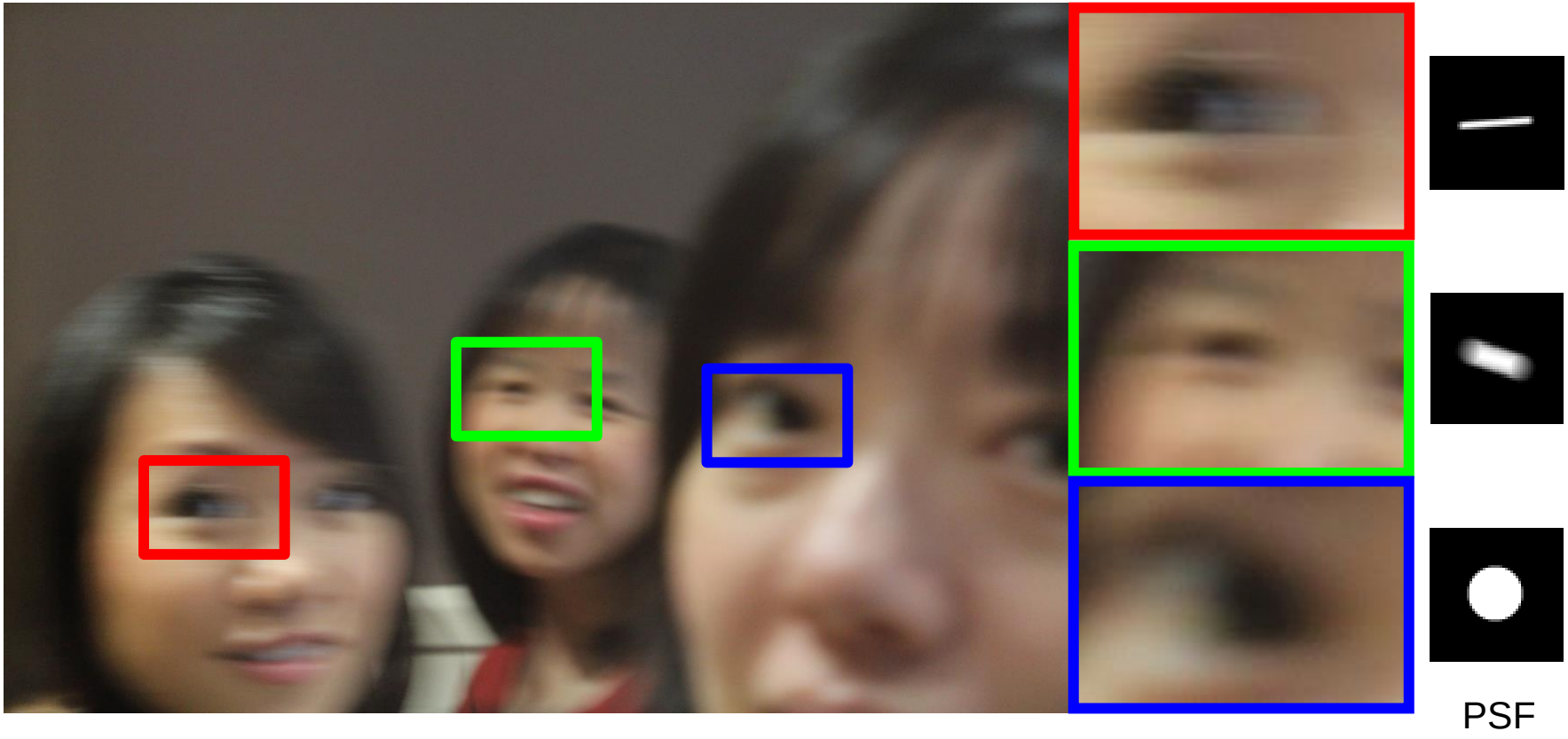
Near-Invariant Blur for Depth and 2D Motion via Time-Varying Light Field Analysis

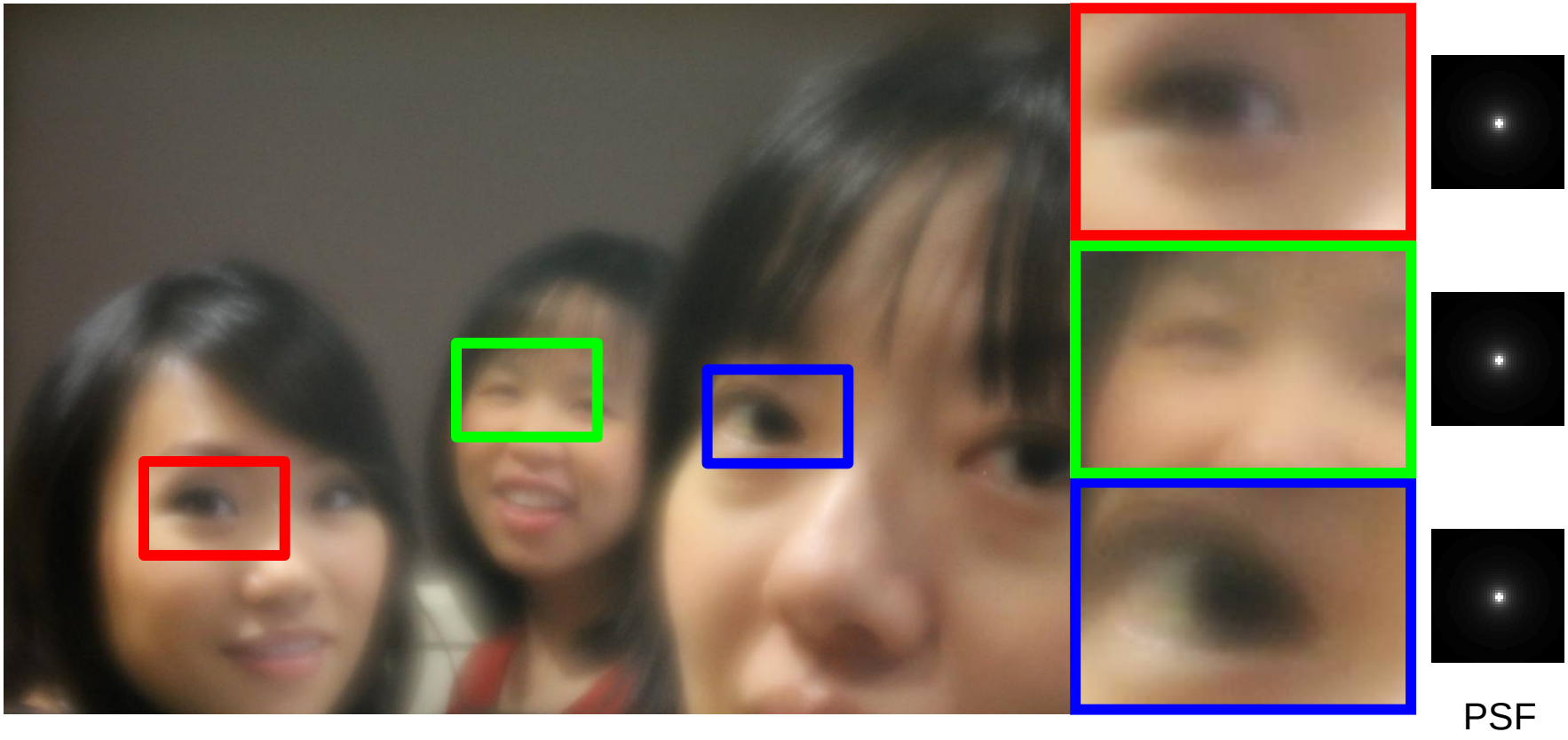
Yosuke Bando[1,2] Henry Holtzman[2] Ramesh Raskar[2]

