20 Hz

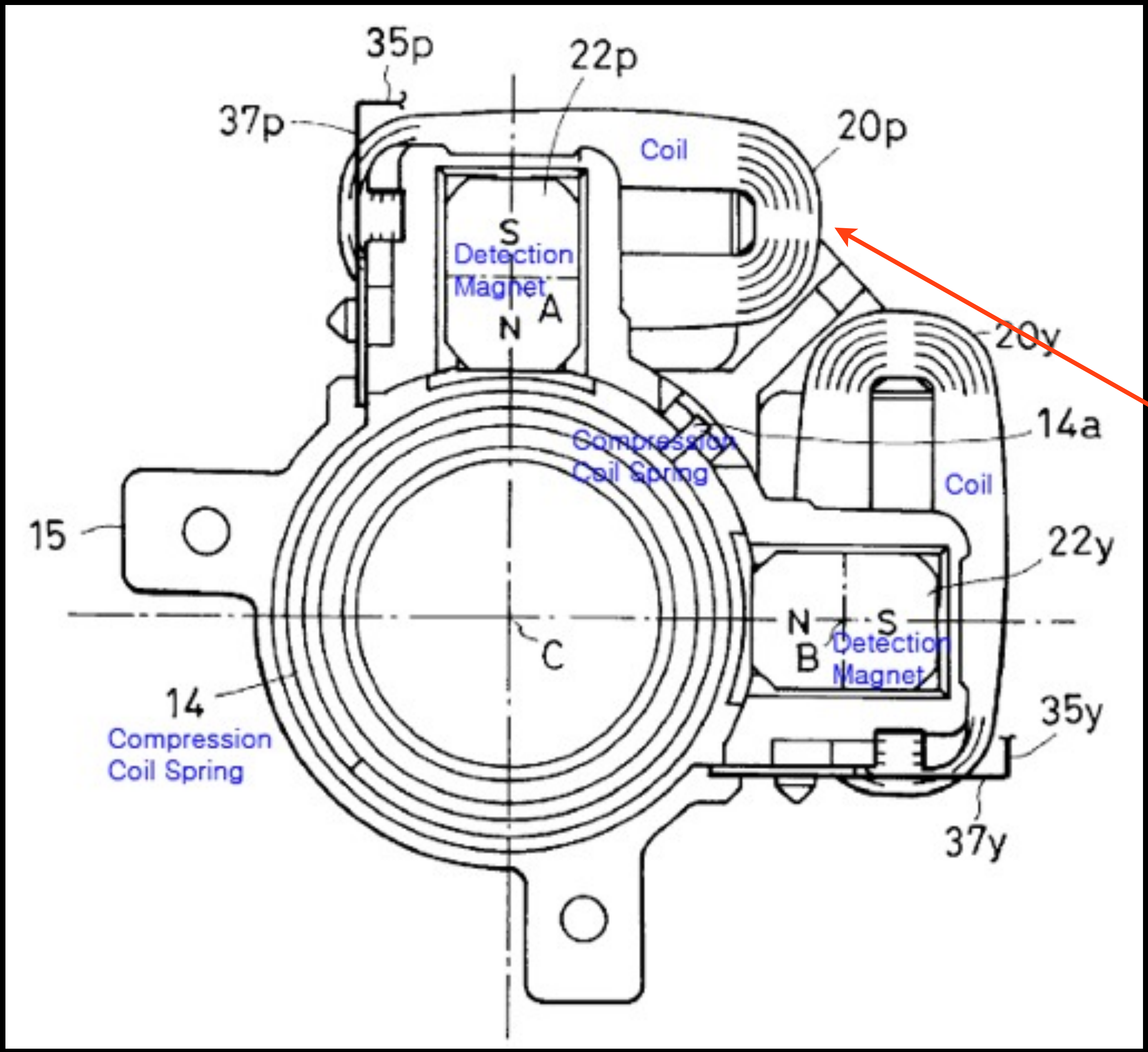


Gyroscopes, not accelerometers, are used. (Decouple linear motion)

Canon EF 28-135mm IS USM

figures stolen from Sung Hee Park

Lens-Shift Image Stabilization

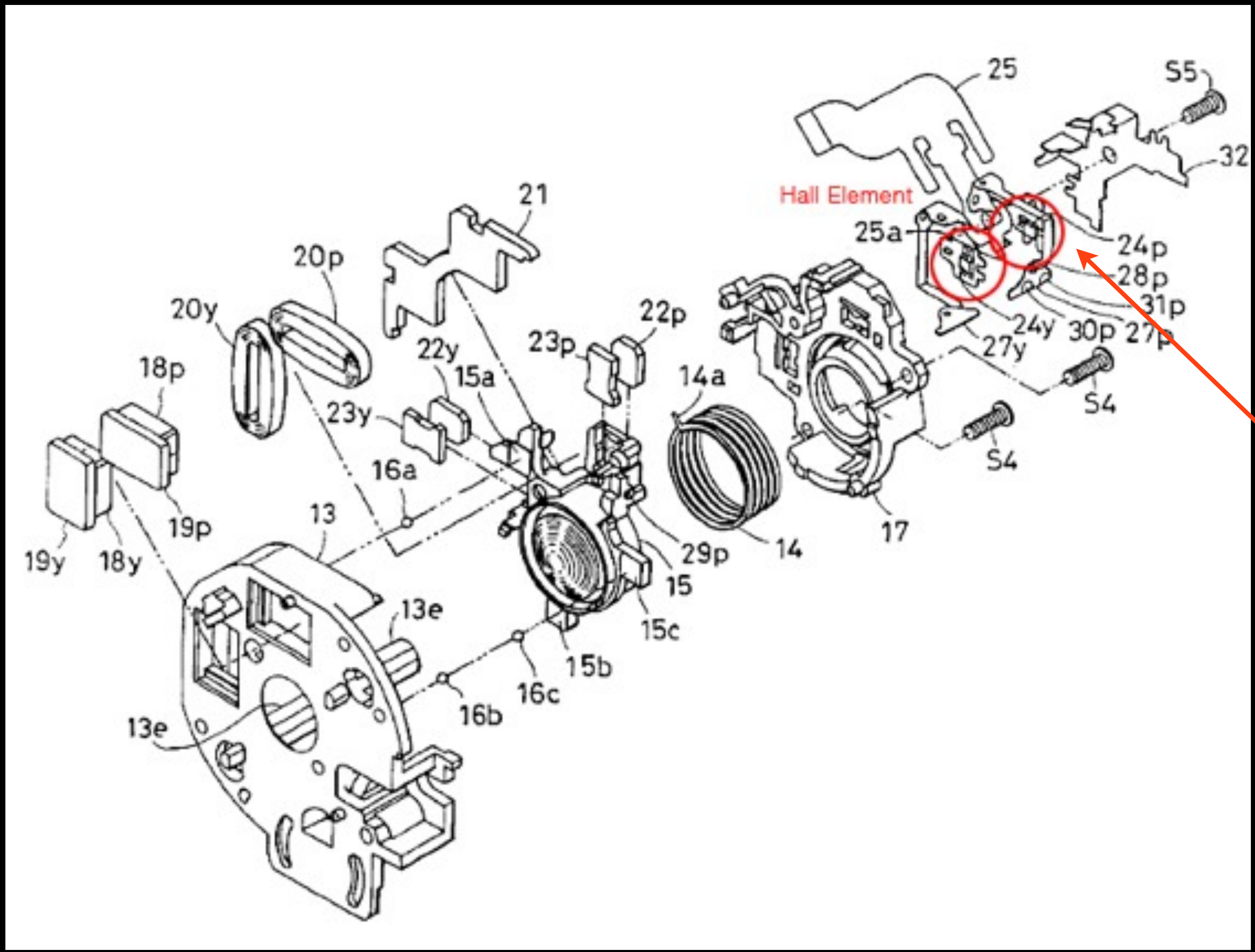


Two voice coils are used for actuation.

Canon EF 28-135mm IS USM

figures stolen from Sung Hee Park

Lens-Shift Image Stabilization



Hall Sensors: varies output voltage in response to change in magnetic field (feedback into control system)

Canon EF 28-135mm IS USM

figures stolen from Sung Hee Park

Lens-Shift Image Stabilization

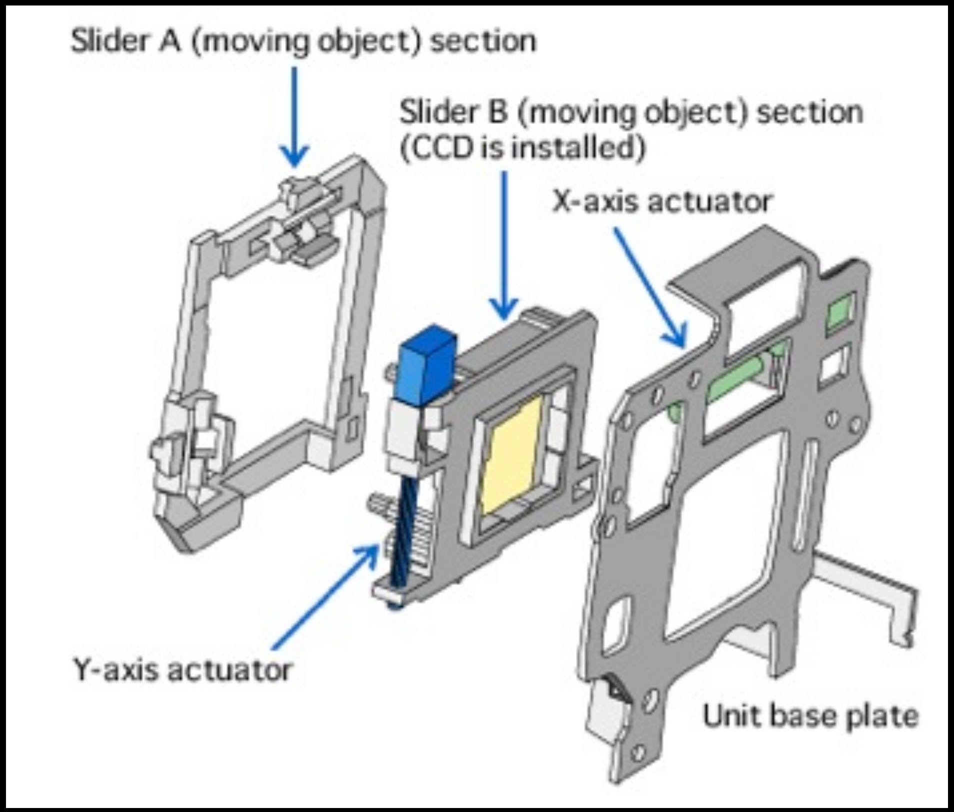
- Video
  - http://www.dpreview.com/reviews/konicaminoltaa2/Images/asmovie.mov

Sensor-Shift Image Stabilization

- Lots of different names, again
  - Anti Shake (Minolta)
  - Super Steady Shot (Sony)
  - Shake Reduction (Pentax)
  - Image Stabilization (Olympus)

content stolen from
Sung Hee Park

Sensor-Shift Image Stabilization



figures stolen from
Sung Hee Park

Sensor-Shift Image Stabilization

Use piezoelectric supersonic linear actuator
(small, precise and responsive.)

