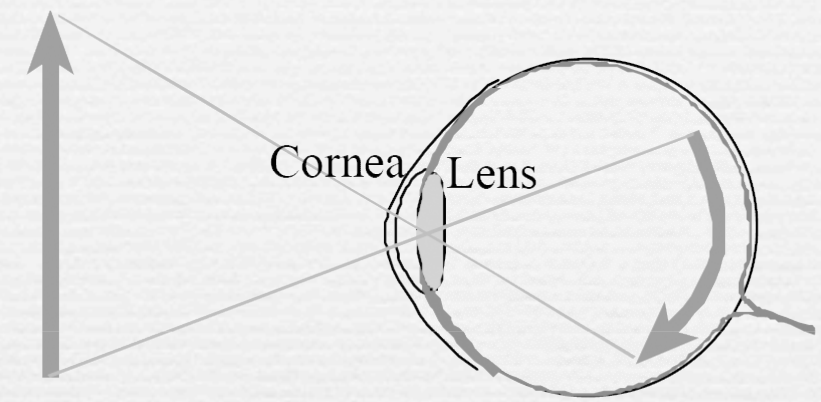
Computation is the New Optics

Frédo Durand
MIT CSAIL

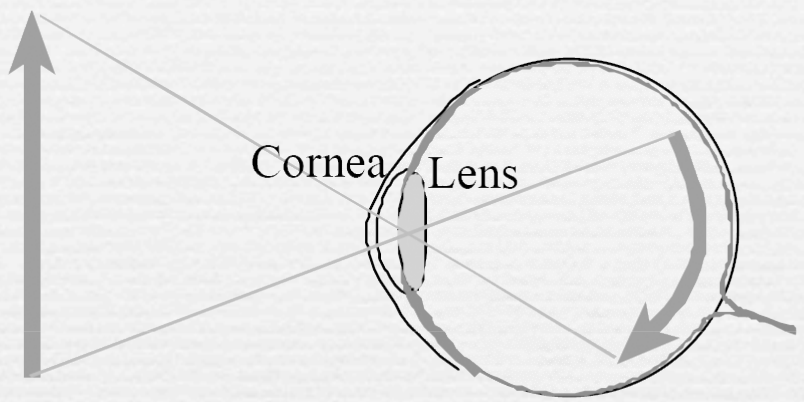
Computation is the new optics

- Naked eye viewing



Computation is the new optics

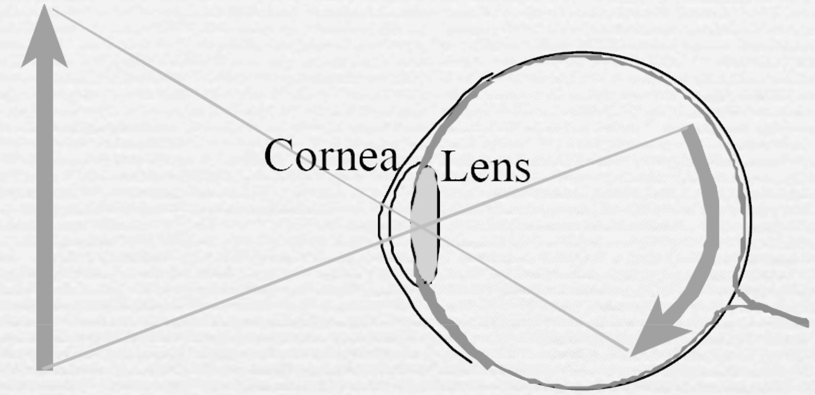
- Naked eye viewing



- Optical enhancement
    - correct vision
    - reduce brightness
    - magnify size
    - reduce distance

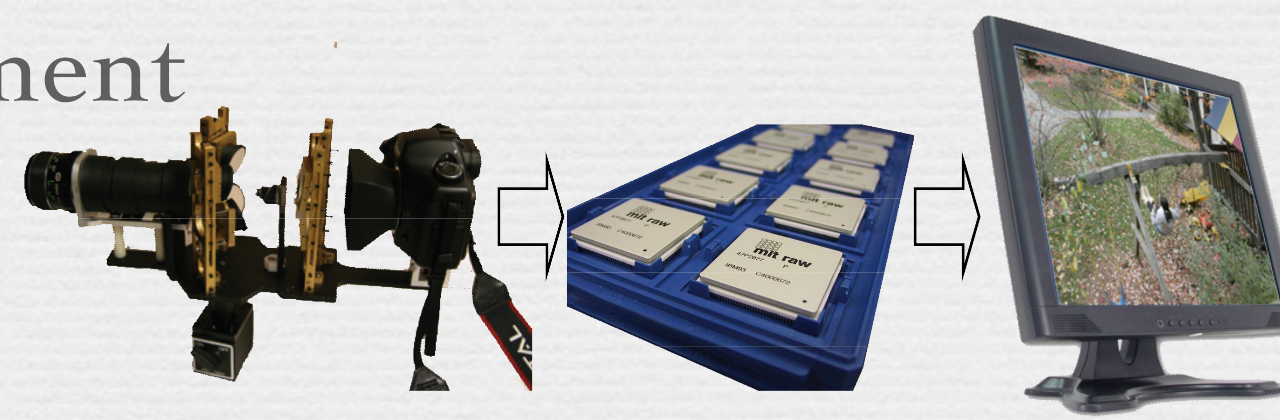
Computation is the new optics

- Naked eye viewing



- Optical enhancement
  - correct vision,
  - reduce brightness,
  - magnify size,
  - reduce distance

- Computational enhancement
  - camera + computation + display

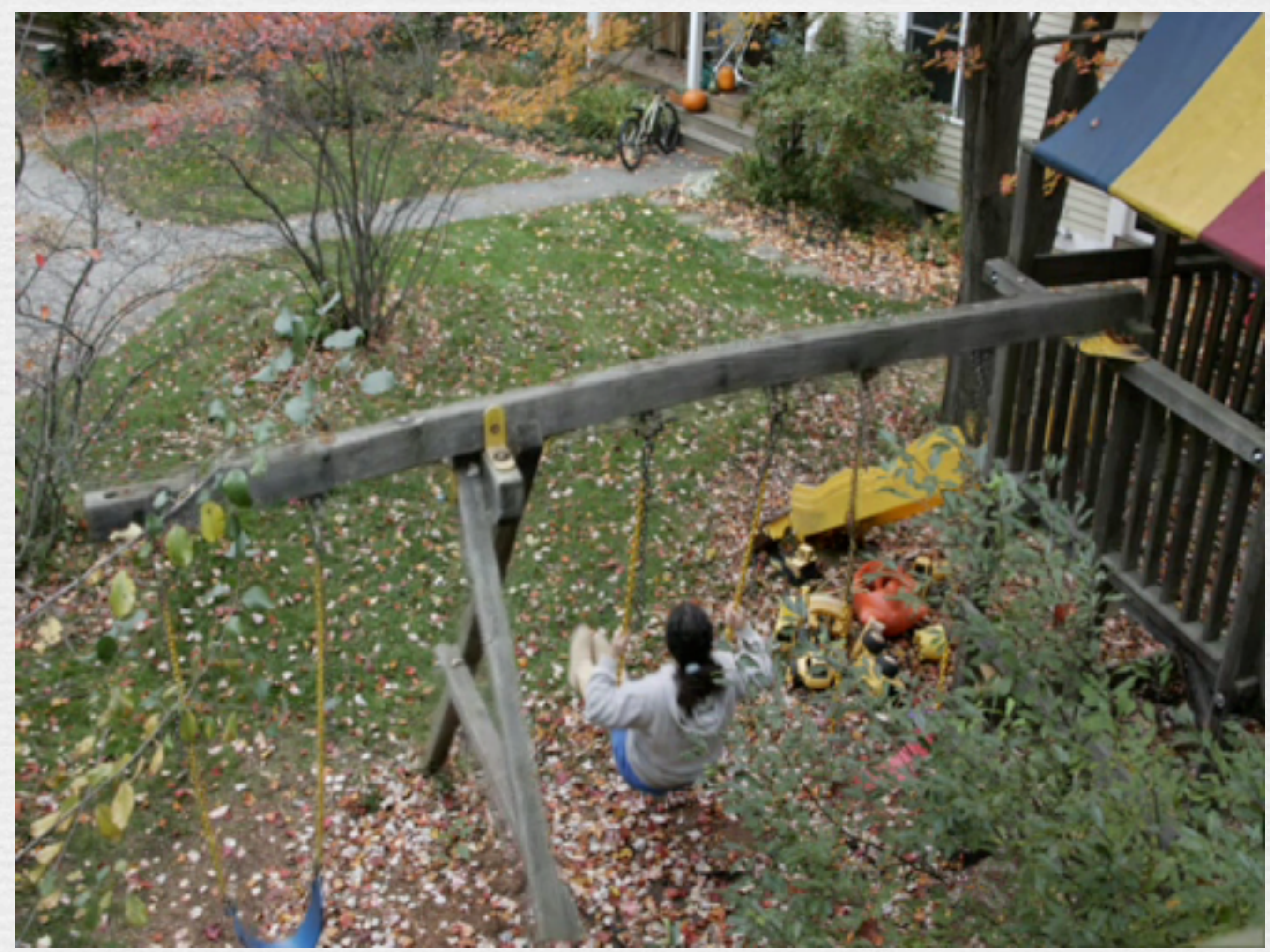


Motion magnification

- *with Liu, Torralba, Freeman & Adelson [Siggraph 2005]*
- Analyze motion in video (robust to occlusion)
- Magnify motion that is hard to see

Motion magnification

- *with Liu, Torralba, Freeman & Adelson [Siggraph 2005]*
- Analyze motion in video (robust to occlusion)
- Magnify motion that is hard to see


Original

Eardrum analysis

- work in Denny Freeman's group (Proceedings National Academy of Sciences)

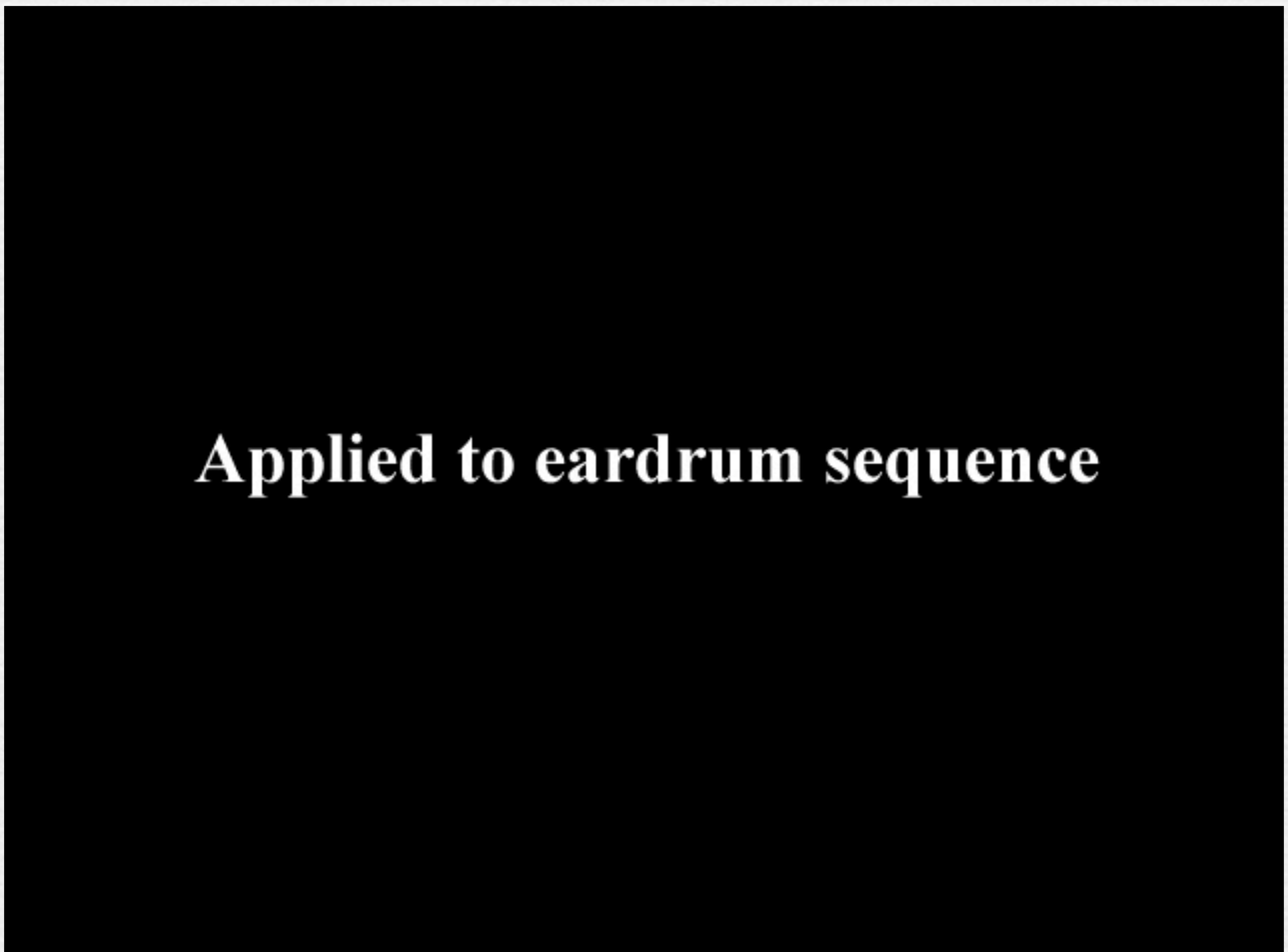


Image capture

- A sensor placed alone in the middle of the visual world does not record an image

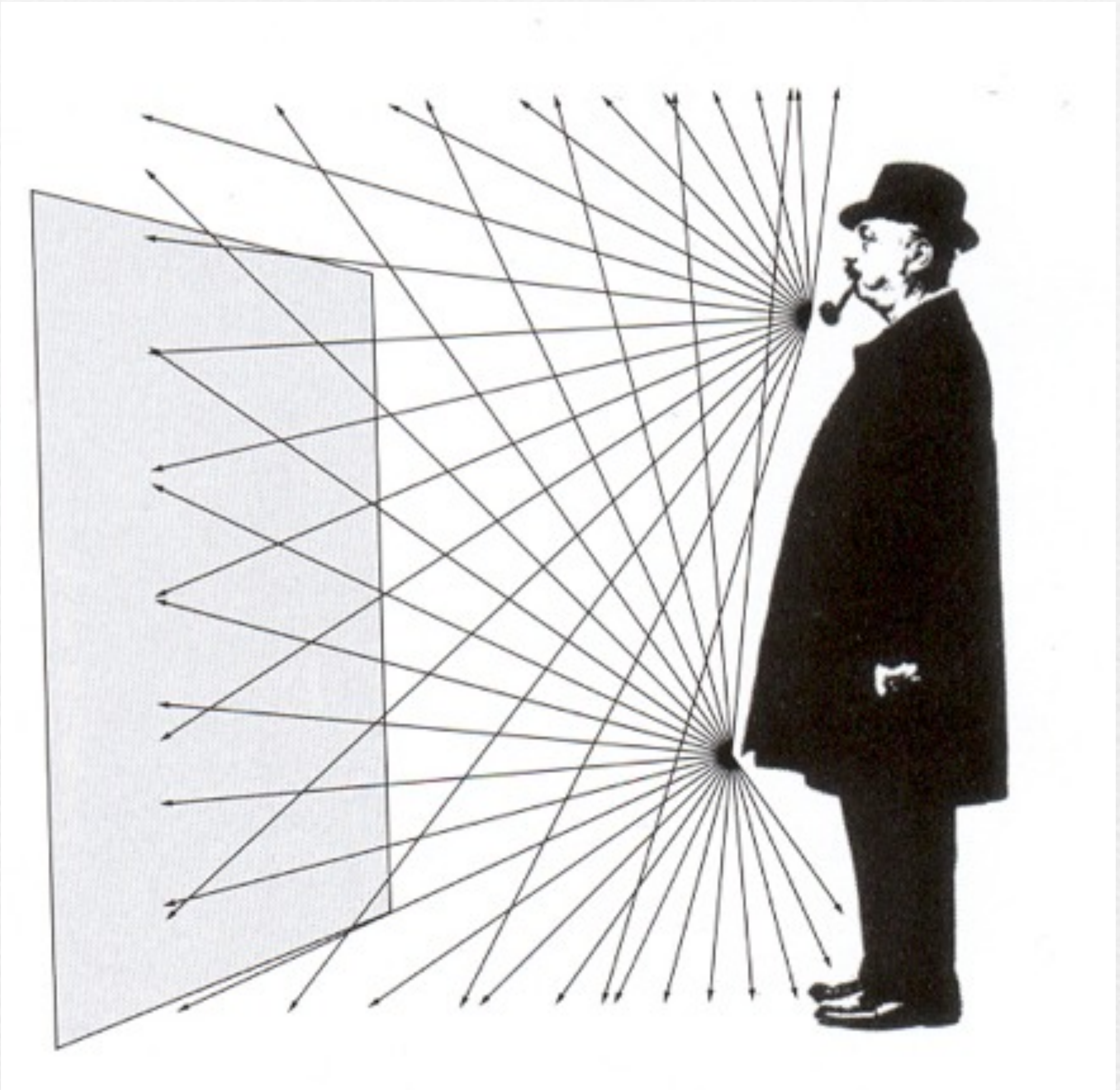


Image formation: optics

- Optics forms an image: selects and integrates light rays

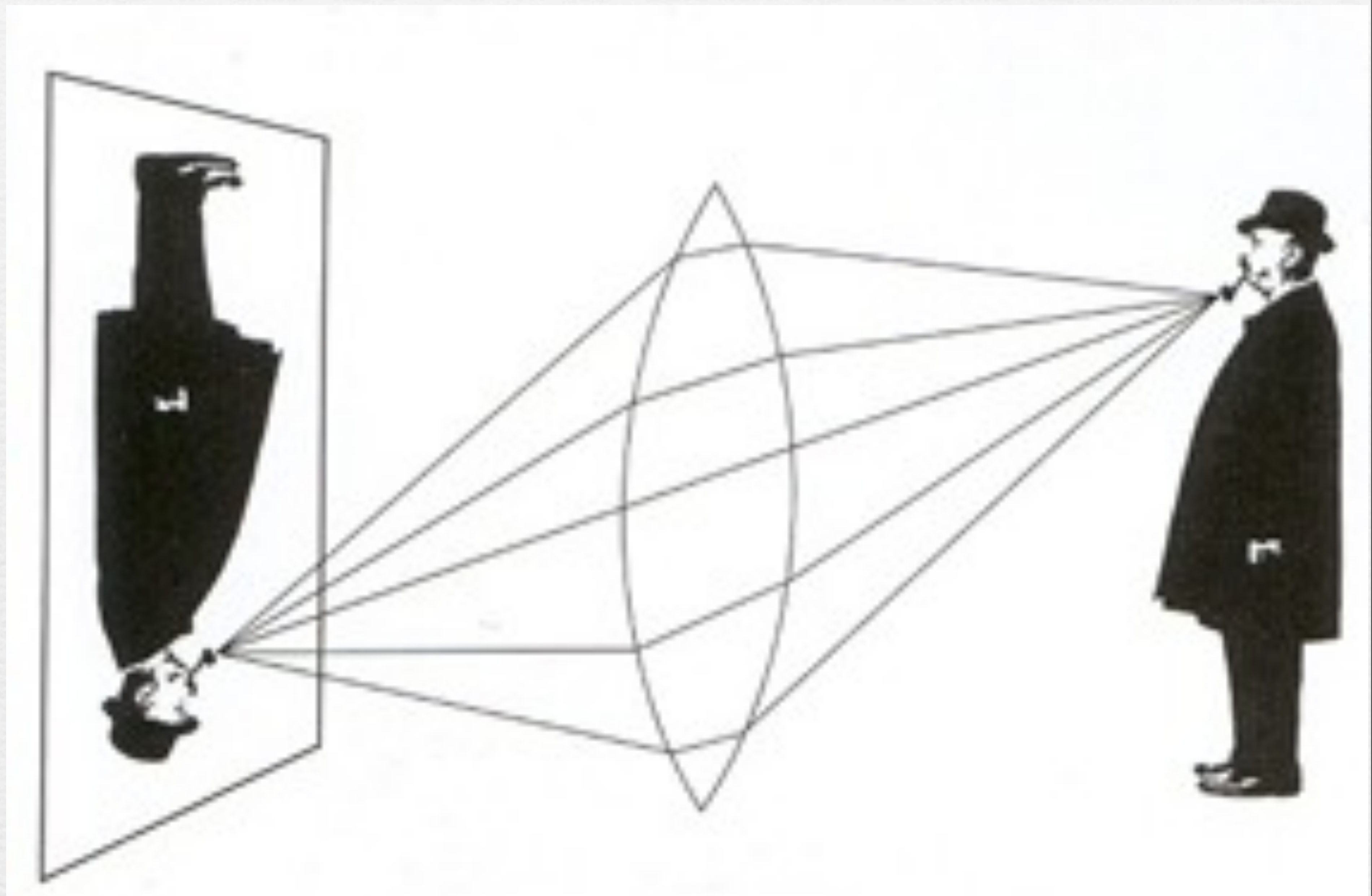
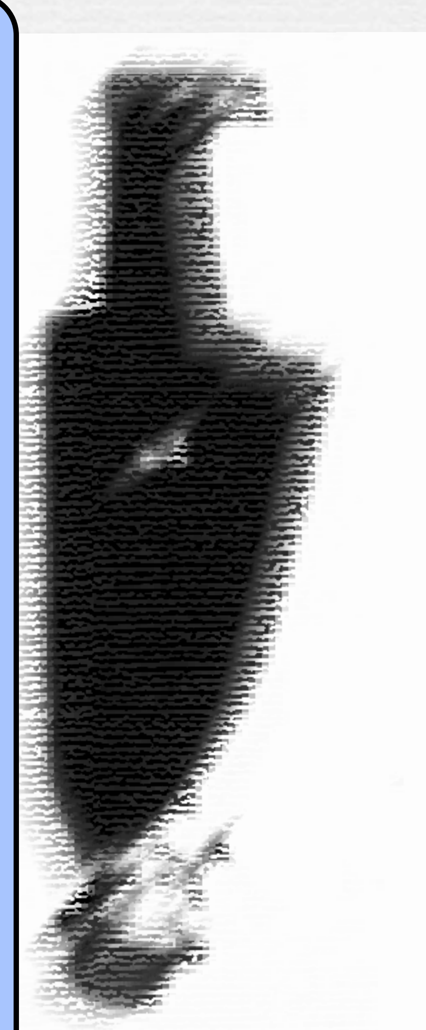


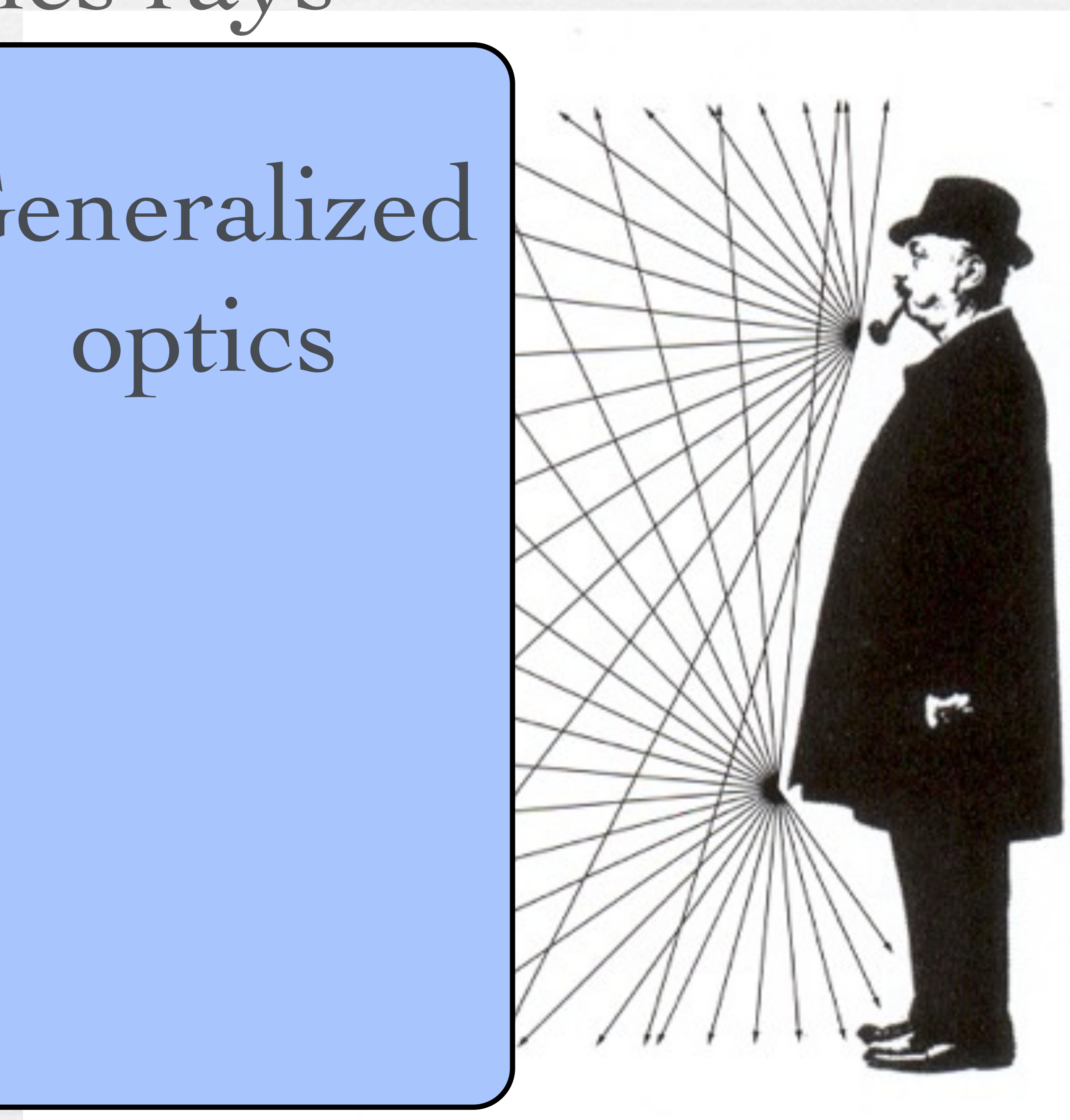
Image formation: computation

- The combination of optics & computation forms the image: selects and combines rays

Computation



Generalized optics



Final image

Intermediate optical image

Related fields

- Computer Vision
  - Extract information from visual array
- Computer graphics
  - Try to reproduce reality
- **Computational Imaging: areas with physics challenges**
  - **Astronomy/telescope**
  - **Radar**
  - **Microscopy**
  - **Medical Imaging**

Plan

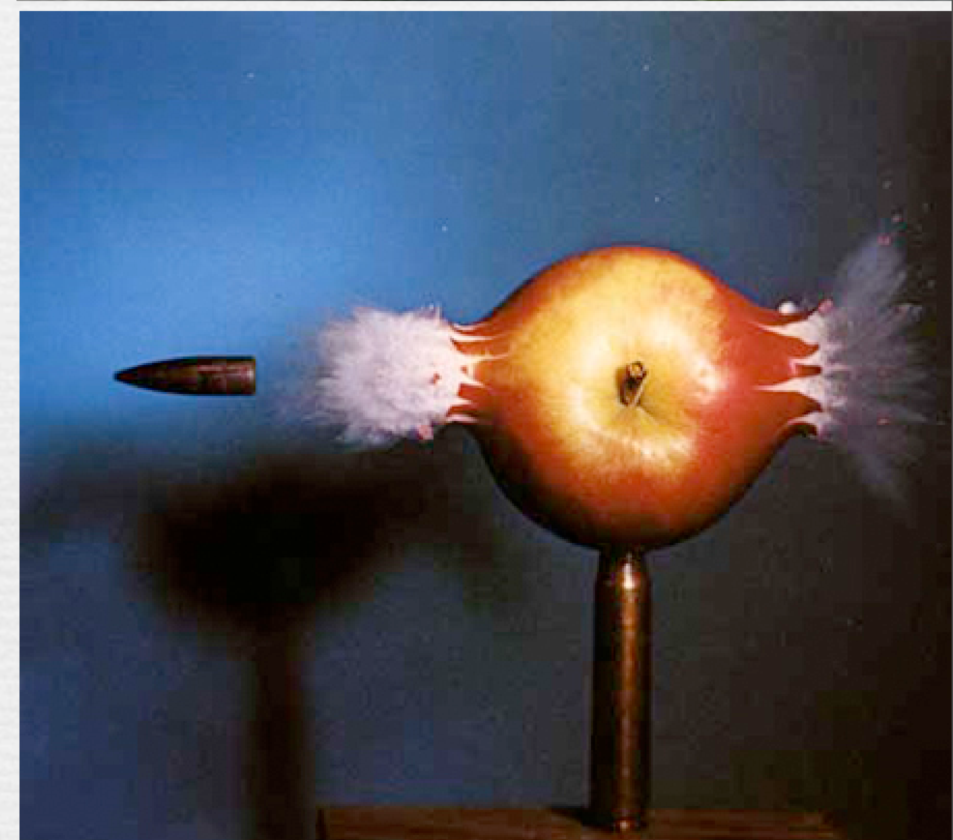
- Introduction of computational photography
  - Enhance our vision
  - Capture visual information
- **Motion Invariant Photography**
- Potpourri
- Big Ideas in Computational Photography

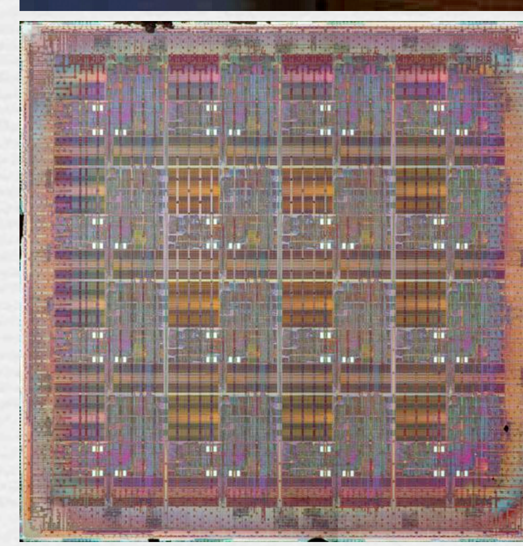
Motion Invariant Photography

Frédo Durand
MIT CSAIL
with Anat Levin, Peter Sand,
Taeg Sang Cho, Bill Freeman

This talk: blur removal

- Blur often reduces image quality
  - Motion blur, diffraction, defocus
- Traditional solution:
  - Faster shutter speed, smaller aperture, bigger aperture
  - Often increases noise (gathers less light)
- Today: computational solution
  - Remove blur given **single** image
  - Imaging hardware + software



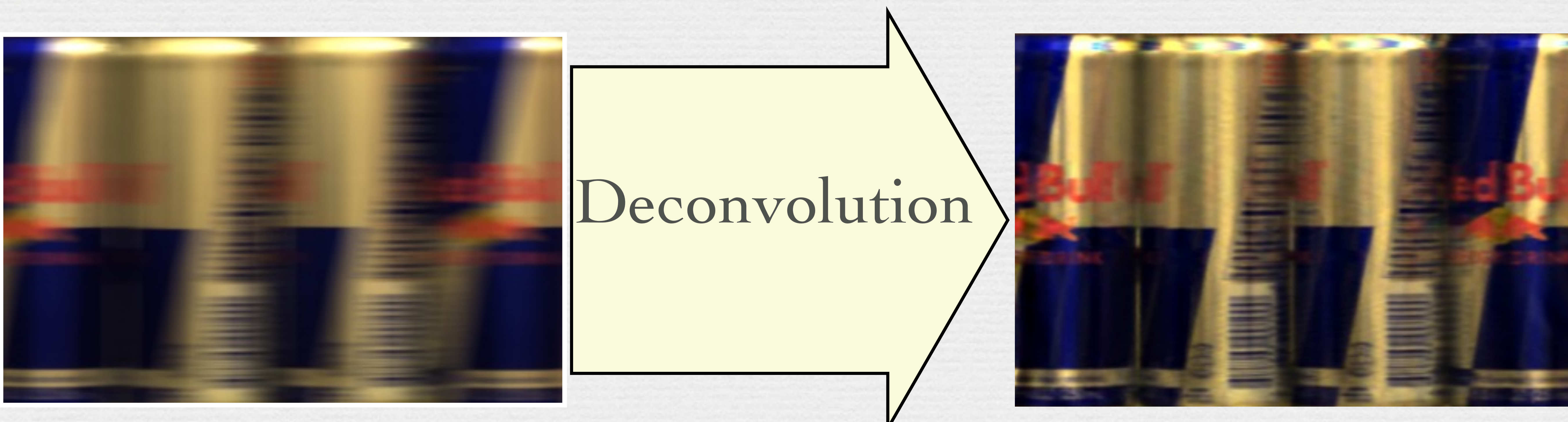


Motion blur


Most of the scene is static
red bull is moving
from left to right

Can we remove the blur?