[1]Toshiba Corporation [2]MIT Media Lab

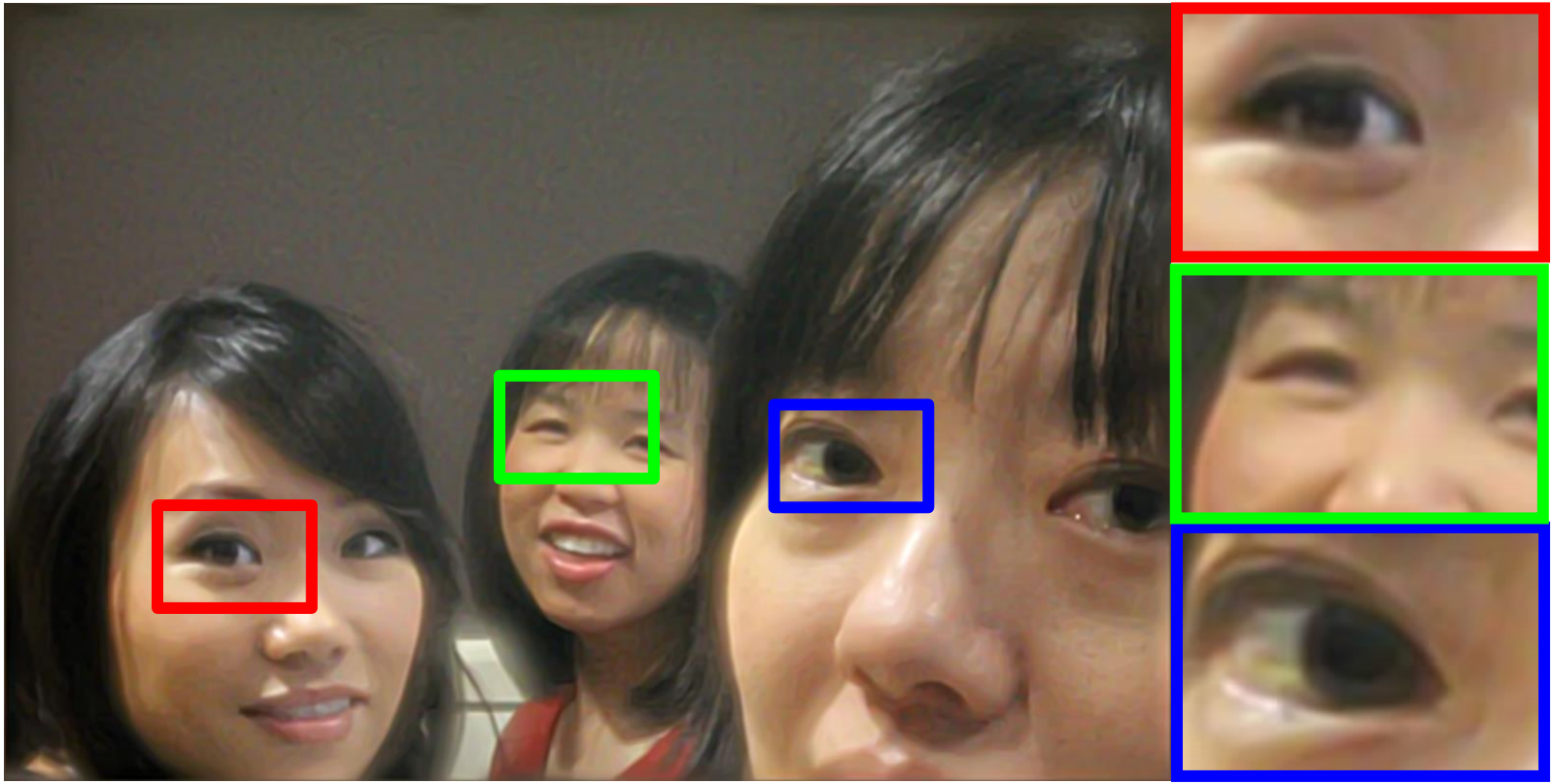
Defocus & Motion Blur



Depth and Motion-Invariant Capture



Deblurring Result



Outline

- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

Outline

- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

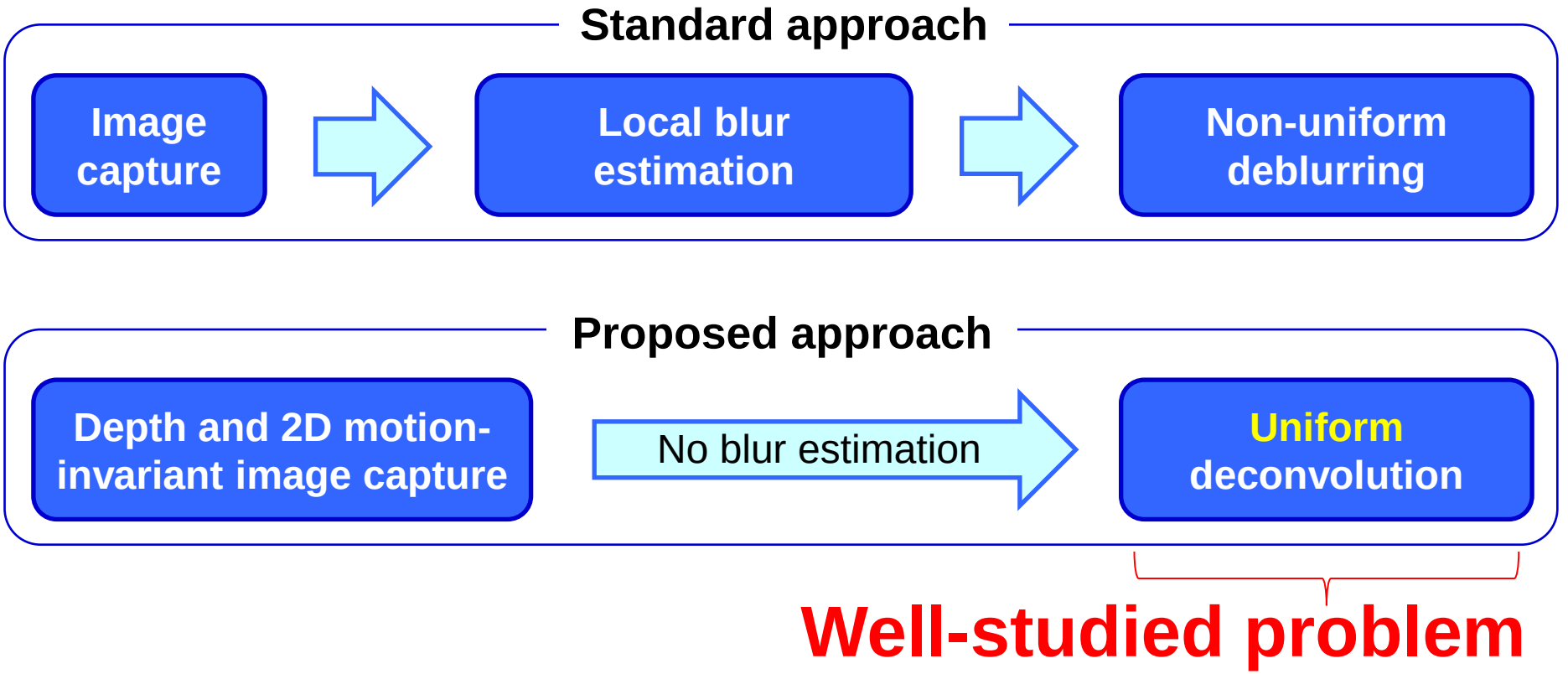
Joint Defocus & Motion Deblurring

Standard approach

Extremely difficult

- Estimate depth and motion from a single image
- Recover lost high-frequency content

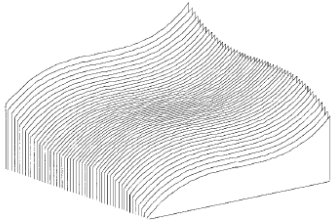
Joint Defocus & Motion Deblurring



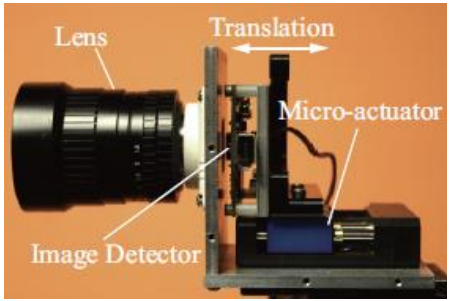
Outline

- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

Depth-Invariant Capture

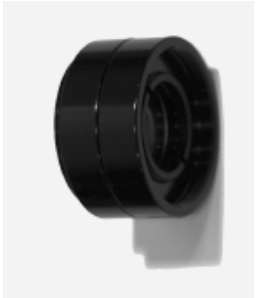


Wavefront coding
[Dowski and Cathey 1995]



Focus sweep
[Hausler 1972;
Nagahara et al. 2008]

Diffusion coding
[Cossairt et al. 2010]



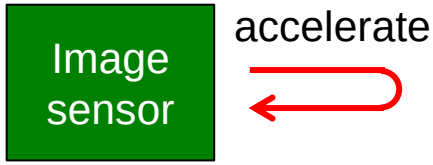
Spectral focus sweep
[Cossairt and Nayar 2010]

Depth-invariant image

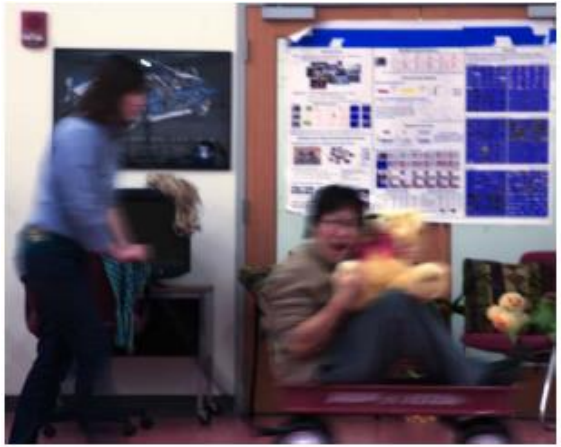
Deblurred

1D Motion-Invariant Capture

- Invariant to object speed
- Motion direction must be fixed
  - Horizontal, for example



[Levin et al. 2008]



Normal camera

Motion-invariant image

Deblurred

Computational Cameras for Deblurring

	Defocus deblurring	Motion deblurring
H p (r		
Invariant capture	**Wavefront coding** [Dowski and Cathey 1995] **Focus sweep** [Hausler 1972; Nagahara et al. 2008] **Diffusion coding** [Cossairt et al. 2010] **Spectral focus sweep** [Cossairt and Nayar 2010]	**Motion-invariant photography** (for 1D motion) [Levin et al. 2008]