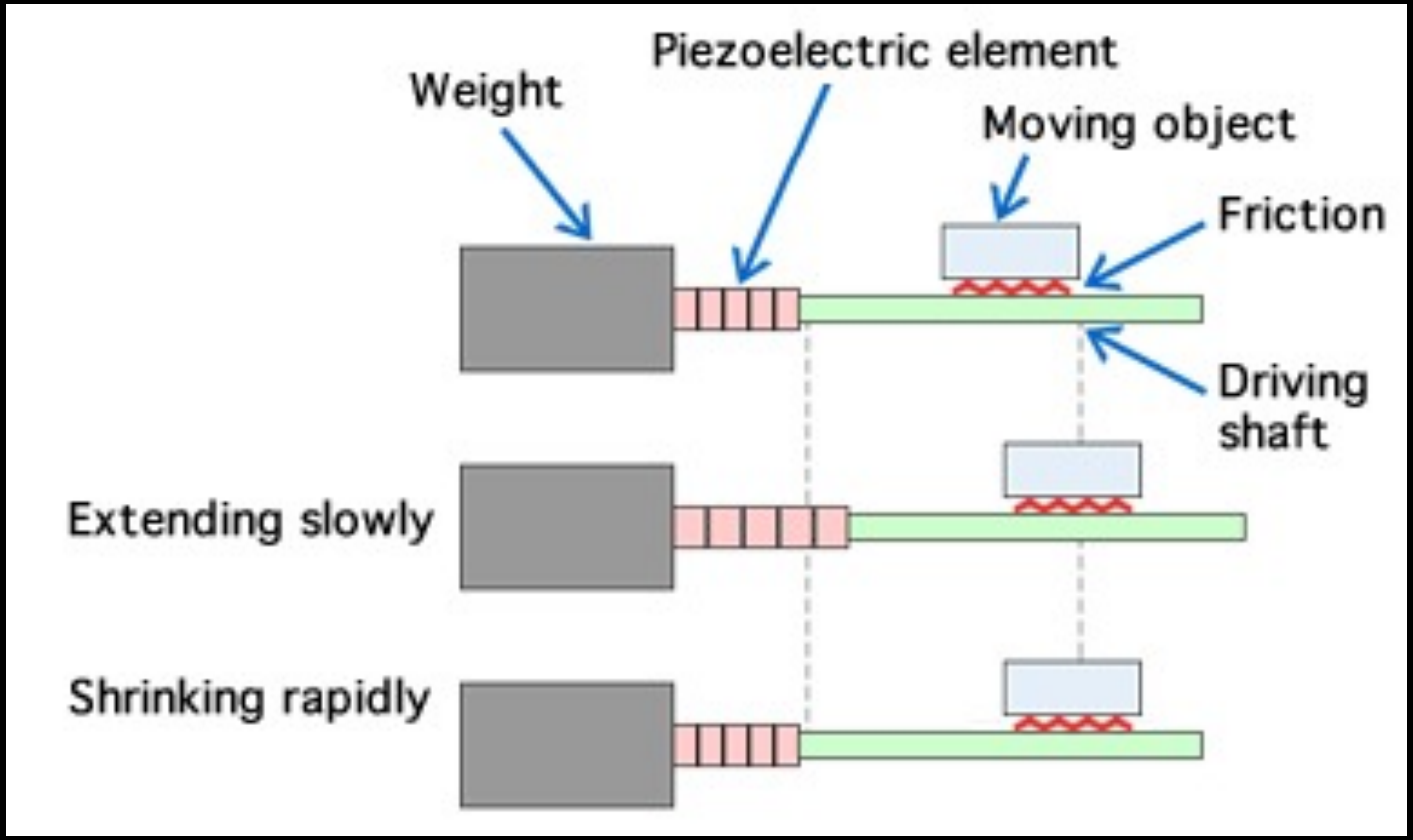


figure stolen from
Sung Hee Park

Sensor-Shift Image Stabilization

- Video
  - http://gizmodo.com/optical-image-stabilizer

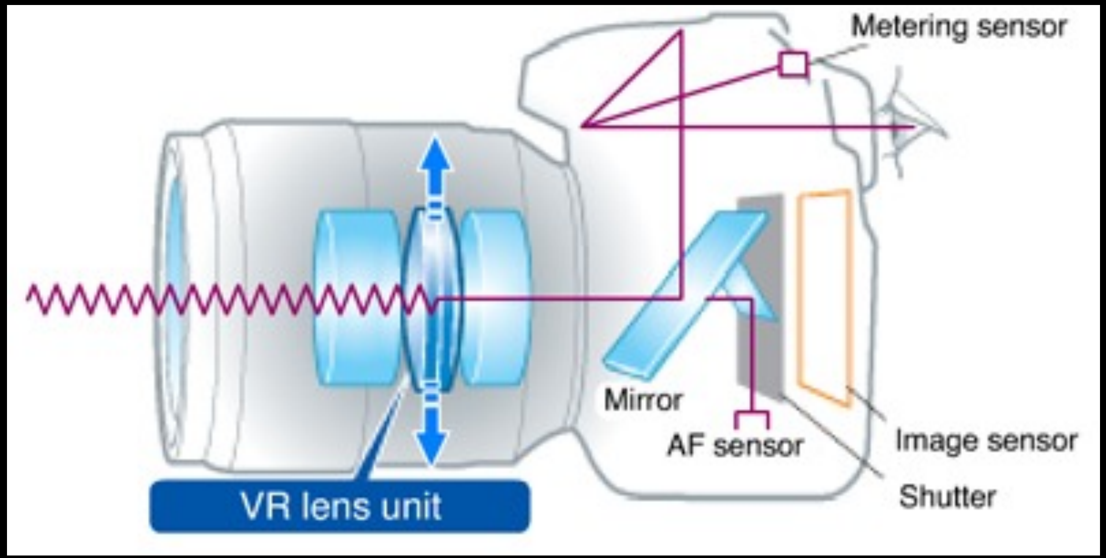
Lens-Shift vs. Sensor-Shift

Lens-Shift

- Stable viewfinder
- Better AF/AW
- Optimized to every lens

Sensor-Shift

- Works for all lens
- Cost-effective
- Better optical performance



figures stolen from
Sung Hee Park

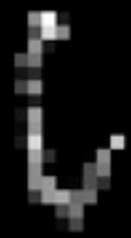
Digital Stabilization

- What if you already incurred blur?
  - Need to "remove" blur

Image Formation

- $I = L \otimes K + N$
  - I : Observation
  - L : Latent image
  - K : Blur kernel
  - N : Noise

L



K

N

I

Image Formation+

Spatially varying blur

- $I = \sum_i (L \otimes K_i \;.*\; M_i) + N$
  - I : Observation
  - L : Latent image
  - K_i : (Many) Blur kernels
  - M_i : Influence map, $\sum_i M_i = 1$
  - N : Noise

Will only discuss spatially-invariant blur for now.

Non-Blind Deconvolution

- $I = L \otimes K + N$
  - I : Observation
  - L : Latent image
  - K : Blur kernel
  - N : Noise

Known

K

I

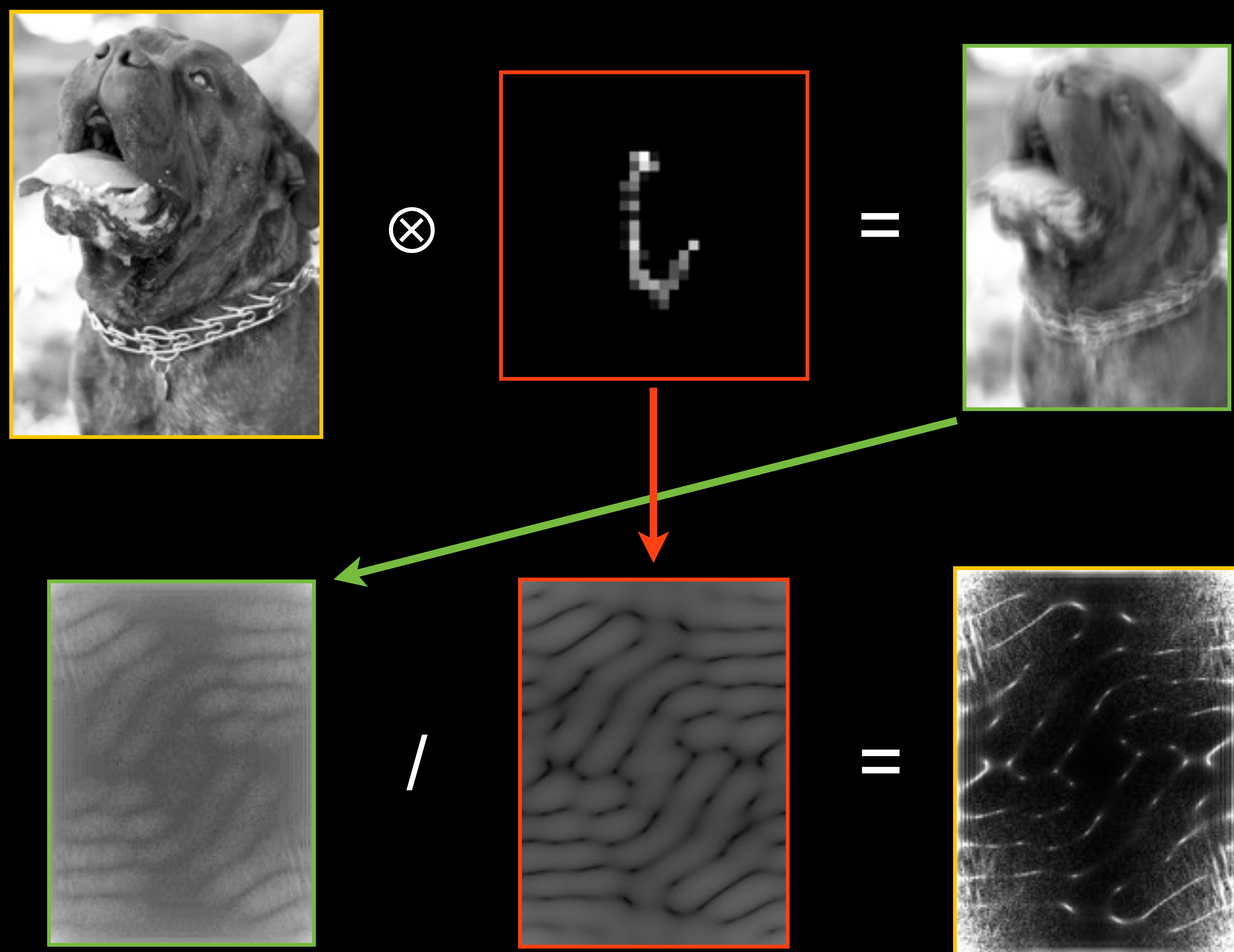
Unknown

L

N

Fourier-Domain Division

Assume no noise.