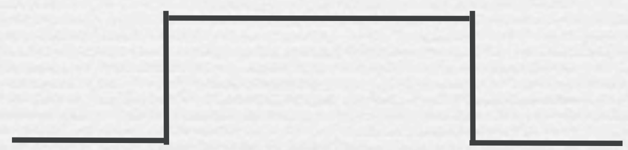
- Given single image with blur
- Blur is mostly a linear process, just invert it
  - called deconvolution
- But we need to know the exact blur
- But the process needs to be invertible
  - Lose as little information as possible



Kernel identification

Input blurry image

$\stackrel{?}{=}$



Correct kernel

$\otimes$

Output from correct kernel

Input blurry image

$\stackrel{?}{=}$

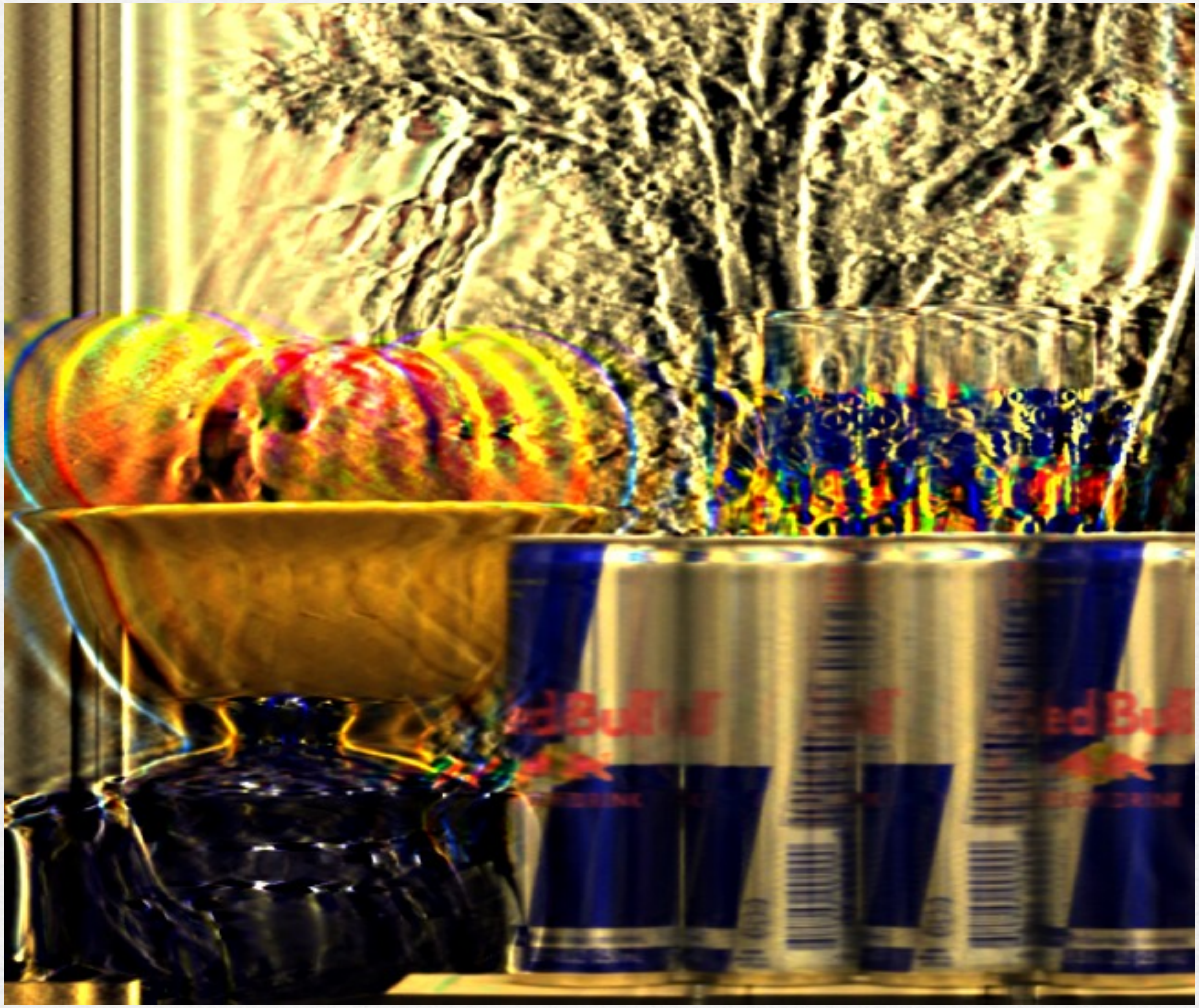
Wrong kernel

$\otimes$

Output from wrong kernel

Kernel identification

- The kernel is spatially varying



Entire image deblurred with kernel corresponding to the cans' velocity

Challenge with deblurring

- Blur destroys information
  - Often box filter

Deblurring given known blur:

blurred input

deblurred

static input

Blur destroys information

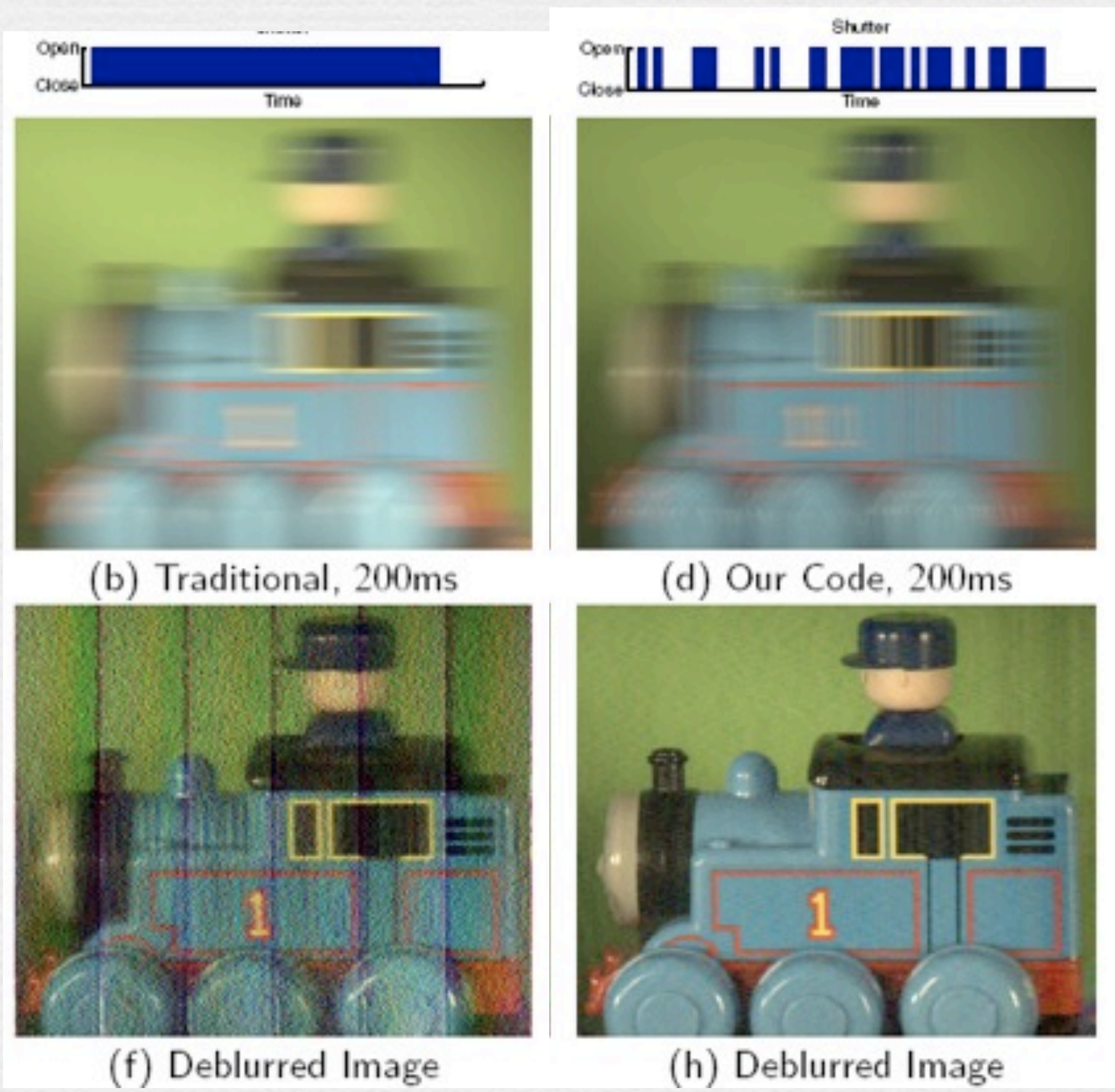
- Blur is a convolution, but sensor has noise
- Fourier domain:
  - Blurred image Y is a multiplication of sharp image X by kernel K plus noise N
  - $Y=XK+N$
- Deconvolution amplifies noise:
  - $X'=Y/K$
    $=X+N/K$
  - When kernel spectrum K is low, noise is amplified

Challenge with DoF and motion

- Blur destroys information
  - Low kernel spectrum is bad
- Kernel identification
  - Spatially varying

Flutter Shutter, Raskar et al 2006

- Close & open shutter during exposure to achieves broad-band kernel.
- But does not address kernel estimation and segmentation



(b) Traditional, 200ms (d) Our Code, 200ms

(f) Deblurred Image (h) Deblurred Image

Our counter intuitive solution:

To reduce motion blur, increase it!

- move camera as picture is taken

- Makes blur invariant to motion- can be removed with spatially uniform deconvolution
  - kernel is known (no need to estimate motion)
  - kernel identical over the image (no need to segment)
- Makes blur easy to invert

Inspiration: depth invariant defocus

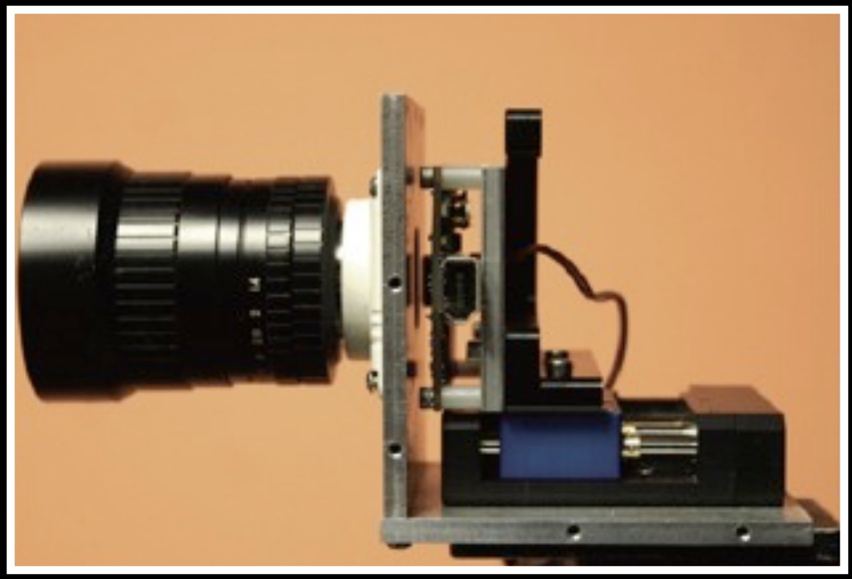
- **Wavefront coding - manipulate optical element**

  Cathey and Dowski 94

- **Vary object/detector distance during integration**

  - Hausler 72

  - Nagahara, Kuthirummal, Zhou, Nayar 08



Motion invariant blur- disclaimers:

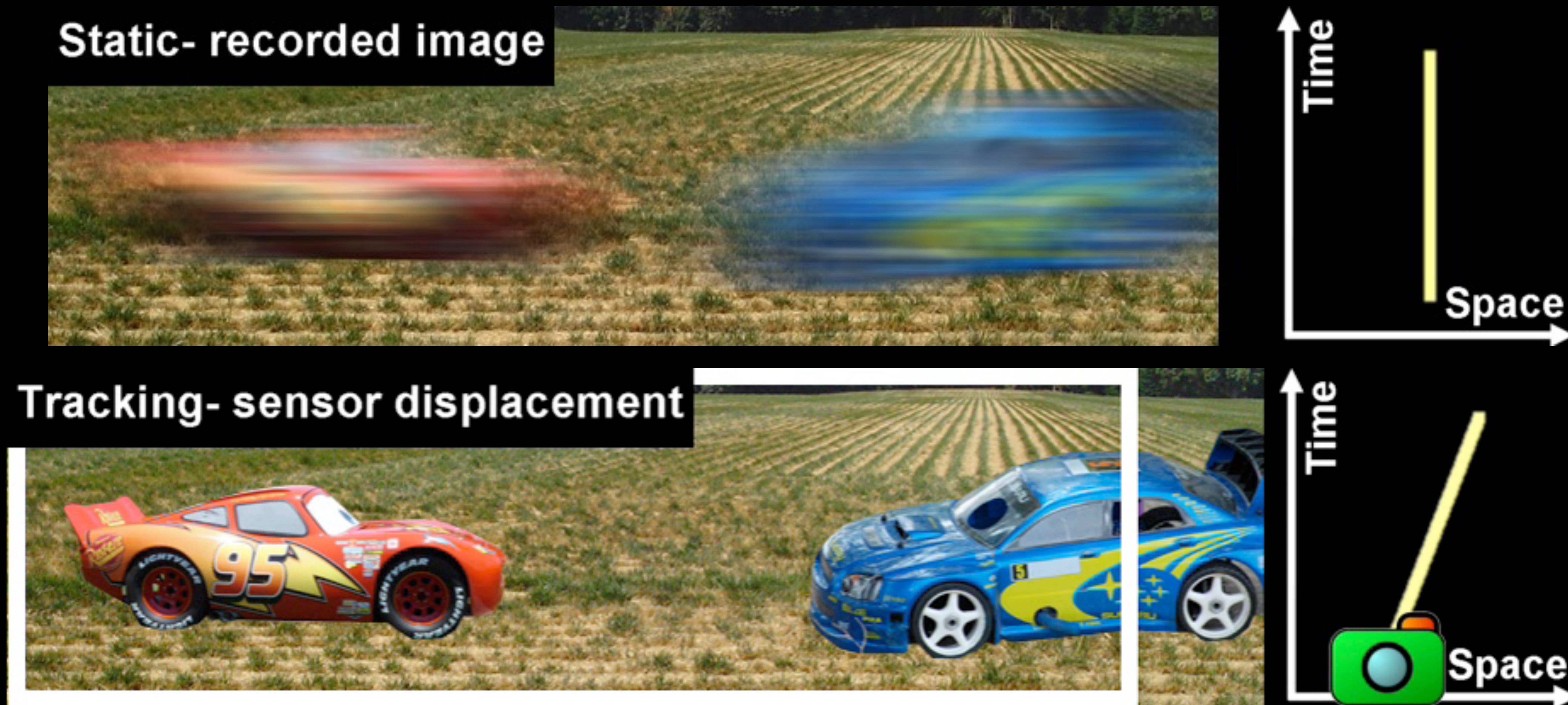
- Assumes 1D motion (e.g. horizontal)
- Degrades quality for static objects

Controlling motion blur

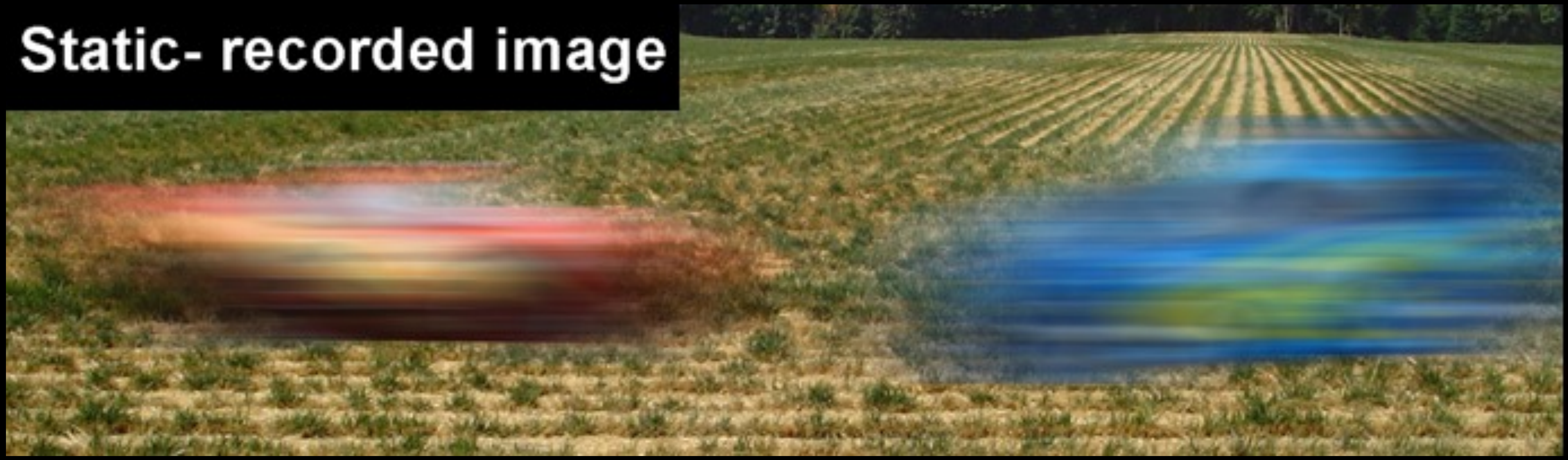
Controlling motion blur

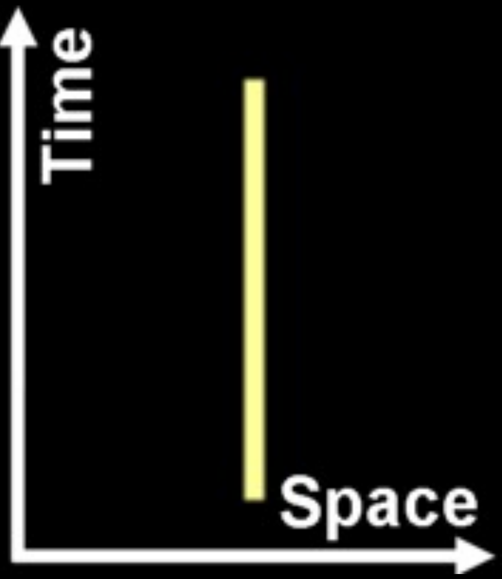
Can we control motion blur?

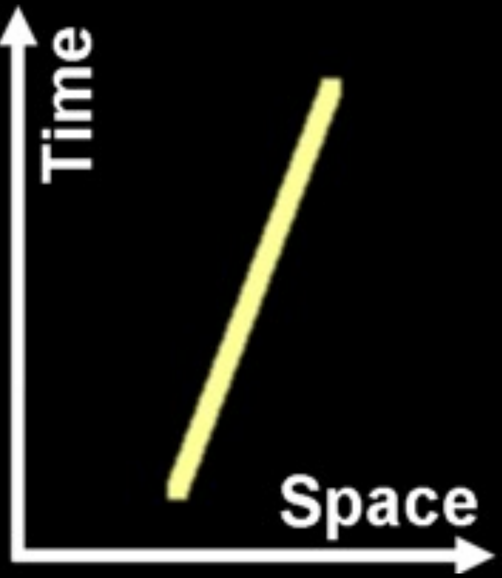
Controlling motion blur



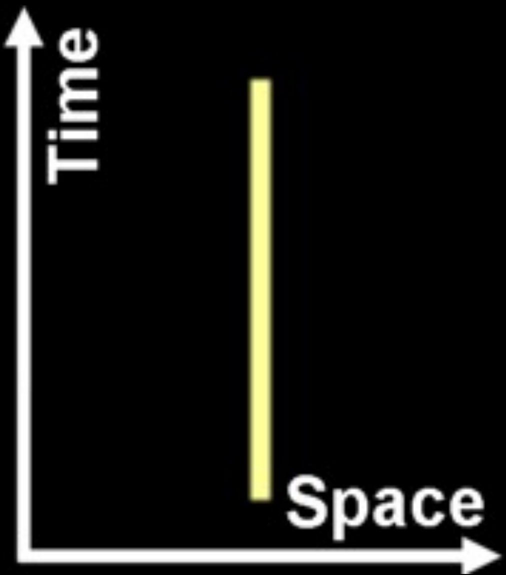
Controlling motion blur

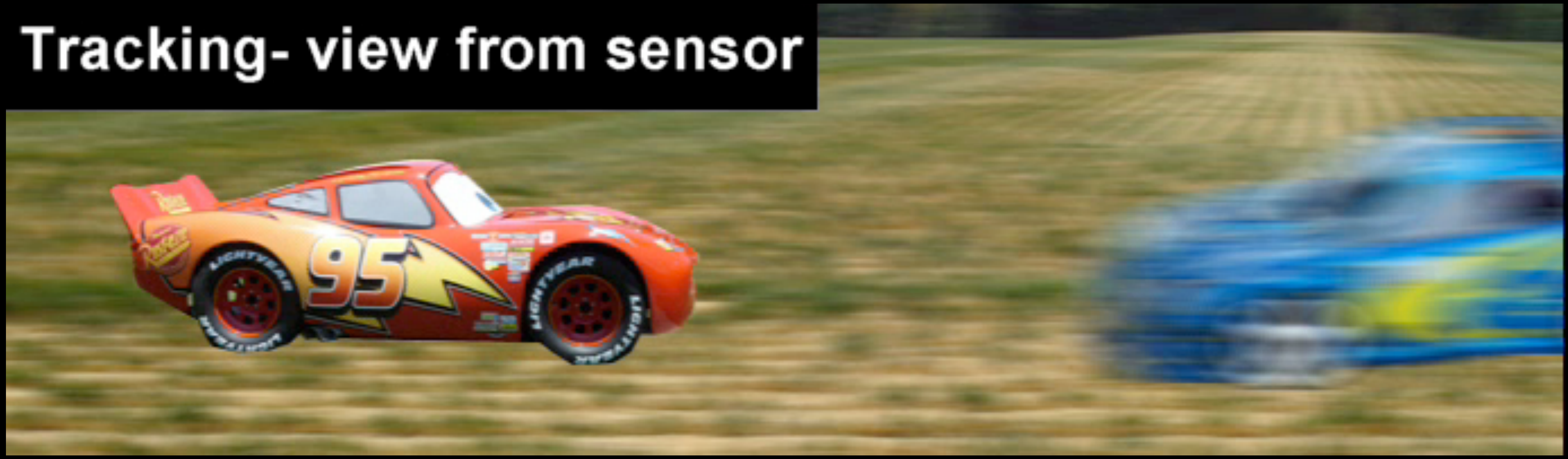


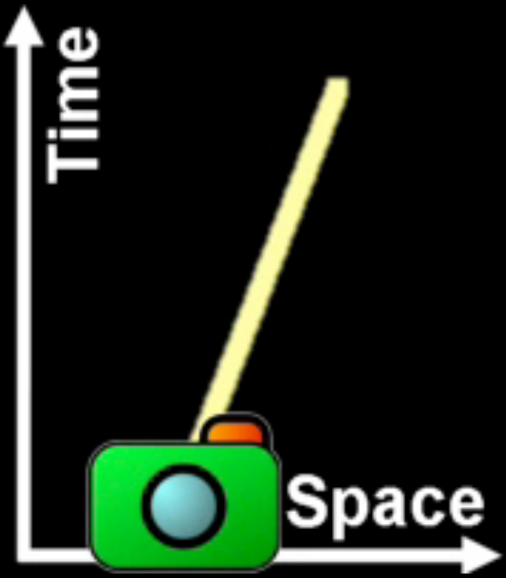




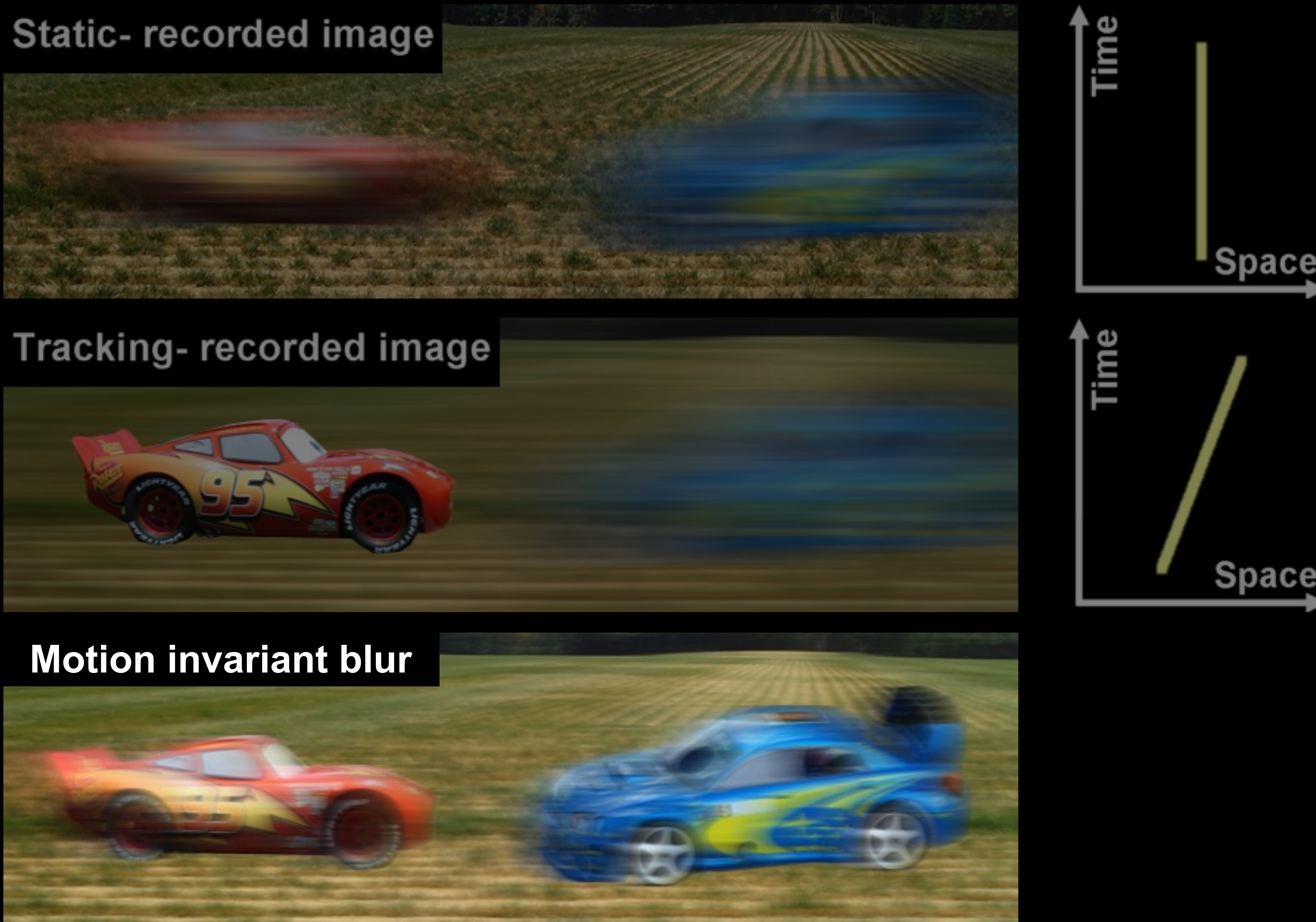
Controlling motion blur







Controlling motion blur



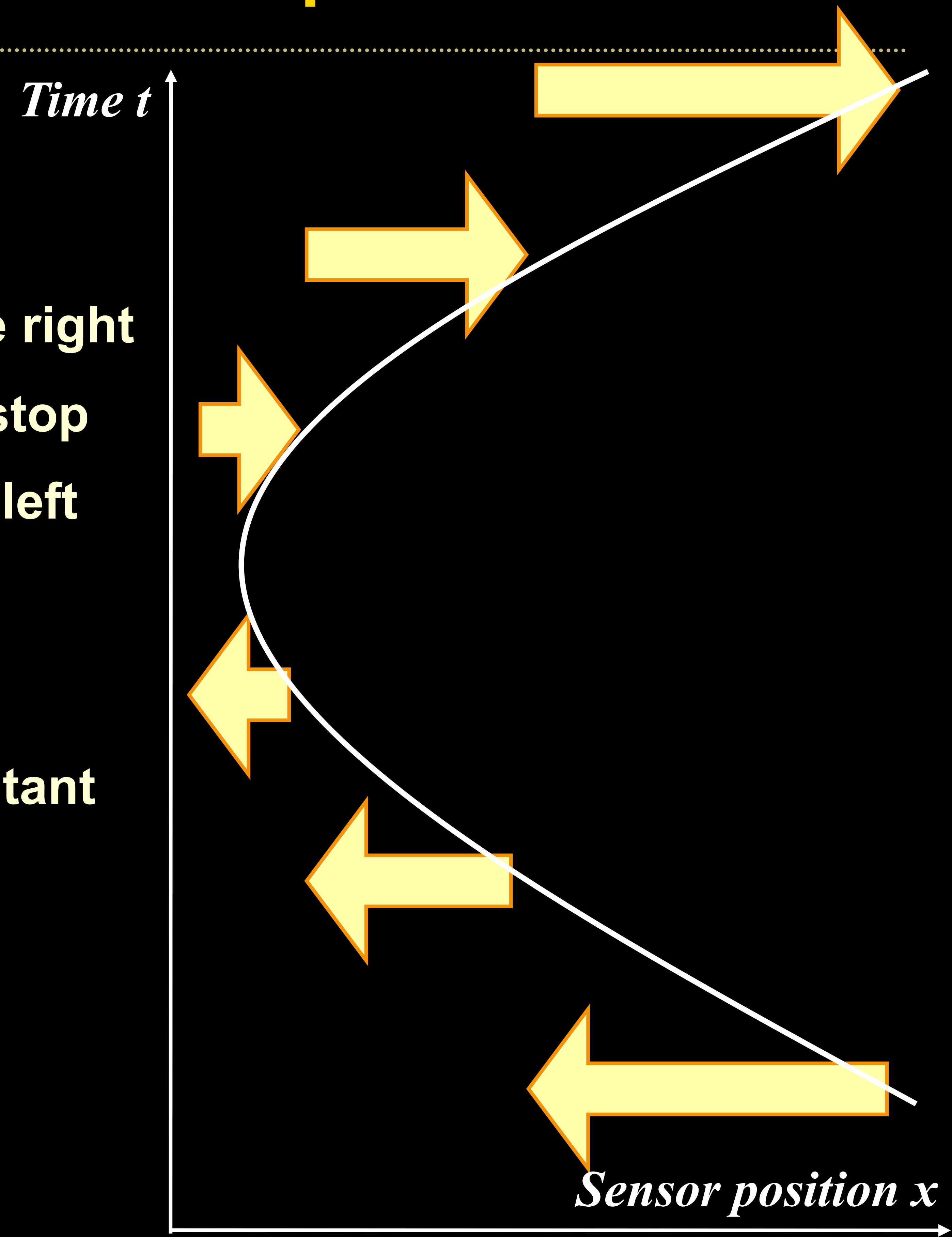
Parabolic sweep

Sensor position $x(t)=a\ t^2$

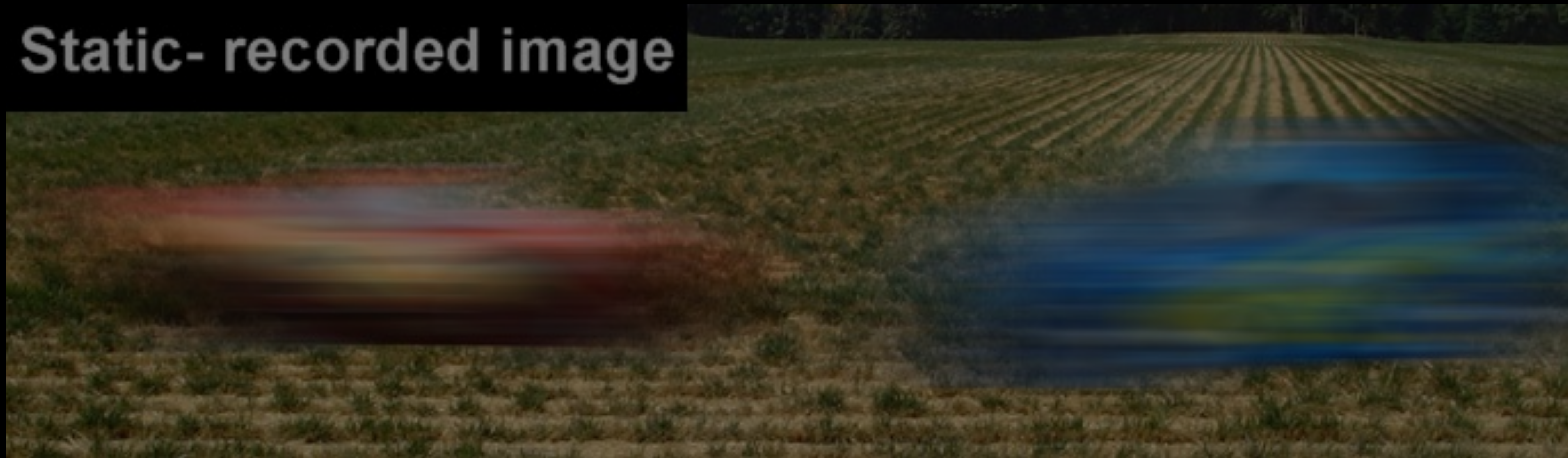
- Start by moving very fast to the right
- Continuously slow down until stop
- Continuously accelerate to the left

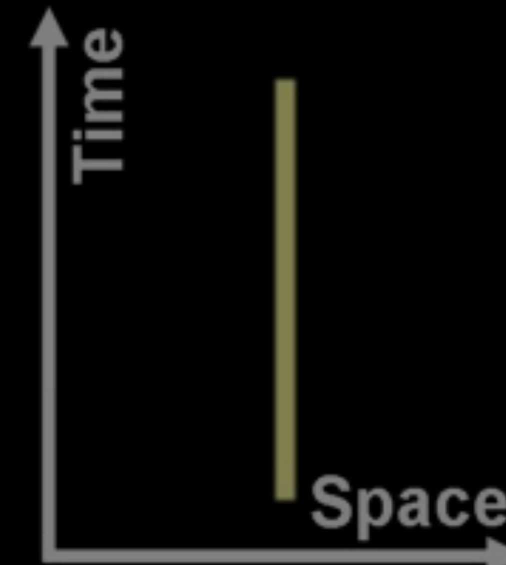
Intuition:

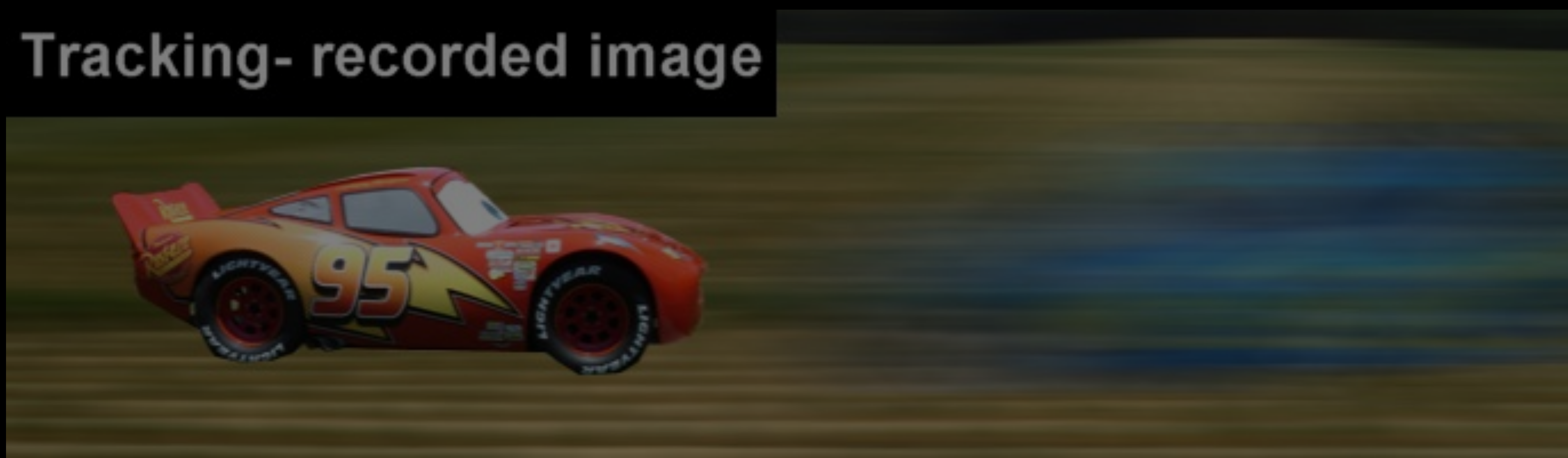
For any velocity, there is one instant where we track perfectly.