No joint defocus and motion deblurring

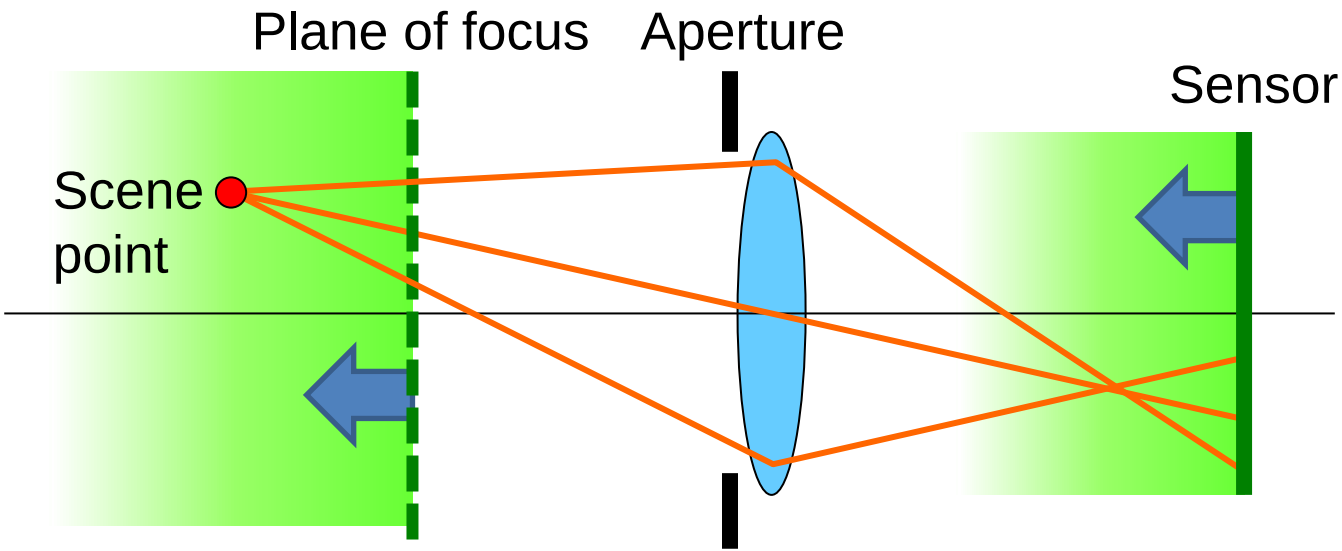
No 2D motion-invariant capture

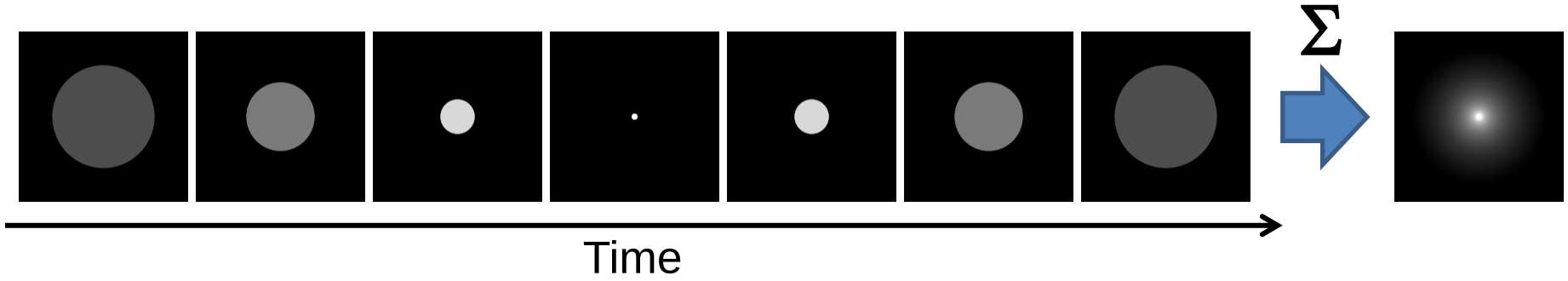
Also nearly 2D motion-invariant

Outline

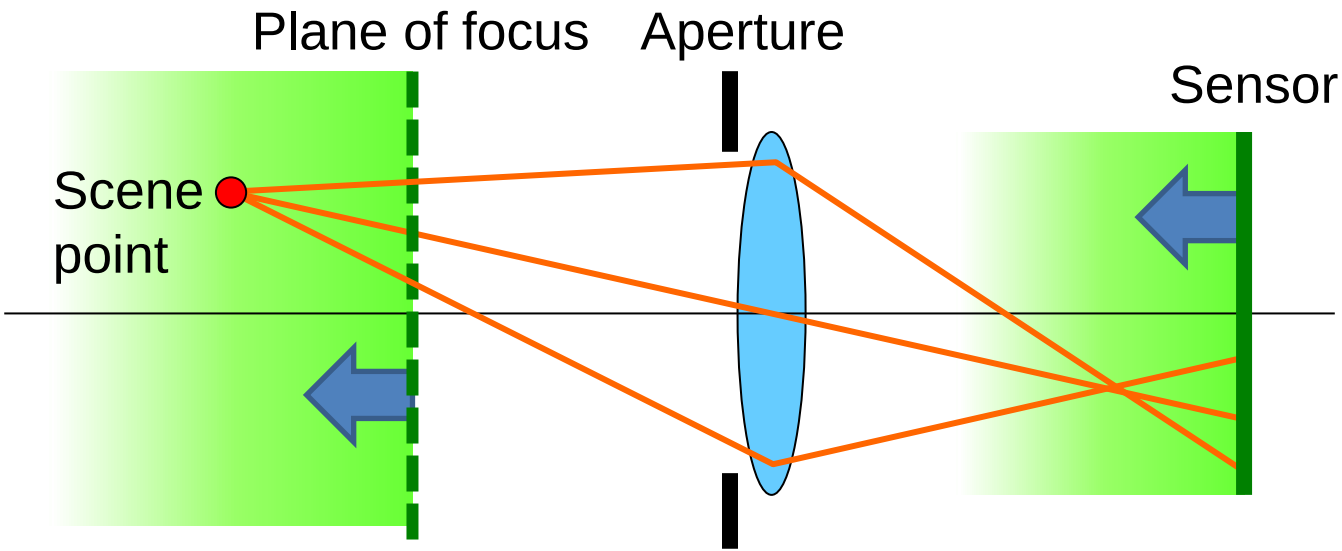
- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