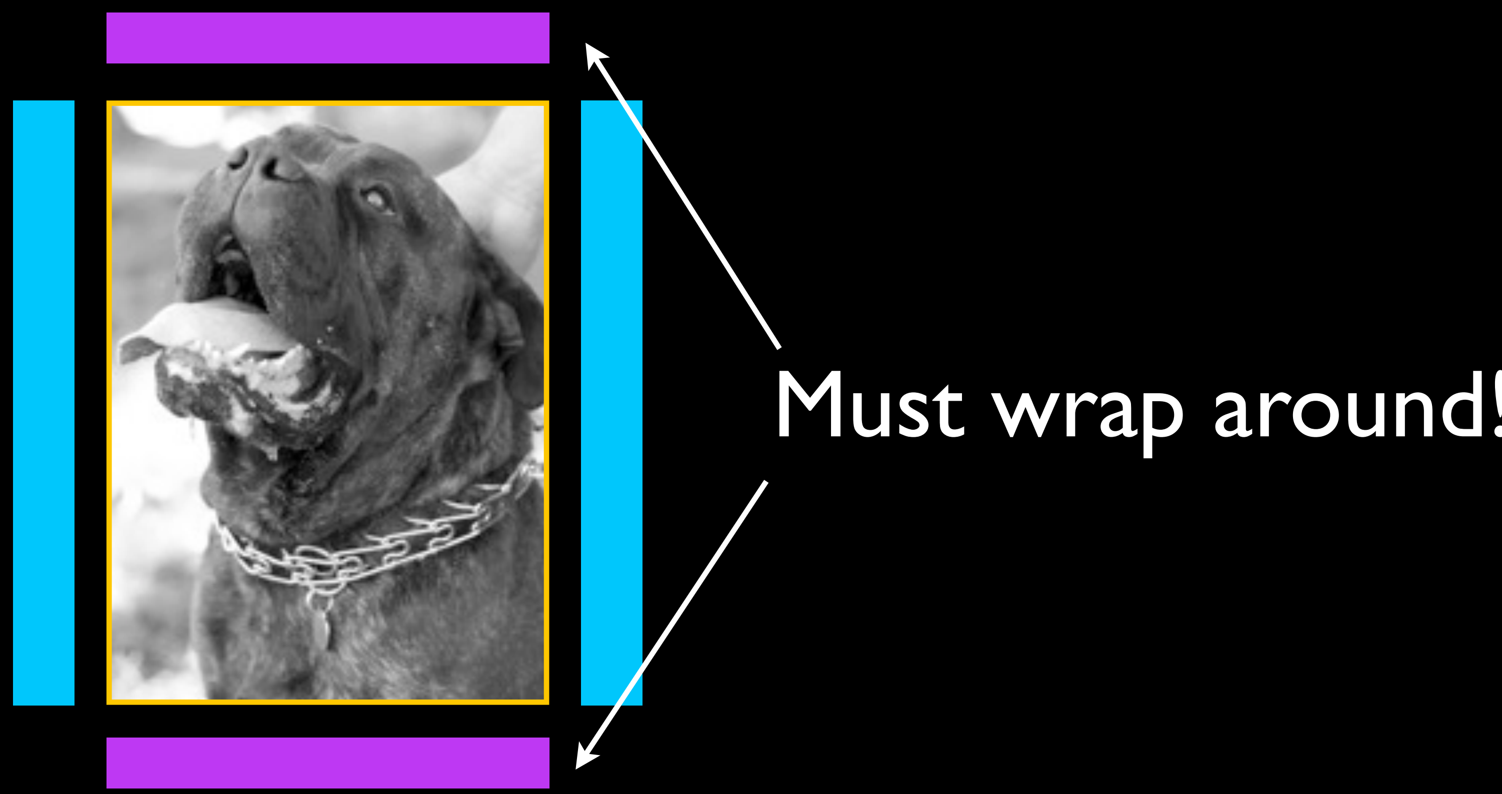
Fourier-Domain Division

Assume no noise.

What went wrong?

Fourier-Domain Division

- Assume periodic signal.
  - Often incorrect.

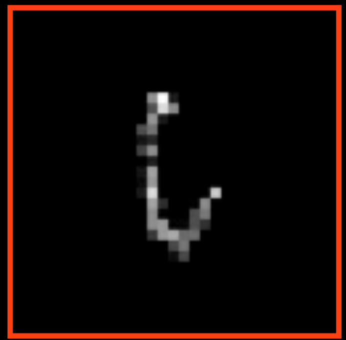


Often fixed by clever padding

Fourier-Domain Division

Try again with periodic image.

$\otimes$

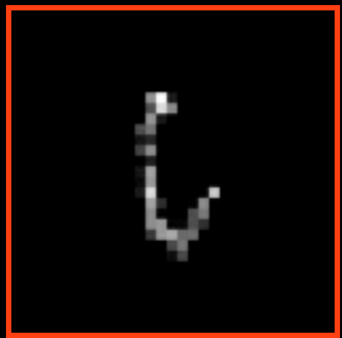


$=$

Looks good!

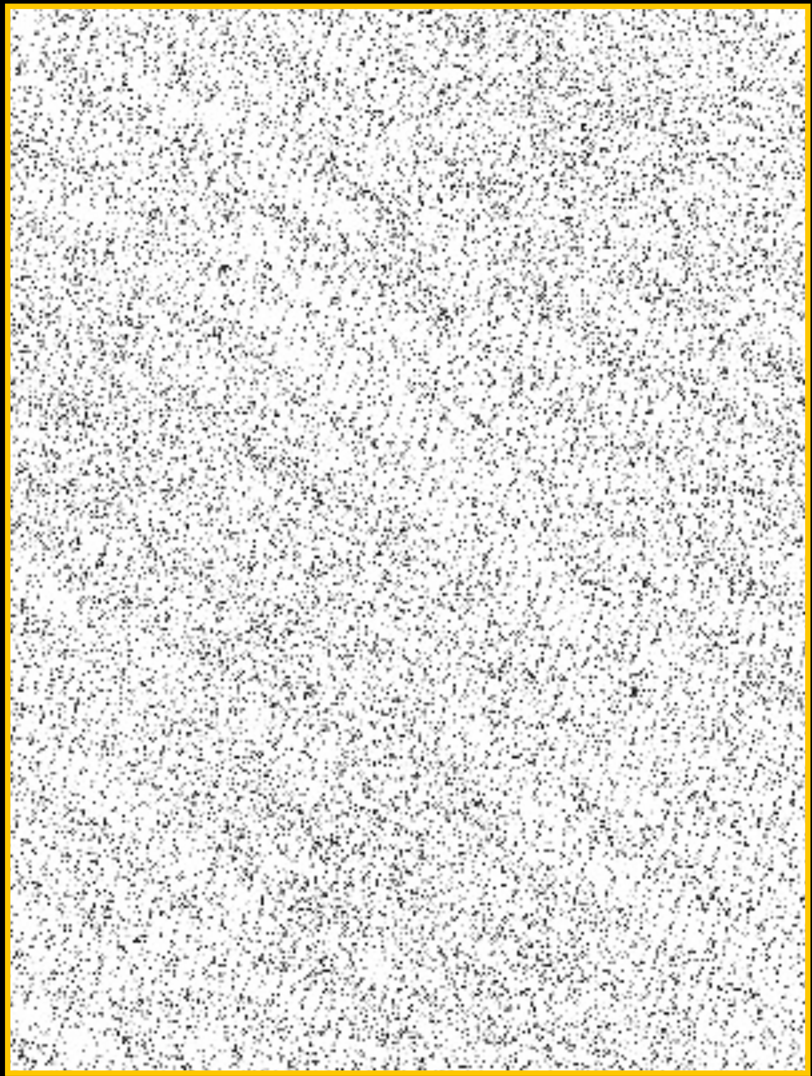
Fourier-Domain Division

Add some noise?

 ⊗  + $\sigma=0.1$ = 

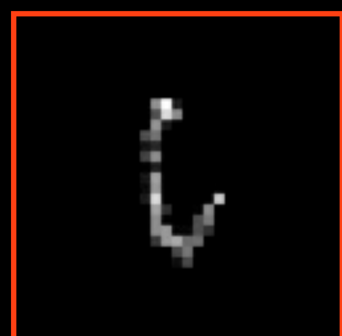
No noise

$\sigma=0.1$

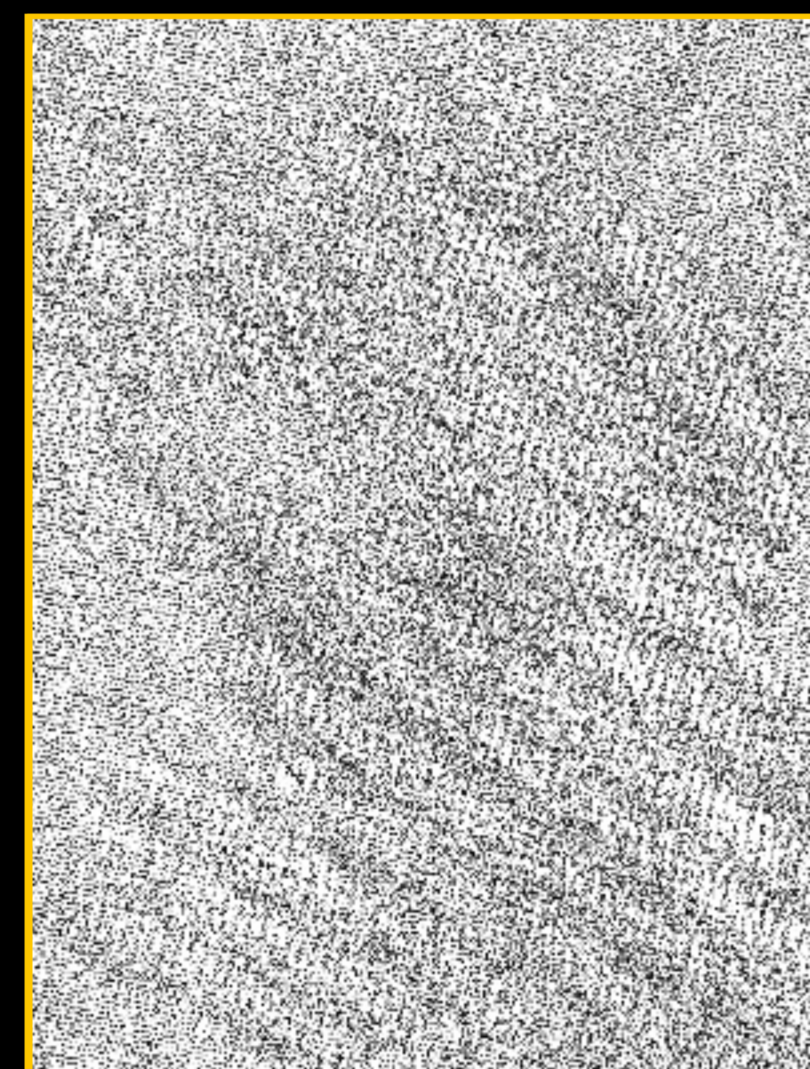


Fourier-Domain Division

Add some noise?

 $\otimes$  + $\sigma=0.04$ = 

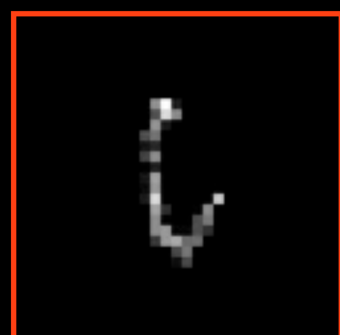
$\sigma=0.04$



Fourier-Domain Division

Add some noise?