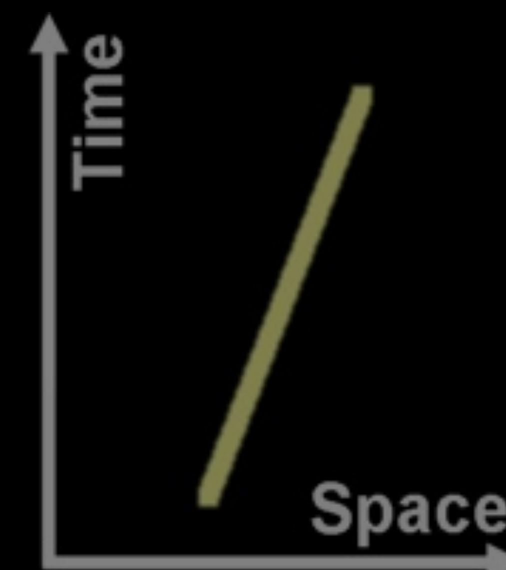


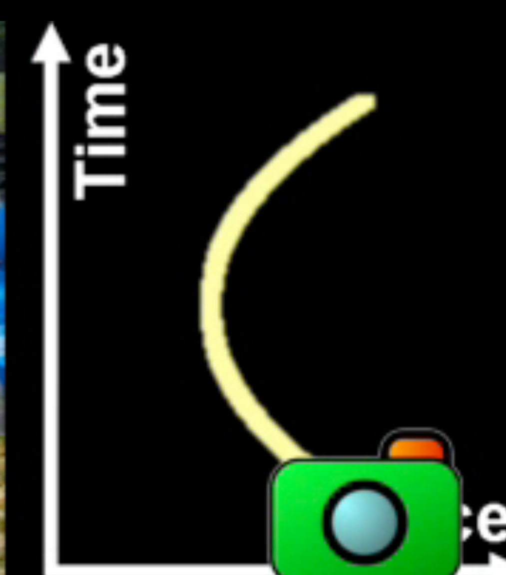
Motion invariant blur



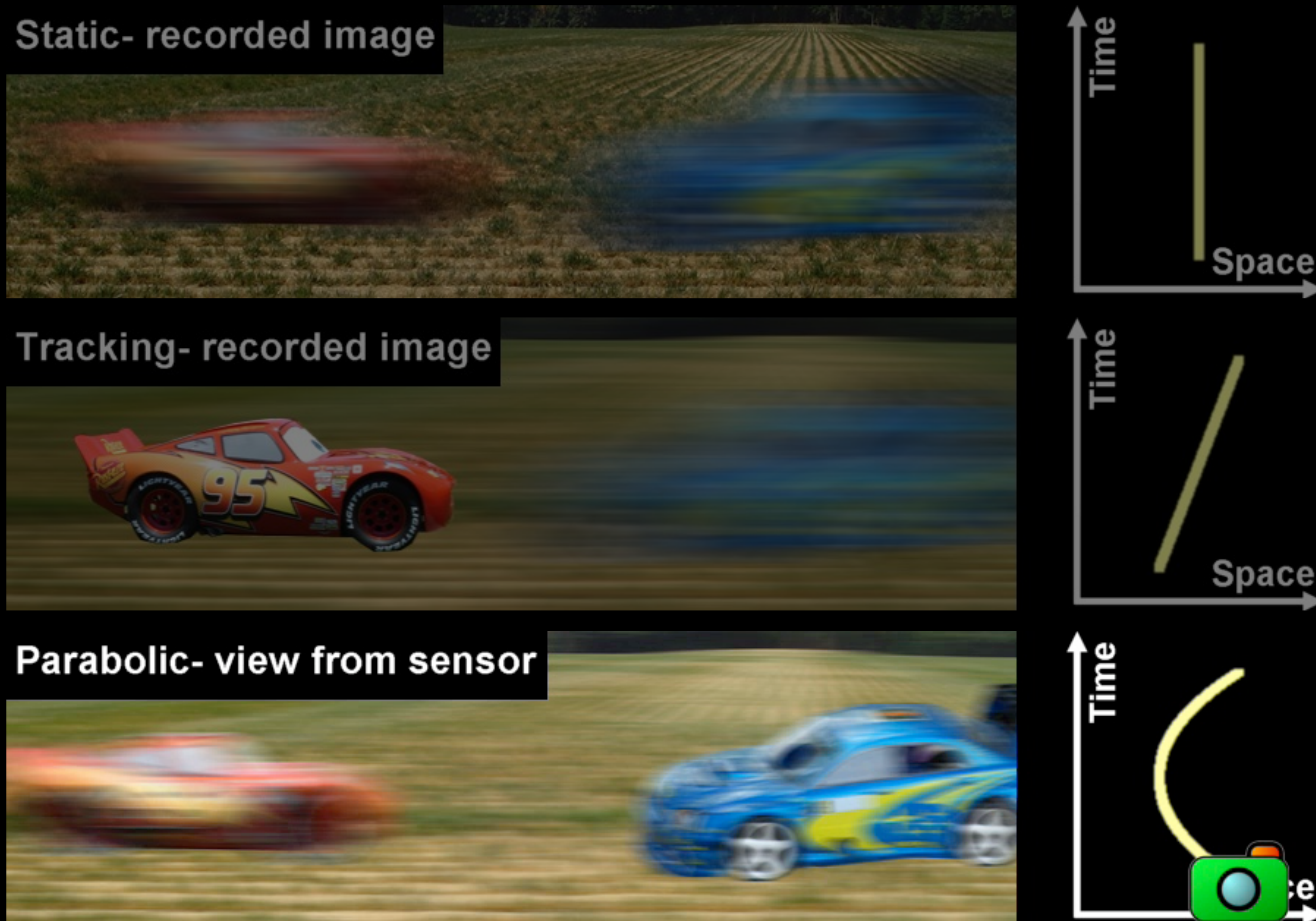




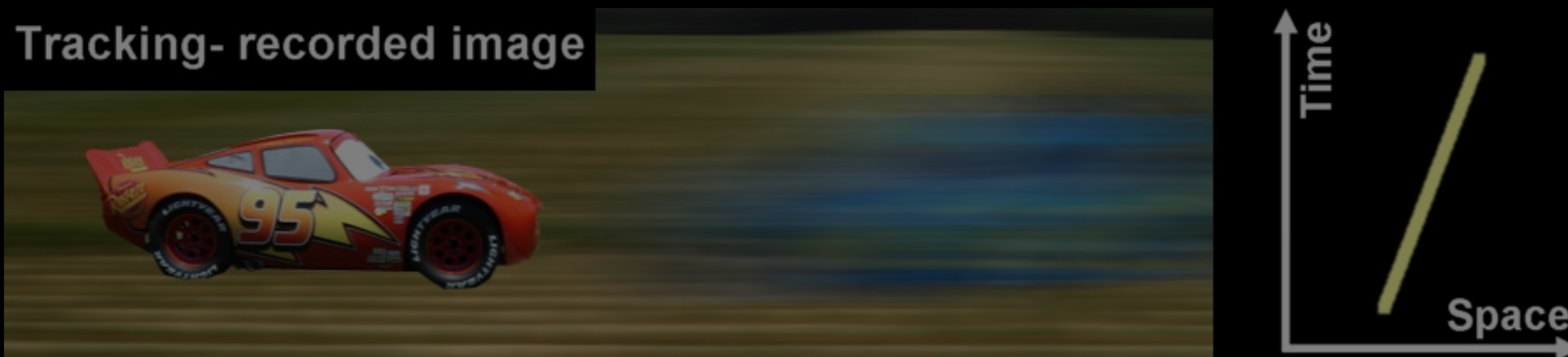




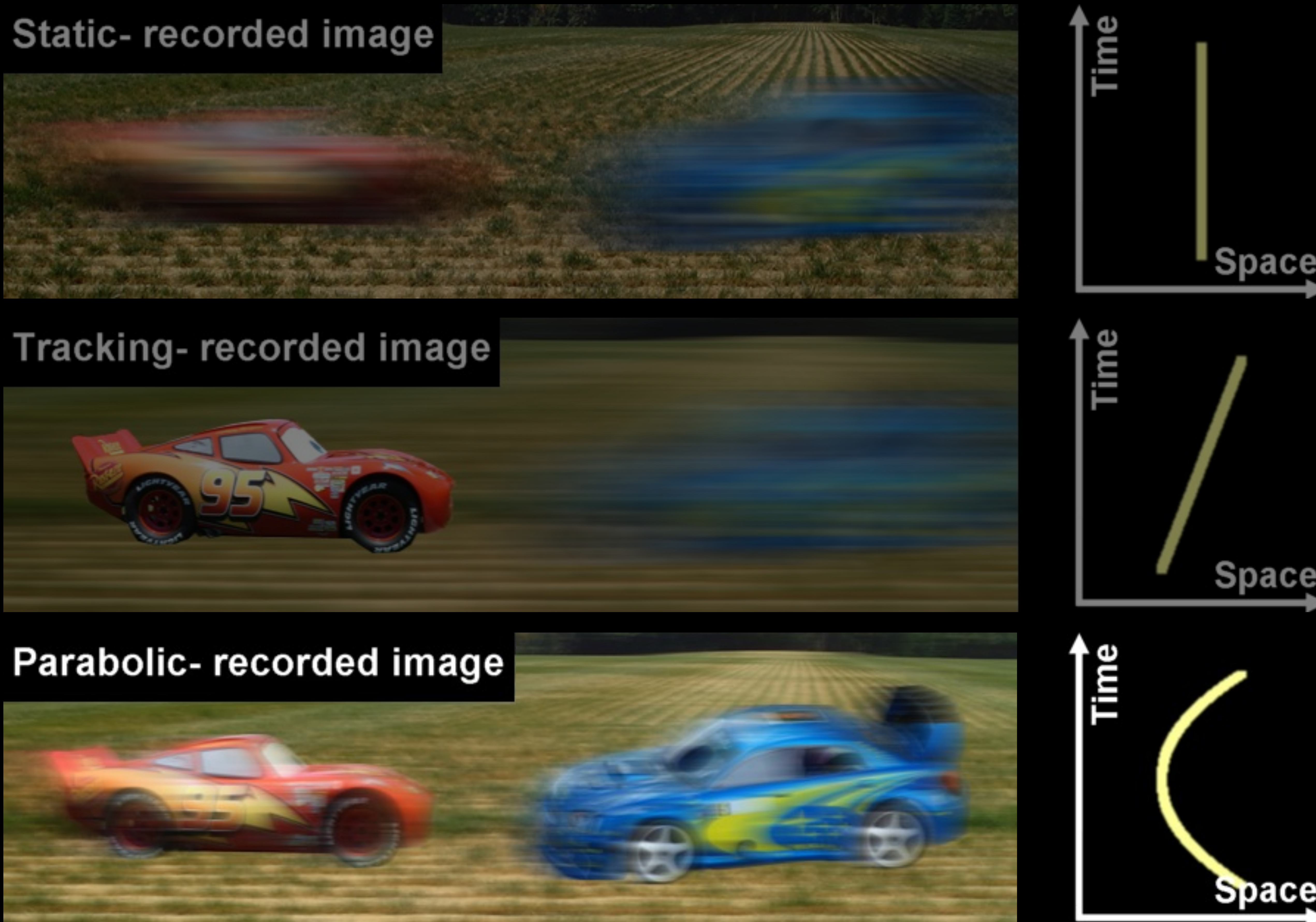
Motion invariant blur



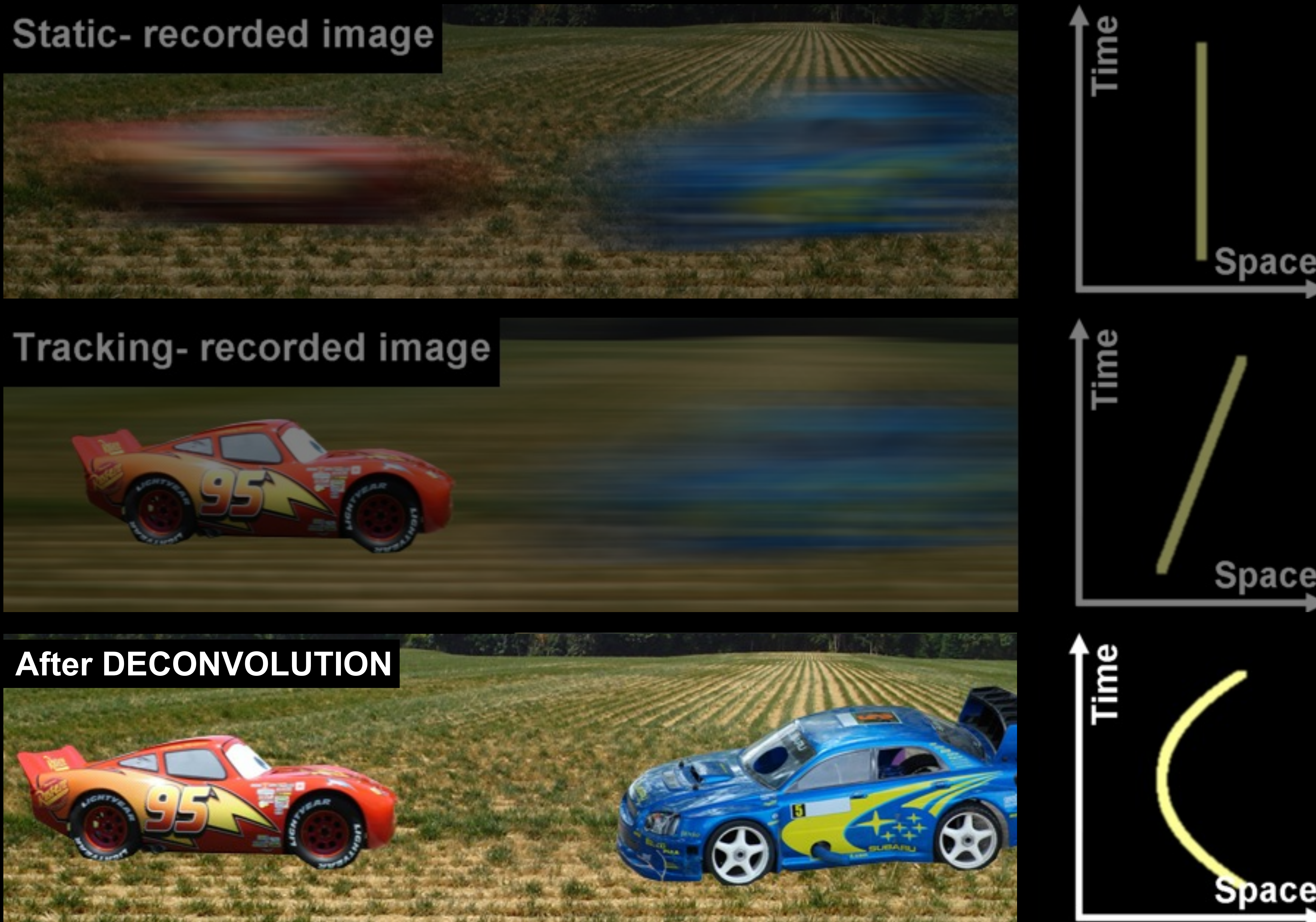
Motion invariant blur

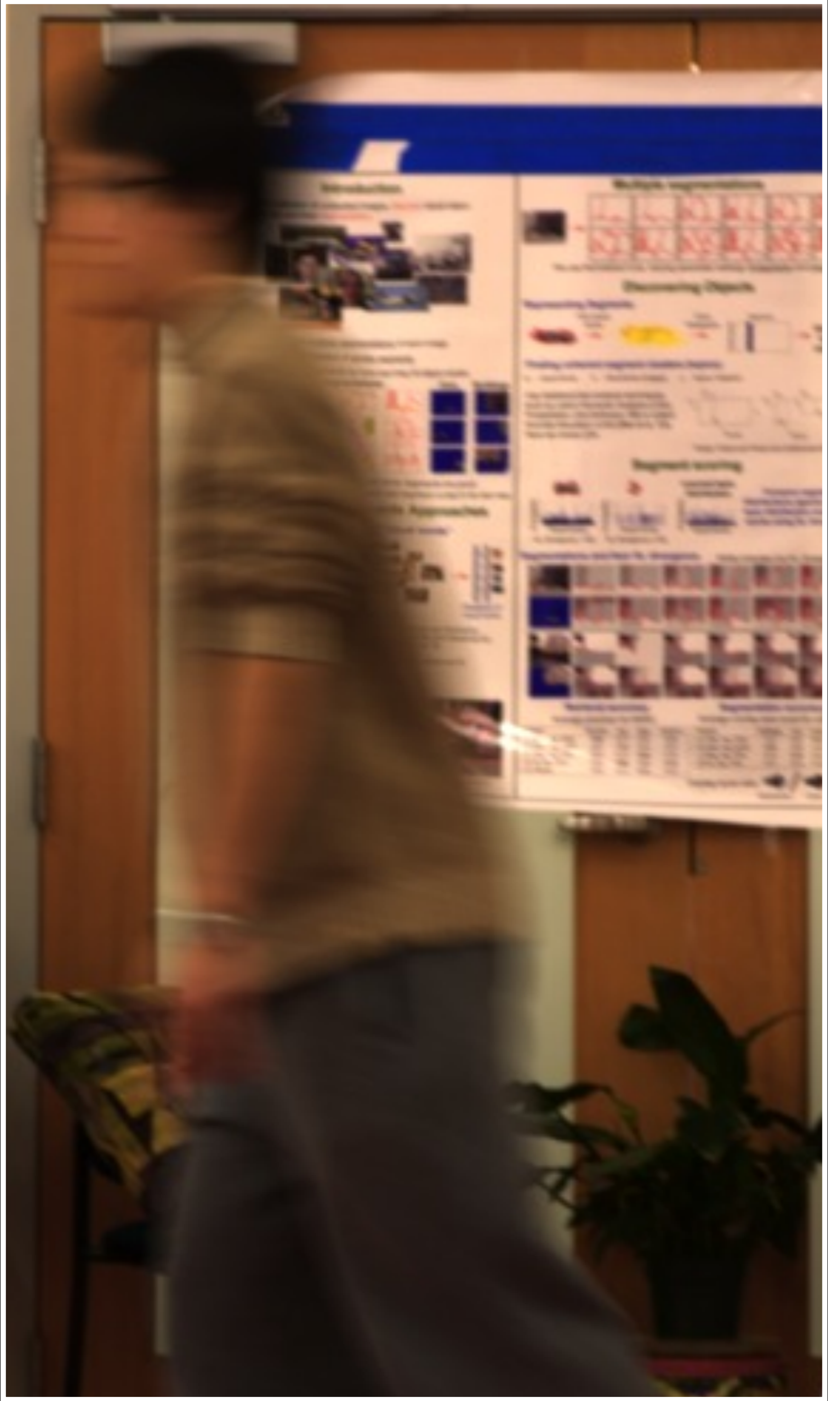


Motion invariant blur



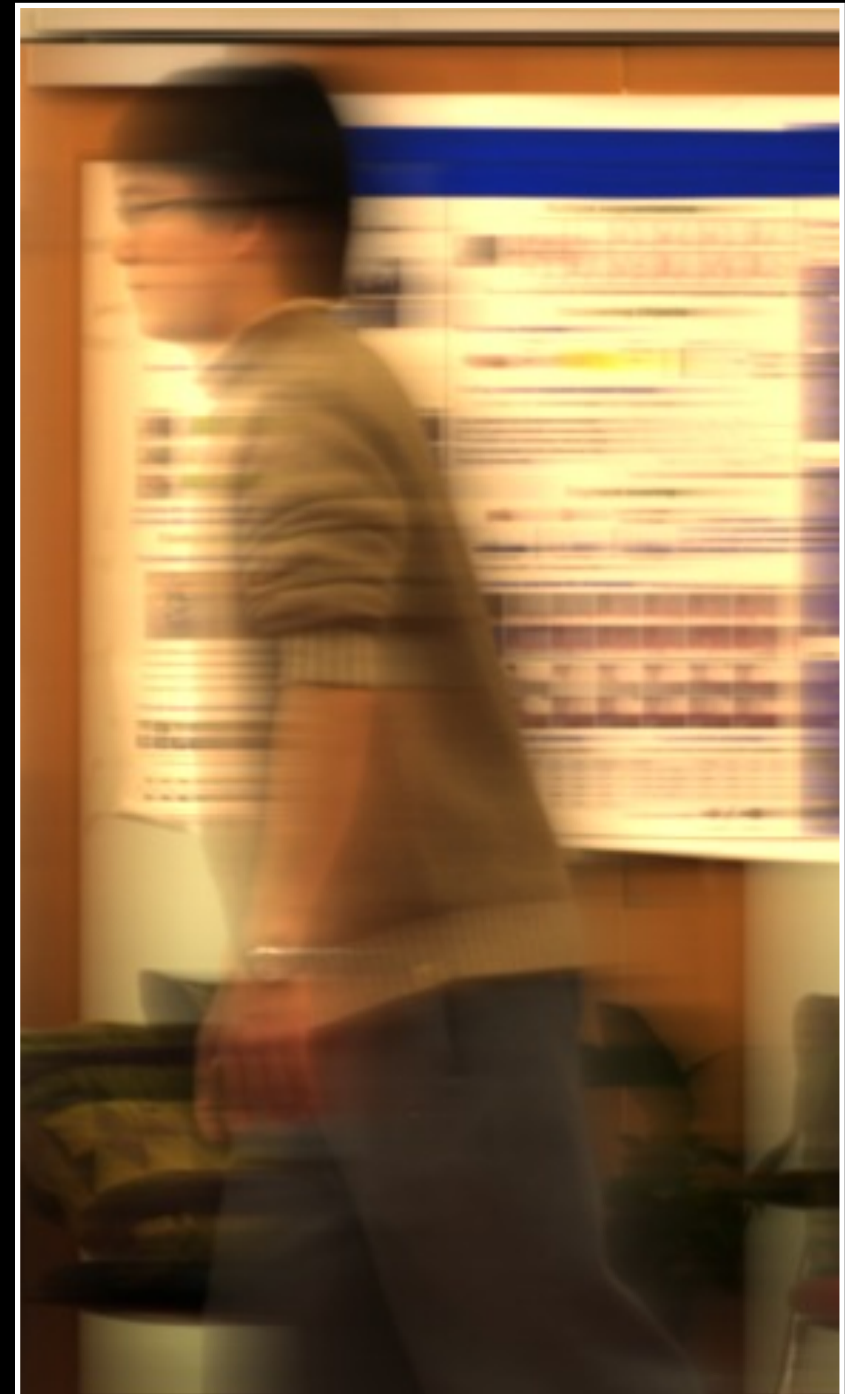
Motion invariant blur





Static camera

Unknown and variable blur kernels



Our parabolic input

Blur kernel is invariant to velocity

Our output after deblurring

NON-BLIND deconvolution

Recall: challenge with motion

- Blur destroys information
  - Often box filter
- Kernel identification
  - Spatially varying
  - **We have addressed this by making the kernel invariant to velocity**

Deblurring and information loss

Assume: we could perfectly identify blur kernel

Which camera has motion blur that is easy to invert?
- Static? Flutter Shutter? Parabolic?

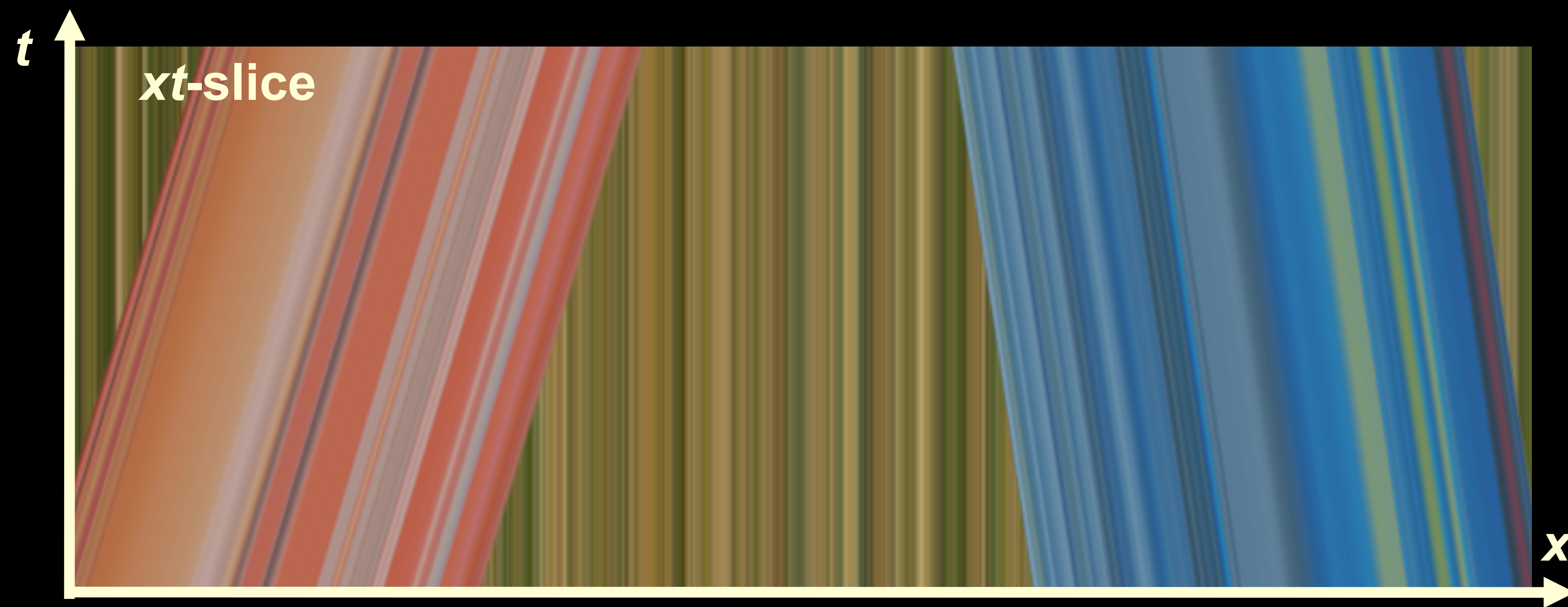
Our papers proves that parabolic motion achieves near optimal information preservation

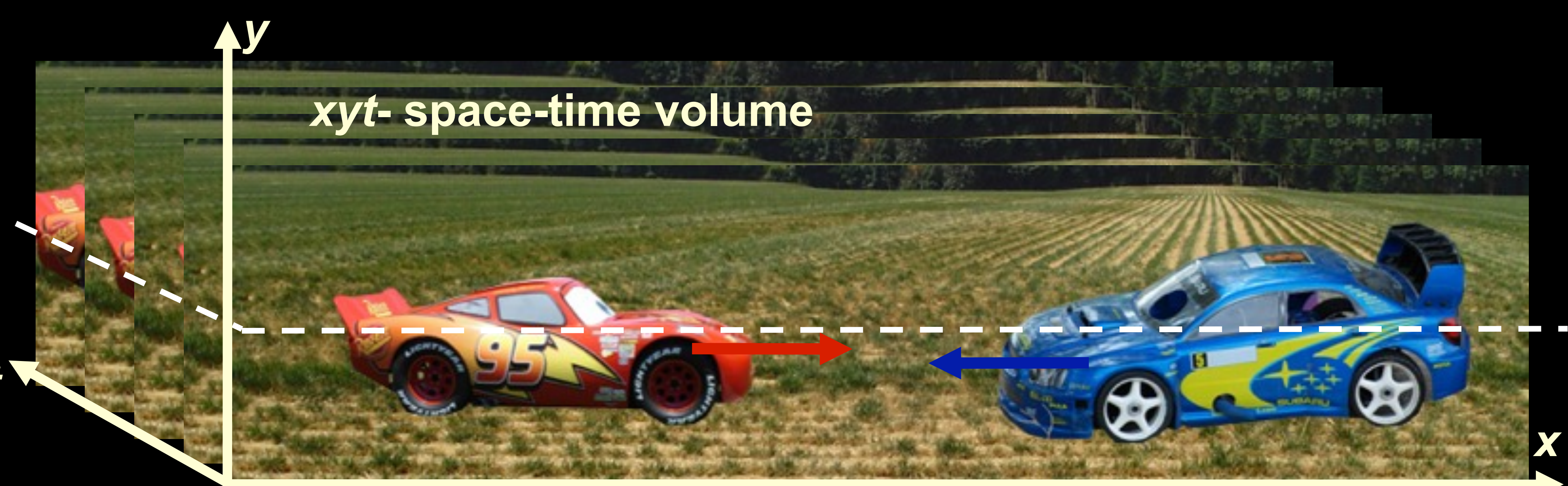
blurred input

deblurred

static input

The space time volume

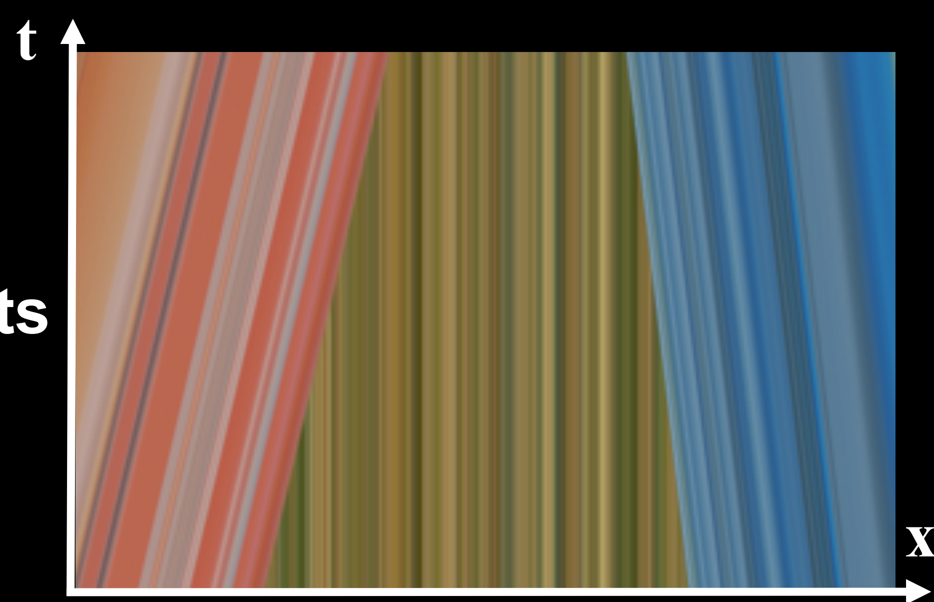


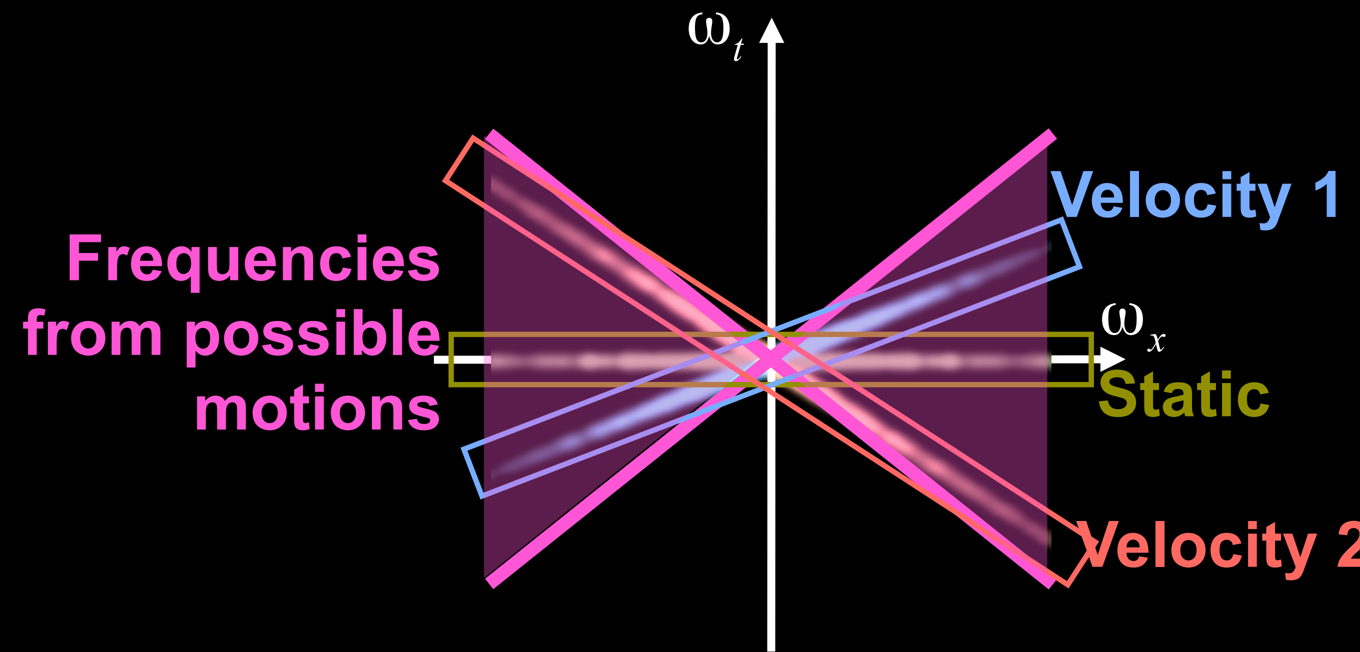


Space-time Fourier domain

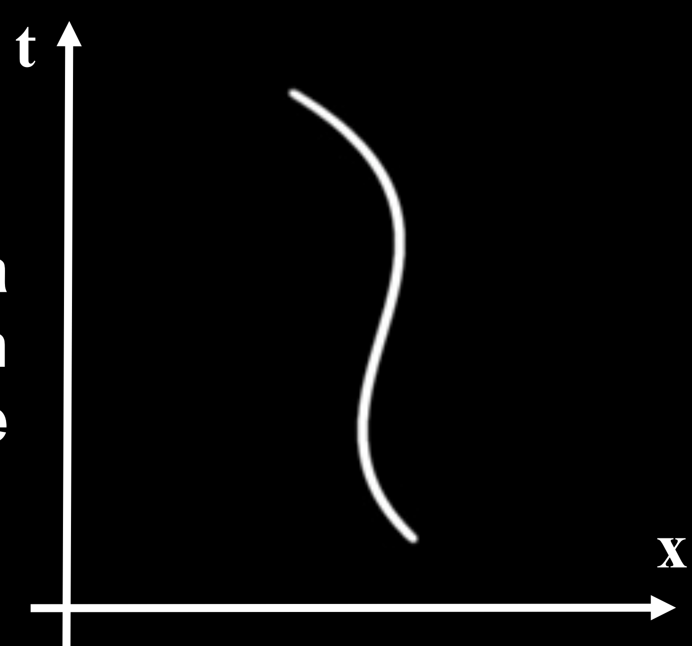
Primal Domain | **Frequency Domain**

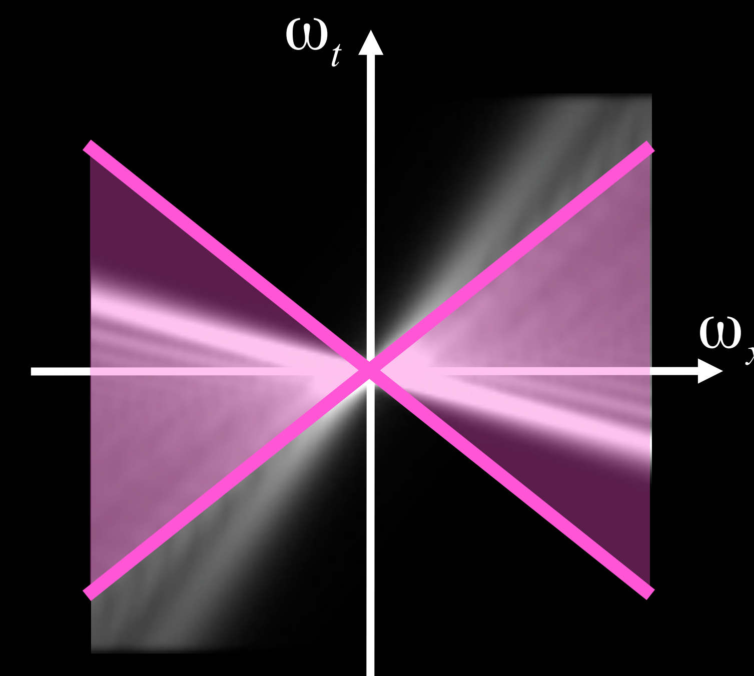
Objects





Camera integration curve





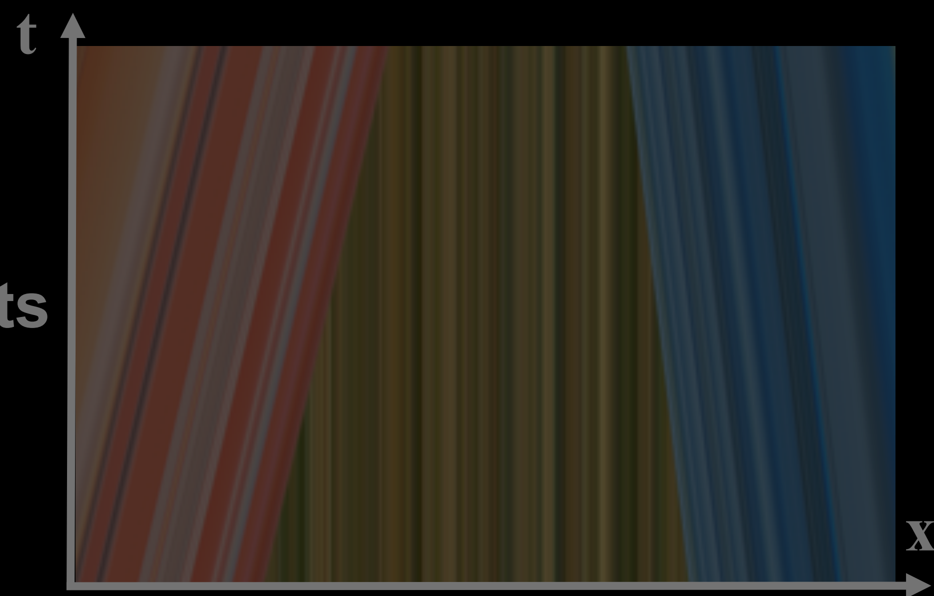
Bounded velocities range=> need to preserve a double wedge in the frequency domain