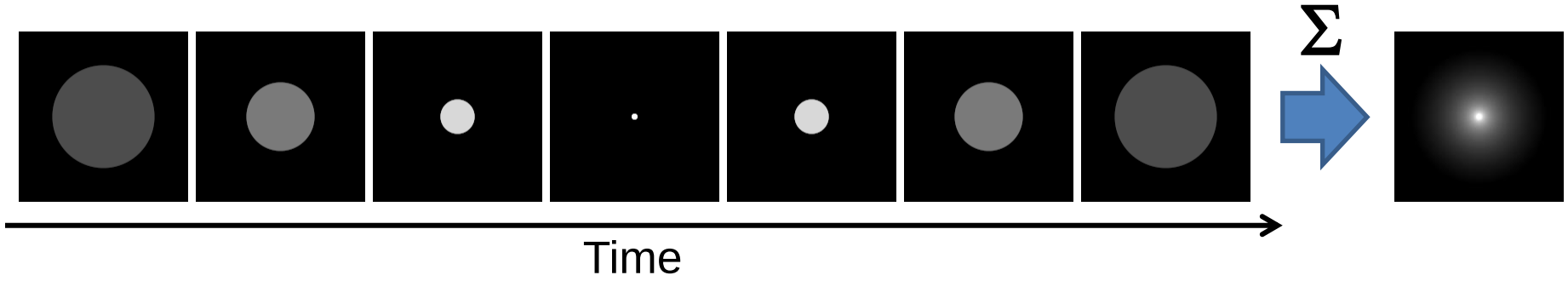
Depth-Invariance for Static Point



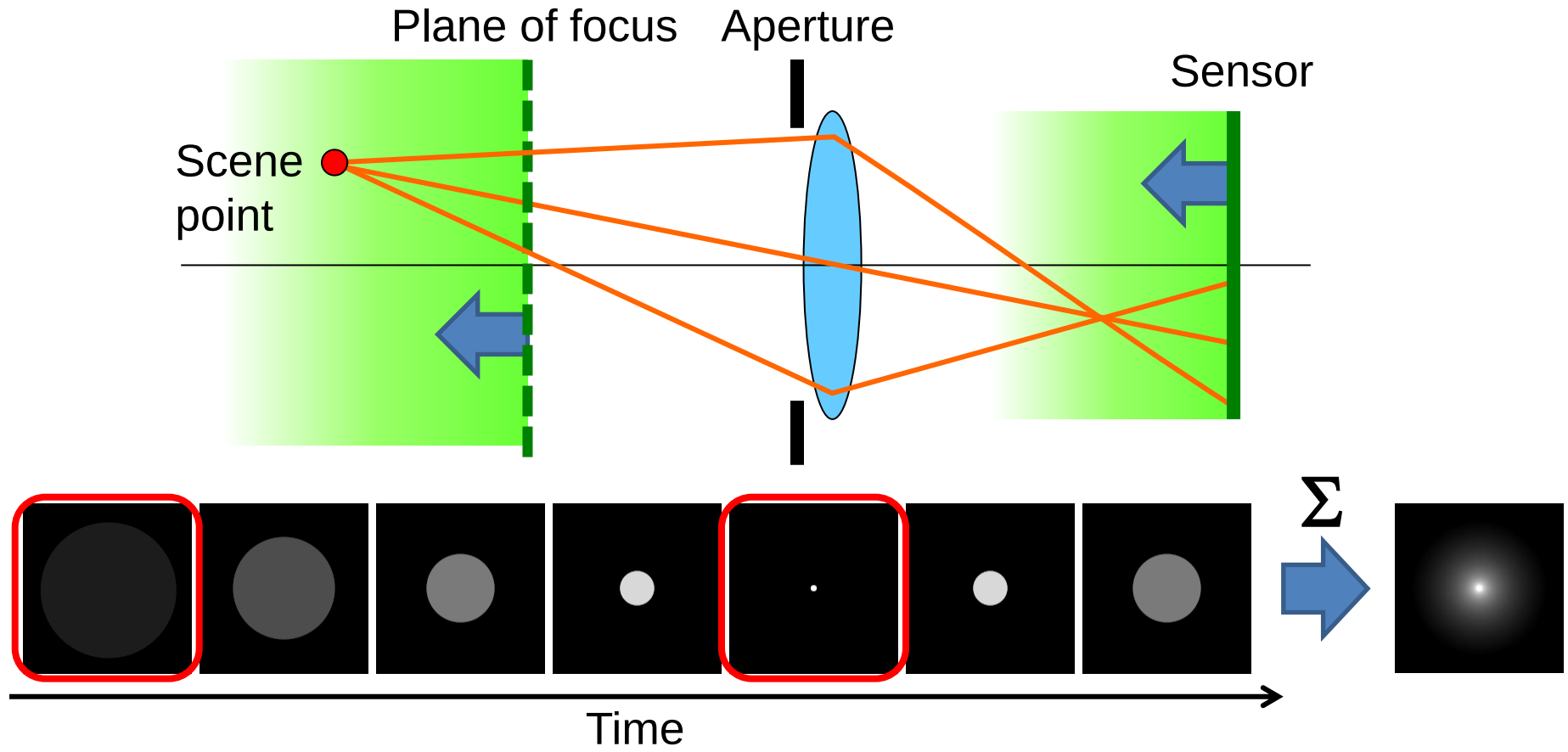


Depth-Invariance for Static Point

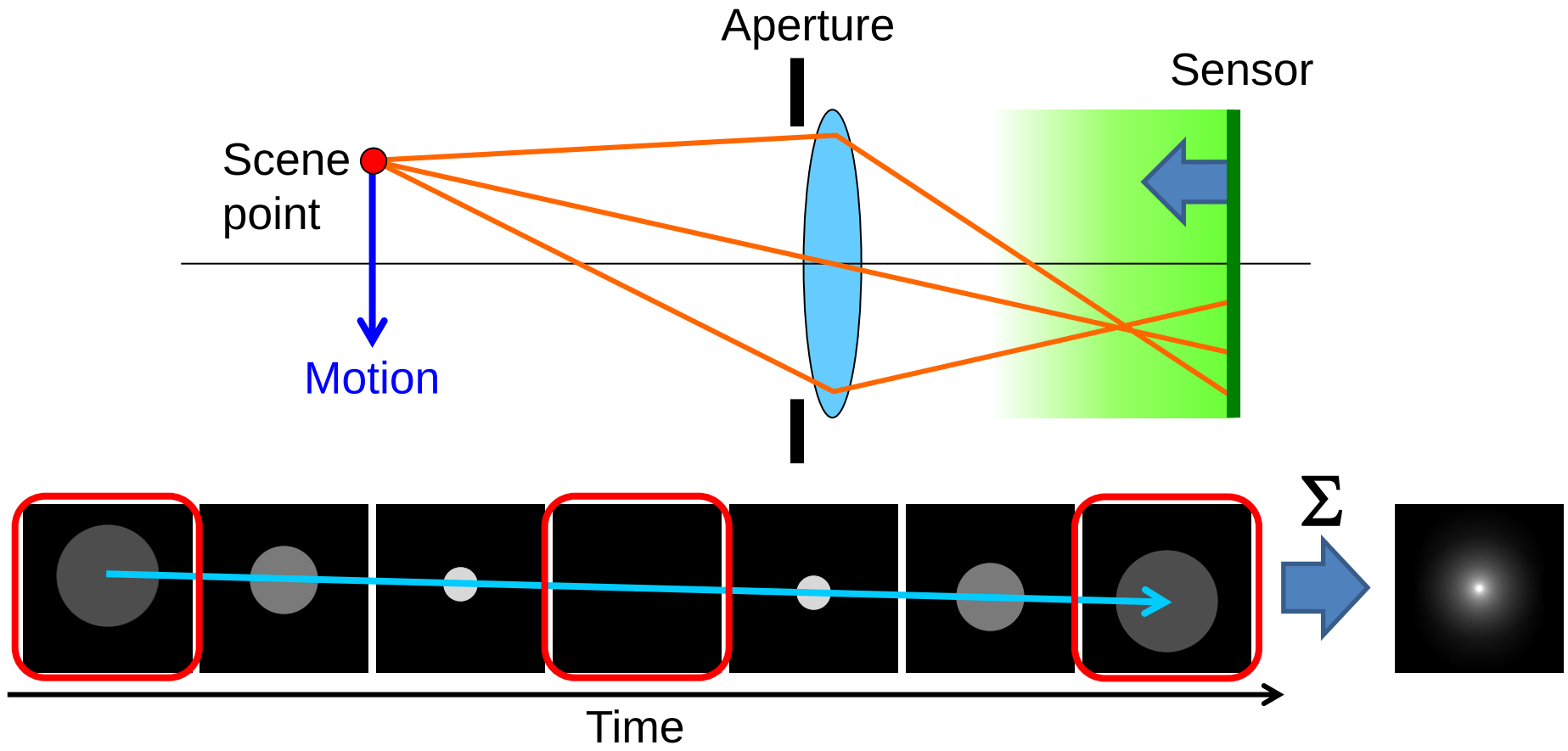




Depth-Invariance for Static Point



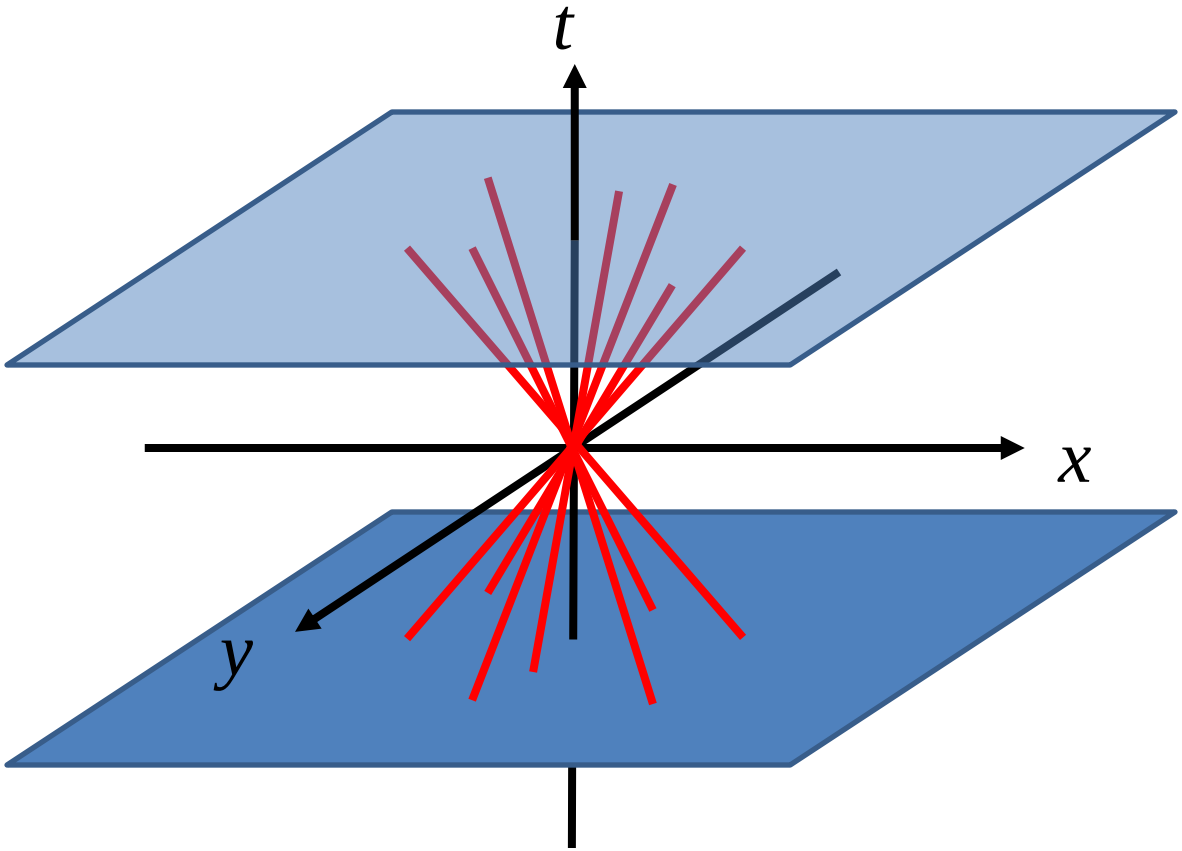
Motion-Invariance for Moving Point



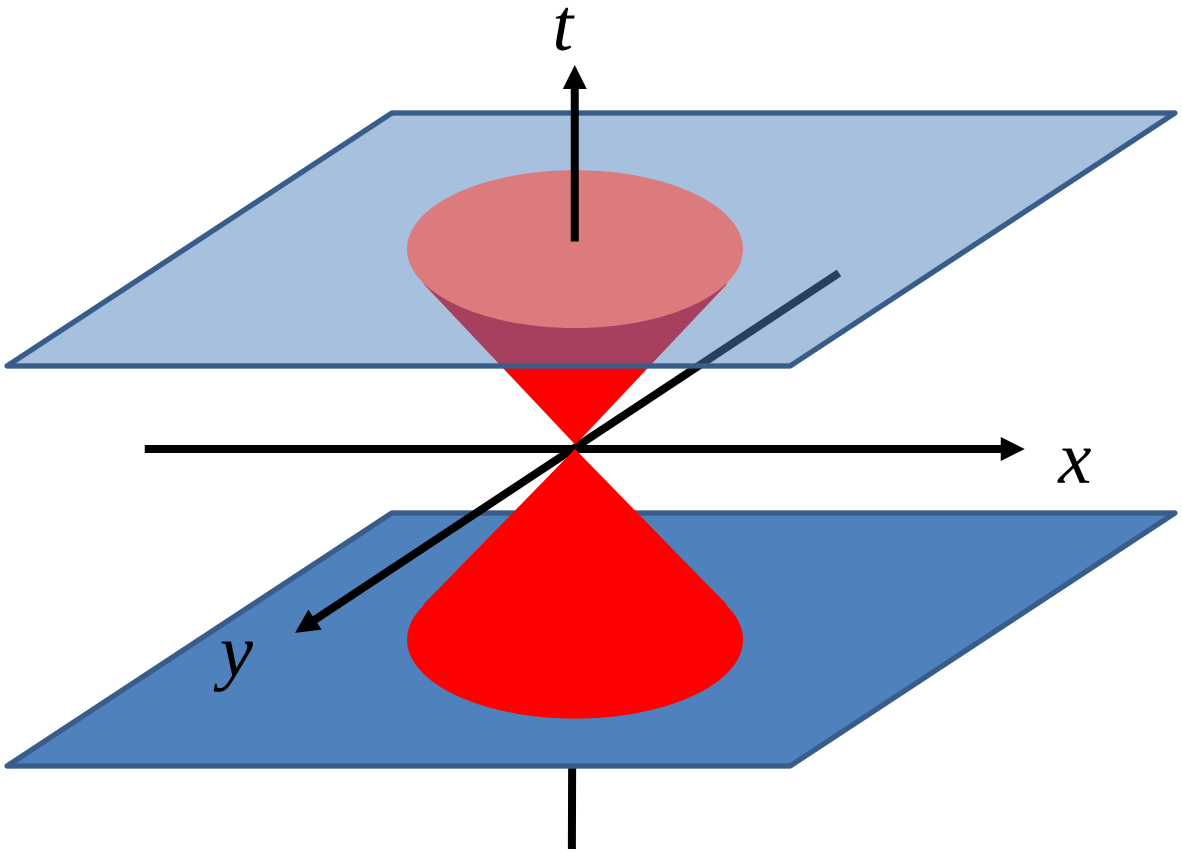
Follow Shot

http://commons.wikimedia.org/wiki/File:Bruno_Senna_2006_Australian_Grand_Prix-3.jpg