$\otimes$

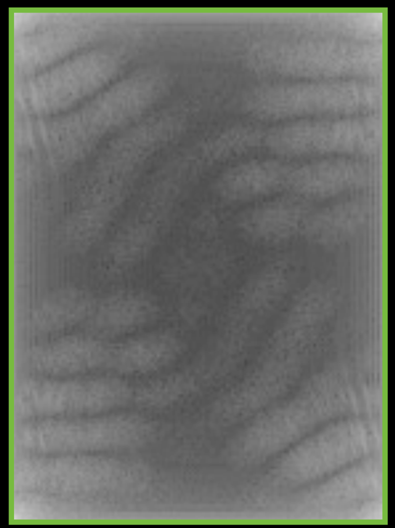


$+$ $\sigma=0.01$ $=$

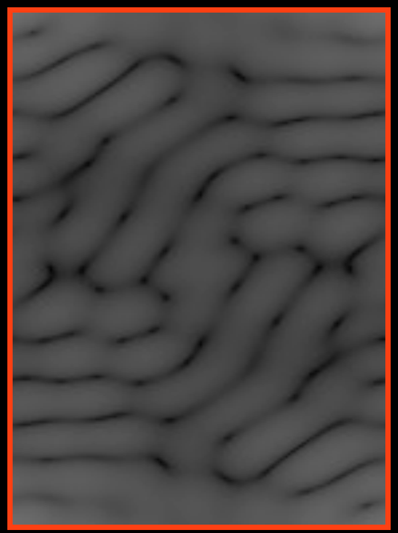
$\sigma=0.04$

Fourier-Domain Division

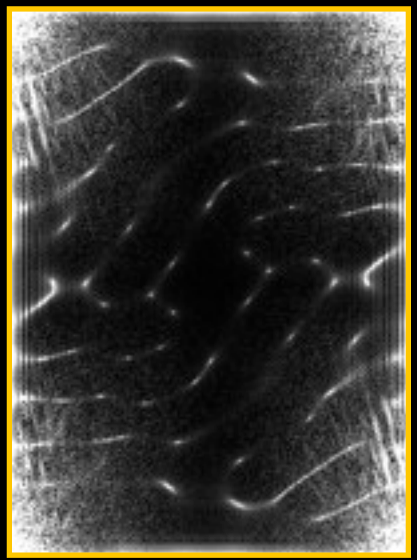
- Dividing by zero is bad.
  - Especially when the numerator is corrupted by noise!



/



=



MAP Estimate

- $I = L \otimes K + N$
- Solve for the maximum likelihood (L)
  - $\log P(L, K \mid I) =$

$$\lambda_1 h(I - L \otimes K) + \lambda_2 f(L)$$

Data Term (typically square-norm)

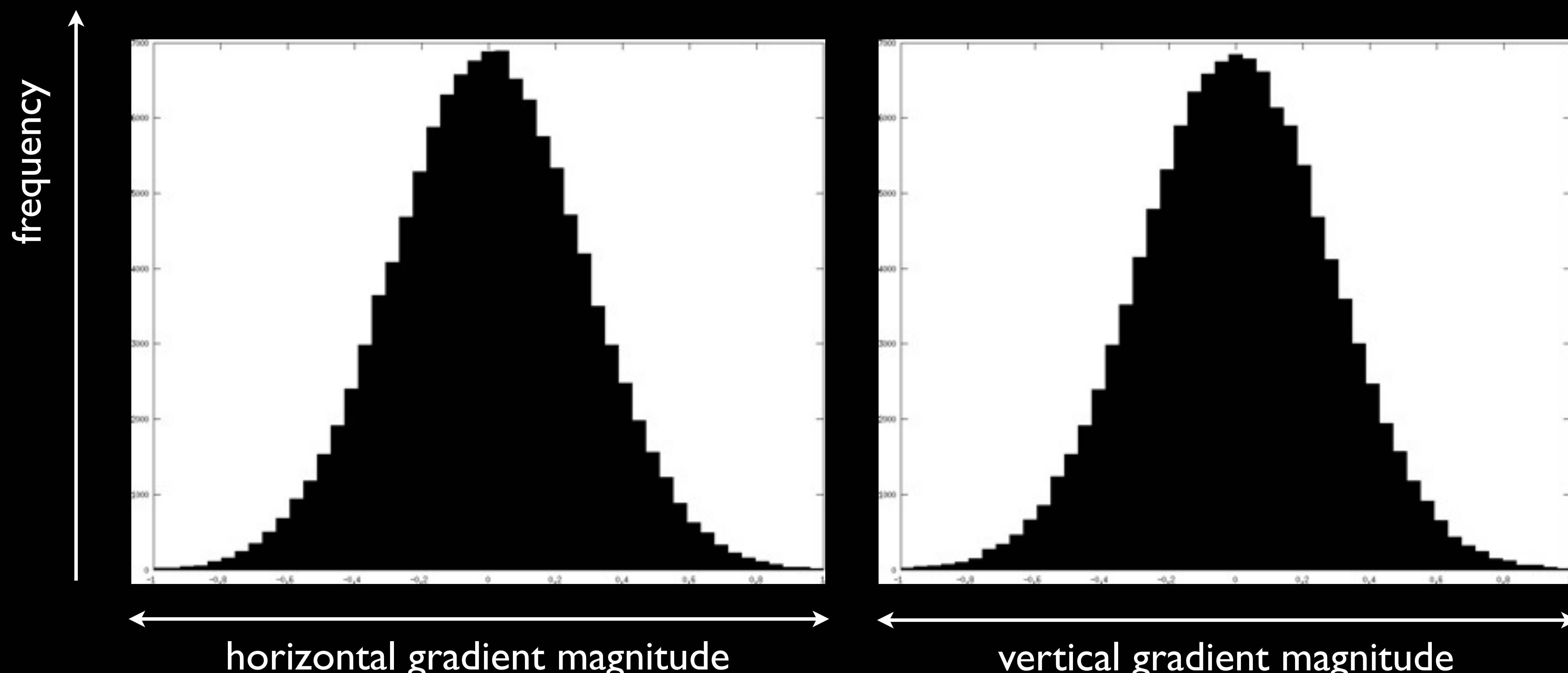
Image Prior

Image Priors

- f(L): should be high for natural images, and low for others.
  - Often based on sparsity of gradients.

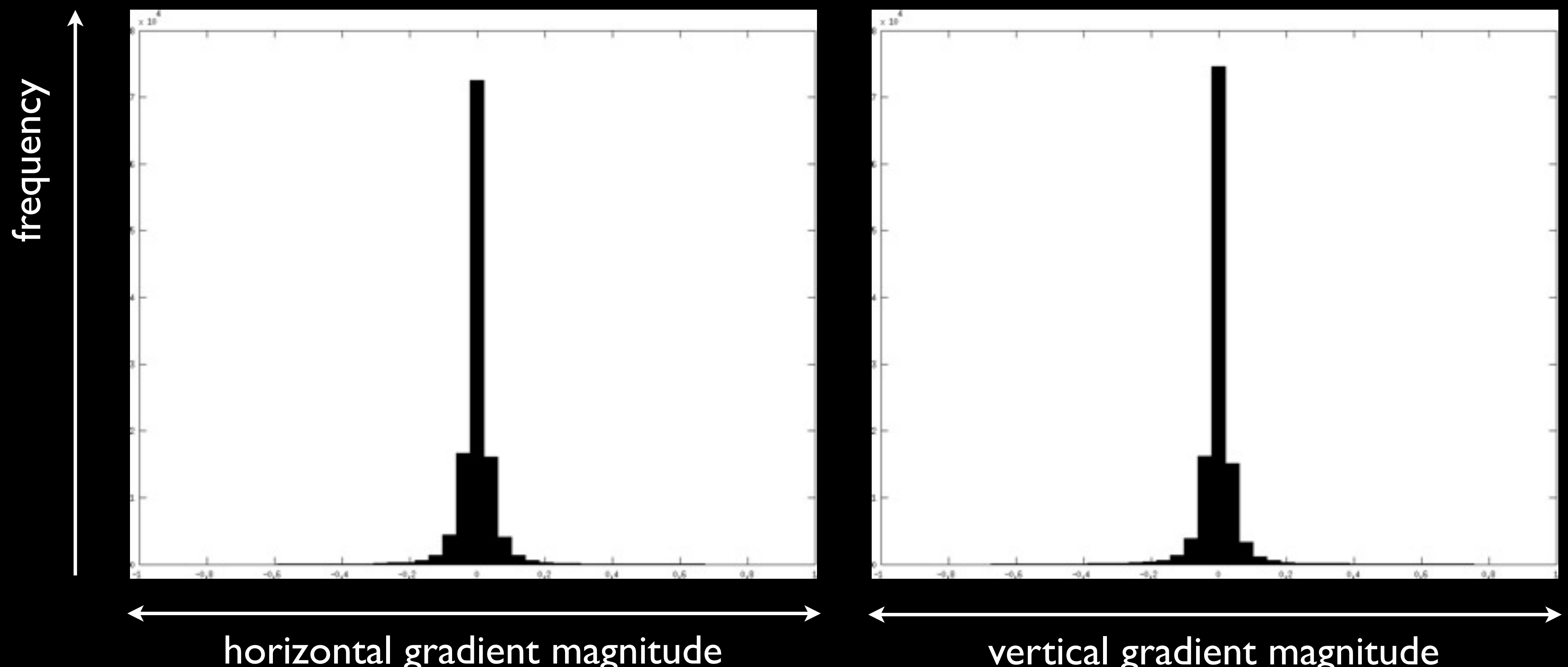
Gradient Statistics

- Noise has plenty of high-magnitude gradients.



Gradient Statistics

- Natural images often have mostly zero gradients.
  - Perhaps we could penalize high gradients?



Gaussian Prior

- Each gradient follows (independently) a Gaussian distribution.
  - Probability of gradient magnitude g:
    - $\mathrm{Prob}(g) = \exp\{ -|g|^2 / 2\sigma^2 \}$
  - Log-likelihood:
    - $f(g) \propto -|g|^2$
    - $f(L) \propto -\sum_{x,y} \nabla L^2 = -\sum_{x,y} (L \otimes d_x)^2 + (L \otimes d_y)^2$

The higher gradient, the less plausible it is!