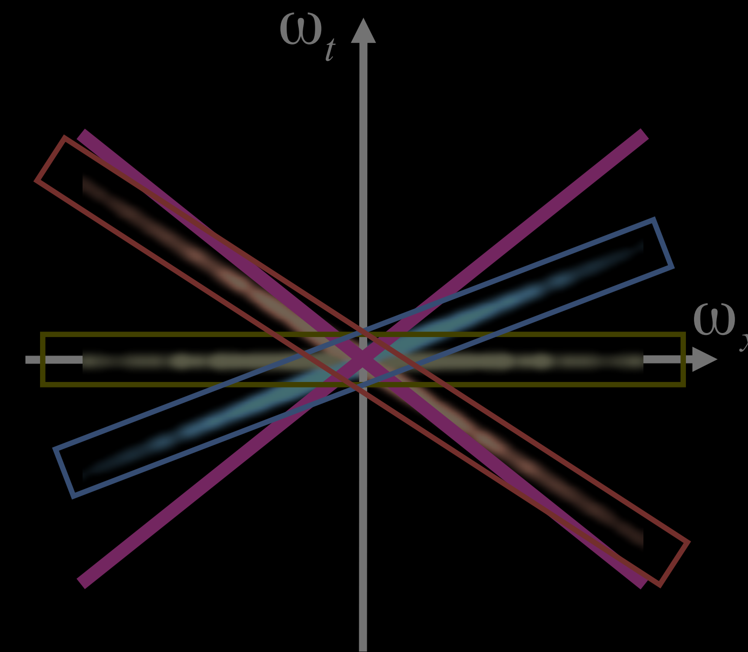
Static camera

Primal Domain

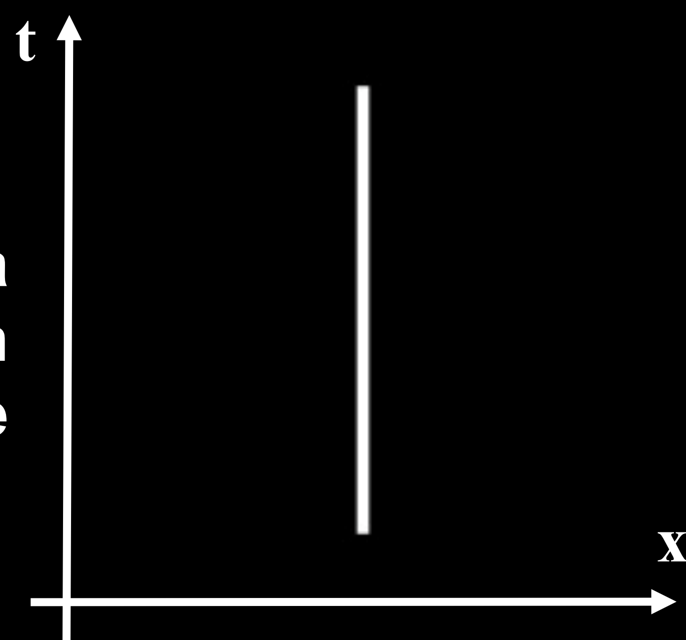
Frequency Domain

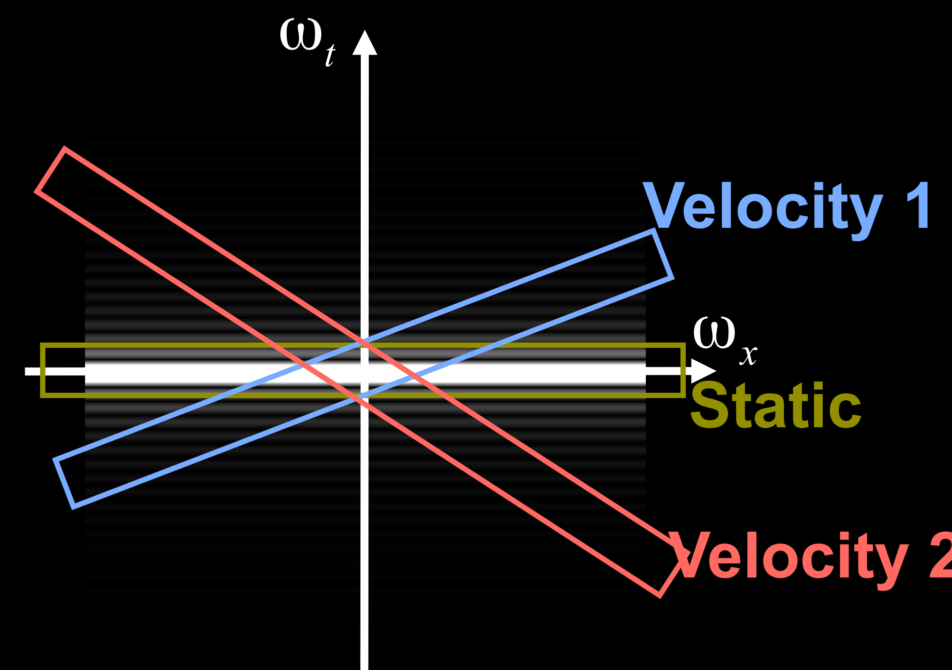
Objects





Camera integration curve





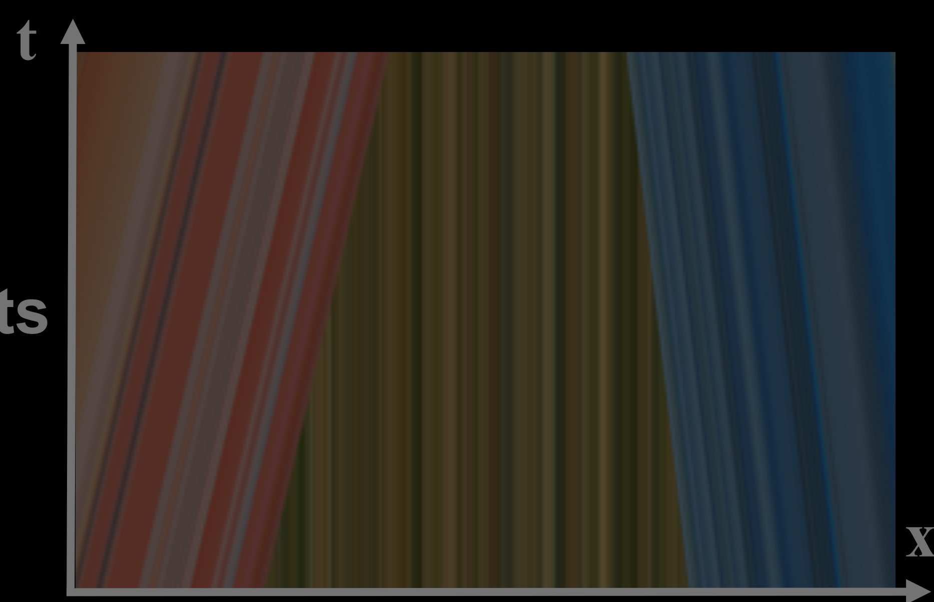
Vertical integration segment

Static object: high response
Higher velocities: low

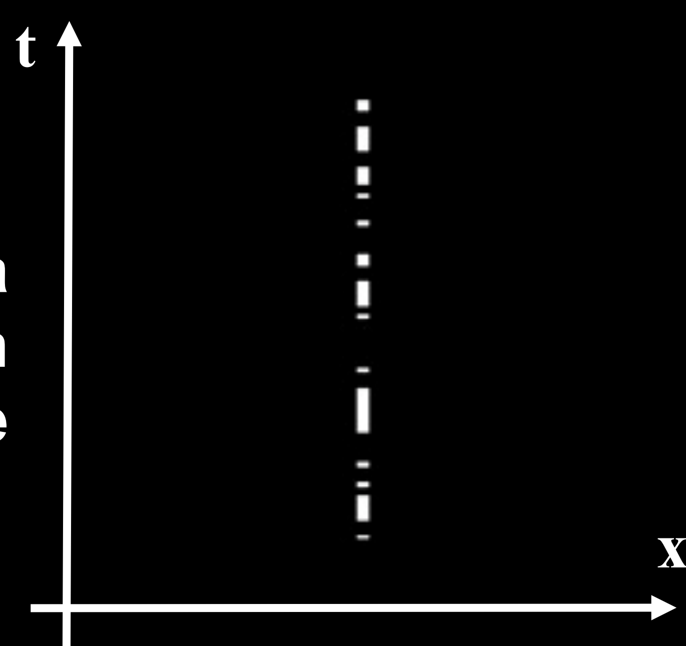
Flutter shutter (Raskar et al 2006)

Primal Domain

Objects

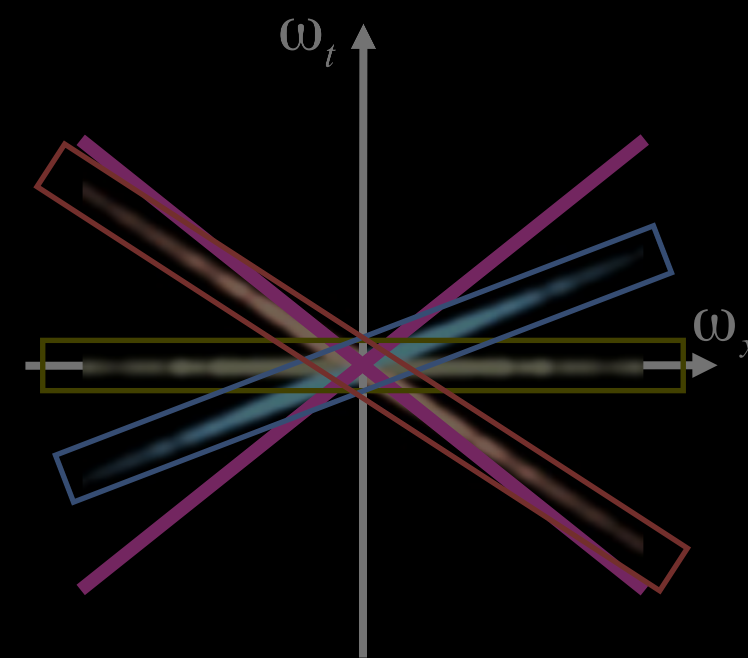


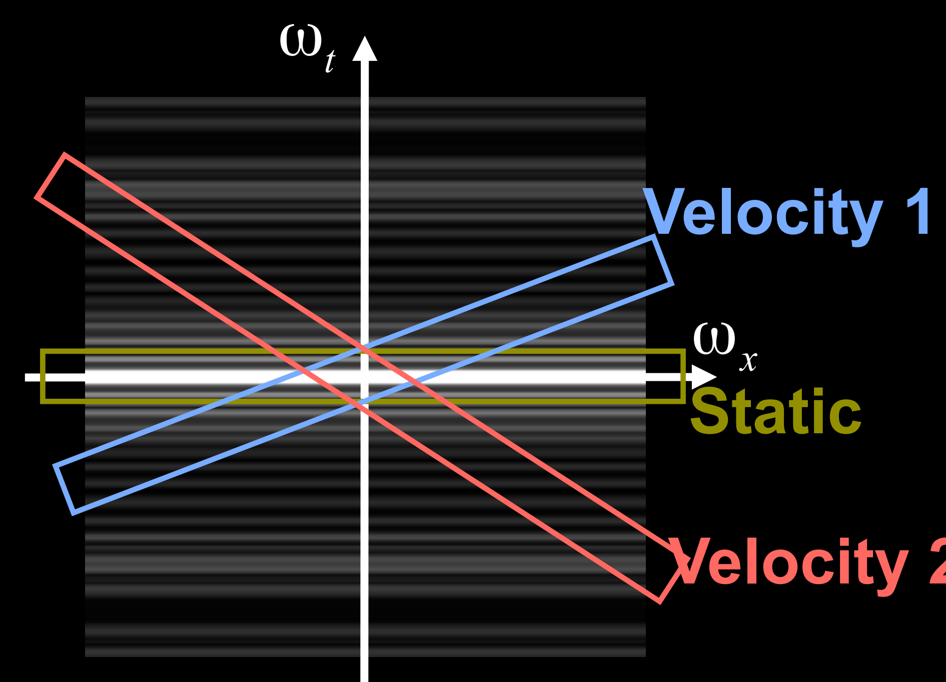
Camera integration curve



Vertical but discontinuous integration segment

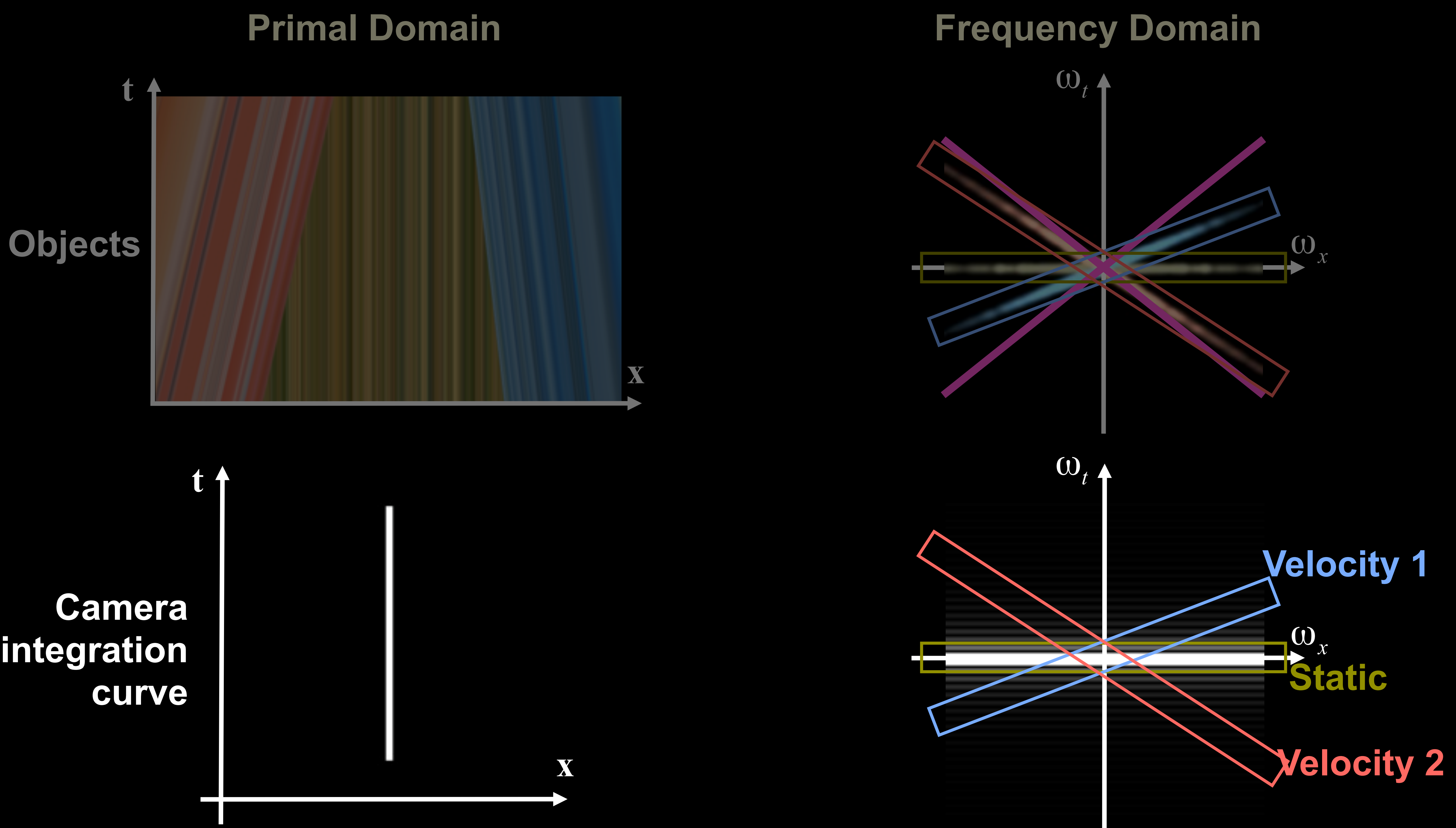
Frequency Domain





Higher velocities: better than static camera

Static camera



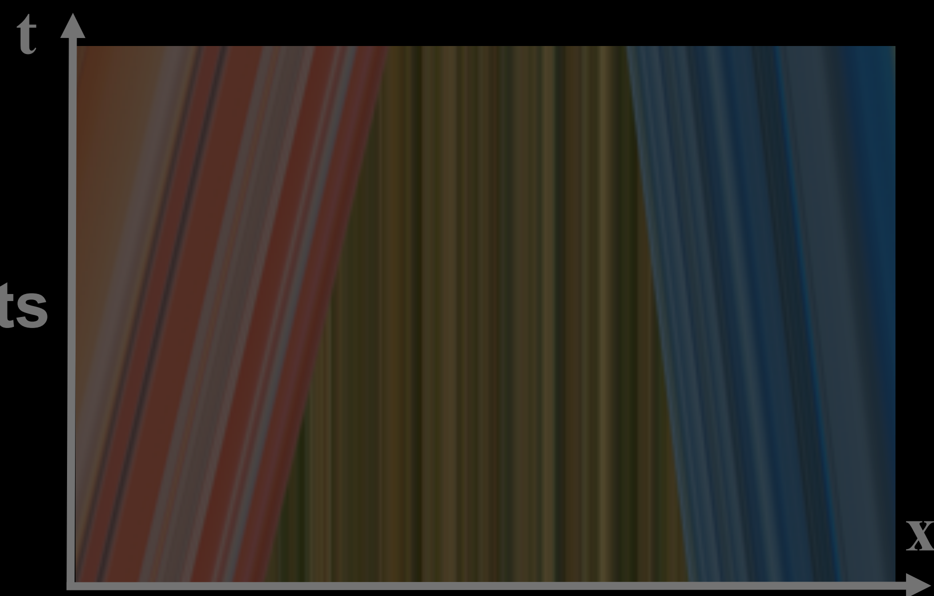
Vertical integration segment

Static object: high response
Higher velocities: low

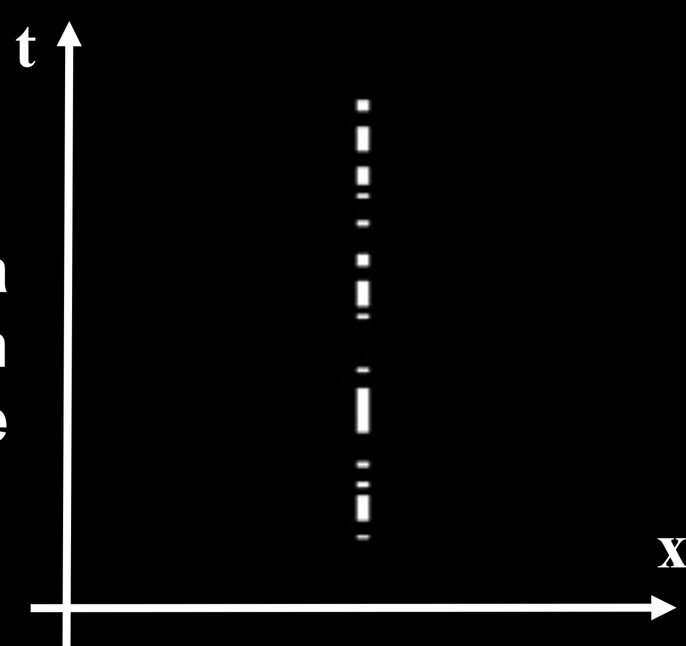
Flutter shutter (Raskar et al 2006)

Primal Domain

Objects

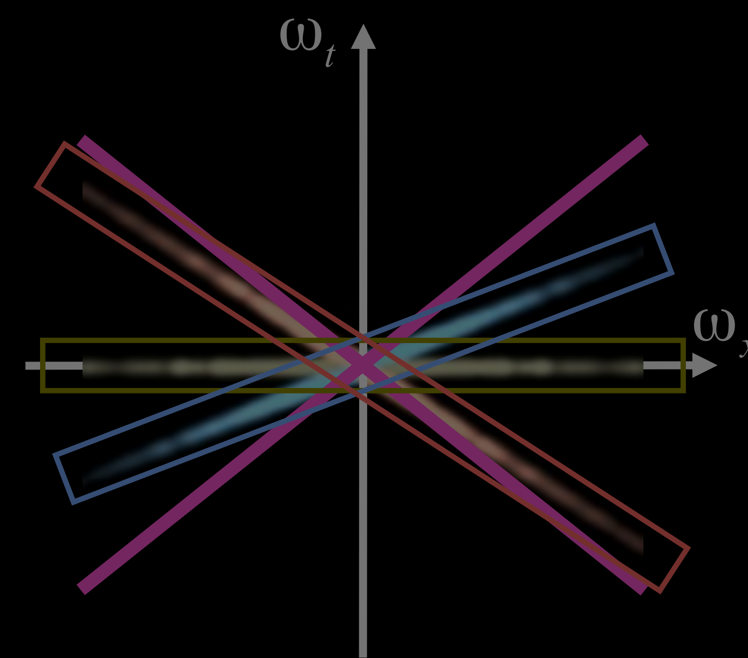


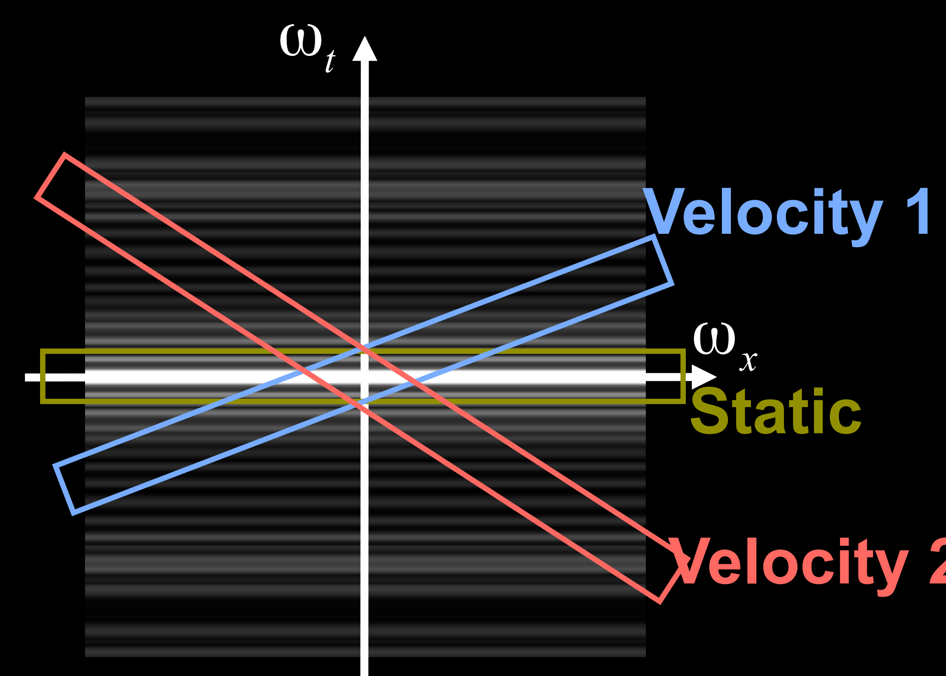
Camera integration curve



Vertical but discontinuous integration segment

Frequency Domain





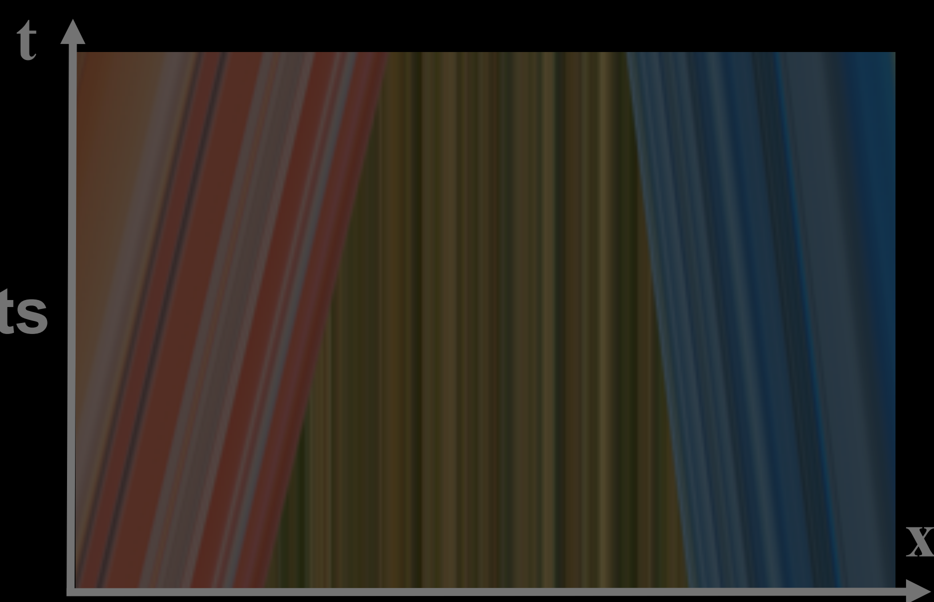
Higher velocities: better than static camera

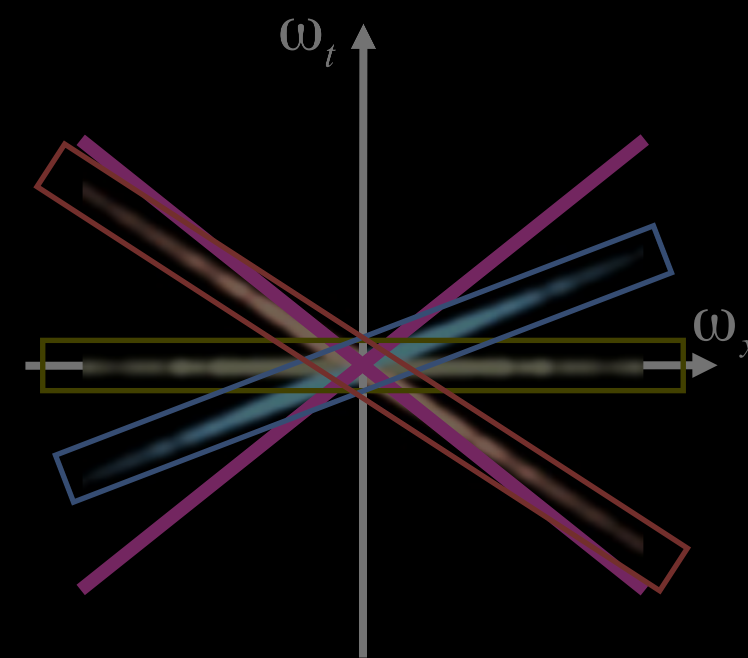
Our parabolic camera

Primal Domain

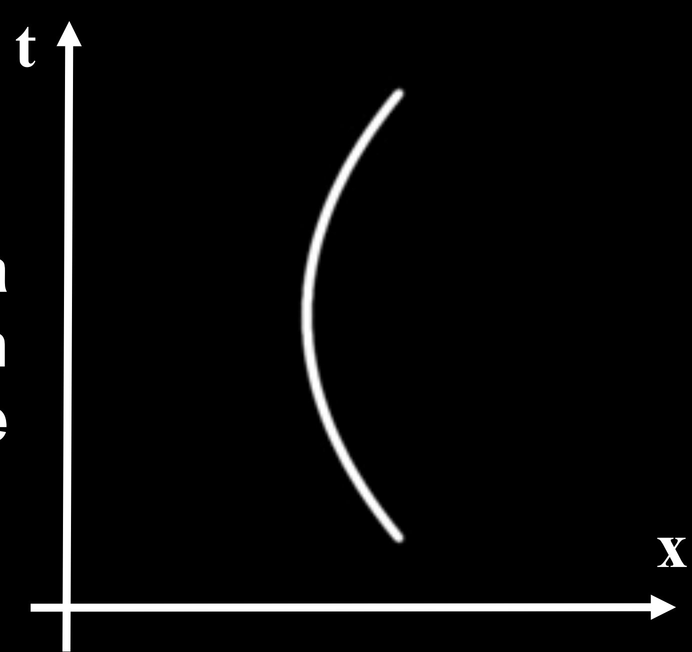
Frequency Domain

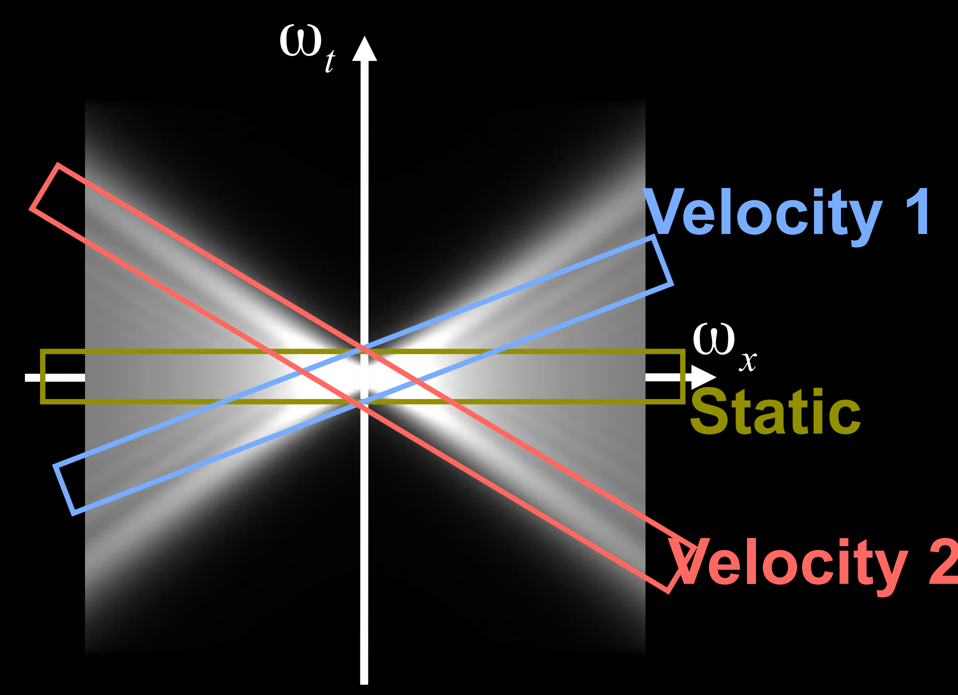
Objects





Camera integration curve





Parabola

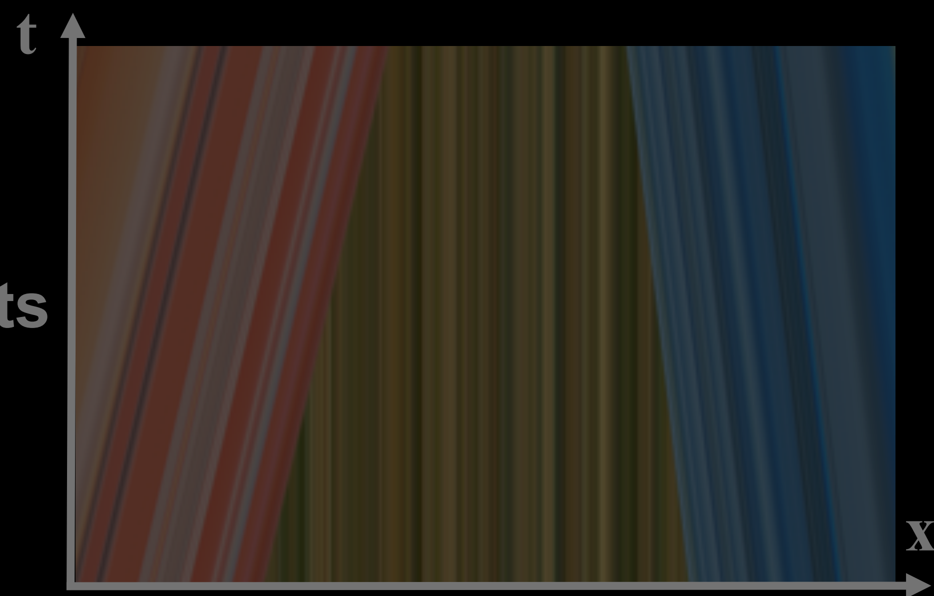
Equal high response in all range

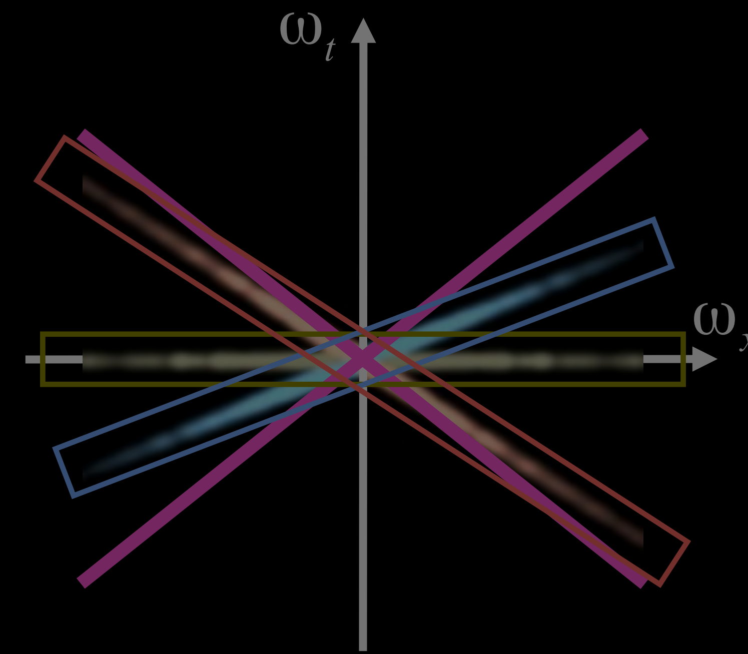
Flutter shutter (Raskar et al 2006)