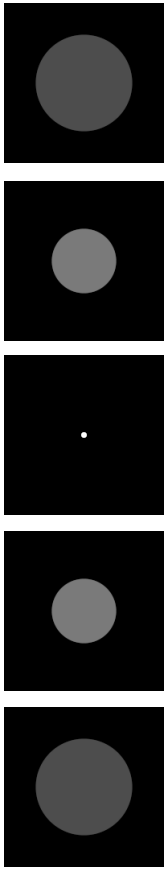
Follow Shots for Various Motions

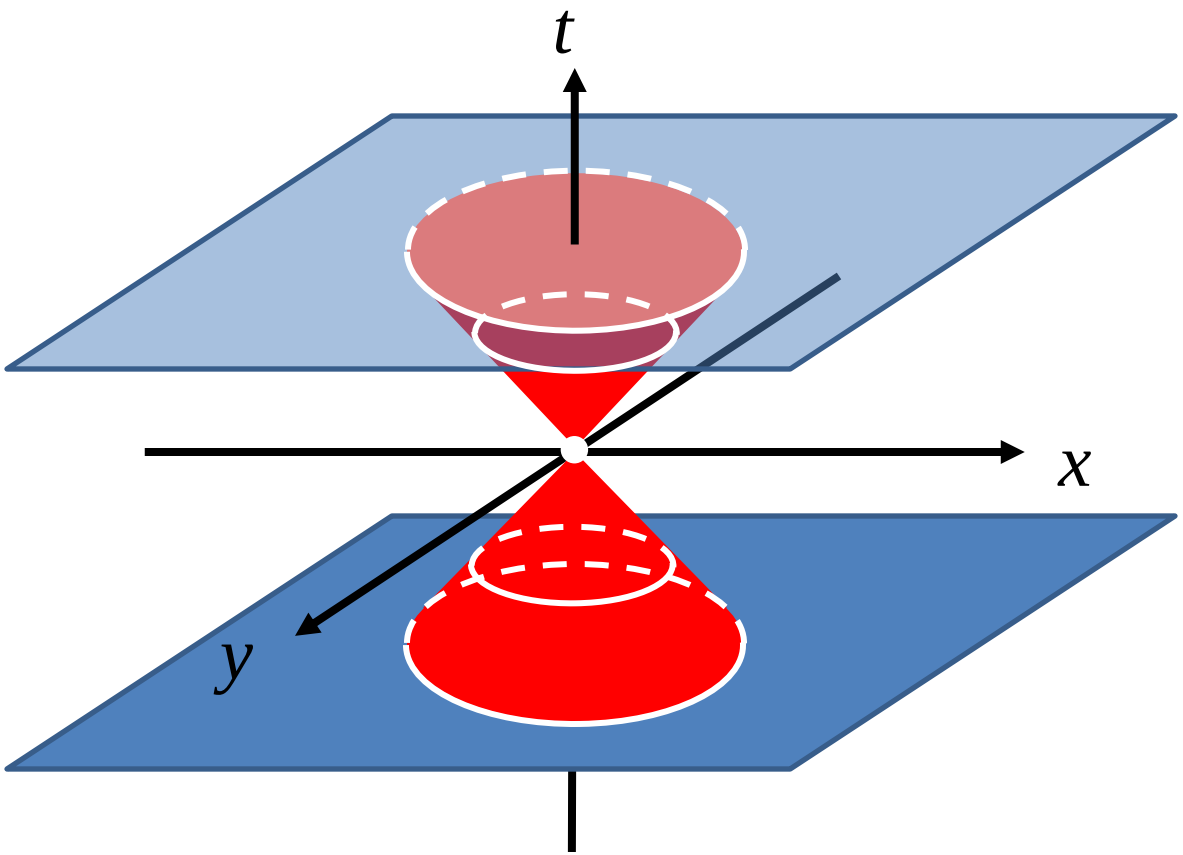


Follow Shots for Various Motions



Follow Shots for Various Motions





Outline

- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

Analysis

- Photo is a projection of a light field [Ng 2005]

$$D(\mathbf{x}_0) = \iint k(\mathbf{x}_0 - \mathbf{x}, -\mathbf{u}) \cdot l(\mathbf{x}, \mathbf{u})\, d\mathbf{x} d\mathbf{u}$$

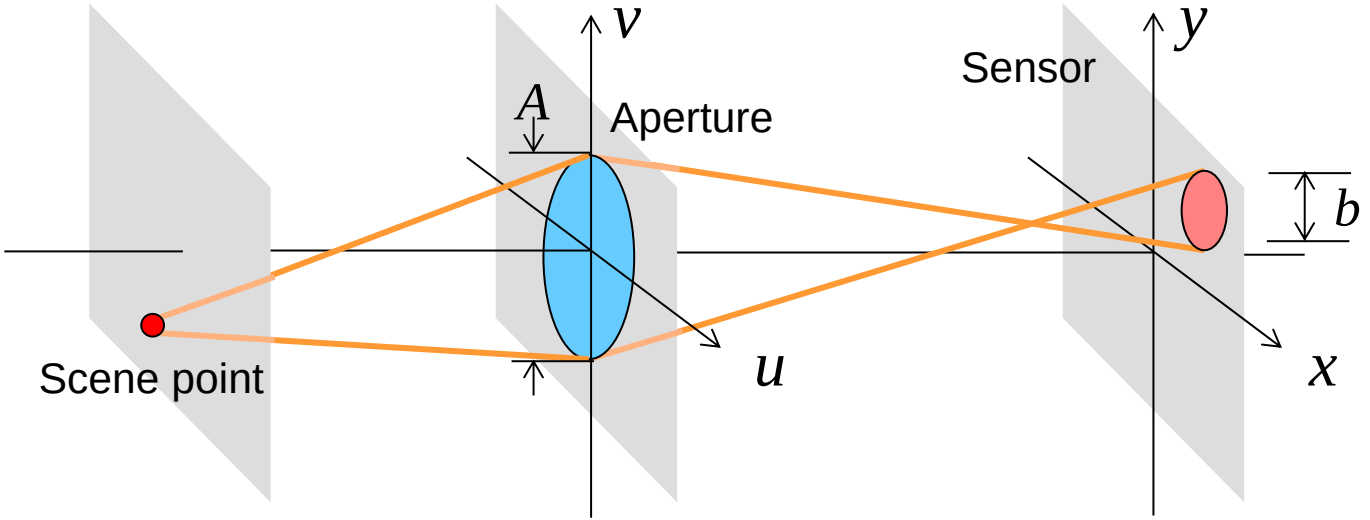
$\mathbf{x} = (x, y)$

$\mathbf{u} = (u, v)$

Defocus-blurred image

Light field kernel

Light field



Analysis

- Photo is a projection of a time-varying light field

$$D(\mathbf{x}_0) = \iint k(\mathbf{x}_0 - \mathbf{x}, -\mathbf{u}) \cdot l(\mathbf{x}, \mathbf{u})\, d\mathbf{x} d\mathbf{u}$$

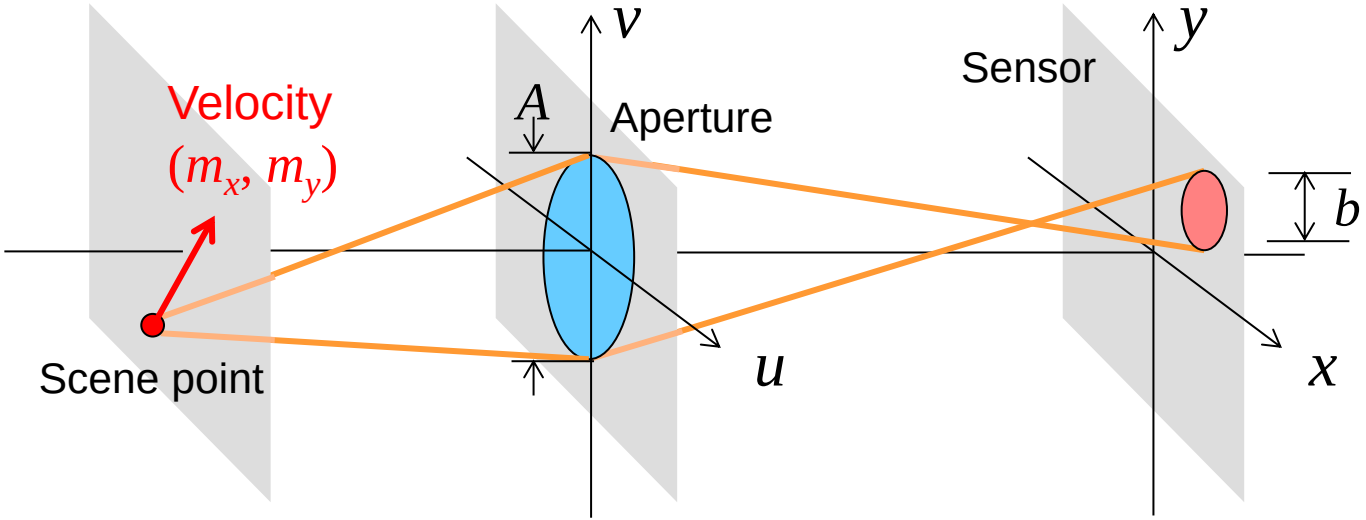
$\mathbf{x} = (x, y)$

$\mathbf{u} = (u, v)$

Defocus-blurred image

Light field kernel

Light field



Analysis

- Photo is a projection of a time-varying light field

$$D(\mathbf{x}_0) = \iiint k(\mathbf{x}_0 - \mathbf{x}, -\mathbf{u}, -t) \cdot l(\mathbf{x}, \mathbf{u}, t)\, d\mathbf{x} d\mathbf{u} dt$$