Gaussian Prior

- Log-likelihood:
  - $f(L) \propto -\sum_{x,y} (L \otimes d_x)^2 + (L \otimes d_y)^2$
- Parseval's relation:
  - $f(L) \propto -\sum |F\{L\}\, F\{d_x\}|^2 + |F\{L\}\, F\{d_y\}|^2$

Gaussian Prior

- Hence, we solve for L that minimizes:
  - $\lambda_1 | F\{I\} - F\{L\} F\{K\} |^2 + \lambda_2(|F\{L\} F\{d_x\}|^2 + |F\{L\} F\{d_y\}|^2)$
- Component-wise quadratic minimization.
  - Easy.
  - $F\{L\} = \lambda_1 F\{I\} F^*\{K\}$ divided by $\lambda_1 |F\{K\}|^2 + \lambda_2(|F\{d_x\}|^2 + |F\{d_y\}|^2))$

Gaussian Prior

λ_1=1, λ_2=0.00

λ_1=1, λ_2=0.01

$$\log P(L, K \mid I) = \lambda_1 h(I - L \otimes K) + \lambda_2 f(L)$$

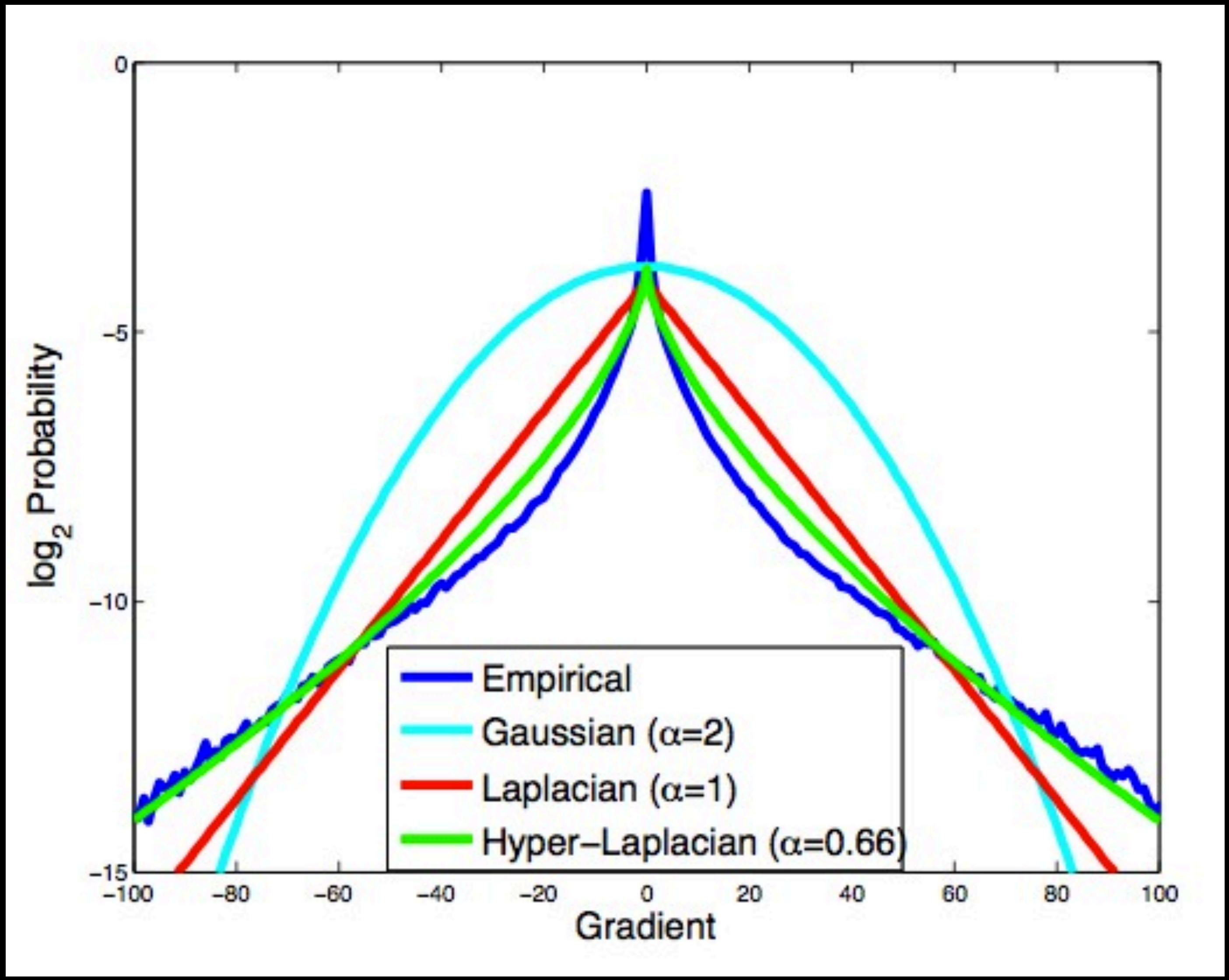
Gaussian Prior

- Just a tiny bit of prior helps regularize!
  - Not quite perfect, though.
    - Ringing artifact
    - Still some noise.

Sparse Prior

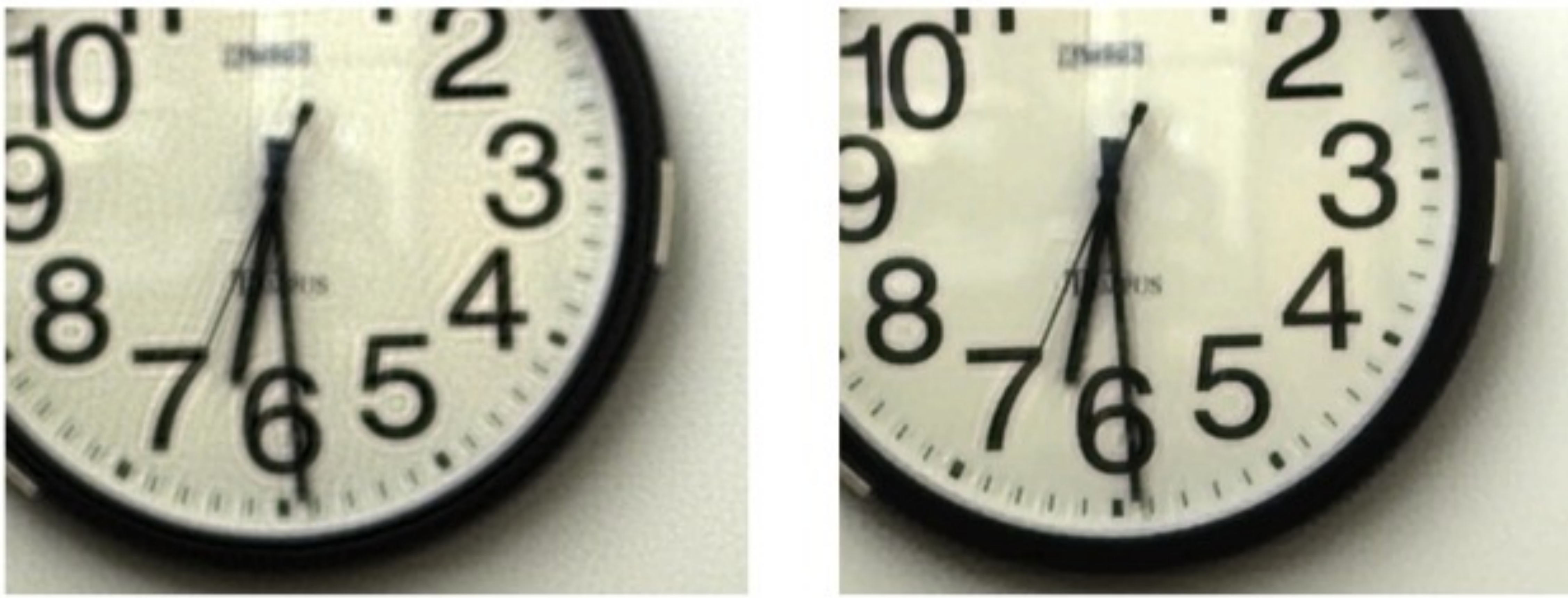
- Each gradient follows (independently) a hyper-Laplacian distribution.
  - Probability of gradient magnitude g:
    - $P(g) = \exp\{ -|g|^{\alpha} / 2\sigma^2 \}$ where $0<\alpha\leqq 1$
  - Log-likelihood:
    - $f(g) \propto -|g|^{\alpha}$
    - $f(L) \propto -\sum_{x,y} |\nabla L|^{\alpha} = -\sum_{x,y} |L \otimes d_x|^{\alpha} + |L \otimes d_y|^{\alpha}$

Gaussian v. Sparse Prior



- Sparse prior is more realistic.
- Gaussian prior makes math easy.

Gaussian v. Sparse Prior



(c) Gaussian prior (d) Sparsity prior

- Sparse prior is more realistic.
- Gaussian prior makes math easy.

Gaussian v. Sparse Prior

- Toy Example
  - Consider three consecutive pixels {0, x, 1}
    - What would Gaussian prior prefer?
      - Minimize $|x-0|^2 + |1-x|^2$. ← Optimal at x=0.5
    - What would sparse prior prefer?
      - Minimize $|x-0|^\alpha + |1-x|^\alpha$, where $0<\alpha\leqq 1$.

Optimal at x=0 or x=1

Blind Deconvolution

- We have so far assumed the blur kernel is known.
  - True for coded aperture, or other calibrated blurs.
  - True if kernel can be calculated somehow.
- Most of the time, the blur is unknown.

Blind Deconvolution

- $I = L \otimes K + N$
- Solve for the maximum likelihood (L, K)
  - $\log P(L \mid K, I) =$

$$\lambda_1 h(I - \underbrace{L \otimes K}_{\text{Data Term}}) + \lambda_2 f(\underbrace{L}_{\text{Image Prior}}) + \lambda_3 g(\underbrace{K}_{\text{Kernel Prior}})$$

  - Every paper follows this recipe.

MAP Estimate: Recipe

- $\log P(L, K \mid I) =$
  $$\lambda_1 h(I - L \otimes K) + \lambda_2 f(L) + \lambda_3 g(K)$$
- Must know:
  - Relative sizes of $\lambda_1, \lambda_2, \lambda_3$
  - Data term h(...)
  - Image prior f(...)
  - Kernel prior g(...)
  - Optimization procedure