Primal Domain

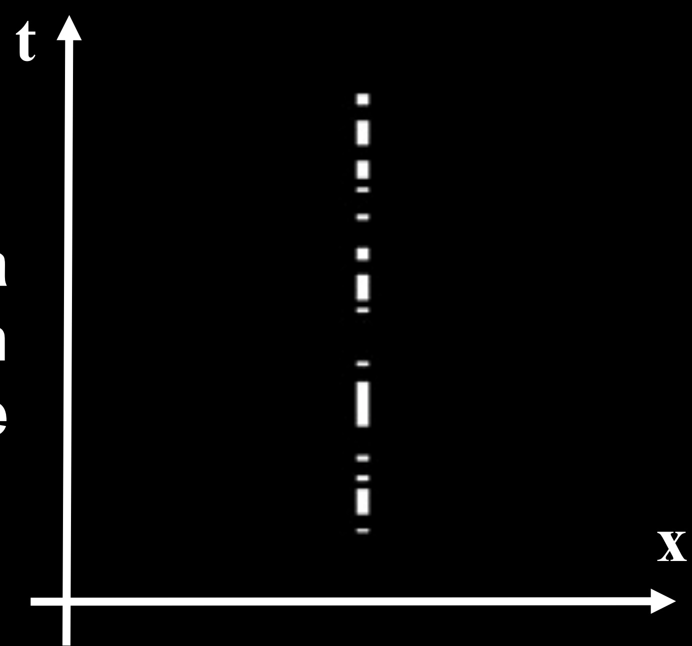
Frequency Domain

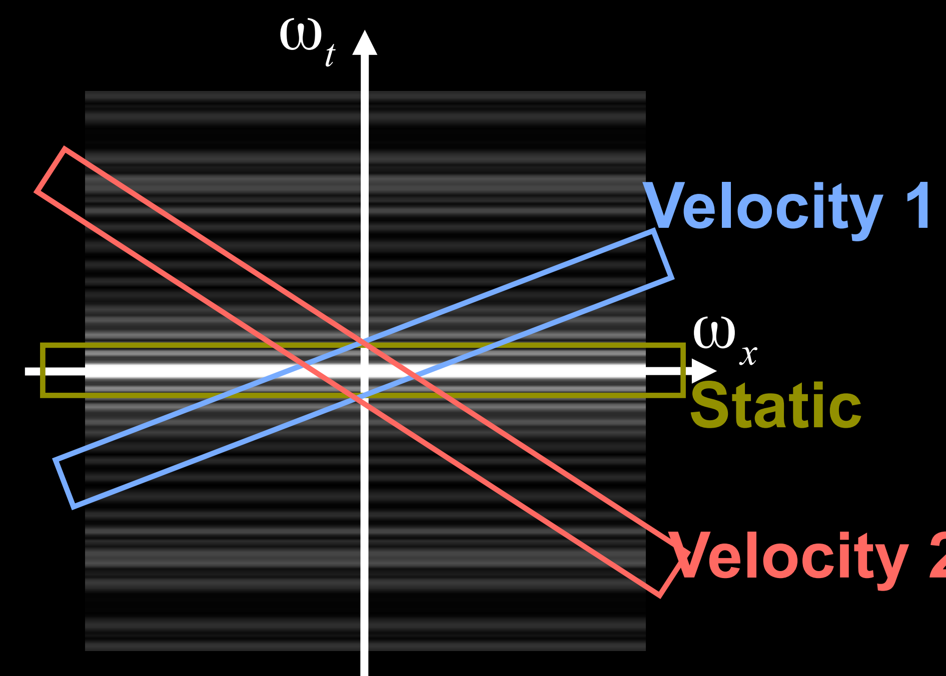
Objects





Camera integration curve





Vertical but discontinuous integration segment

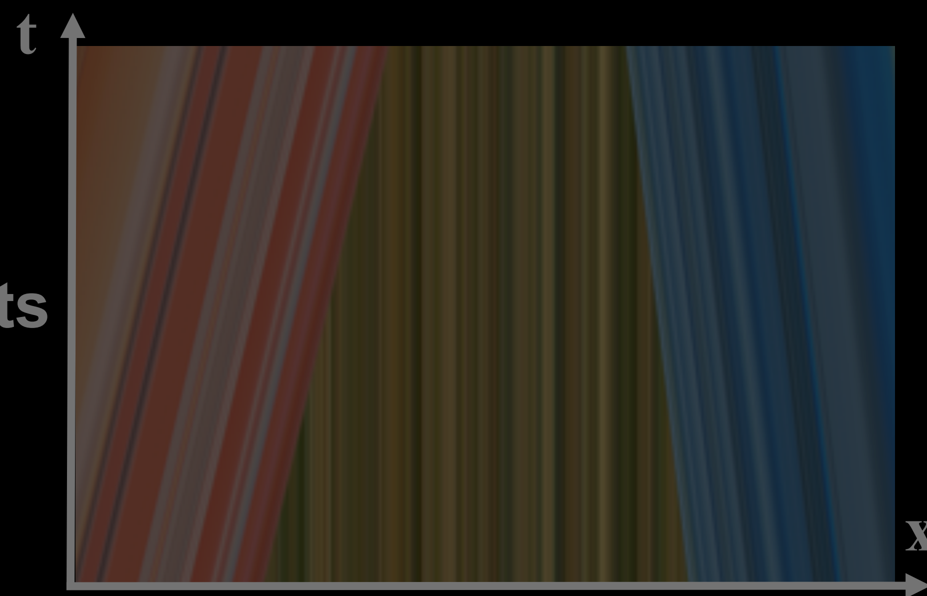
Higher velocities: better than static camera

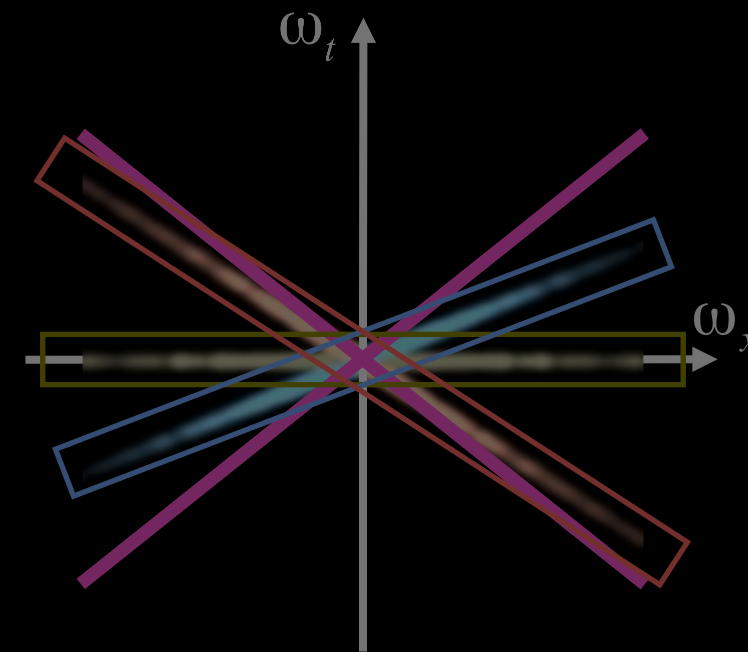
Our parabolic camera

Primal Domain

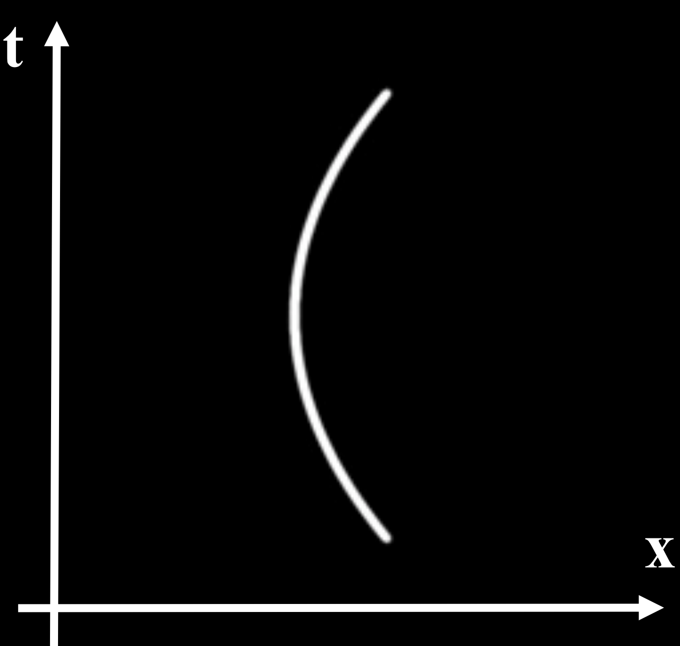
Frequency Domain

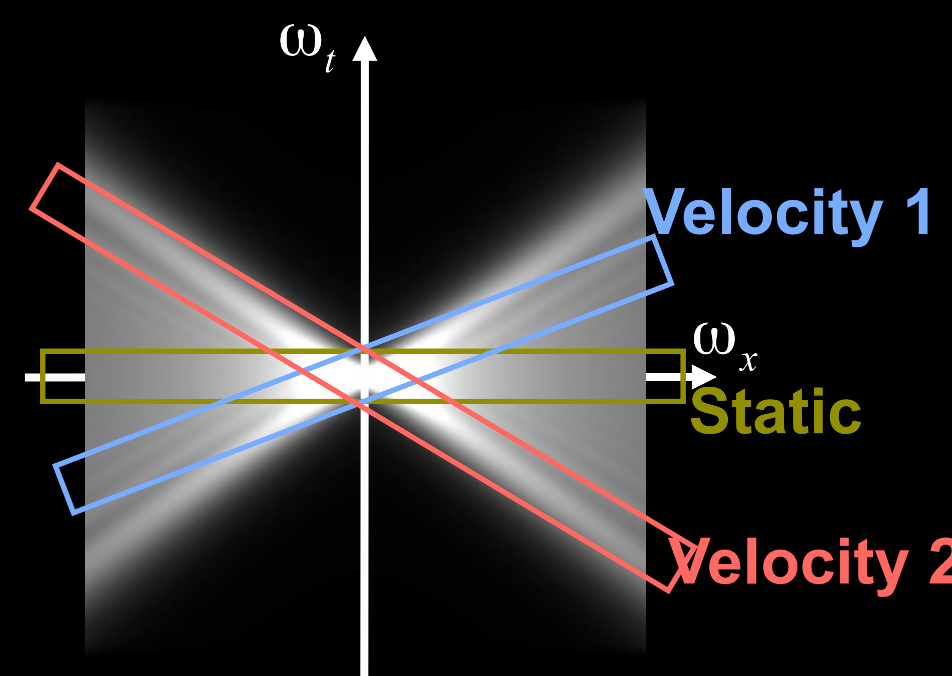
Objects





Camera integration curve





Parabola

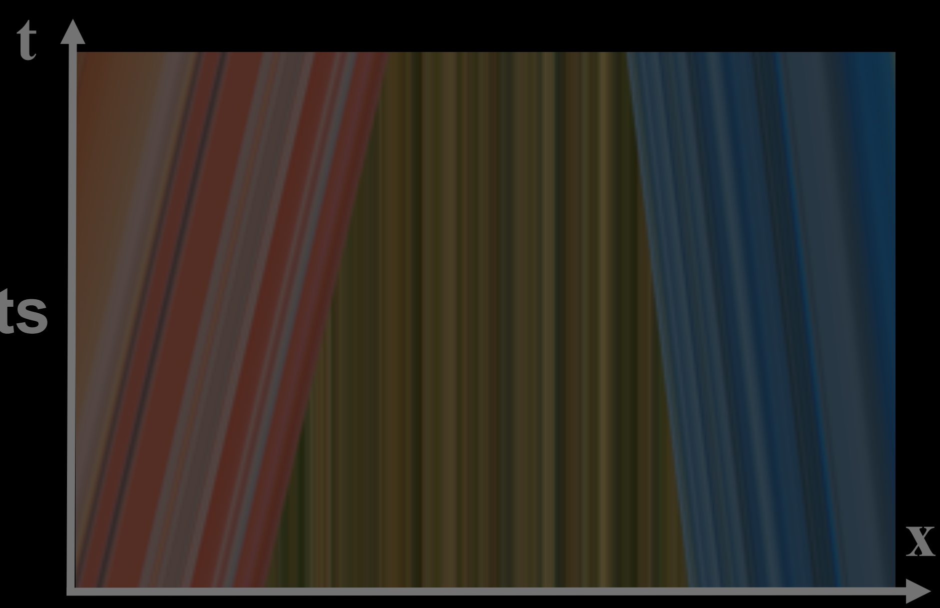
Equal high response in all range

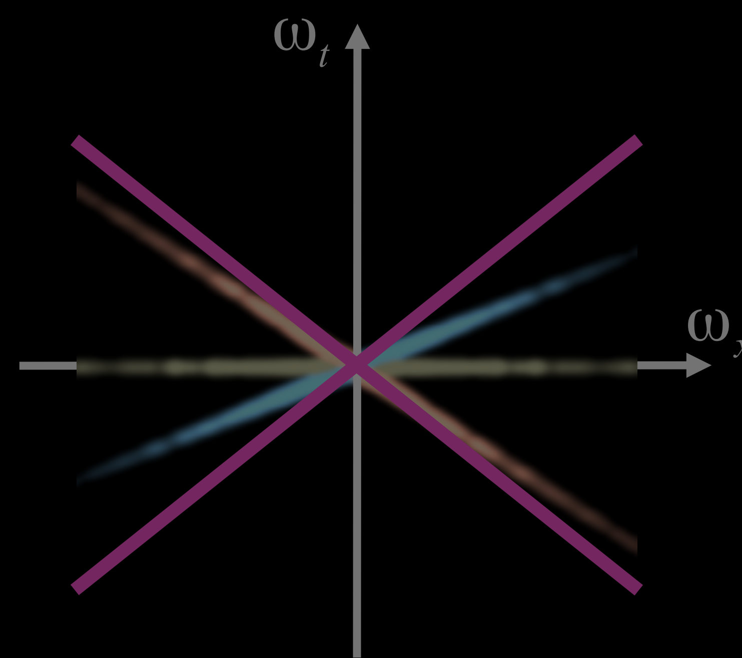
Information budget

Primal Domain

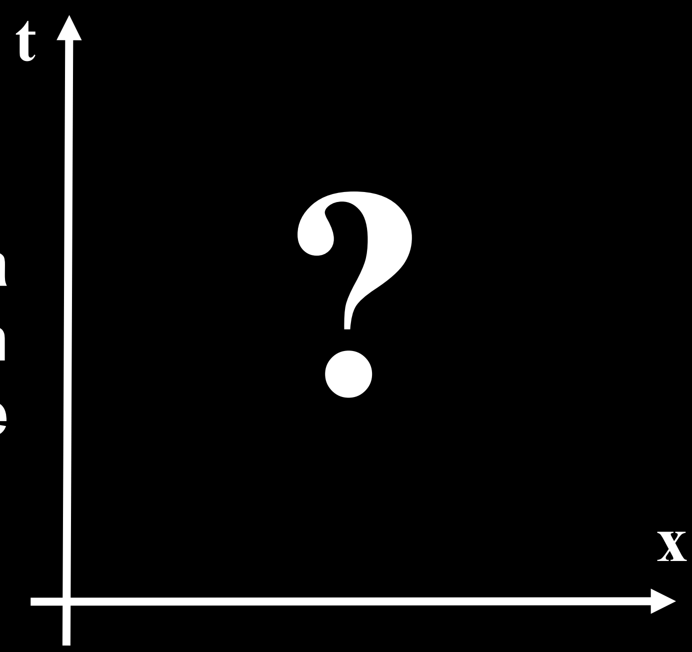
Frequency Domain

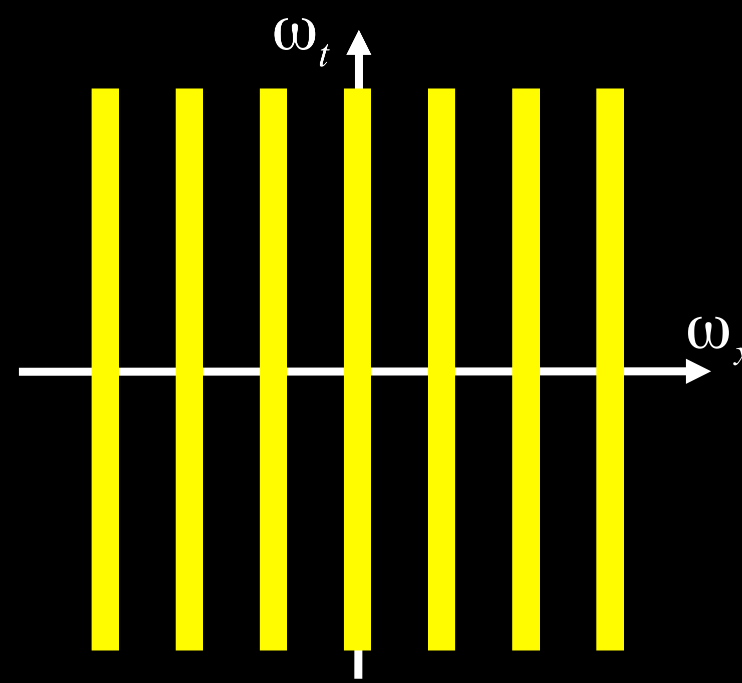
Objects





Camera integration curve

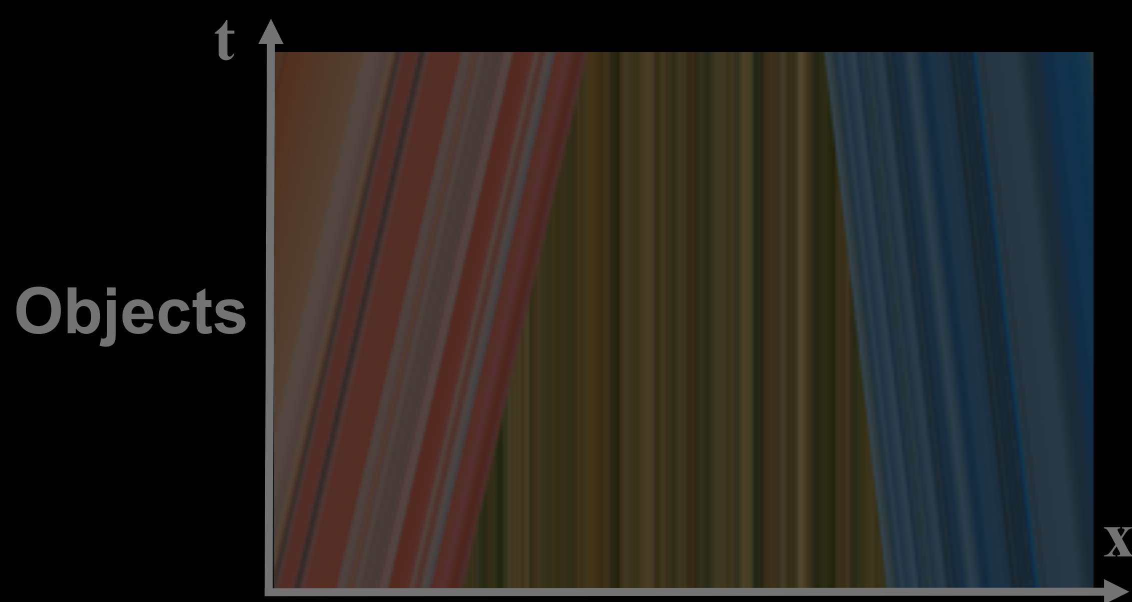


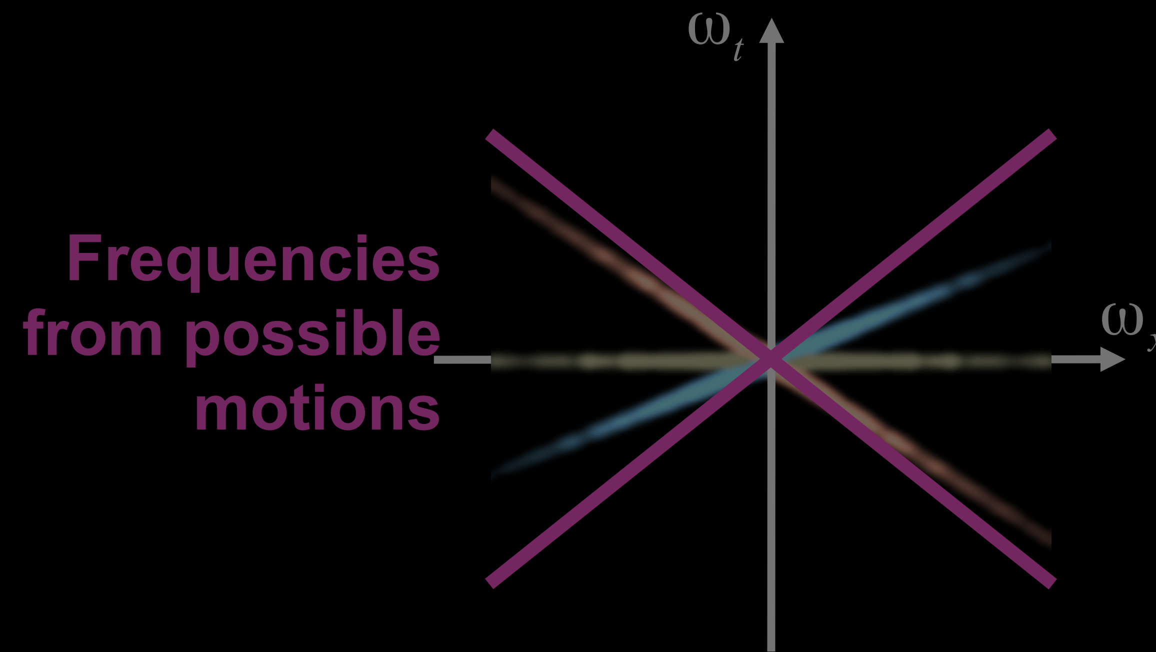


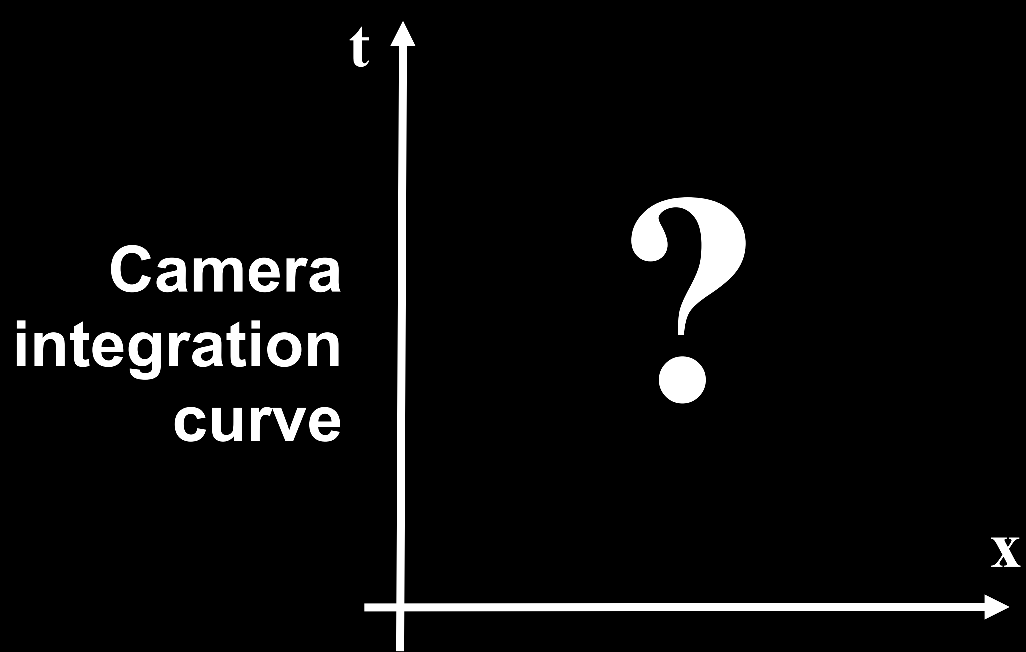
Bounded number of photons ➡ **Bounded budget per column (norm of power spectrum)**

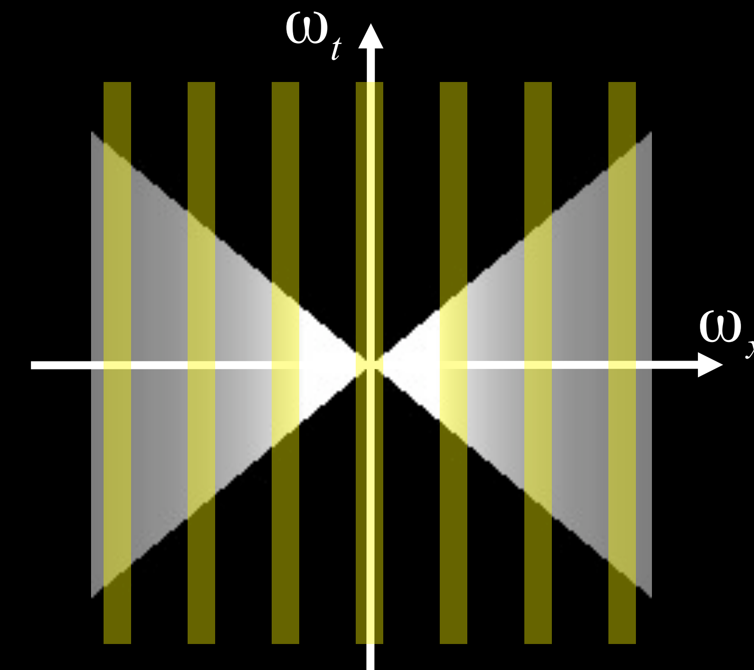
Upper bound given velocity range

Primal Domain | **Frequency Domain**



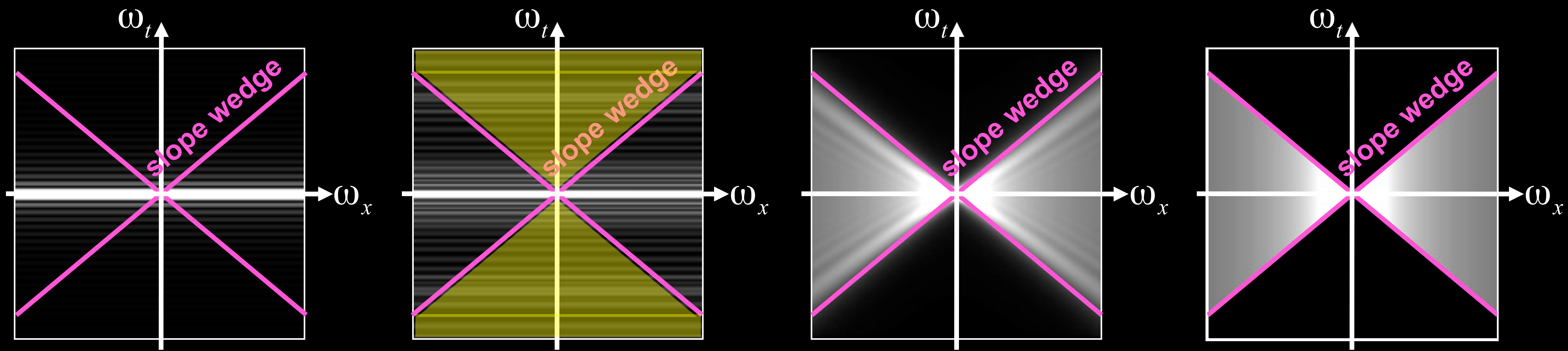






For each column, distribute budget uniformly within wedge $|K(\omega_x, \omega_t)|^2 \le \frac{1}{|\omega_x|}$

Cameras and information preservation



Static

Flutter shutter

Parabolic

Upper bound

Constant horizontally

Spends frequency "budget" outside wedge

Handles 2D motion

Near optimal "budget" usage at all frequencies

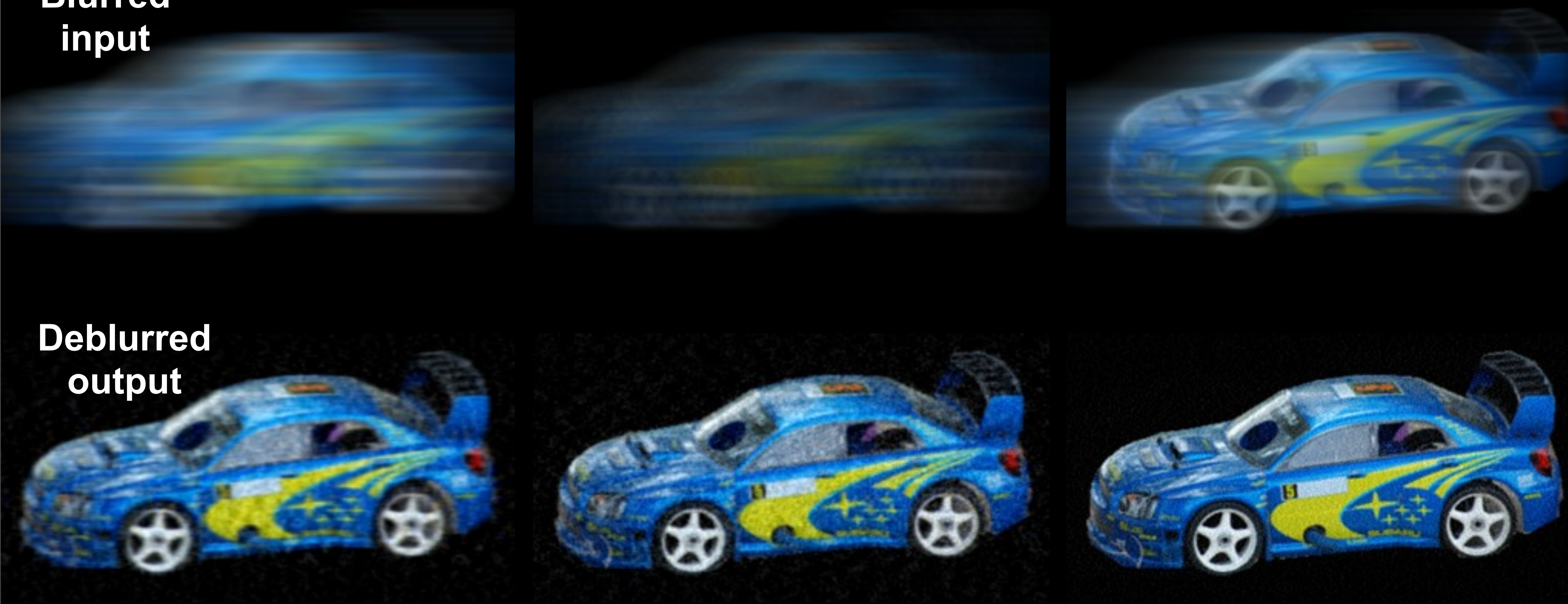
Bounded "budget" per column ω_x

Comparing camera reconstruction

Static | **Flutter Shutter** | **Parabolic**

Blurred input

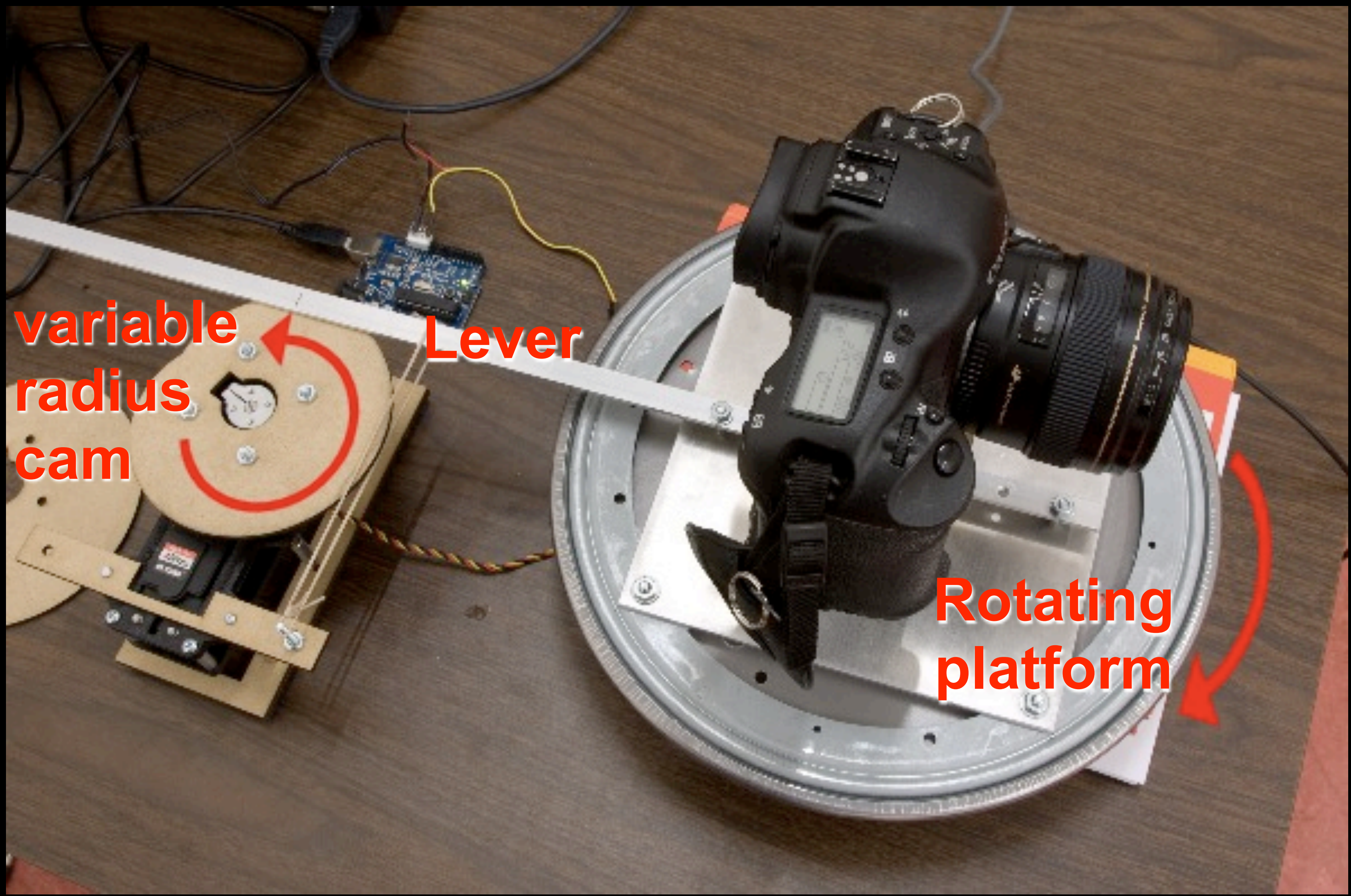
Deblurred output



Note: synthetic rendering, exact PSF is known

Hardware construction

- Ideally move sensor
  (requires same hardware as existing stabilization systems)
- In prototype implementation: rotate camera