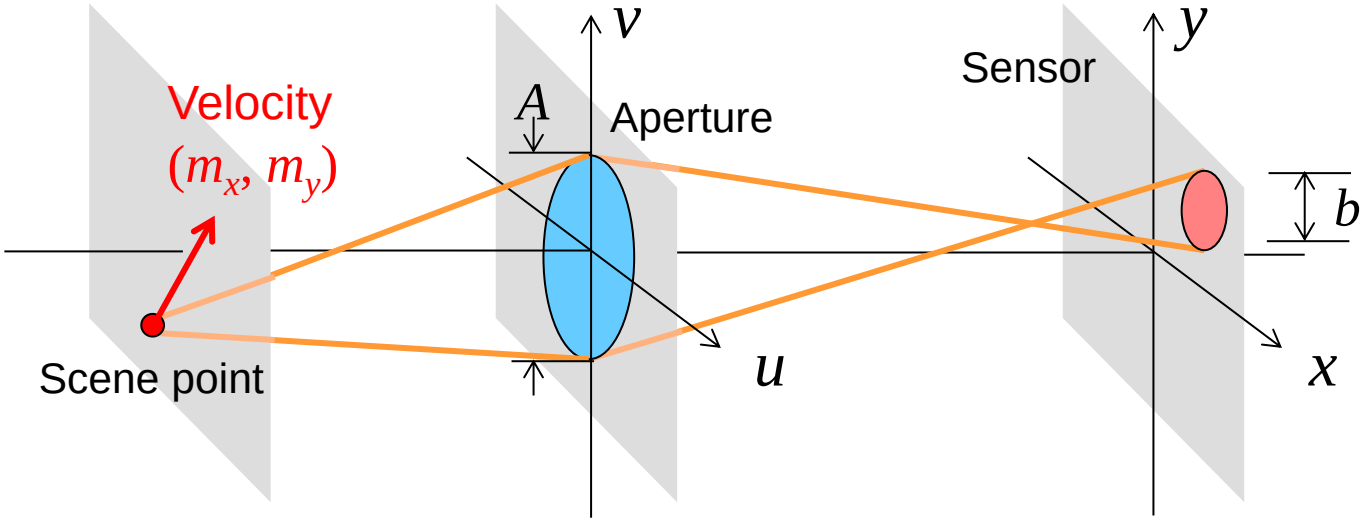
$\mathbf{x} = (x, y)$

$\mathbf{u} = (u, v)$

Defocus/motion-blurred image

Time-varying light field kernel

Time-varying light field



Time-Varying Light Field Analysis

- Photo is a projection of a time-varying light field

$$D(\mathbf{x}_0) = \iiint k(\mathbf{x}_0 - \mathbf{x}, -\mathbf{u}, -t) \cdot l(\mathbf{x}, \mathbf{u}, t)\, d\mathbf{x} d\mathbf{u} dt$$

$\mathbf{x} = (x, y)$

$\mathbf{u} = (u, v)$

Defocus/motion-blurred image

Time-varying light field kernel

Time-varying light field

Lambertian scene at depth s with velocity $\mathbf{m} = (m_x, m_y)$

$$\phi_{s,\mathbf{m}}(\mathbf{x}) = \iint k(\mathbf{x} + s\mathbf{u} + \mathbf{m}t, \mathbf{u}, t)\, d\mathbf{u} dt$$

Joint defocus & motion blur PSF

Magnitude of 2D Fourier transform

$$|\hat{\phi}_{s,\mathbf{m}}(\mathbf{f_x})|^2 = |\hat{k}(\mathbf{f_x}, -s\mathbf{f_x}, -\mathbf{m} \cdot \mathbf{f_x})|^2$$

Modulation transfer function (MTF)

Analysis Procedure and Findings

- For each existing computational camera for deblurring
  - 1. derive a kernel equation describing the optical system
  - 2. calculate its Fourier transform to obtain the MTF
  - 3. compare it with the theoretical upper bounds

Camera design	High freq. preserv. $\min_{s,\mathbf{m}} \lvert\hat{\phi}_{s,\mathbf{m}}(\mathbf{f_x})\rvert^2$	MTF invariance $\min \lvert\hat{\phi}_{s,\mathbf{m}}\rvert^2 / \max \lvert\hat{\phi}_{s,\mathbf{m}}\rvert^2$
Upper bound	$\frac{2A^3T}{3SM\lvert\mathbf{f_x}\rvert^2}$	1
Focus sweep	$\frac{2A^3T}{3\sqrt{3}SM\lvert\mathbf{f_x}\rvert^2}$	$\frac{2}{3}$