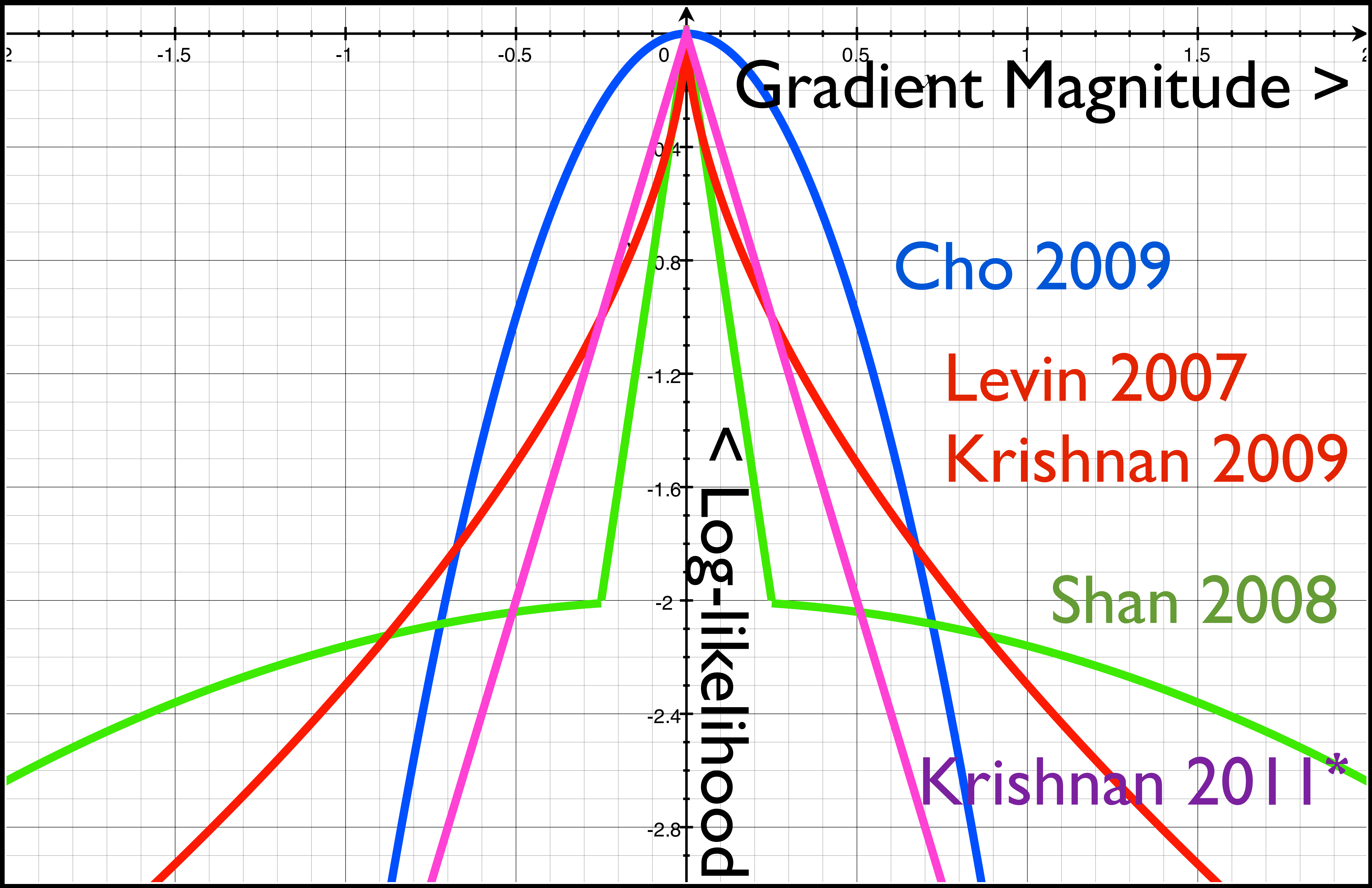
Data Term : $h(I - L \otimes K)$

- Penalize deviation from observed data.
  - $h(z) = |z|^2$ (Fergus 2005, Jia 2007, Krishnan 2010)
    - Most obvious. Corresponds to Gaussian noise
  - $h(z) = |\nabla z|^2$ (Cho 2009)
    - Cheap if you are already computing gradients.
  - $h(z) = |z|^2 + |\nabla z|^2 + \ldots$ (Shan 2008)
    - Constrain multiple orders of derivatives.

Image Prior : f(L)

- Gradients are sparse. Penalize high gradient.
  - $f(L) = \sum | d_x L |^2 + | d_y L |^2$ (Cho 2009)
  - $f(L) = \sum | d_x L |^\alpha + | d_y L |^\alpha$ (Levin 2007, Krishnan 2009)
  - $f(L) = \sum | d_x L |^\beta + | d_y L |^\beta$ (Shan 2008)
    - β=1 for small gradient, β=2 for large gradient
  - $f(L) = \dfrac{\sum | d_x L |^1 + | d_y L |^1}{(\sum | d_x L |^2 + | d_y L |^2)^{0.5}}$ (Krishnan 2010)

Image Prior : Illustration



Kernel Prior : $g(K)$

- Blur kernel is typically sparse.
  - $g(K) = \sum | d_x K |^2 + | d_y K |^2$ (Cho 2009)
  - $g(K) = \sum | d_x K |^1 + | d_y K |^1$ (Shan 2008, Krishnan 2011)
- Enforce contiguity?
  - No one seems to do this explicitly...*

Optimization

- In the end, we have an objective function in terms of L and K.
  - Quadratic in simplest form (Cho 2009)
    - Standard linear system to solve. We saw this earlier.
  - Mixture of quadratic and L1-norm (Shan 2008)
  - Highly nonlinear (Krishnan 2011)
    - Need fancier methods.

Challenges

- L and K are both unknown.
  - Solve for one, and then the other. Repeat.
- K is too loosely constrained.
  - Use coarse-to-fine scheme.
- Iterative algorithms are slow.
  - Too bad. Good luck with CG.

Generic Pseudocode

(Fergus 2005, Shan 2008, Cho 2009, Krishnan 2011)

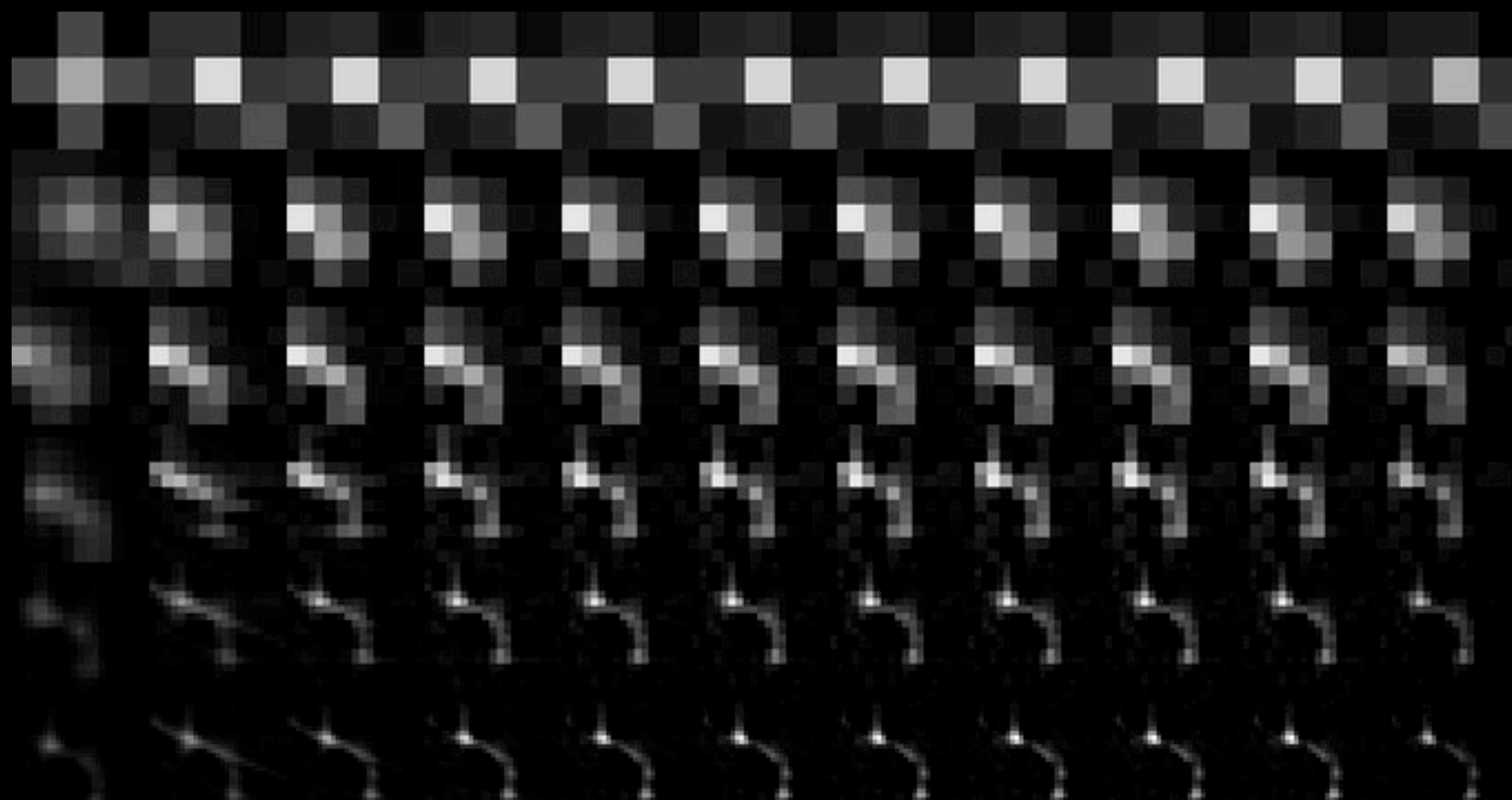
- From coarse to fine,
  - Resample L, K, I to current scale.
  - Fix L, and solve for K.
    - Typically some sort of iterative solver.
  - Fix K, and solve for L.
    - Non-blind deconvolution.