Linear rail

Static camera input-
Unknown and variable blur

Our parabolic input- Blur
is invariant to velocity

Linear rail

Static camera input-
Unknown and variable blur

Our output after deblurring-
NON-BLIND deconvolution

Human motion– no perfect linearity

Input from a static camera

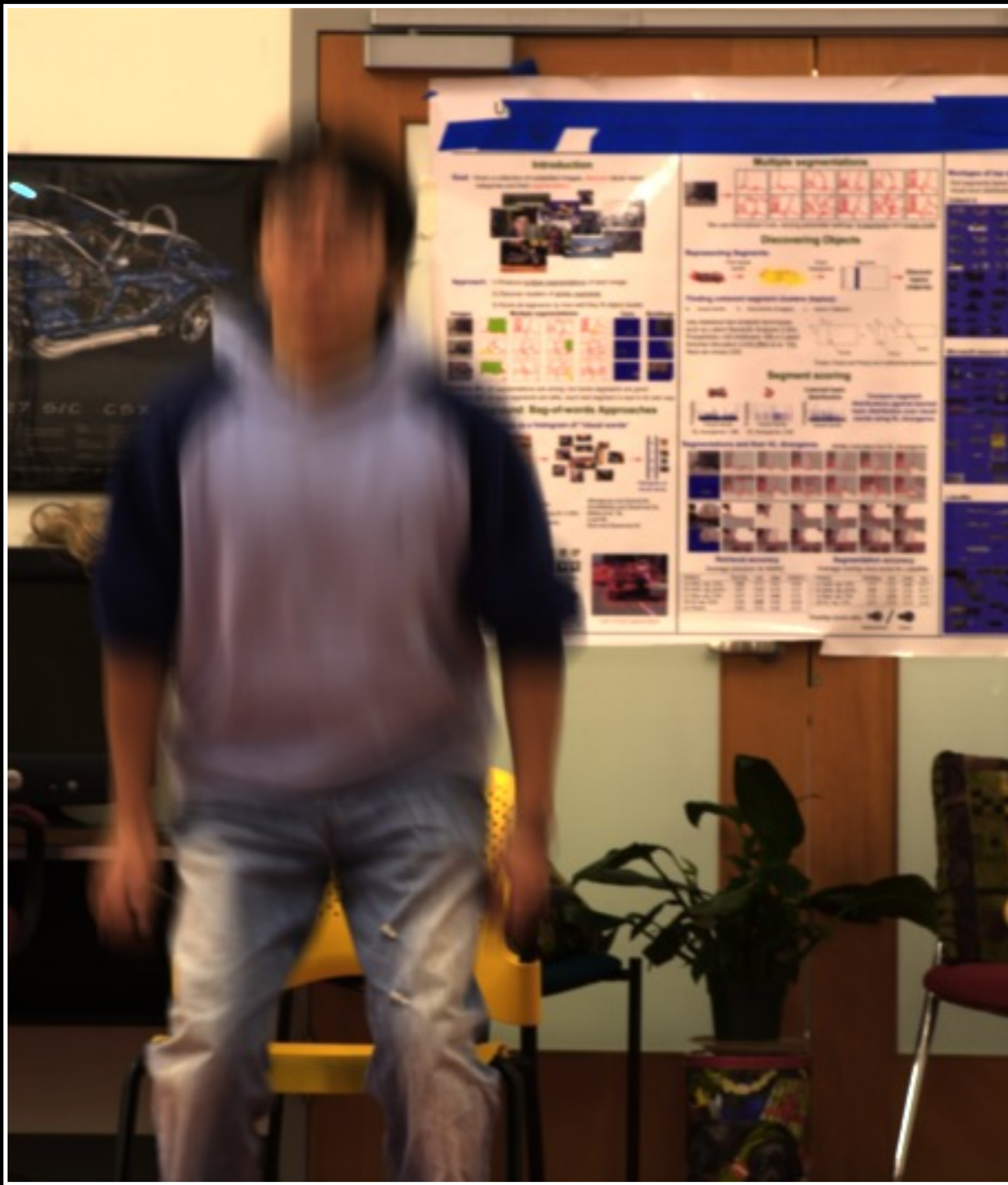
Deblurred output from our camera

Violating 1D motion assumption– forward motion

Input from a static camera

Deblurred output from our camera

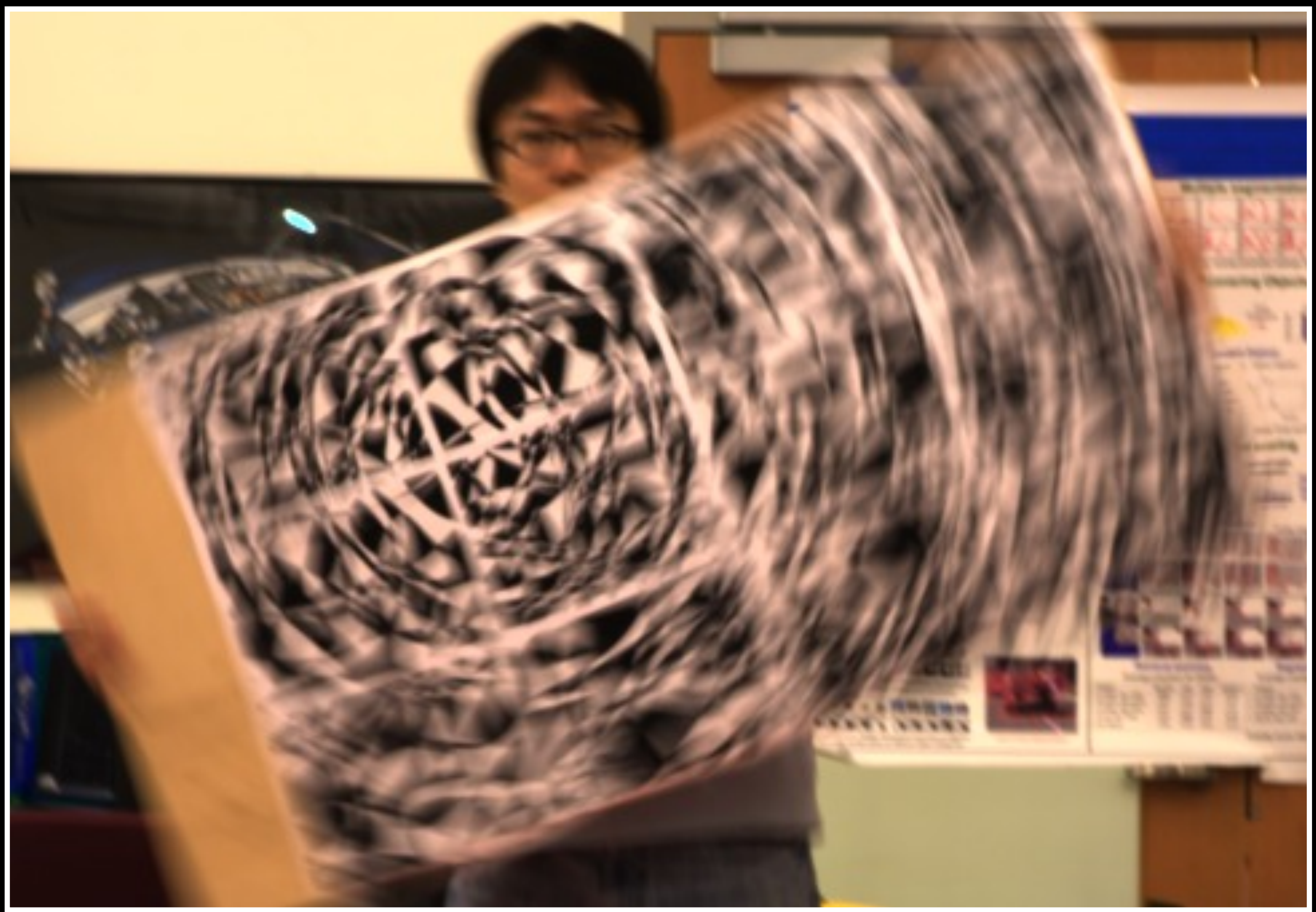
Violating 1D motion assumption- stand-up motion



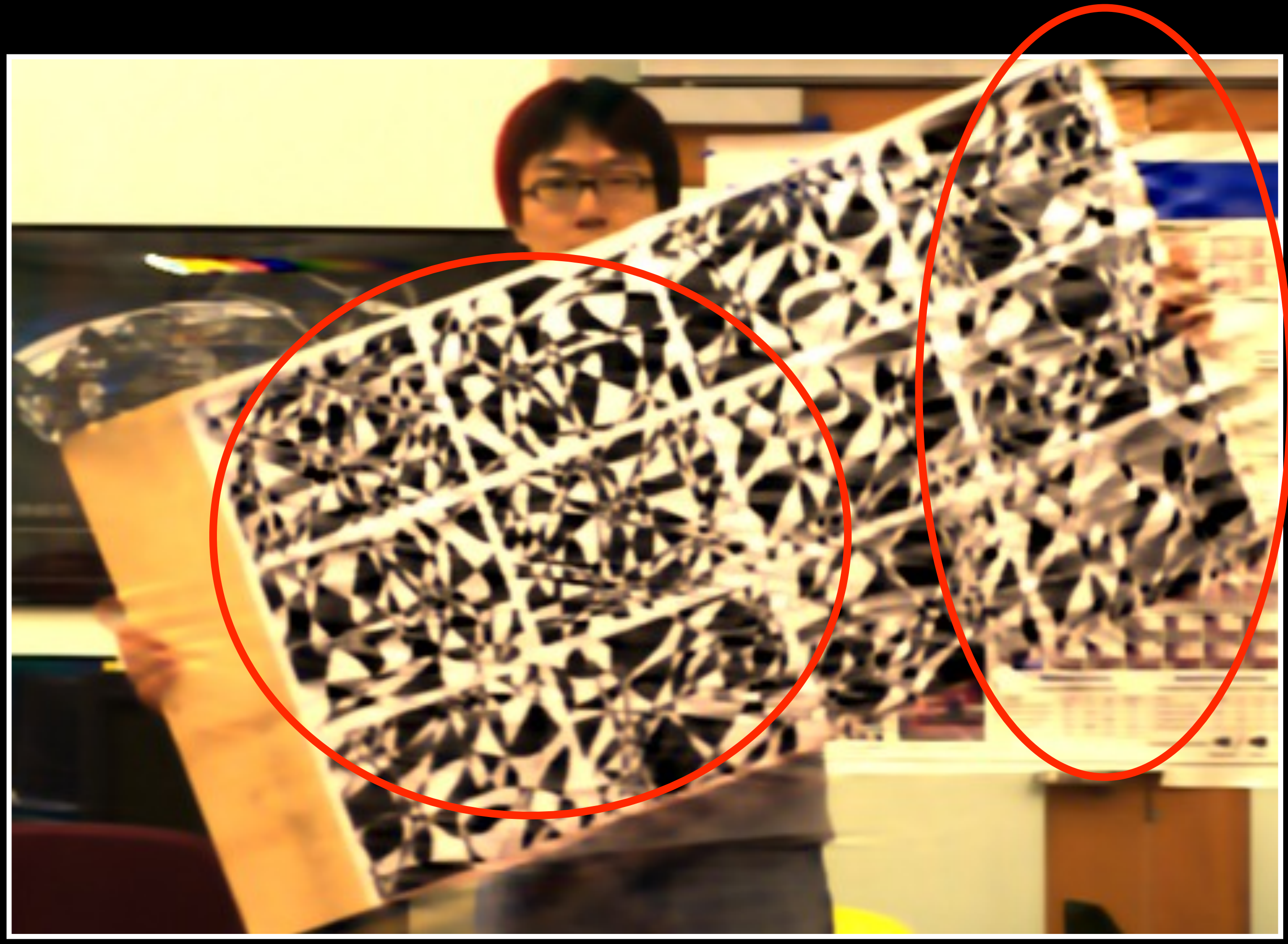
Input from a static camera

Deblurred output from our camera

Violating 1D motion assumption- rotation



Input from a static camera



Deblurred output from our camera

Parabolic curve - issues

- Spatial shift- but does not affect visual quality in deconvolution
- Parabola tail clipping: not exactly the same blur
- Motion boundaries break the convolution model
- Assumes: Object motion horizontal

  Object motion linear up to 1st order approximation

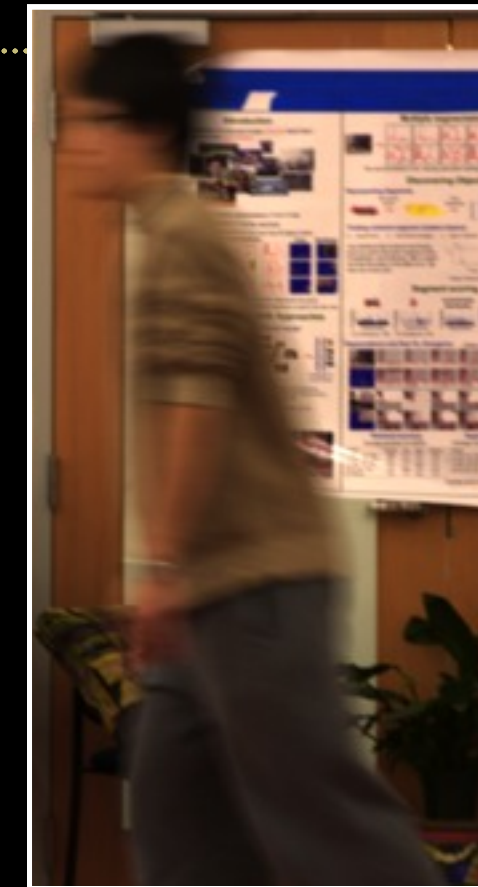
Conclusions

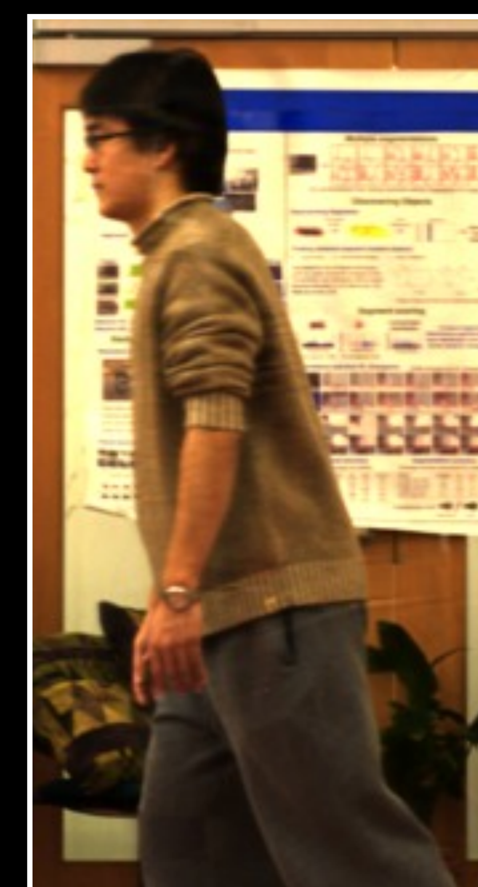
- Camera moved during exposure, parabolic displacement

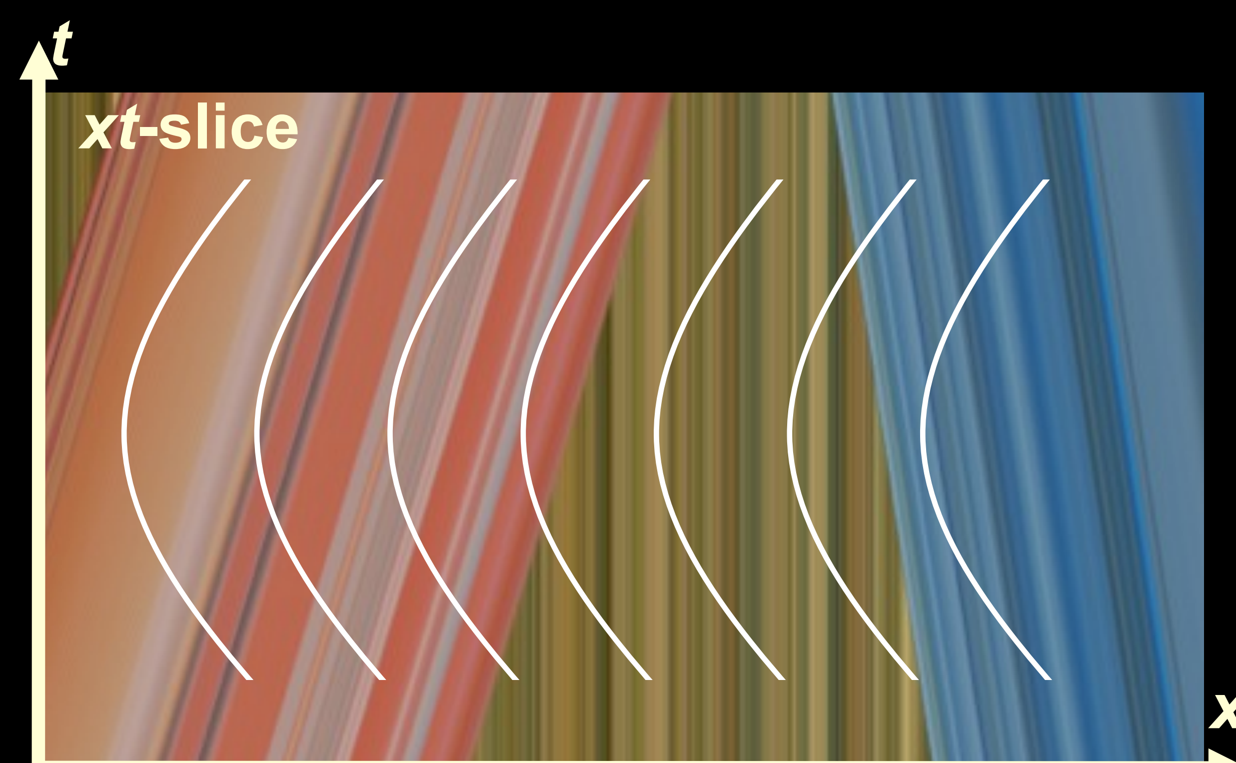
- Blur invariant to motion:
  - Same over all image (no need to segment)
  - Known in advance (no kernel identification)

- Easy to invert (near optimal frequency response)

- For 1D motion
  - Somewhat robust to 1D motion violation
  - Future work: 2D extensions







Acknowledgments:
NSF CAREER award 0447561
Royal Dutch/Shell Group NGA
NEGI-1582-04-0004
Office of Naval Research MURI
MSR New Faculty Fellowship
Sloan Fellowship

Plan

- Introduction of computational photography
  - Enhance our vision
  - Capture visual information
- **Motion Invariant Photography**
  - **Move sensor to make kernel**
    - **invariant**
    - **high frequency response**
  - **Upper bound**
- Potpourri
- Big Ideas in Computational Photography

Plan

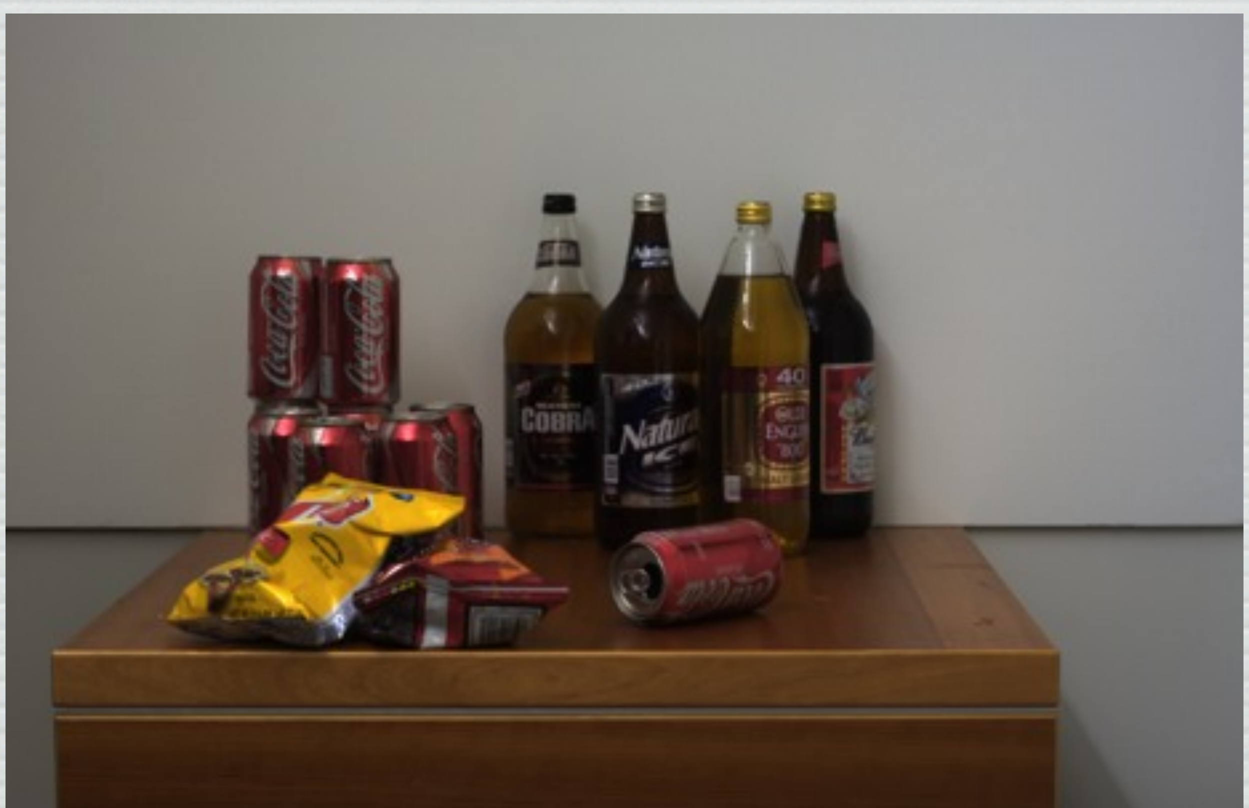
- Introduction of computational photography
  - Enhance our vision
  - Capture visual information
- Motion Invariant Photography
  - Move sensor to make kernel
    - invariant
    - high frequency response
  - Upper bound
- **Potpourri**
- Big Ideas in Computational Photography

Image and Depth from a Conventional Camera with a Coded Aperture



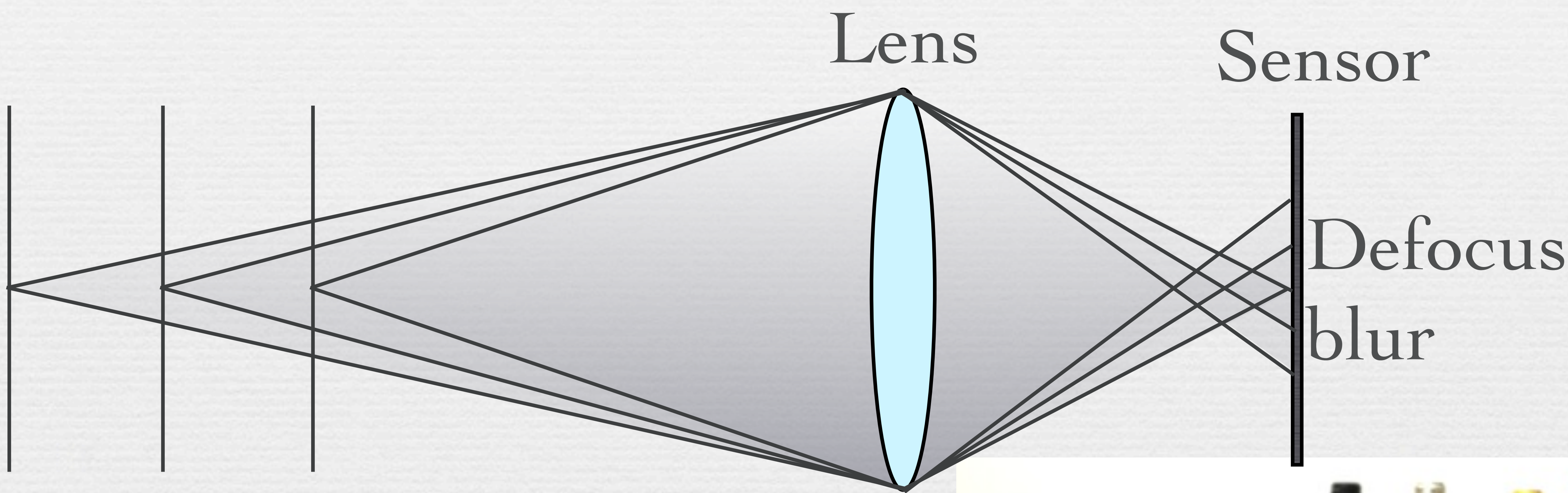
with Levin, Fergus, Freeman [Siggraph 2007]

RGB & coarse depth from single image

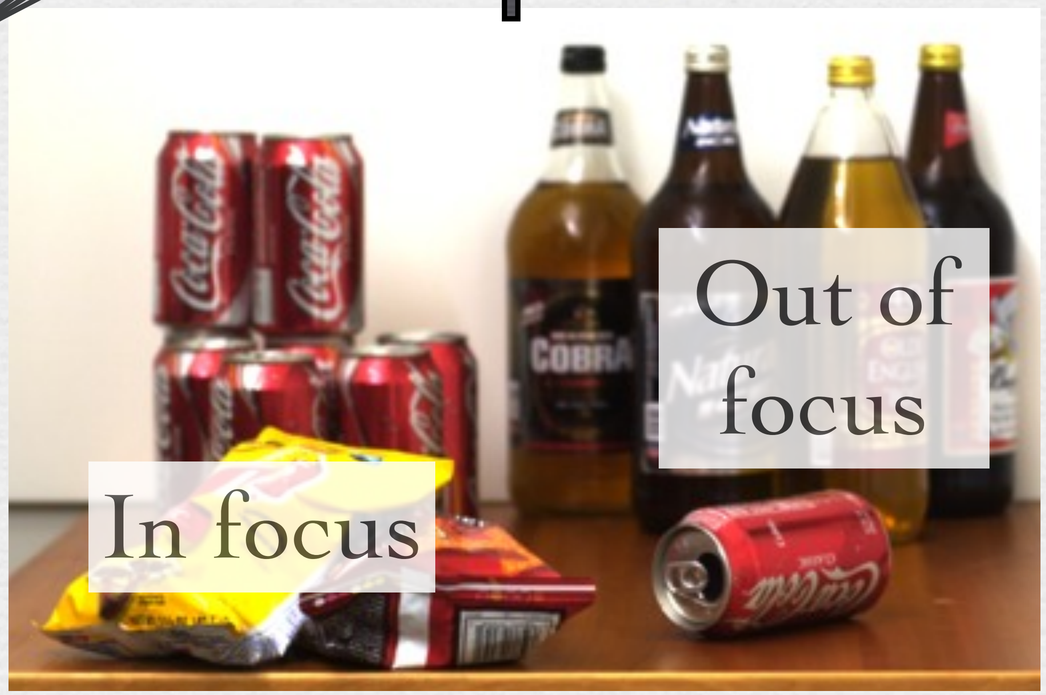


Defocus & depth

- Objects far from focusing distance are blurrier

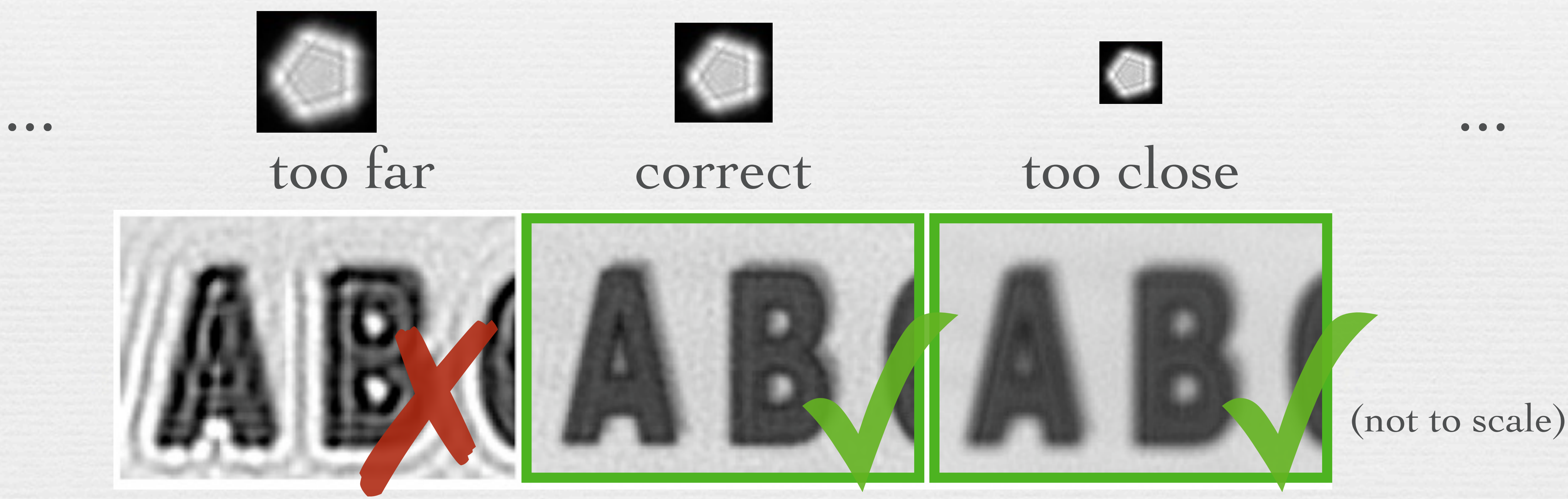


- **By inferring blur, we can infer depth**



Challenge: hard to infer depth

- For each candidate depth,
  try to deconvolve with corresponding kernel



- **“Too close” not so different from “correct”**

Opposite solution: Coded aperture

- Increase kernel variation:
  Put a mask (code) on aperture plane (diaphragm)
  - ➡ more structured blur
  - ➡ easier to identify kernel/depth
  - ➡ easier to remove blur

Conventional

Coded

Why code helps

- Wrong blur is making more mess

too far correct too close

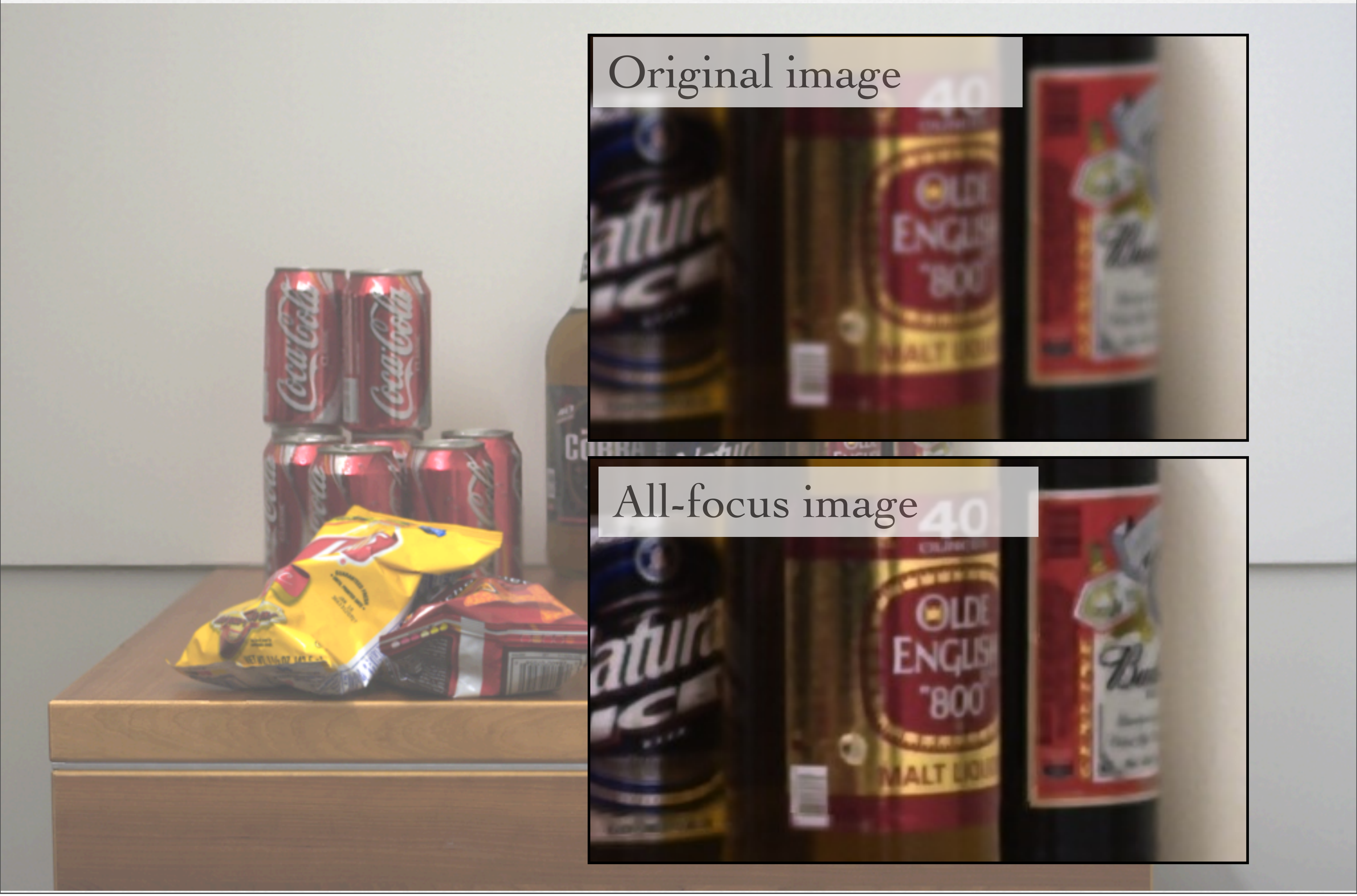
Conventional aperture

Coded aperture

Input

Deconvolved (all-focus)

Close up

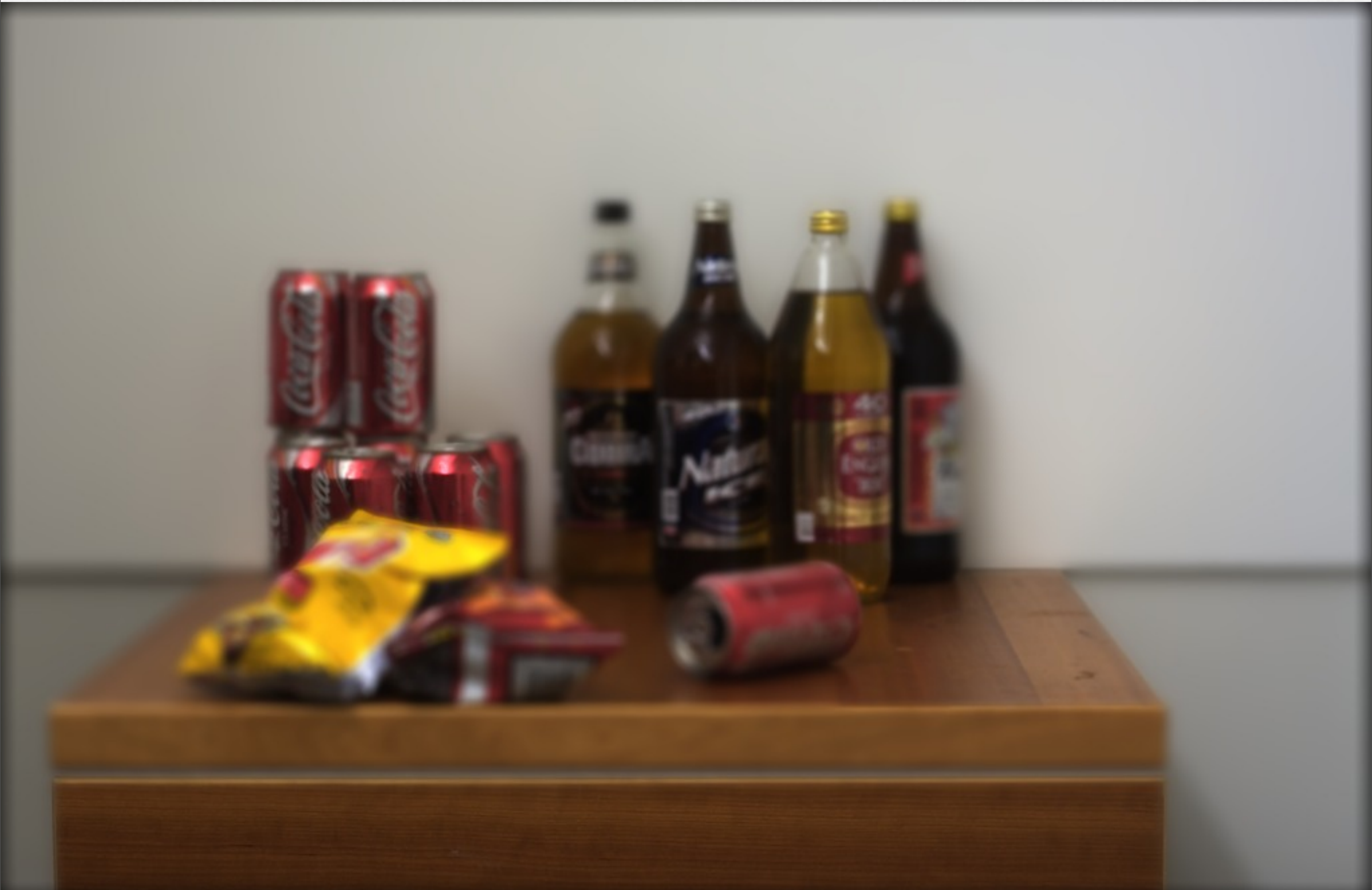


Depth

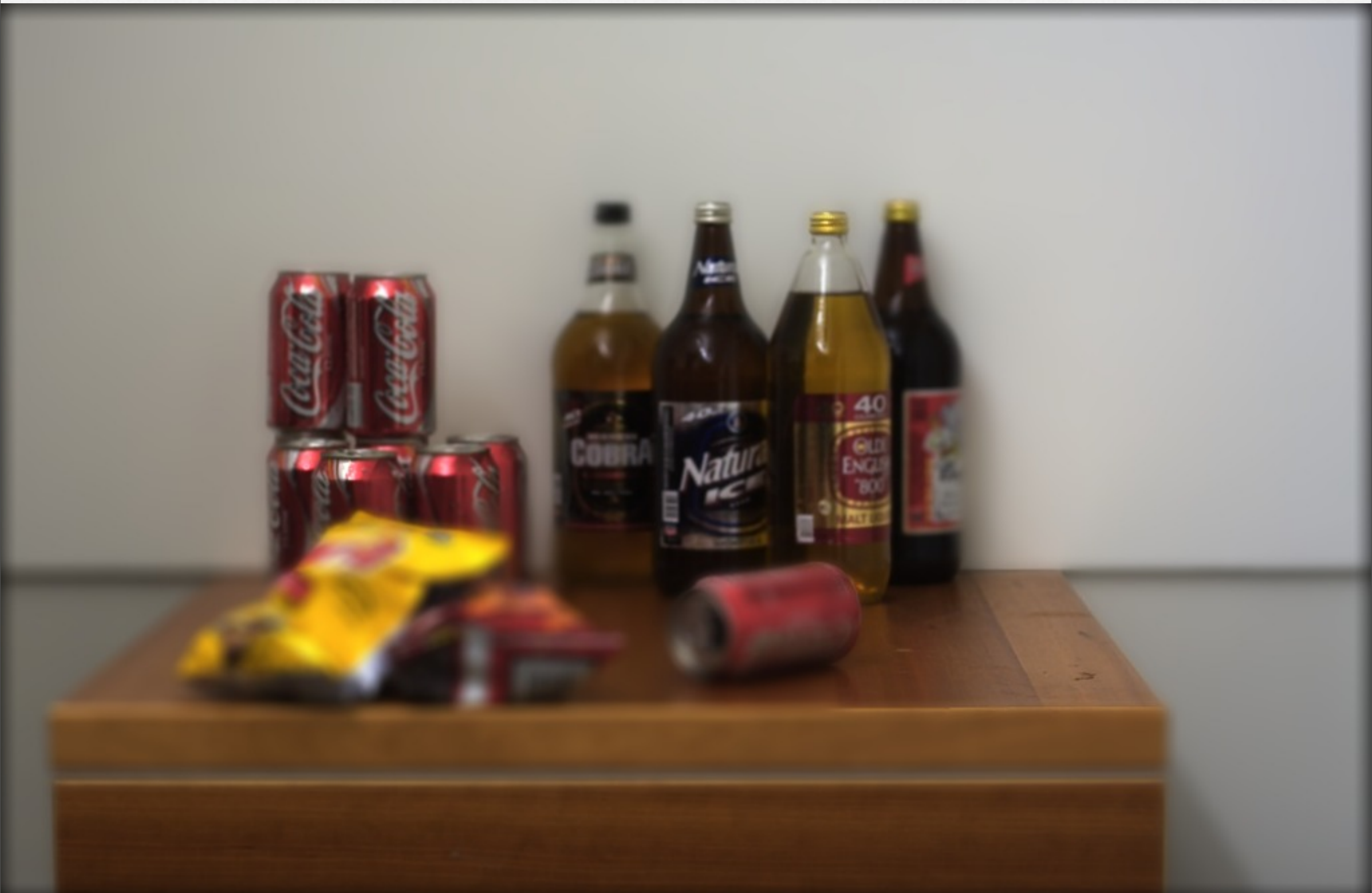
Deconvolved (all-focus)

Refocusing (from single image!)