58%

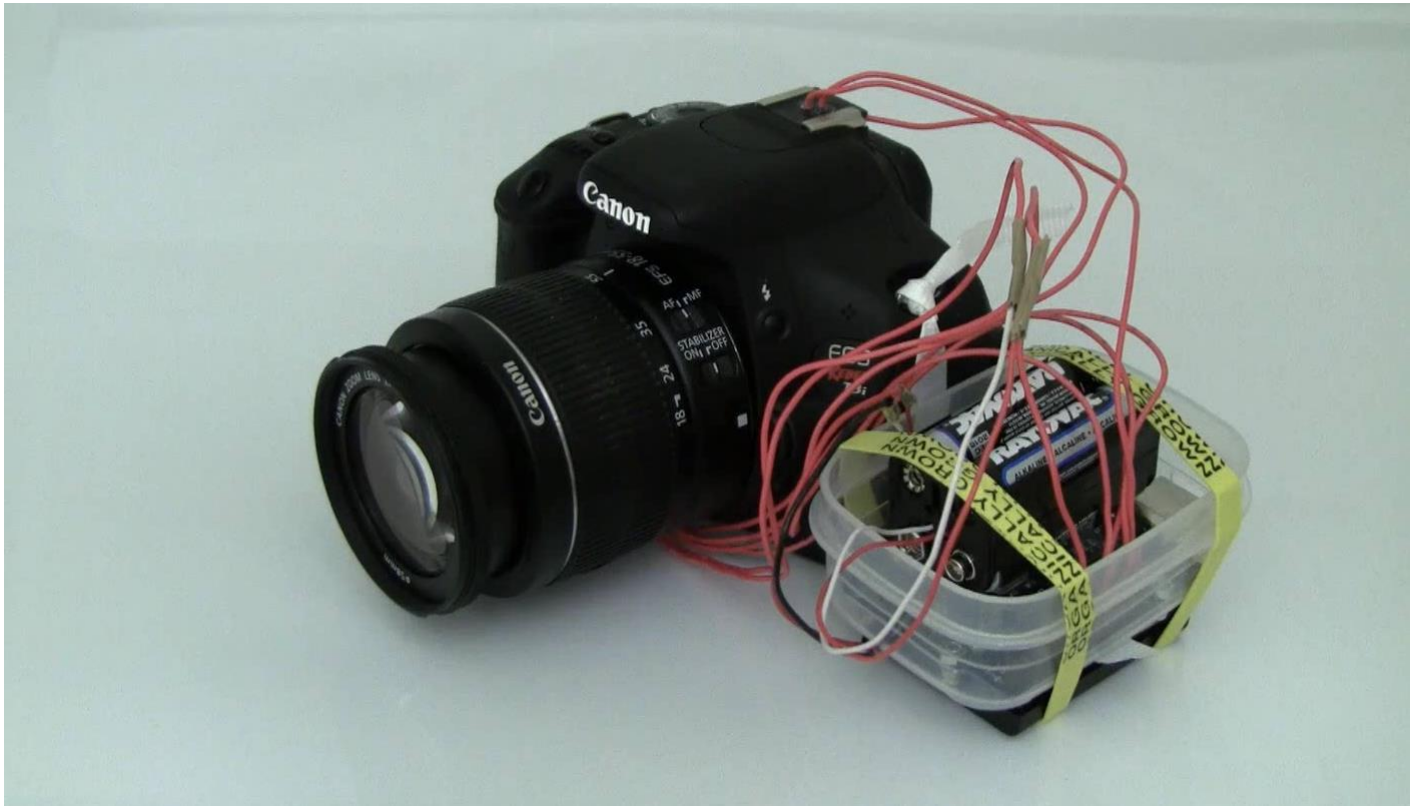
66%

Better than any other existing computational cameras for deblurring

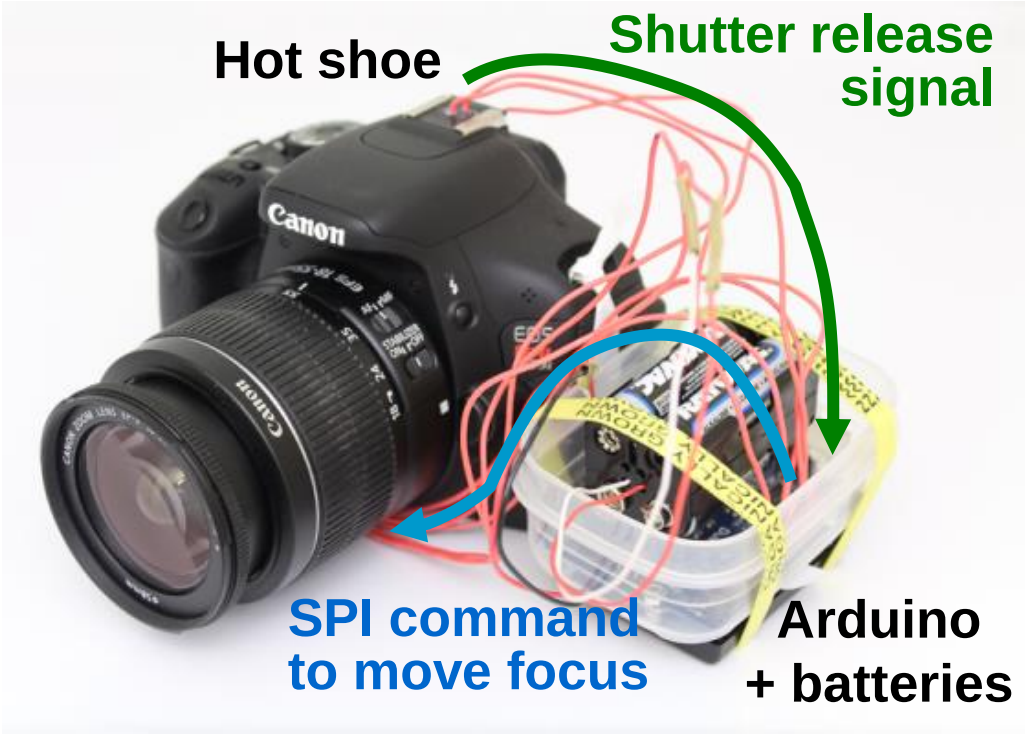
Outline

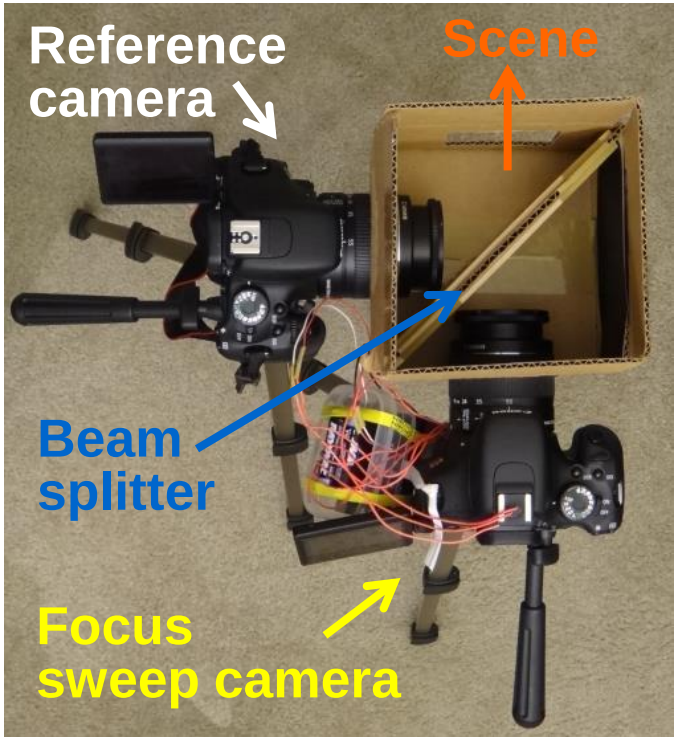
- Motivation
- Related Work
- Intuitions
- Analysis
- Results
- Conclusions