Coarse-to-Fine

True kernel

CG iterations >

Coarse-to-fine >

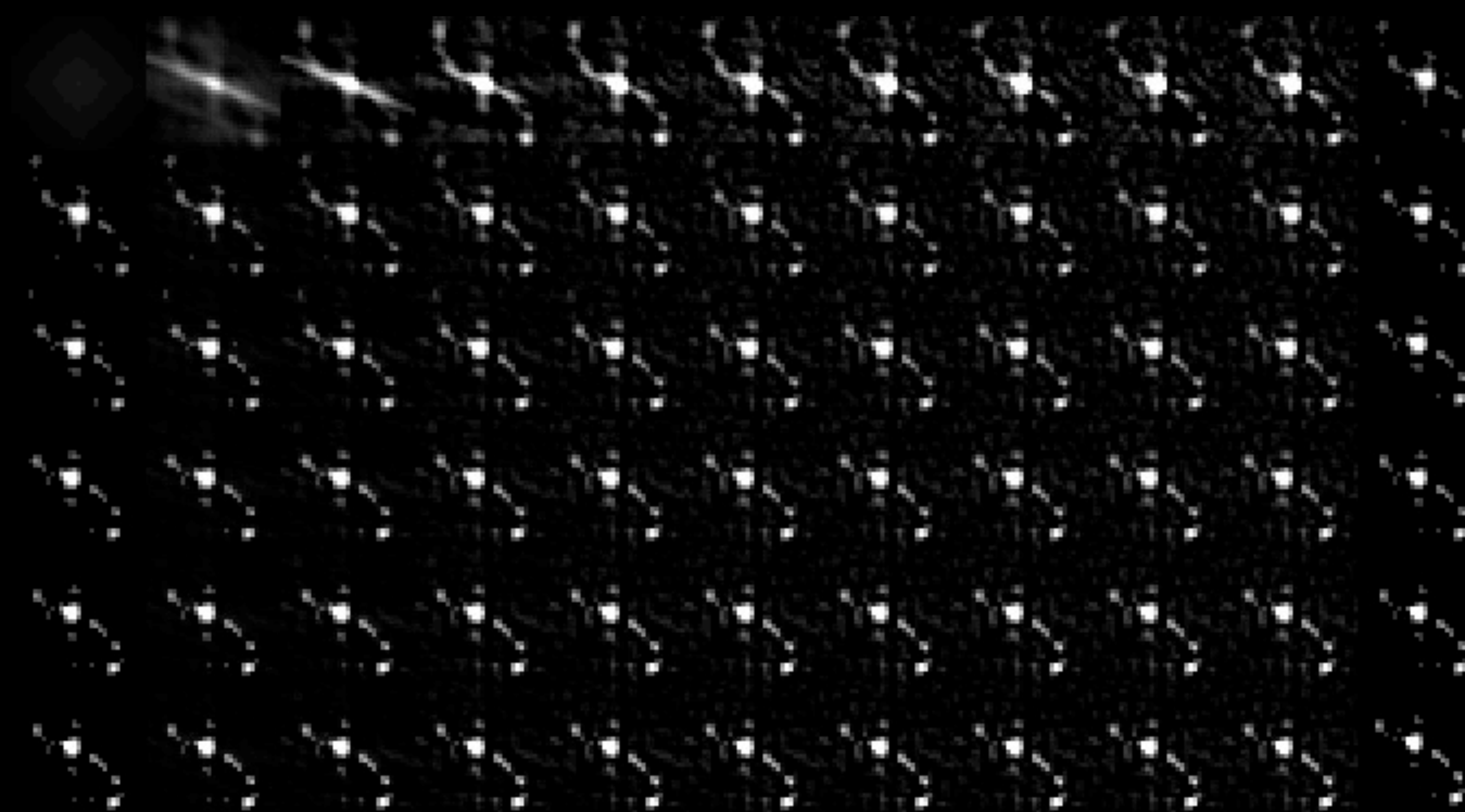


Without Coarse-to-Fine

True kernel

CG iterations >

Outer Iterations >

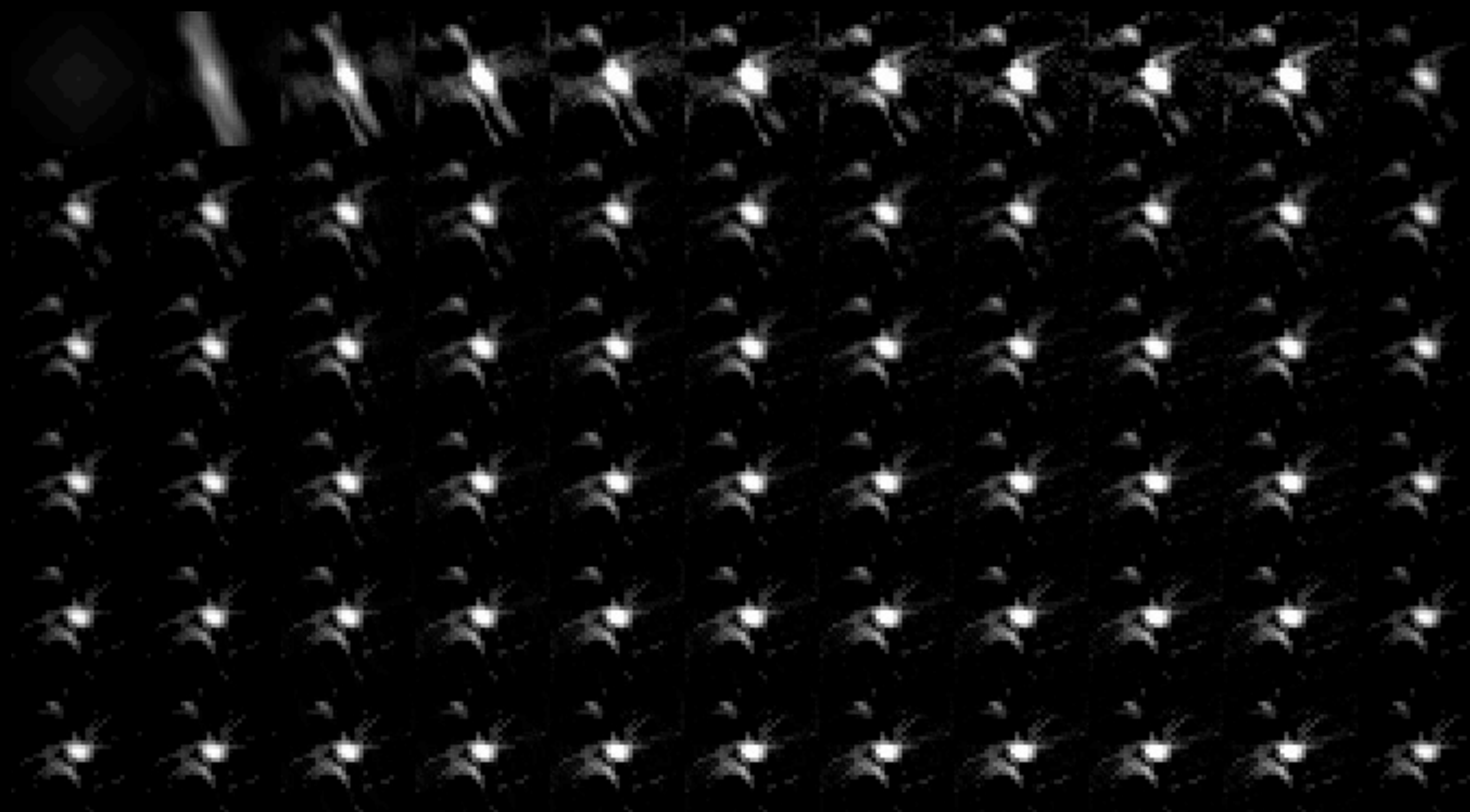


Without Coarse-to-Fine

True kernel

CG iterations >

Outer Iterations >



Case Study

- Cho and Lee, 2009
    - (Comparatively) Very fast.
    - Quality comparable to others. How?

Case Study : Cho 2009

- $\log P(L, K \mid I) = \lambda_1 h(I - L \otimes K) + \lambda_2 f(L) + \lambda_3 g(K)$
- h is quadratic.
- L is quadratic.
- K is quadratic.

Optimizer's paradise!

Pseudocode

- From coarse to fine,
  - Resample L, K, I to current scale.
  - Fix L, and solve for K.
    - Conjugate gradient.
  - Fix K, and solve for L.
    - Fourier-domain division

In Fourier domain as well

Bad. Creates ringing

Very fast

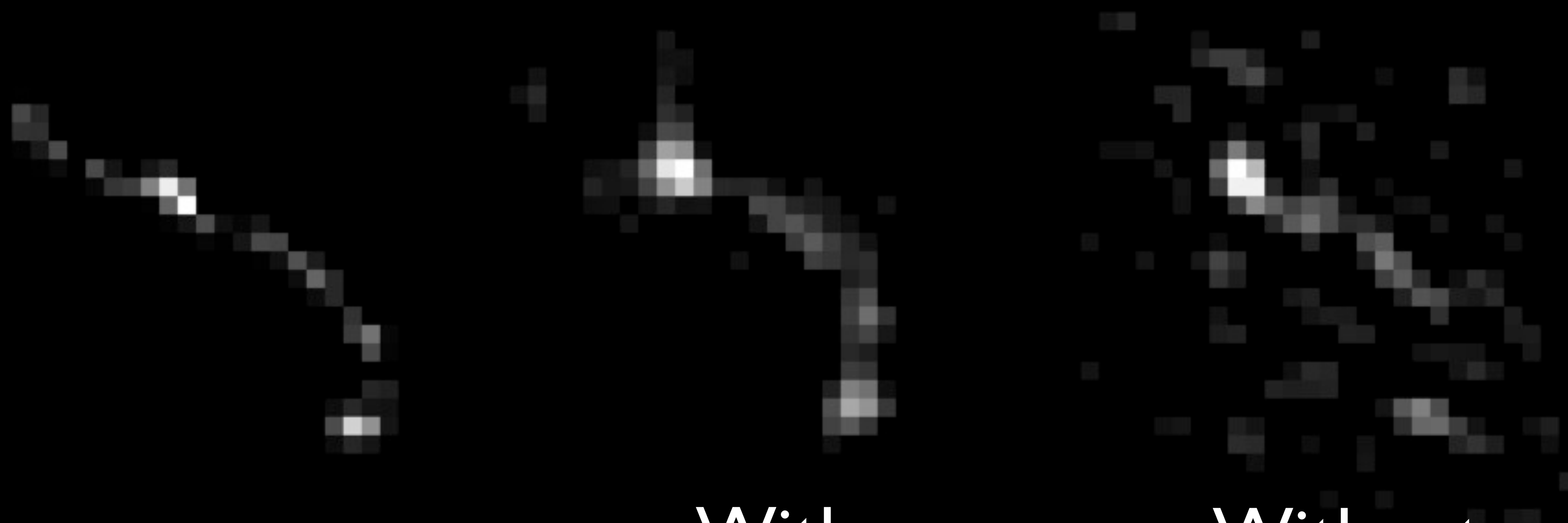
Pseudocode

- From coarse to fine,
  - Resample L, K, I to current scale.
  - Fix L, and solve for K.
  - Bilateral-filter and shock-filter L.
    - Conjugate gradient.
  - Fix K, and solve for L.
    - Fourier-domain division
- Use a nice non-blind deconv. for final result.

De-Ringing

Deconvolved result from previous scale (L)

De-Ringing



True kernel

With de-ringing

Without de-ringing

Some Results

Some Results

Some Results

Performance

Method	Implementation	Speed
Fergus 2006	Matlab	546 sec.
Shan 2008	Binary	121 sec.
Cho 2009	**Binary**	**8 sec.**
Krishnan 2011	Matlab	280 sec.

All tests on ~0.5MP images with 31x31 kernel

Parameters, Parameters

- $\log P(L, K \mid I) =$
  $$\lambda_1 h(I - L \otimes K) + \lambda_2 f(L) + \lambda_3 g(K)$$
- So, what's $\lambda_1, \lambda_2, \lambda_3$?
- St.dev for the bilateral filter?
- Time constant for shock filter?
- How to traverse coarse-to-fine?
  - Max kernel size? Step size?