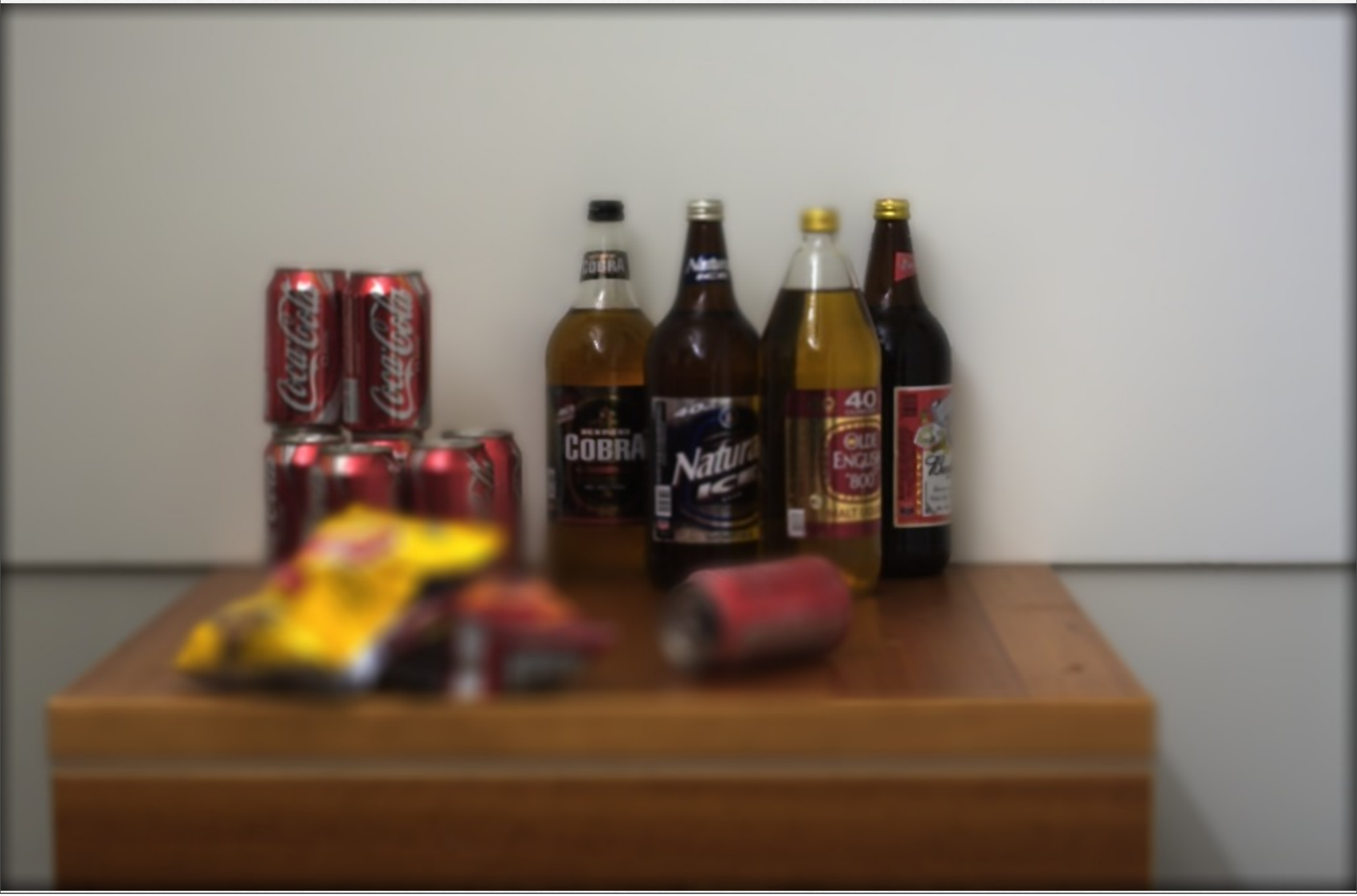
Refocusing (from single image!)



Refocusing (from single image!)



Refocusing (from single image!)



Results

Input

Deconvolved

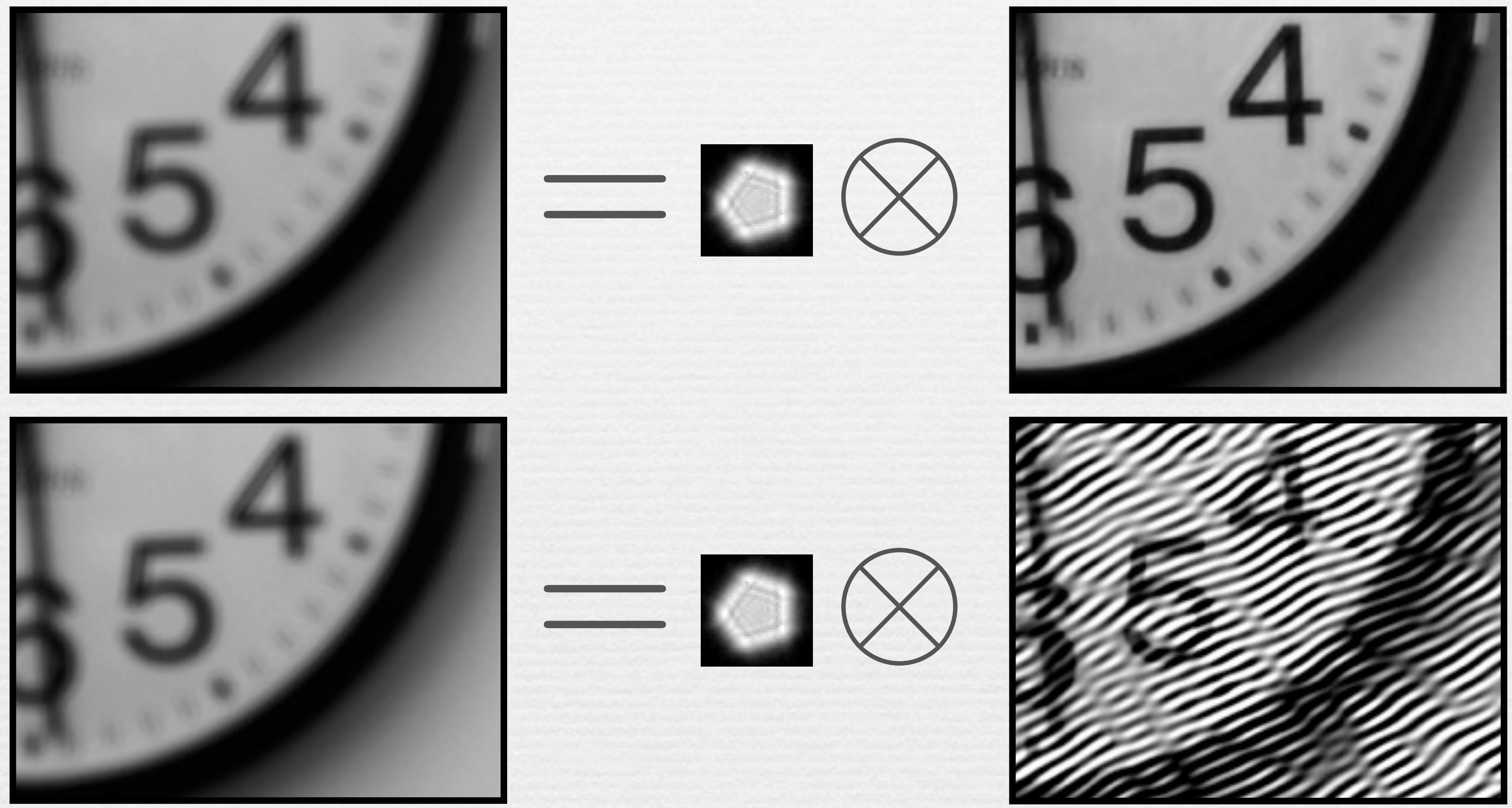
Refocusing (from single image!)

Refocusing (from single image!)

Refocusing (from single image!)

Deconvolution given kernel

- Ill-posed



- Traditional algorithms lead to ringing

Richardson-Lucy
deconvolution

Idea: Natural image prior

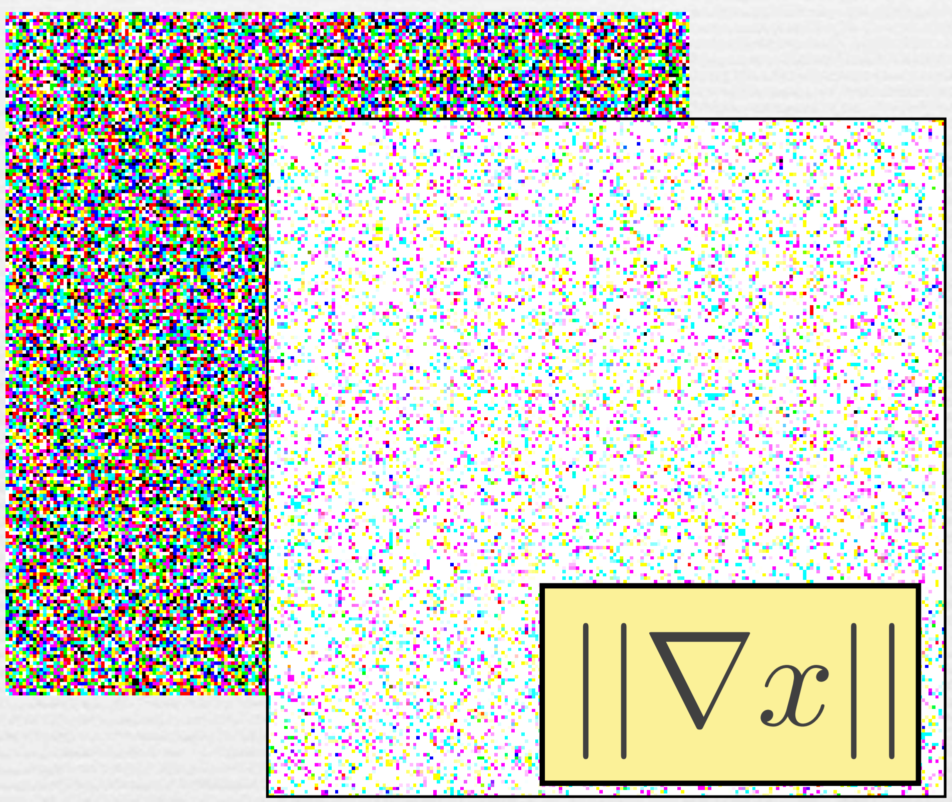
- Random 2D arrays of colors don't look like the world around us

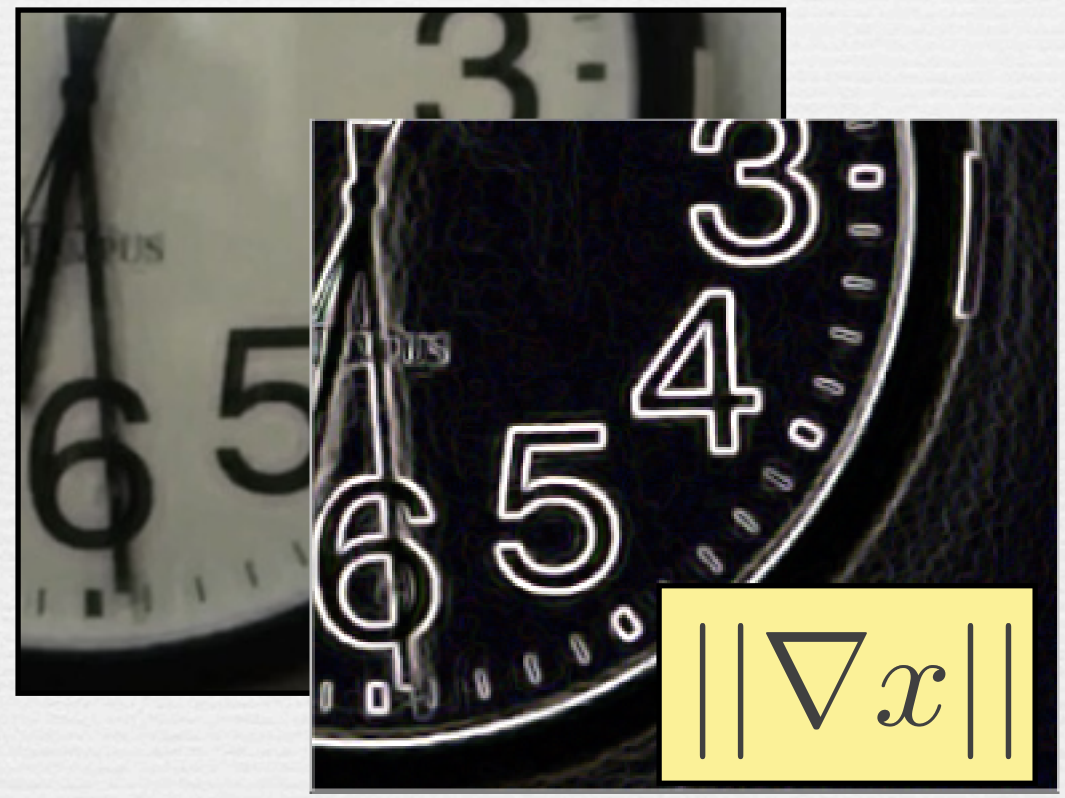
ADD CITATIONS

- If we can characterize natural images, we can bias algorithms to output better results

Sparse derivatives prior

- Natural images have sparse derivative
  (the gradient is small almost everywhere)

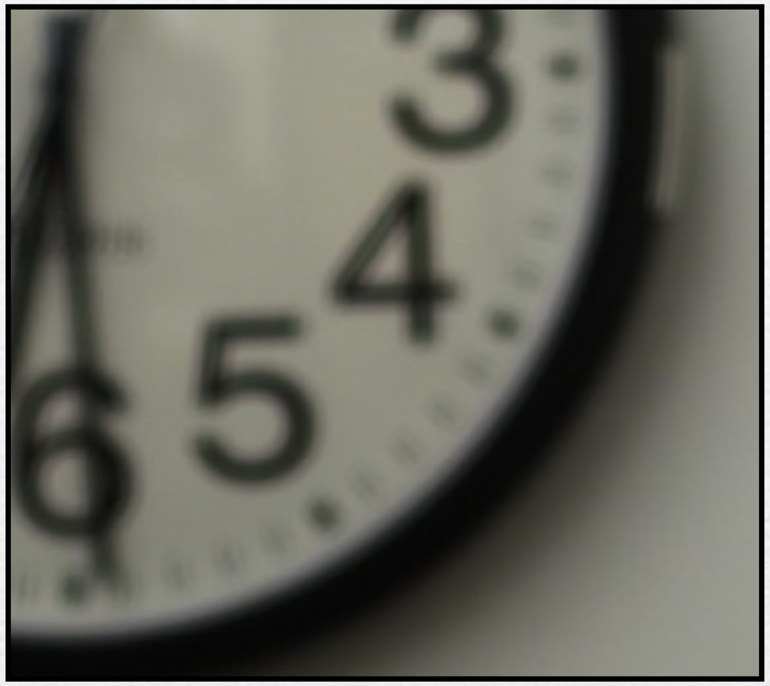




- Add an optimization term
  (a.k.a. regularization)
  Pay penalty where gradient is non-zero

$$\sum ||\nabla x||^{0.8}$$

Sparsity prior for deconvolution



Input

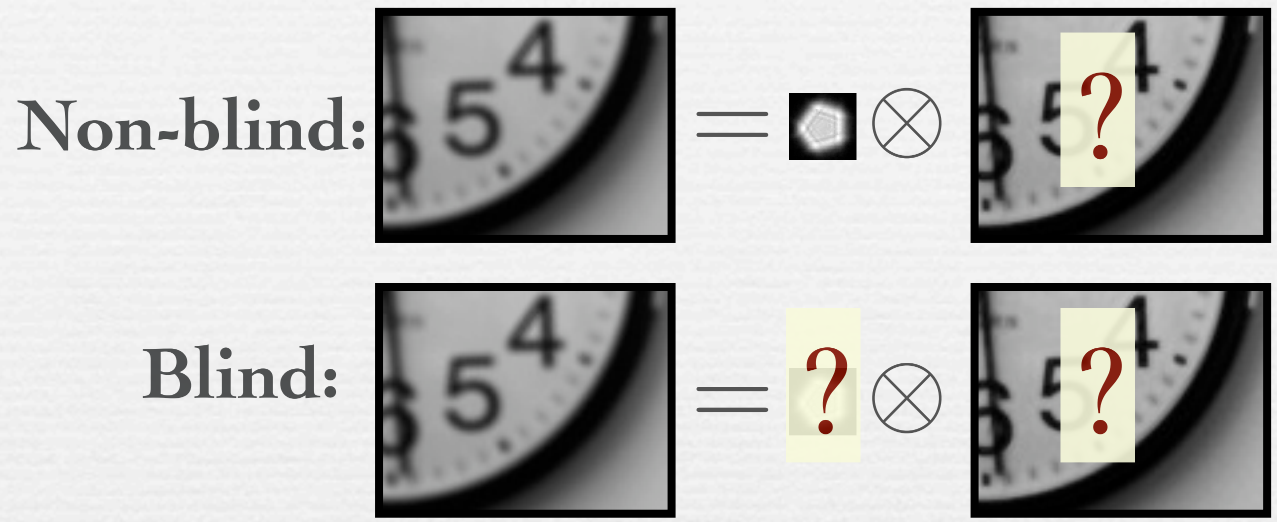
$$\sum ||\nabla x||^{0.8}$$

Richardson-Lucy

Our sparse prior

Blind Deconvolution

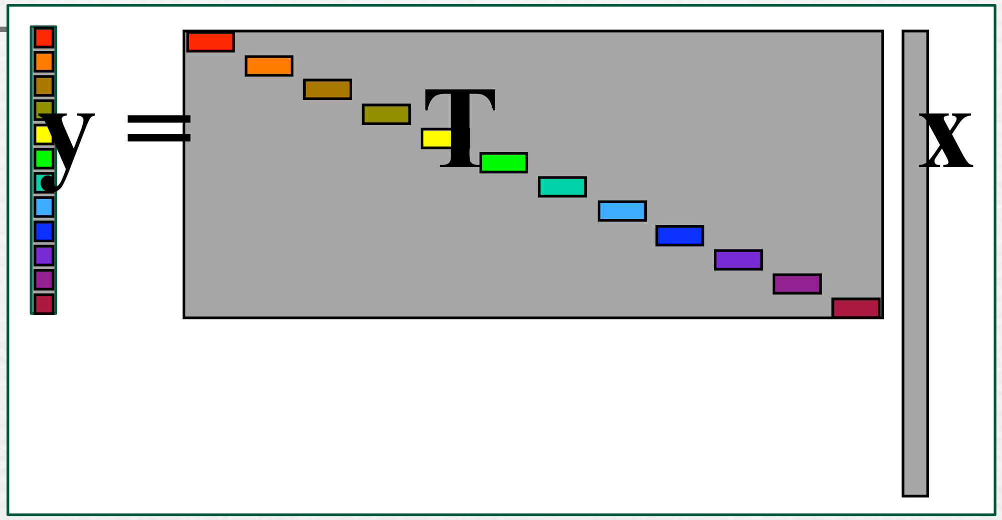
- *[Levin et al. CVPR 09]*

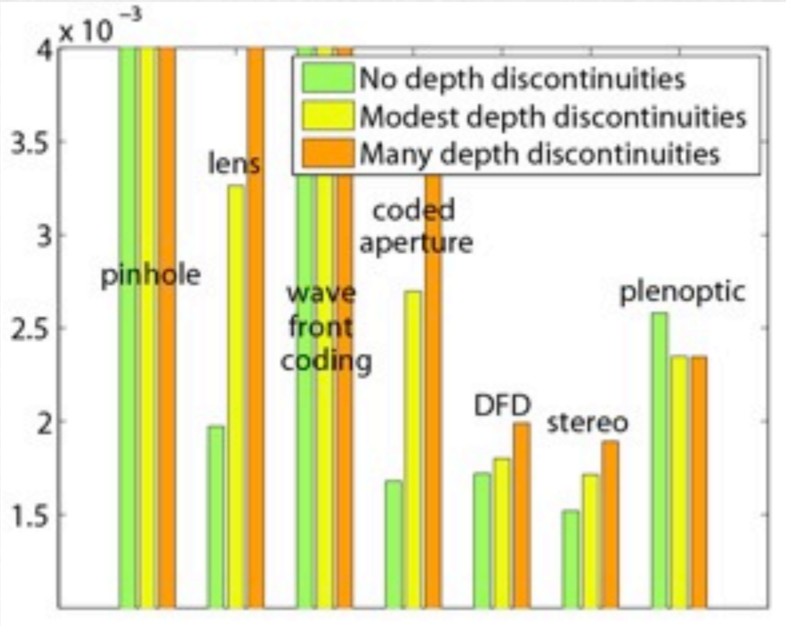


- Even more ill-posed:
  ambiguity between blur kernel & image
- Usually resolved using sparse derivative prior
- But we show common wisdom to be misleading
  - Prior can't tell sharp images
  - More important: pose problem properly

Bayesian lightfield imaging

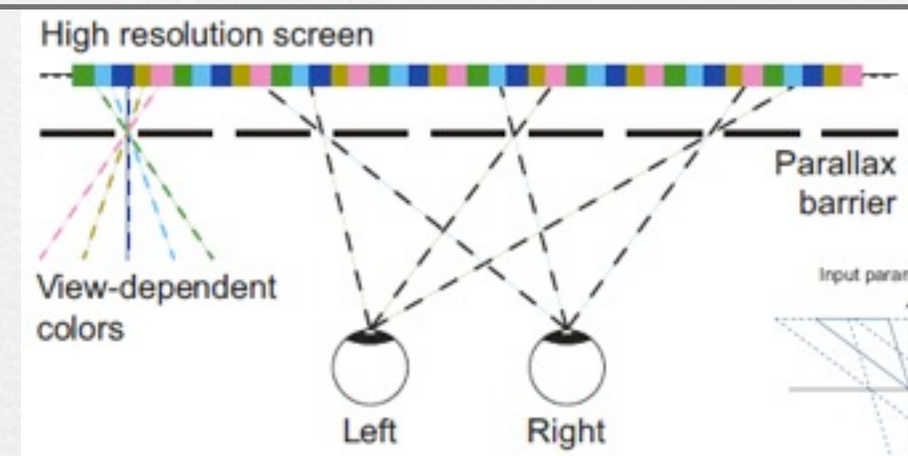
- *[Levin et al. ECCV 08]*
- Model imaging as linear light field projection
- New prior on light field
- Camera decoding expressed as a Bayesian inference problem
- Framework and software for comparison across camera configurations in flatland

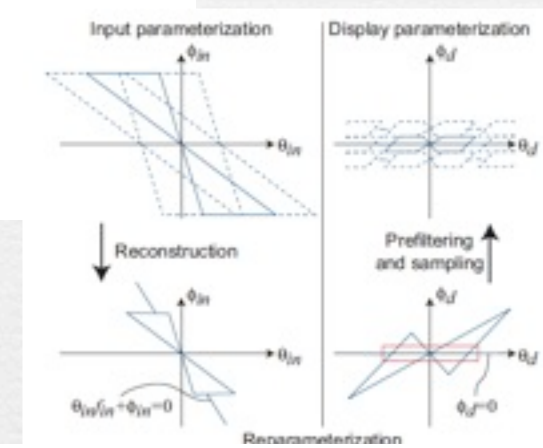


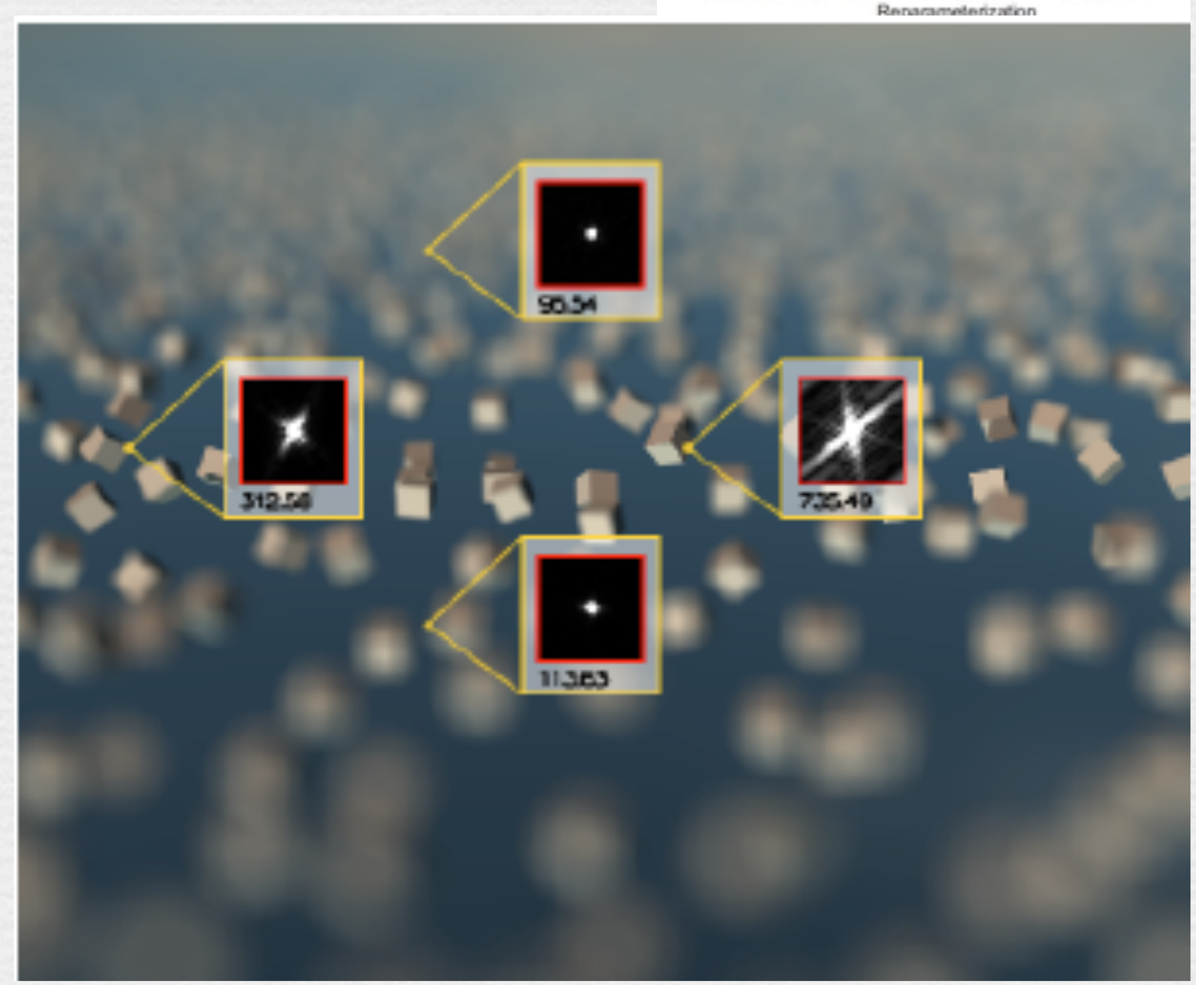


Beyond photography

- 3D Displays
    - Fourier analysis of light field for antialiasing
    - *[Zwicker et al. 06]*
- Rendering
    - Motion blur, depth of field
    - Frequency analysis of light field or time space for adaptive sampling and improved reconstruction
    - e.g. [Durand et al. 05, Soler et al. 09]







Plan

- Introduction of computational photography
  - Enhance our vision
  - Capture visual information
- Motion Invariant Photography
  - Move sensor to make kernel invariant, high frequency response
  - Upper bound
- Potpourri
- **Big Ideas in Computational Photography**

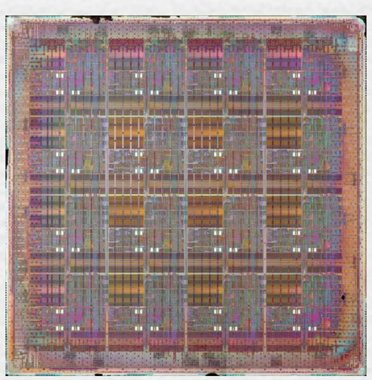
Big ideas in Comp. Photo.