Prototype Focus Sweep Camera

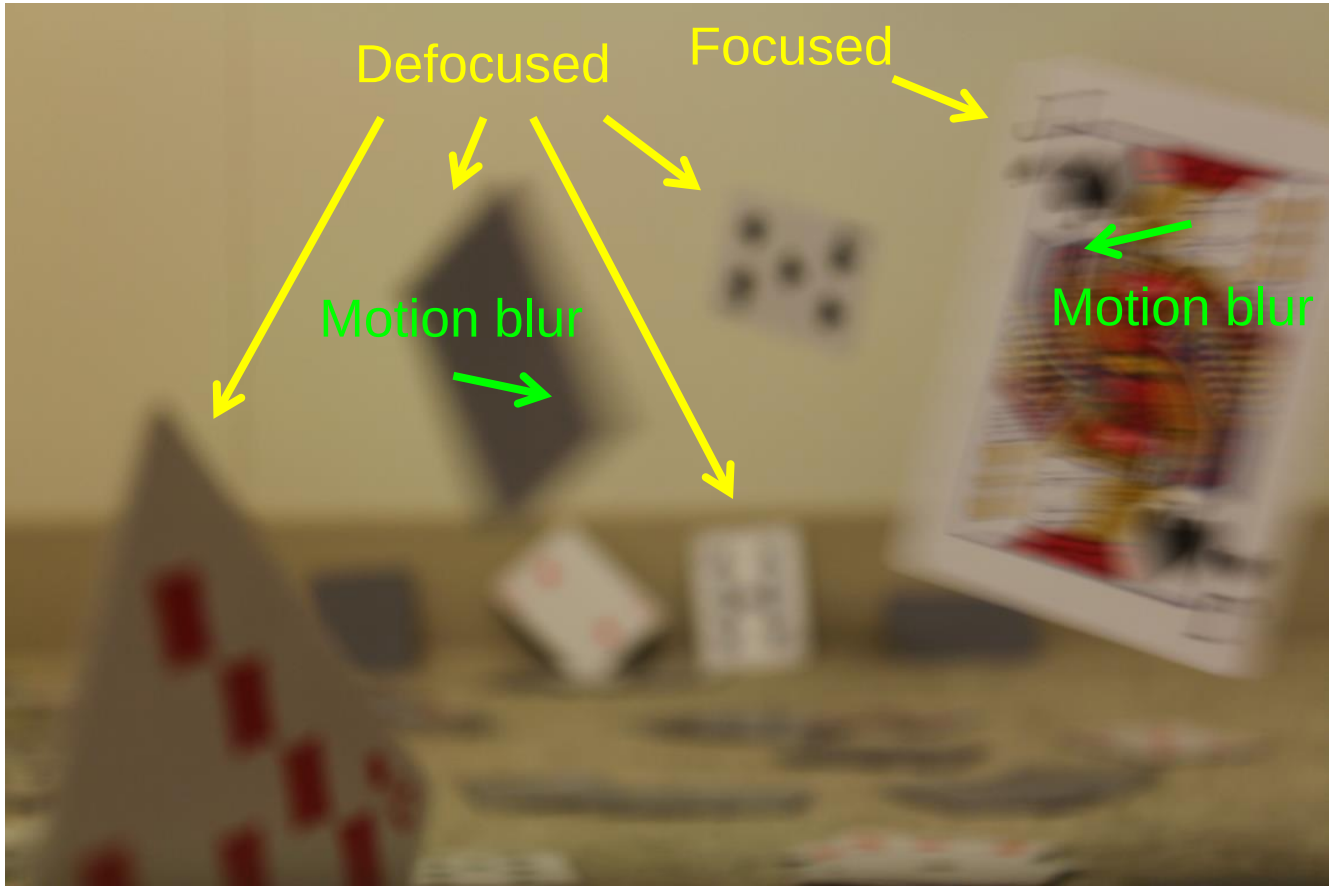


Prototype Camera & Setup

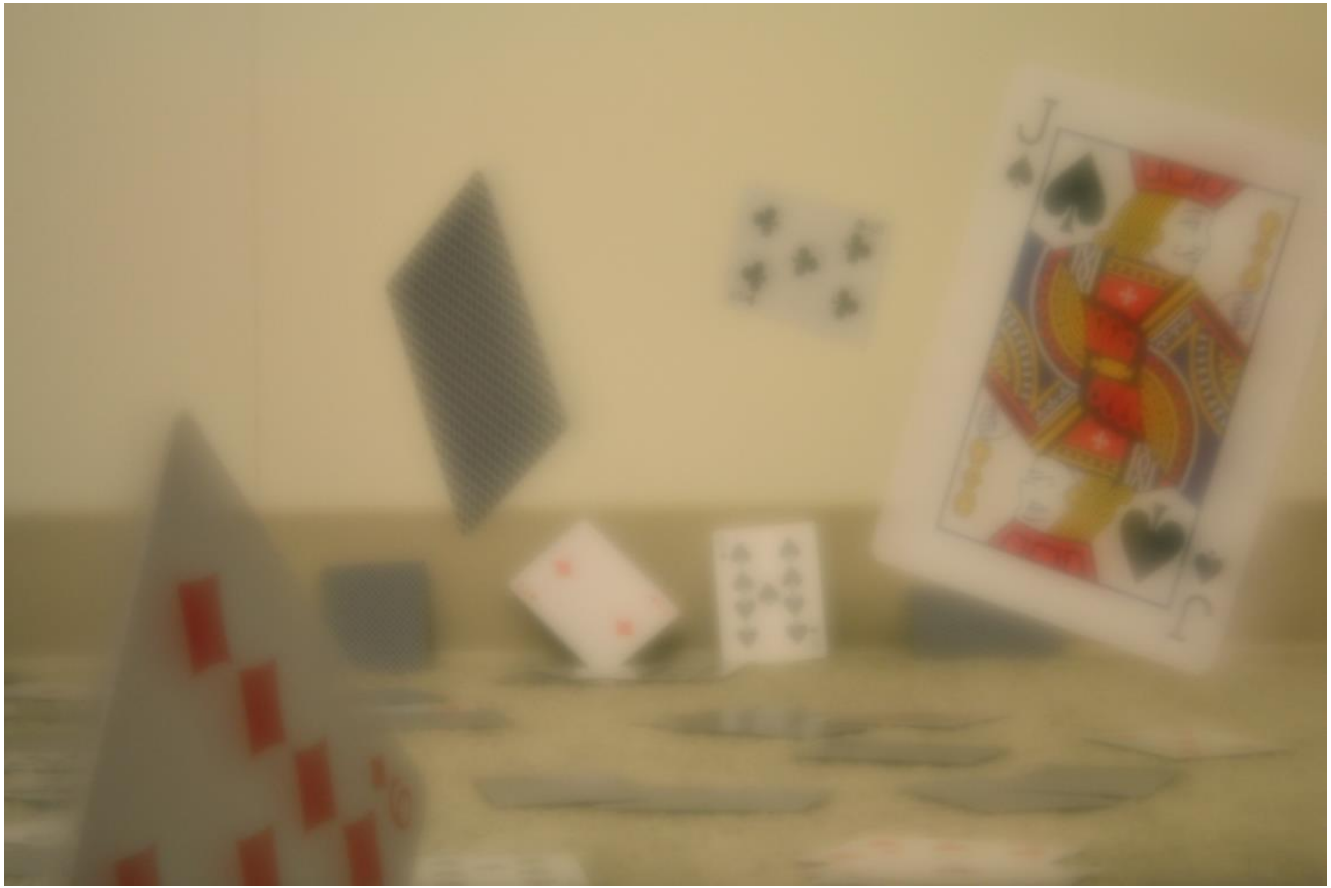




Normal Camera Image

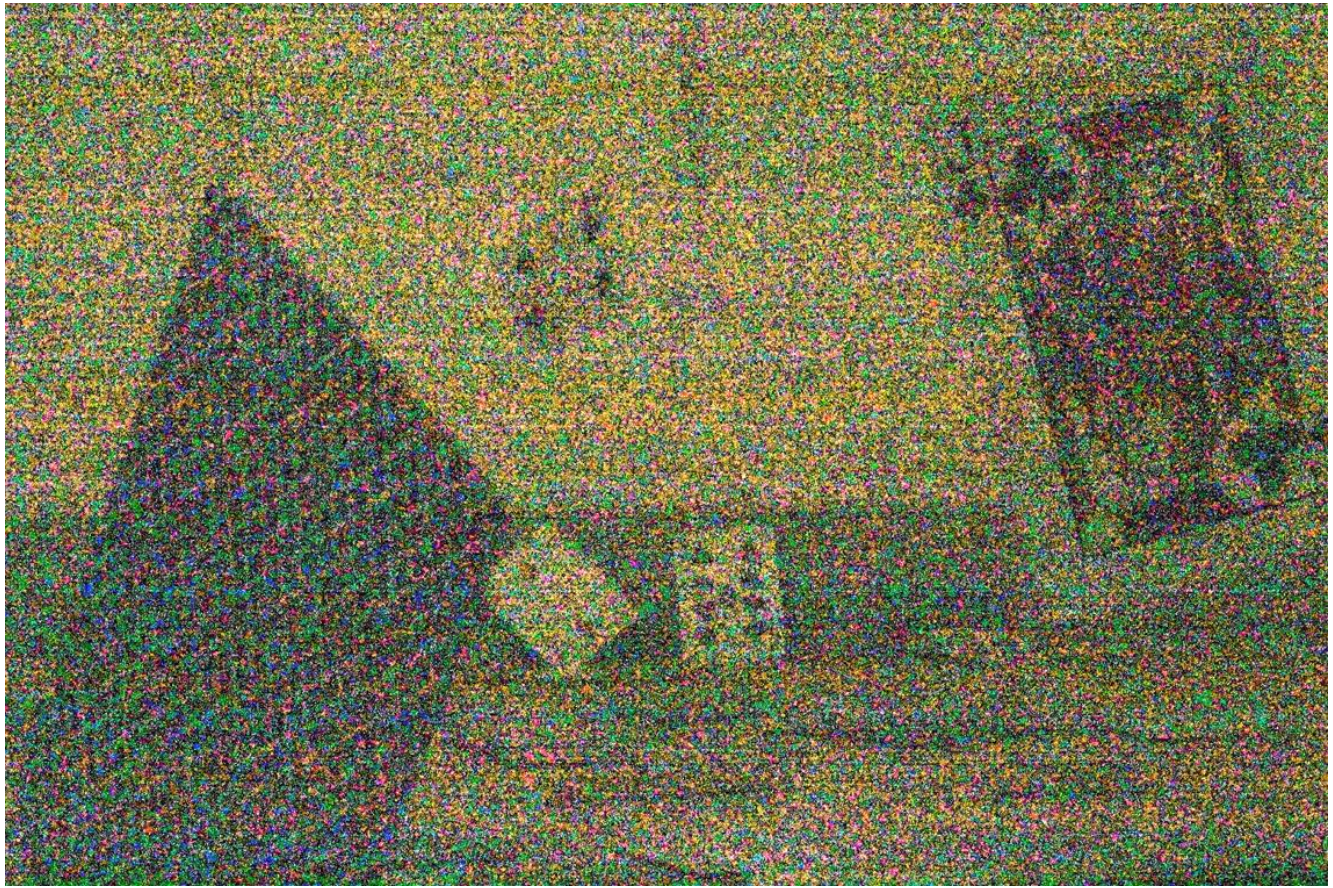


Focus Sweep Image

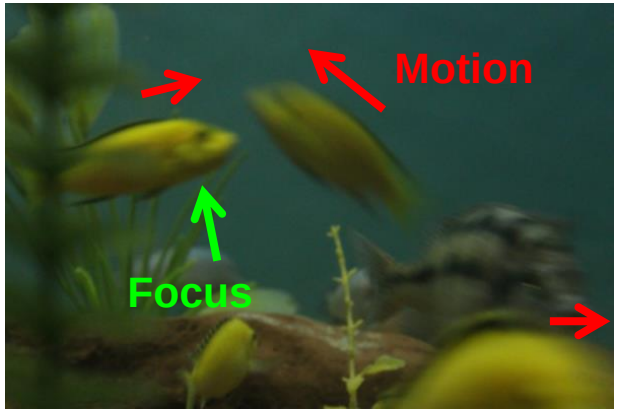


Deconvolution Result

Short Exposure Narrow Aperture Image



More Examples



N/A

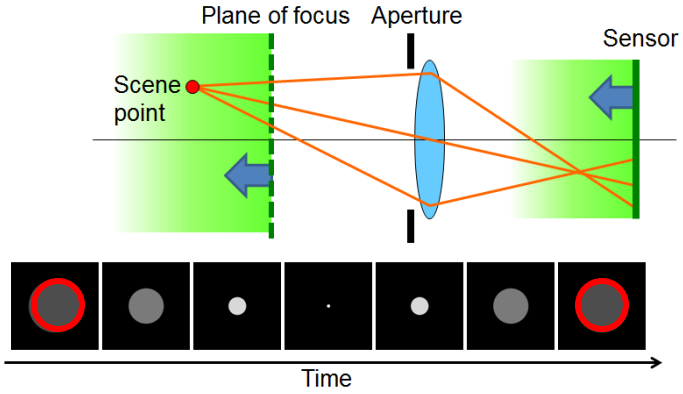
Standard camera

Focus sweep

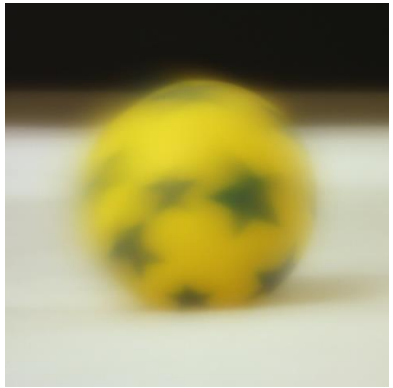
Deconvolution results

Limitations

- Object depth and speed ranges must be bounded
- Depth and speed ranges cannot be adjusted separately
- Object motion must be in-plane linear
- Camera shake cannot be handled



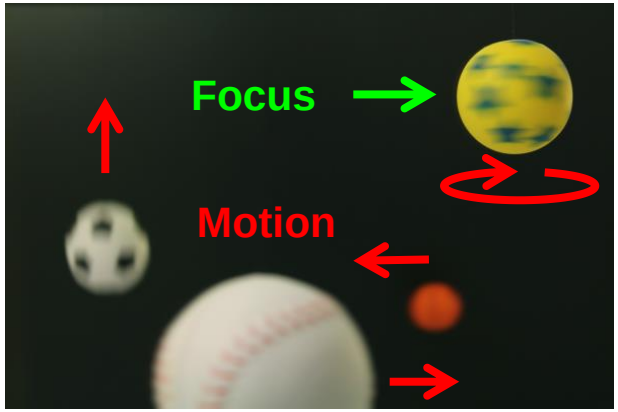
Standard camera

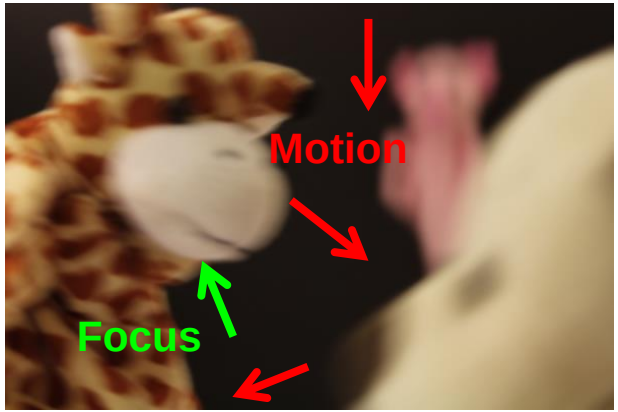


Focus sweep

Deconvolved

Rotation & Z Motion





Standard camera

Focus sweep

Deconvolution results

Summary

- Simple approach to joint defocus & motion deblurring
  - No need for estimating scene depth or motion
  - Also preserves high-frequency image content
  - Theoretically near-optimal
  - Has practical implementation (just firmware update)

Standard camera

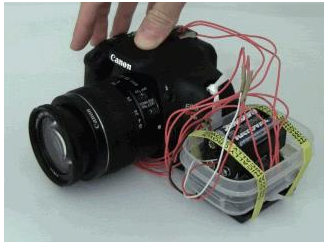
Focus sweep

Deconvolution results

Summary

- Simple joint defocus & motion deblurring
  - No depth or motion estimation
  - Preserves high-frequency
  - Theoretically near-optimal
  - Practical implementation

- http://www.media.mit.edu/~bandy/invariant/
  - How to control the lens
  - How to achieve perfect invariance
    - Computational Cameras & Displays 2013



Acknowledgments

- Yusuke Iguchi
- Noriko Kurachi
- Matthew Hirsch,
- Matthew O'Toole
- Douglas Lanman
- Cheryl Sham
- Sonia Chang
- Shih-Yu Sun
- Jeffrey W. Kaeli
- Bridger Maxwell
- Austin S. Lee
- Saori Bando