Parameters, Parameters

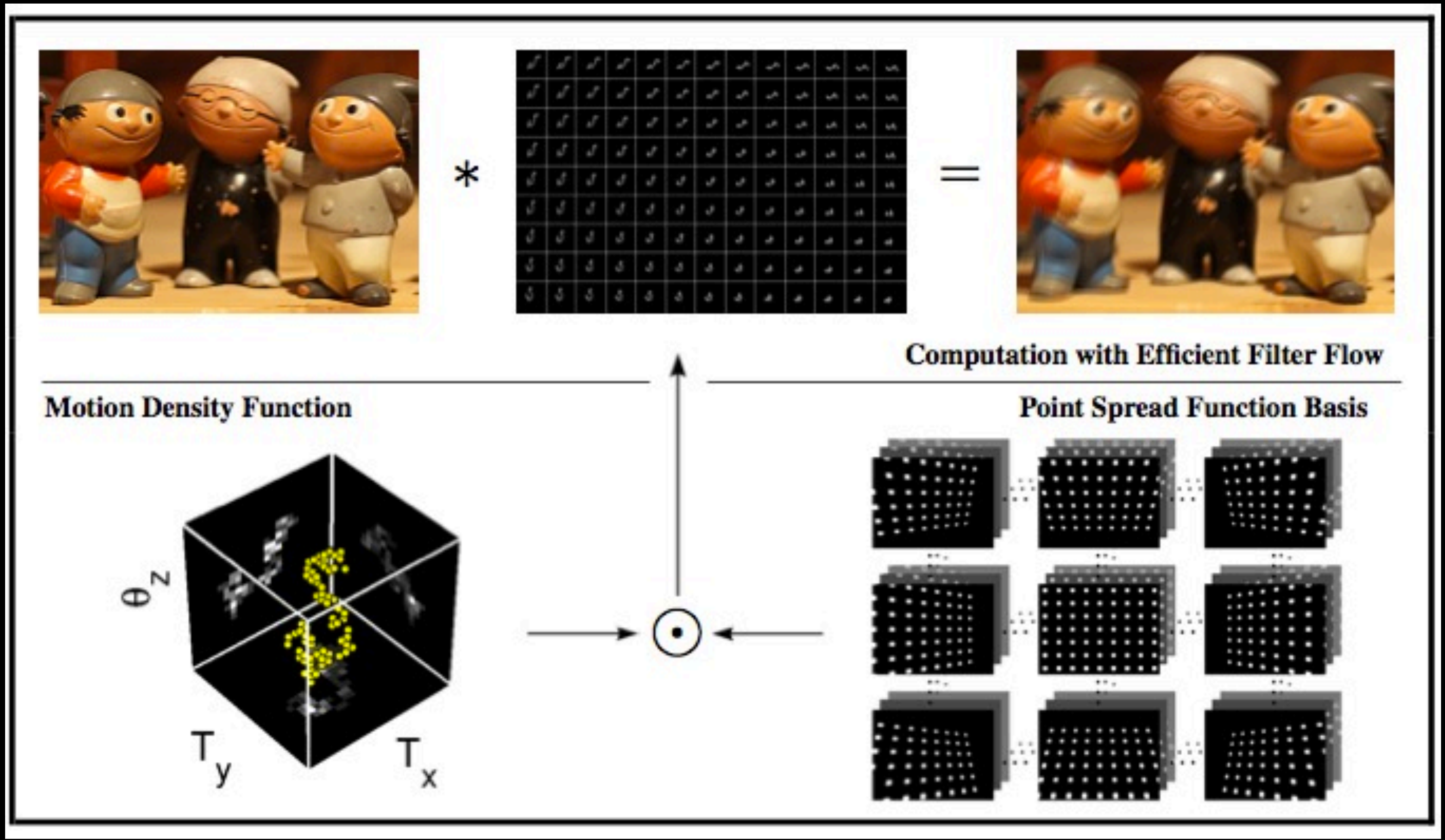
- Demo script from Shan 2008
  - deblur in1.png out1.png 27 27 0.010 0.2 1 0 0 0 0 3.5
    deblur in2.png out2.png 27 27 0.008 0.2 1 0 0 0 0 0.0
- Demo script from Cho 2009
  - deblur in1.jpg out1.jpg 49 47 0.5 0.0005
    deblur in2.jpg out2.jpg 61 43 0.5 0.0005
    deblur in3.jpg out3.jpg 33 33 0.5 0.001
    deblur in4.jpg out4.jpg 35 49 0.5 0.0005
    deblur in5.jpg out5.jpg 65 93 0.5 0.0002

Video

- First 30 seconds of
  - http://www.youtube.com/watch?v=xxjiQoTp864

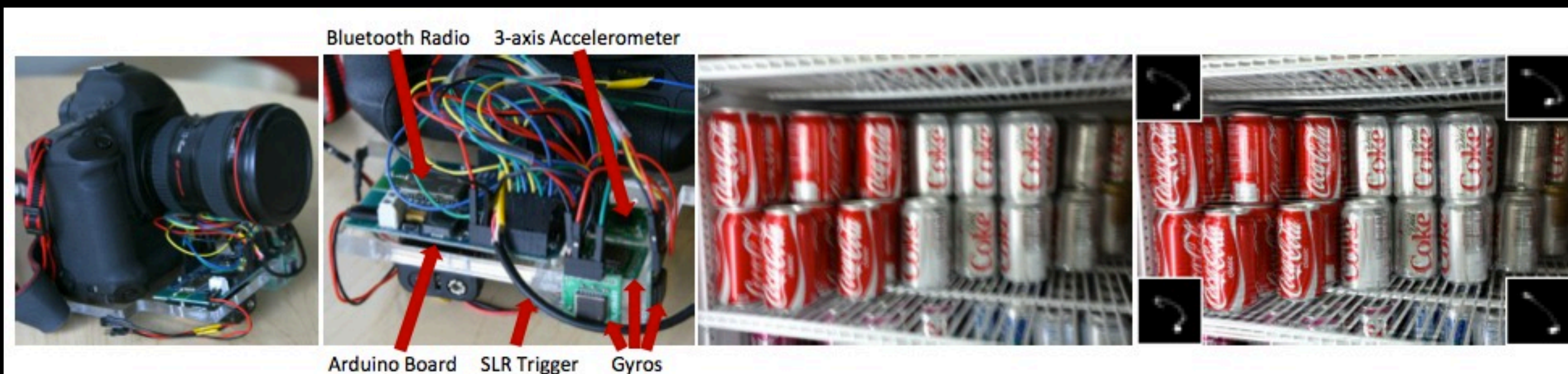
Other Twists

- Non-Uniform Blur
  - Treat as locally uniform deconvolution

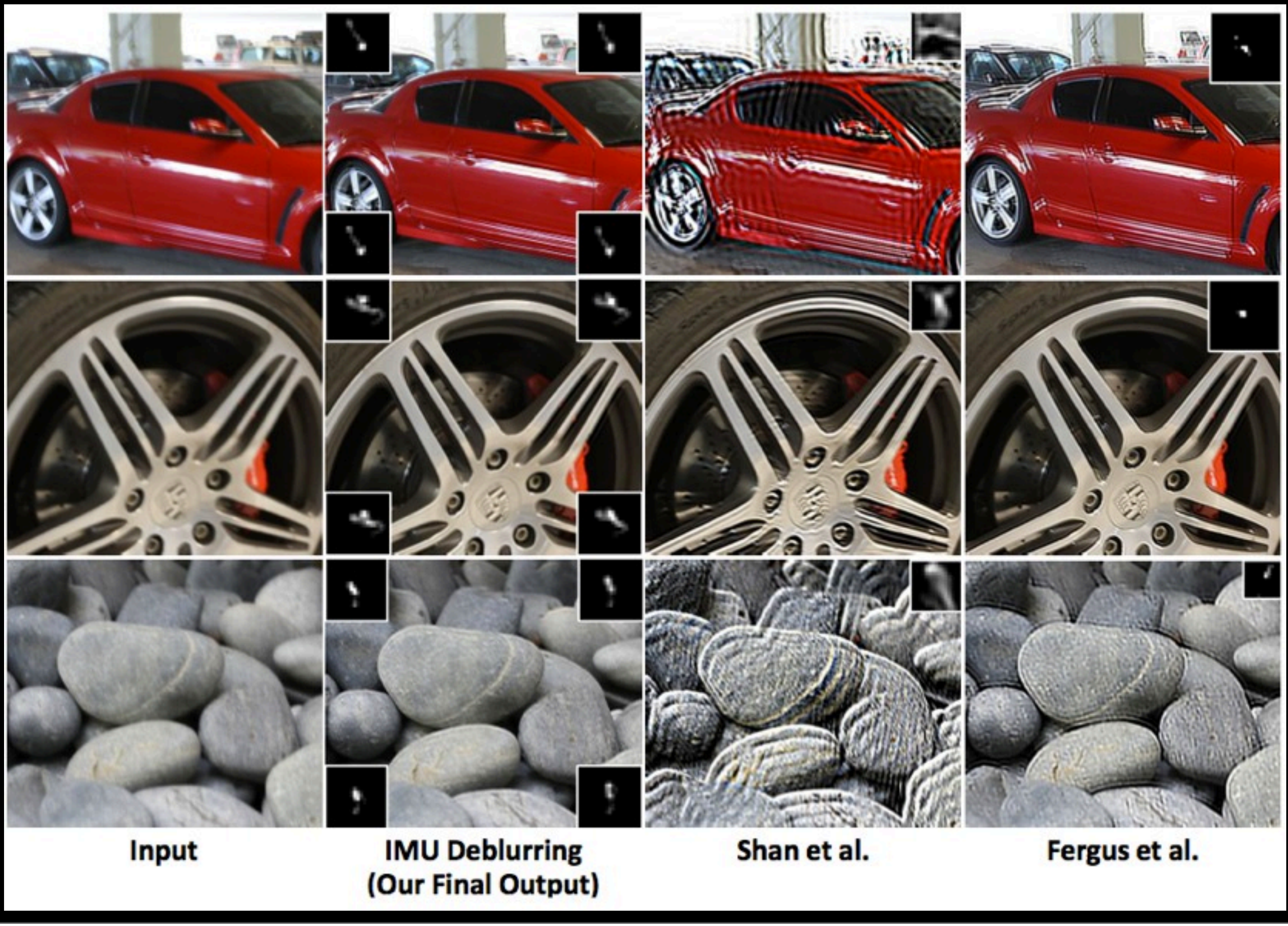


Other Twists

- Use gyros to figure out kernel



Other Twists



Alternatives

- Take a short exposure and denoise.
- Align-and-average
  - People are studying the tradeoffs now.

Questions?

References

- Camera-Motion and Mobile Imaging (Xiao et al., SPIE 2007)
- Camera-Motion and Effective Spatial Resolution (Xiao et al., ICIS 2006)
- Lens barrel having image shake correcting function and optical device having same (Noguchi, U.S. Patent 2003. #6,631,042, 2003
- Canon_EF_mount (Wikipedia) - http://en.wikipedia.org/wiki/Canon_EF_mount
- Image_stabilization (Wikipedia) - http://en.wikipedia.org/wiki/Image_stabilization
- Canon EOS 40D White Paper - http://www.usa.canon.com/uploadedimages/FCK/Image/White%20Papers/EOS_40D_WP.pdf
- Canon Camera Museum - http://www.canon.com/camera-museum/index.html
- Image Stabilization Technology Overview - http://www.invensense.com/shared/pdf/ImageStabilizationWhitepaper_051606.pdf
- Vibration Reduction - http://imaging.nikon.com/products/imaging/technology/vr
- Piezoelectric supersonic linear actuator - http://www.konicaminolta.com/about/research/core_technology/picture/antiblur.html
- Olympus E-System Technology - http://www.olympusamerica.com/cpg_section/oima_slr_technology.asp
- Pentax K100D Shake Reduction Technology - http://www.pentaximaging.com/files/scms_docs/SHAKE_REDUCTION_FACT_SHEET.pdf

References

- Removing Camera Shake from a Single Photograph (Fergus et al., SIGGRAPH 2006)
- Image and Depth from a Conventional Camera with a Coded Aperture (Levin et al., SIGGRAPH 2007)
- High Quality Motion Deblurring from a Single Image (Shan et al., SIGGRAPH 2008)
- Fast Motion Deblurring (Cho and Lee, SIGGRAPH Asia 2009)
- Fast Image Deconvolution using Hyper-Laplacian Priors (Krishnan and Fergus, NIPS 2009)
- Non-Uniform Deblurring for Shaken Images (Whyte et al., CVPR 2010)
- Image Deblurring using Inertial Measurement Sensors (Joshi et al., SIGGRAPH 2010)
- Blind Deconvolution Using a Normalized Sparsity Measure (Krishnan et al., CVPR 2011)
- Fast Removal of Non-Uniform Camera Shake (Hirsch et al., ICCV 2011)