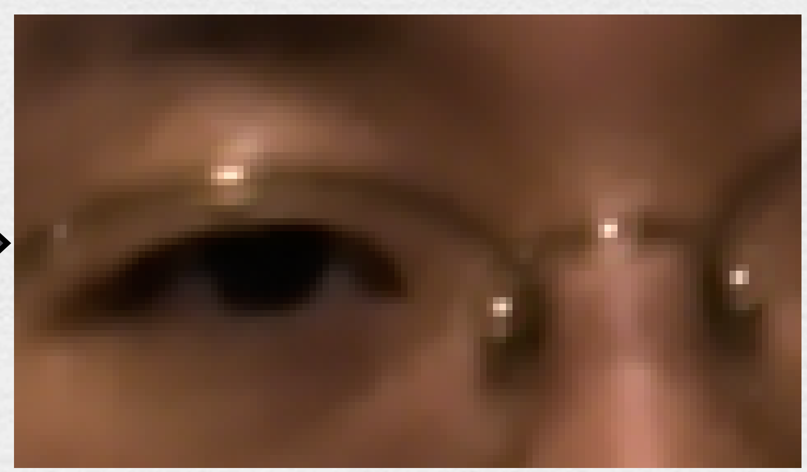
- Goals:
  - Beat physics, better image quality/quantity
  - More data (depth, etc.)
  - Seeing the unseen
  - Creative choices during post-process
  - New visual media

Coded Imaging

Optics encodes information

Computation decodes

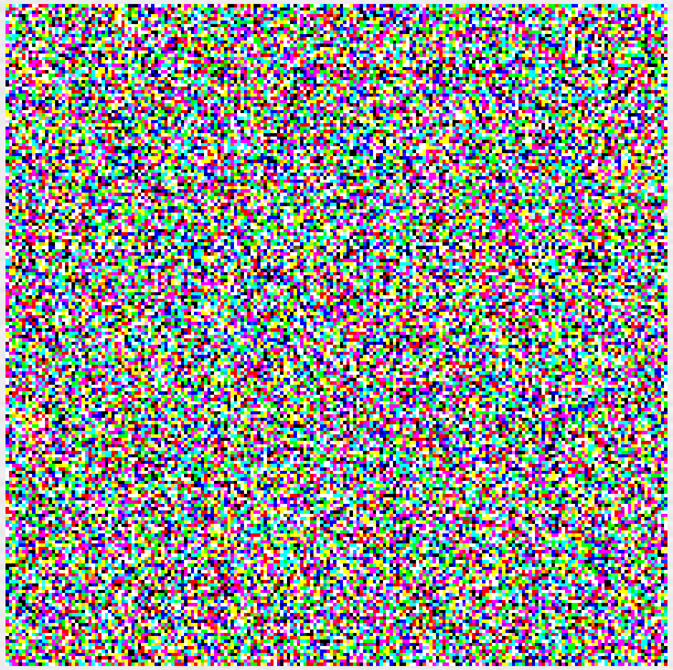




- e.g.
  - motion-invariant
  - coded aperture
  - flutter shutter
  - wavefront coding
  - compressive sensing
  - heterodyning
  - warp-unwarp

Natural signal prior

- Statistics that distinguish images of the world from random signals
- Use to “bias” algorithms to output more likely results or to disambiguate ill-posed problems
- Extension of regularization
- e.g.
  - Denoising
  - Deconvolution
  - Compressive sensing
  - Light field prior

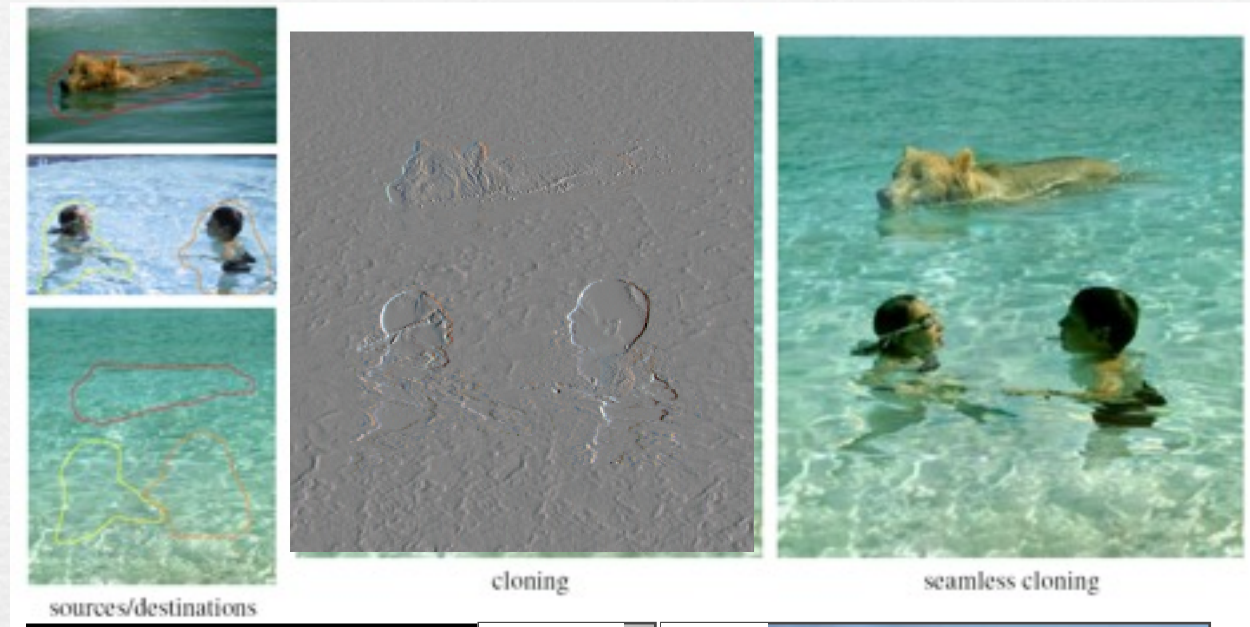


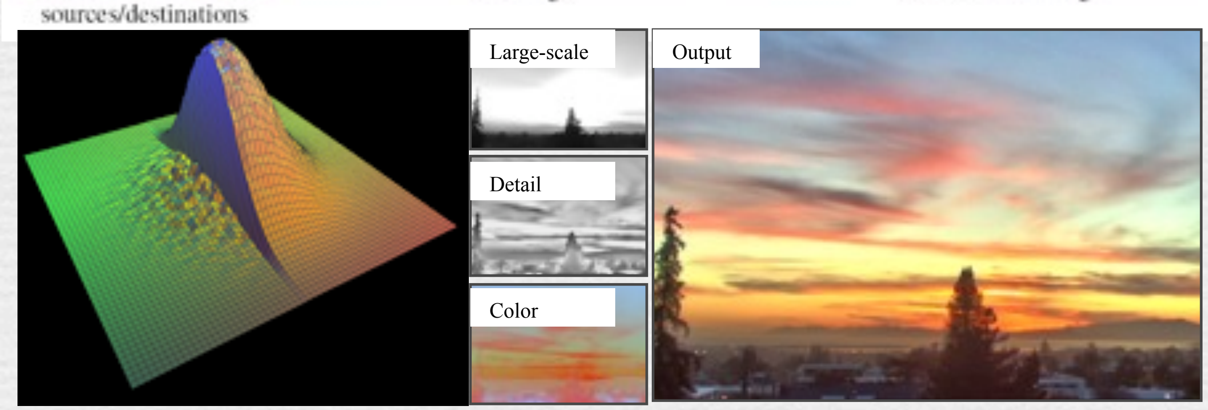
Random

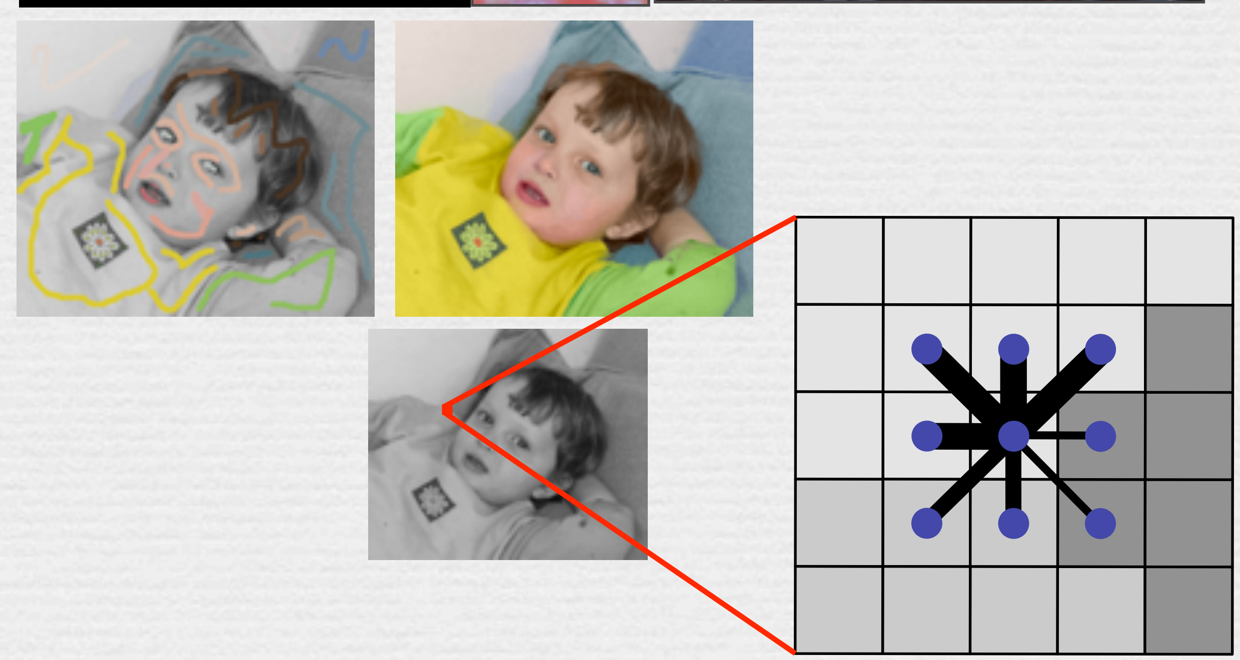
“Natural” image

Edges matter but are not binary

- Sparse derivative image prior
- Gradient domain (seamless cloning, tone mapping, convert2gray)
- Bilateral filter for decomposition
- Non-homogenous regularization for scribble propagation

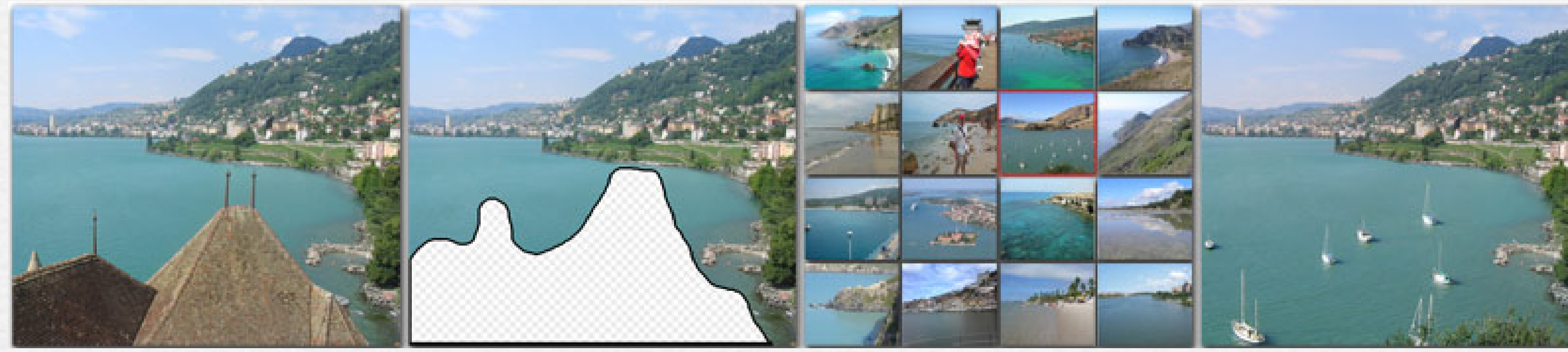






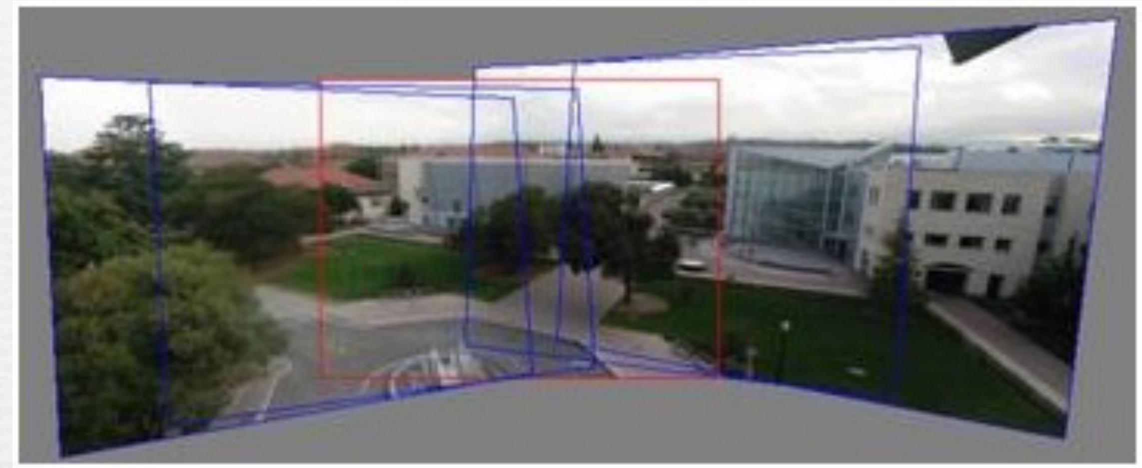
Leverage millions of images

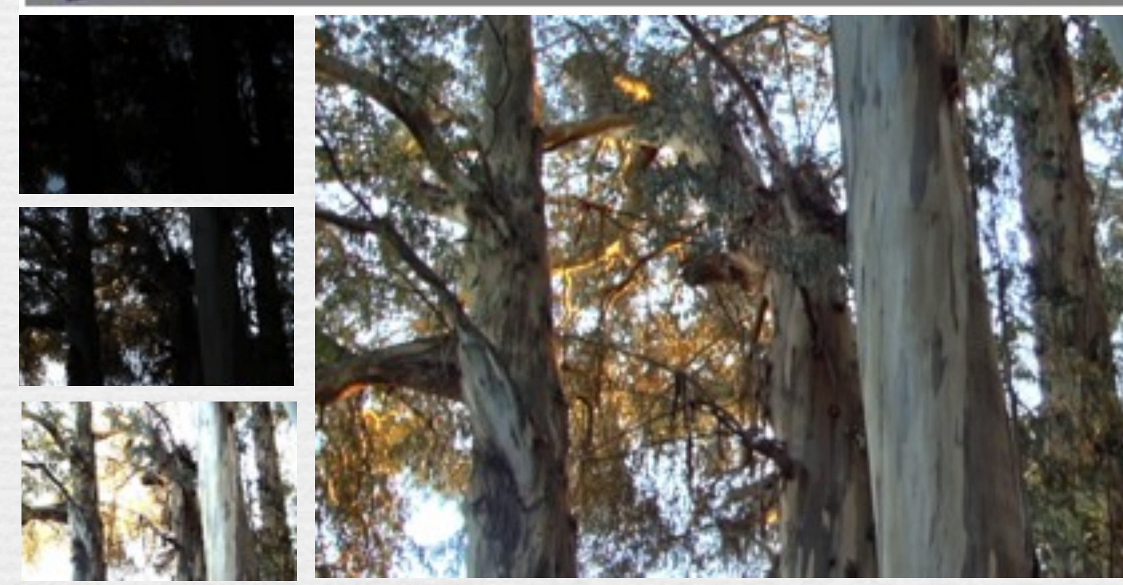
- The ultimate prior?
- Reconstruct the world



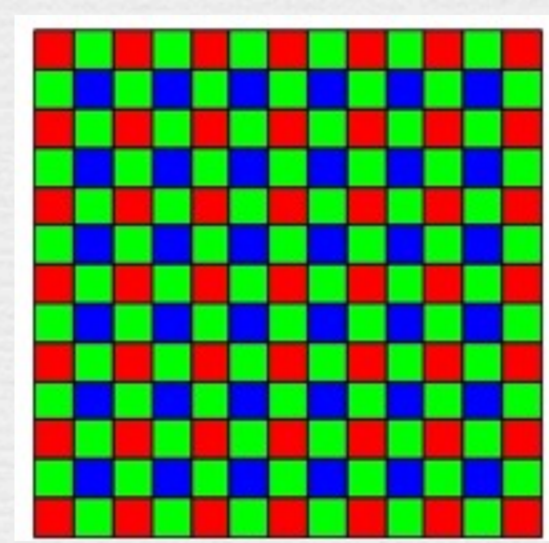
Hays & Efros 07

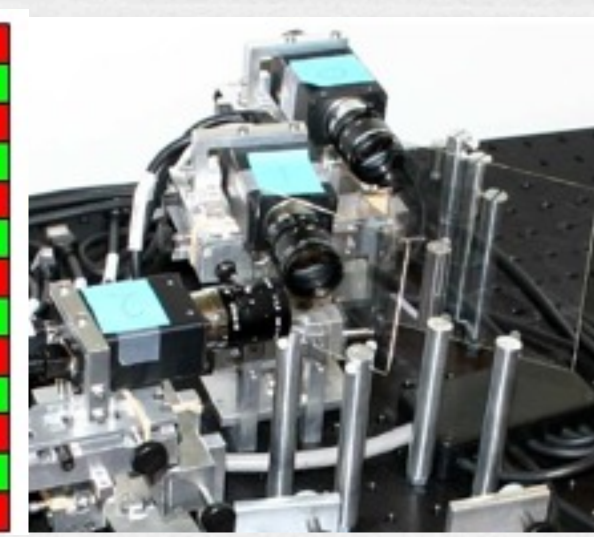
Multiple-exposure & multiplexing





- Expand capabilities by combining multiple images
- Multiplex through time, assorted pixels, beam splitters, camera array
- e.g.
  - Panorama stitching
  - High-dynamic-range imaging
  - Focus stacks
  - Photomontage
  - Super-resolution





The raw data is high dimensional

- Light field: 4D (space-angle)
- Time space: 3D
- +Fourier

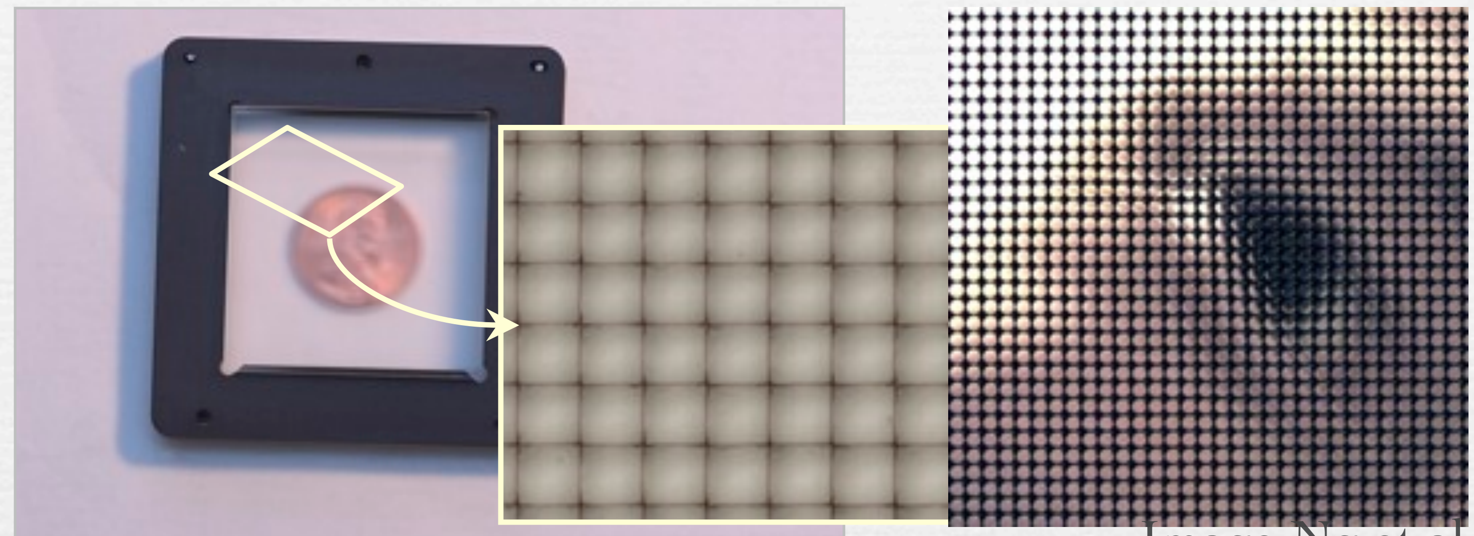
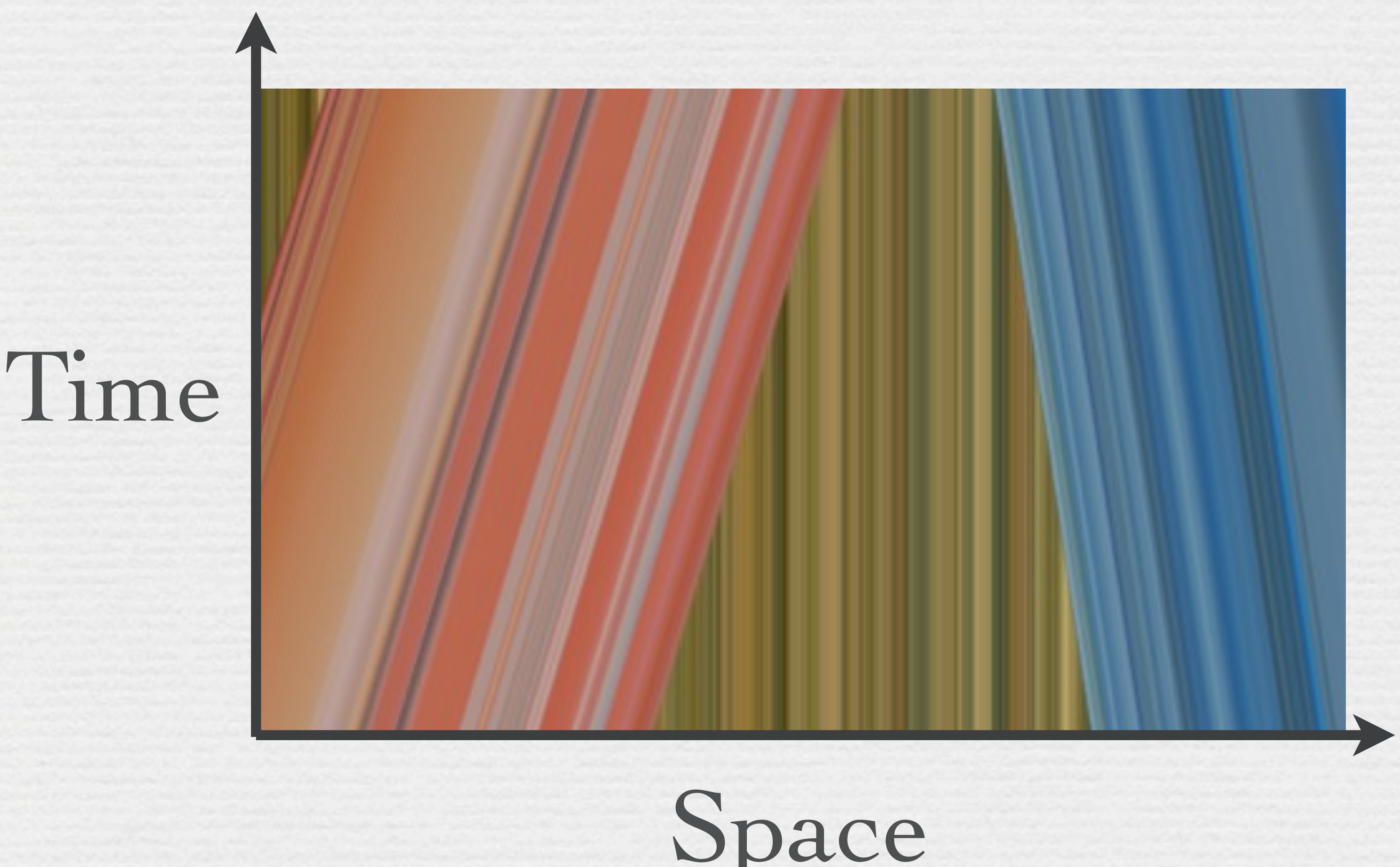
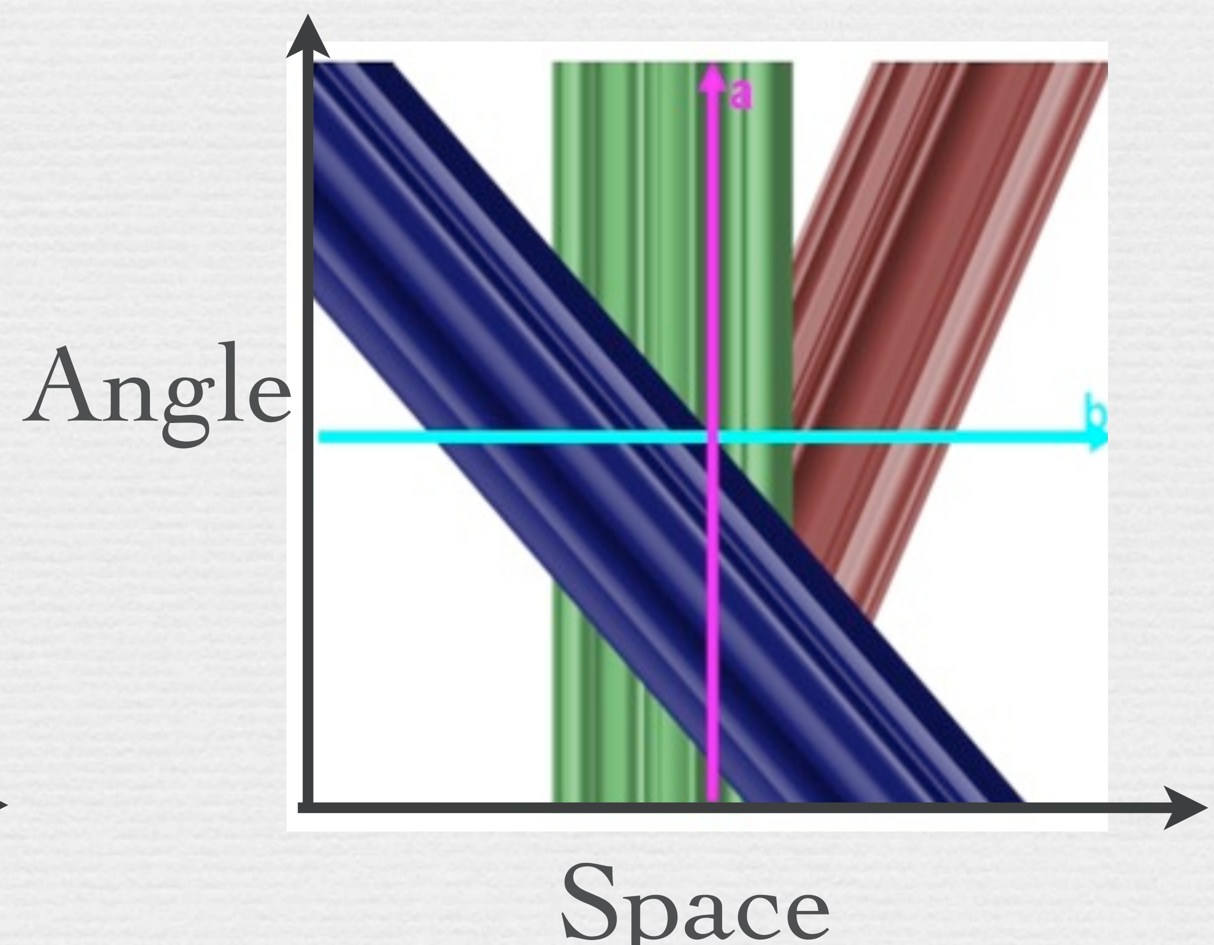


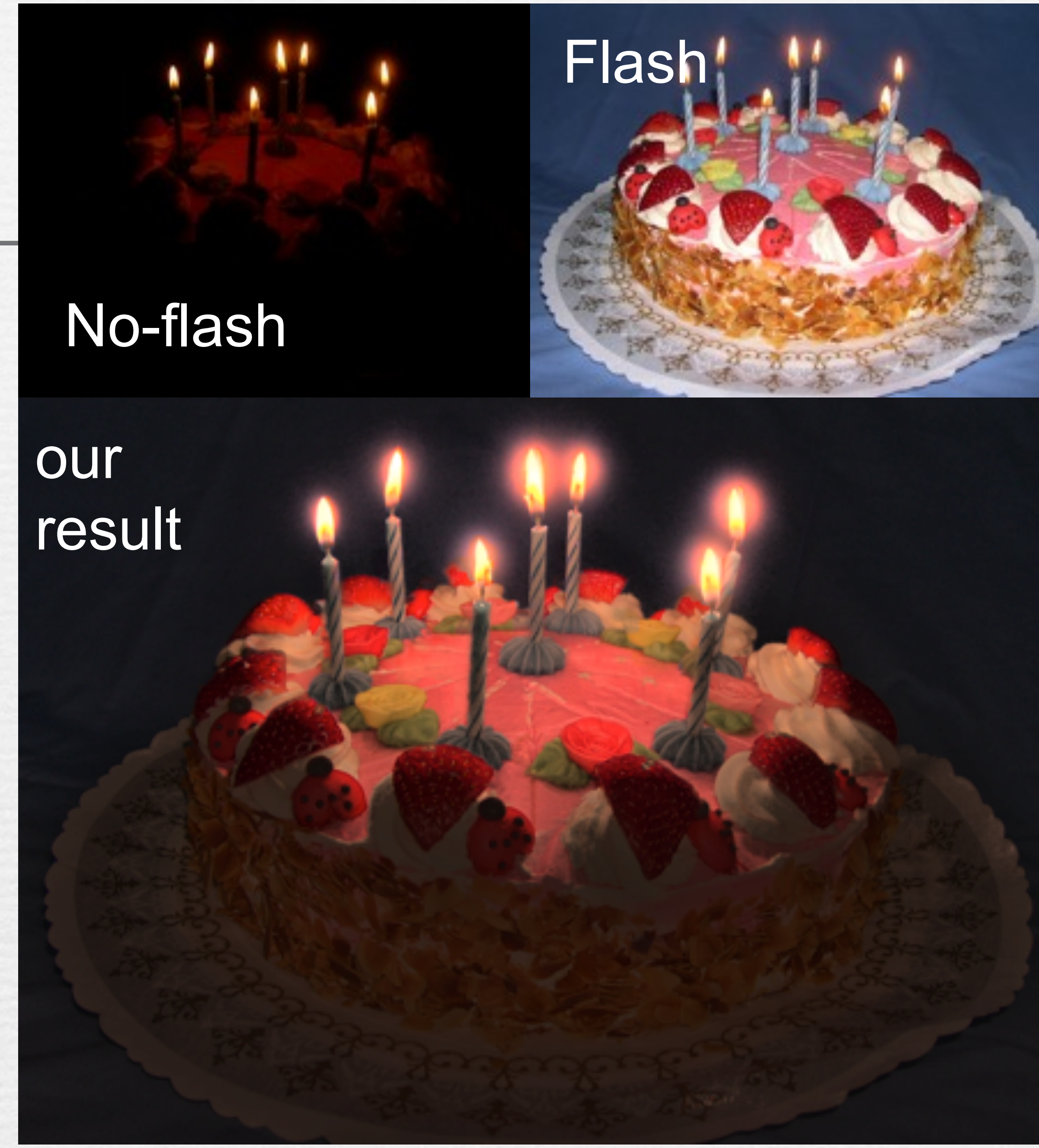
Image Ng et al.

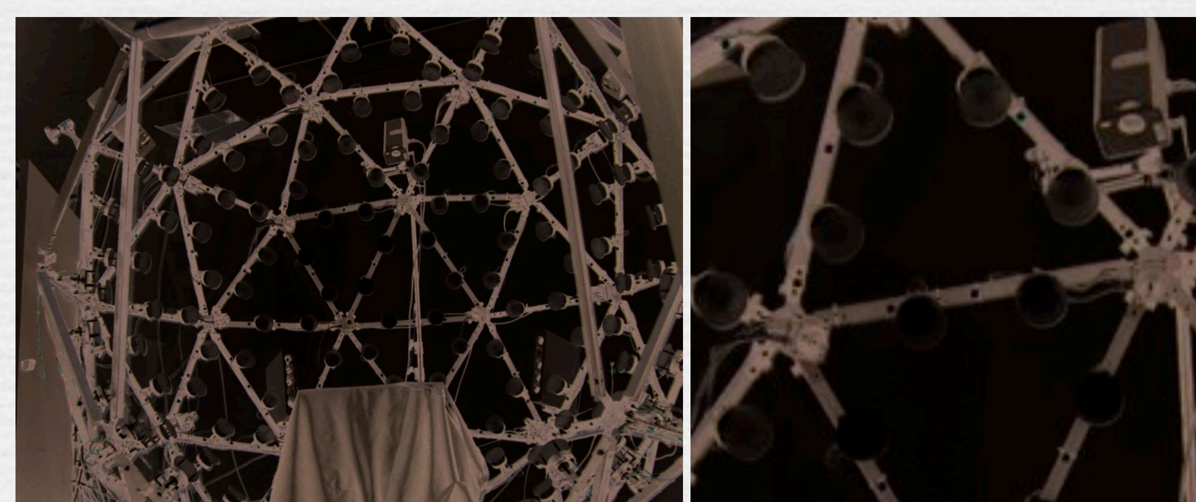




Active imaging

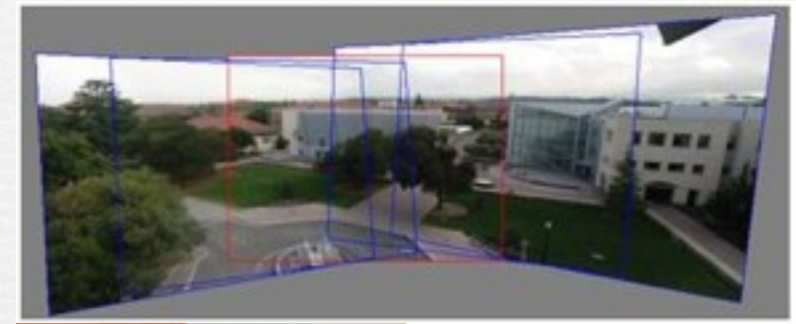
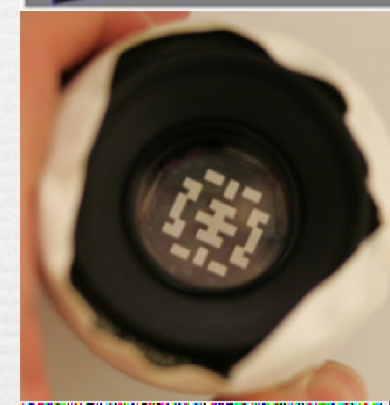
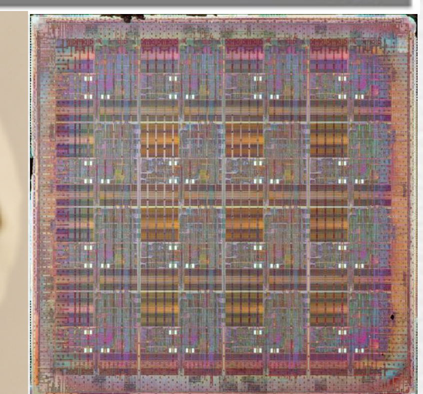
- Modulate light to facilitate information gathering
- e.g.
  - Flash/no flash
  - Light stages
  - Dual imaging
  - Structured-light scanning

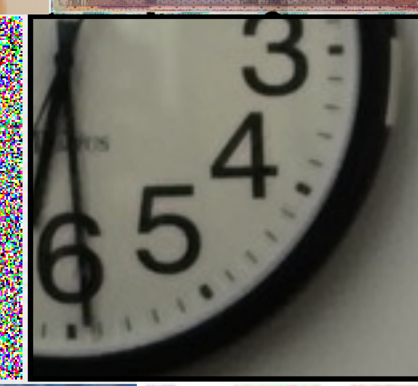
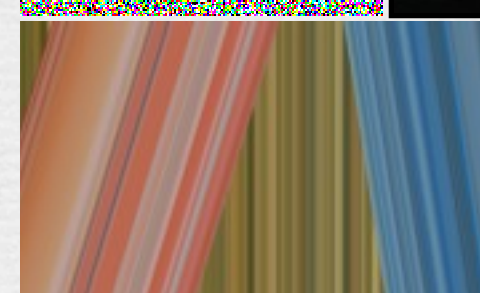


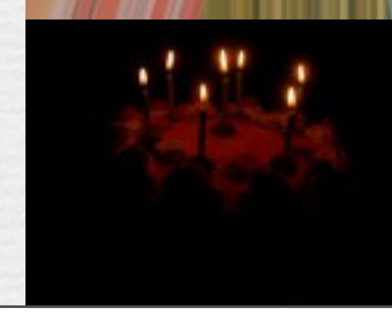


Recap: Big ideas in comp. photo.

- Coded imaging
- Natural signal prior
- Edges matter but should not be detected
- Leverage millions of images
- Raw data is high-dimensional (ligh field, space-time)
- Active Imaging




Ecosystem

- Computational Photography
  - Computer graphics
  - Computer vision
  - Traits: Geometrical optics, light field, ignore diffraction
- Computational Imaging
  - Optics
  - Electrical Engineering
  - Traits: Fourier optics, wave nature of light, often simpler processing
- Start interacting
  - Workshop in Charlotte
  - OSA Frontiers in optics
  - IEEE International Conference on Computational Photography

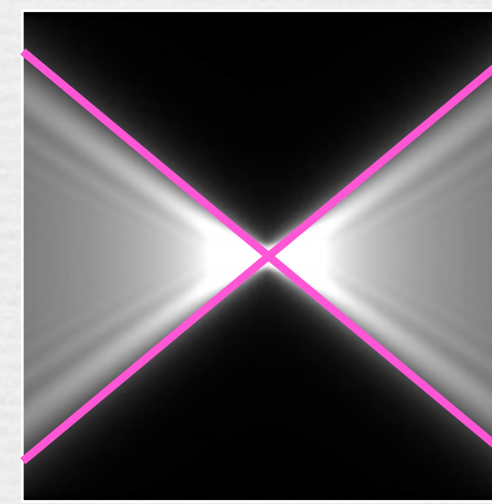
Ongoing work & important challenges: Fundamentals of computational imaging

- Understand information available in the world, necessary for a task, captured by a camera
- Frequency analysis of light field, space time, image
- Effect of noise, fundamental limits
- Unifying frameworks, comparison of strategies

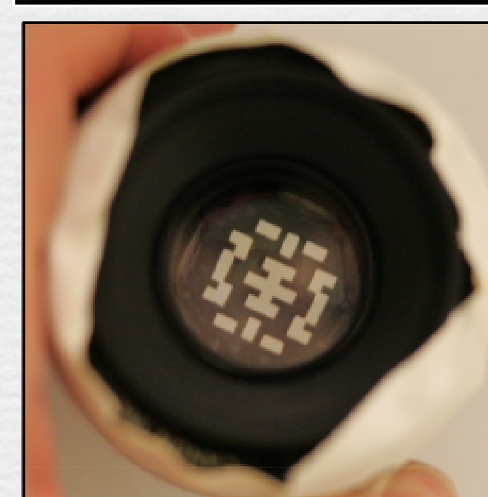
Summary

- Computational photography
  - Enhance our vision
  - Capture visual information

- Motion Invariant Photography
  - Move sensor to make kernel invariant, high frequency response
  - Upper bound



- Potpourri:
  - Coded aperture, sparse derivative, light field camera framework, blind deconvolution, display, rendering



- Big Ideas in Computational Photography
  - Coded imaging, raw data is high-dimensional, prior, edges/gradients, millions of images, active imaging

Acknowledgements

- Anat Levin, Tim Cho, Peter Sand, Rob Fergus, Bill Freeman, Paul Green, Sam Hasinoff, Yair Weiss, Cyril Soler, Kartic Subr, Sillion
- NSF, Microsoft, Sloan foundation, Burge & Libby Jamieson, Lincoln Lab

Challenges & opportunities

- Theory, frameworks, comparisons, optimality
- Diffraction, wave optics
- Putting it all together
  (engineering, system, applications)
- Better priors
  - Kernel identification
  - High-quality inversion
- Video
- Real-time enhancement (e.g. motion magnification)
- Applied visual perception
- Intrinsic images
- Matting
- Scene and object recognition
- Extract and leverage 3D reconstruction

Commercialization

- Computational photo with existing cameras
  - HDR
  - Panoramas
  - Photomontage
  - Poisson/Healing brush
  - Photosynth
- Co-design of optics and computation
  - Is beginning slowly:
    post processing removal of optical aberration
  - Niche areas (e.g. iris recognition)