# **CS6640** Computational Photography

## 17. Video Stabilization

# iPhone video


[Karpenko et al. 2012]

# Steadicam


John E Fry

- **Engineering solution to jittery motion**
- **Camera mounted on elastic arm supported by vest**
  - Operator does not touch camera
  - Aiming via center-of-gravity gimbal
- **Invented 1976 by Garrett Brown**

# Video from small modern cameras

- **Problem #0: people are not being that careful**

  most of us are not-so-practiced camera operators

  consumers just want to point and shoot anyway
- **Problem #1: small and light cameras**

  great for pocketability

  but low mass and moment of inertia → lots of motion
- **Problem #2: these cameras have rolling shutters**

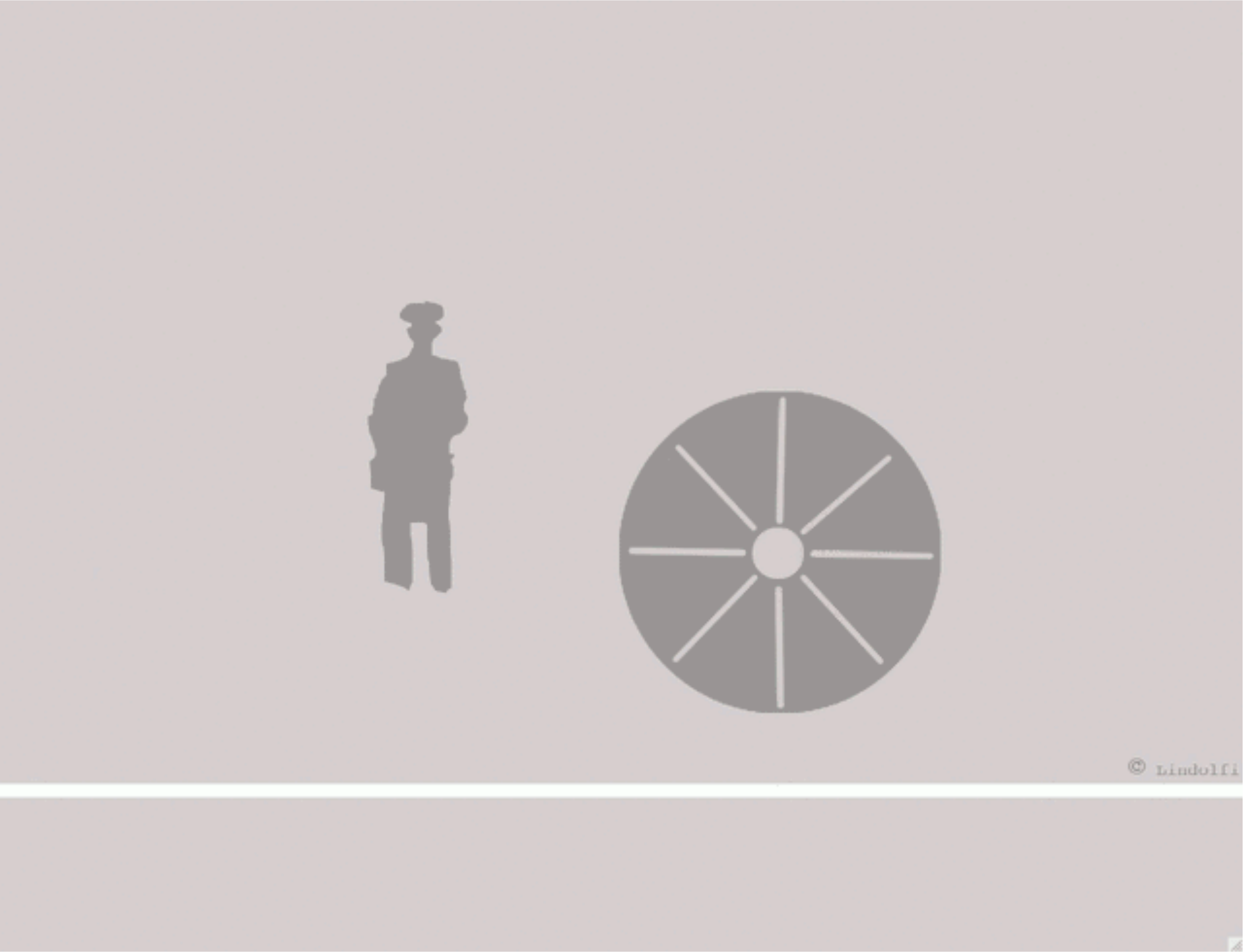
  recall CMOS sensors generally expose and read out sequentially

  (having a global shutter requires somewhere to hide charges)

# Focal plane shutter effects


Jacques Henri Lartigue, 1912

# Focal plane shutter effects




Matlab animation by Bert Otten aka. Lindolfi

**worth a look:**
http://www.largeformatphotography.info/forum/showthread.php?31903-Jacques-Henri-Lartigue-and-his-camera

# Rolling shutter effects

Soren Ragsdale

**more:**
http://www.flickr.com/groups/1485036@N20/

# Video from vehicles

# Compensating for camera motion

- **Basic approach: estimate camera motion and compensate**
- **Shaky camera: resample each frame to simulate fixed camera**
- **Rolling shutter: resample within frames**

  this is not really different, just a more detailed camera model
- **Moving camera: remove unwanted motion but keep intended motion**
- **Basic breakdown of methods: 2D vs. 3D**

  2D methods just model the motion of the image

  3D methods model the motion of the camera
- **Second dimension: measure vs. estimate**

  just like with blur removal, measured camera motion can help

# Agenda

- **Basic 2D**

  Irani et al. 1994: stabilizing with a homography

  Litvin et al. 2003: affine stabilization, mosaicking

  Matsushita et al. 2006: inter-frame and inpainting

- **Full 3D**

  Liu et al. 2009: content-preserving warps

- **Hybrid (2D computations with 3D in mind)**

  Liu et al. 2010: subspace model of image motion

  Ringaby & Forssén 2012: simple method using 3D rotation constraint

- **Measured motion**

  Hanning et al. 2011, Karpenko et al. 2011: using gyros on iPhone 4

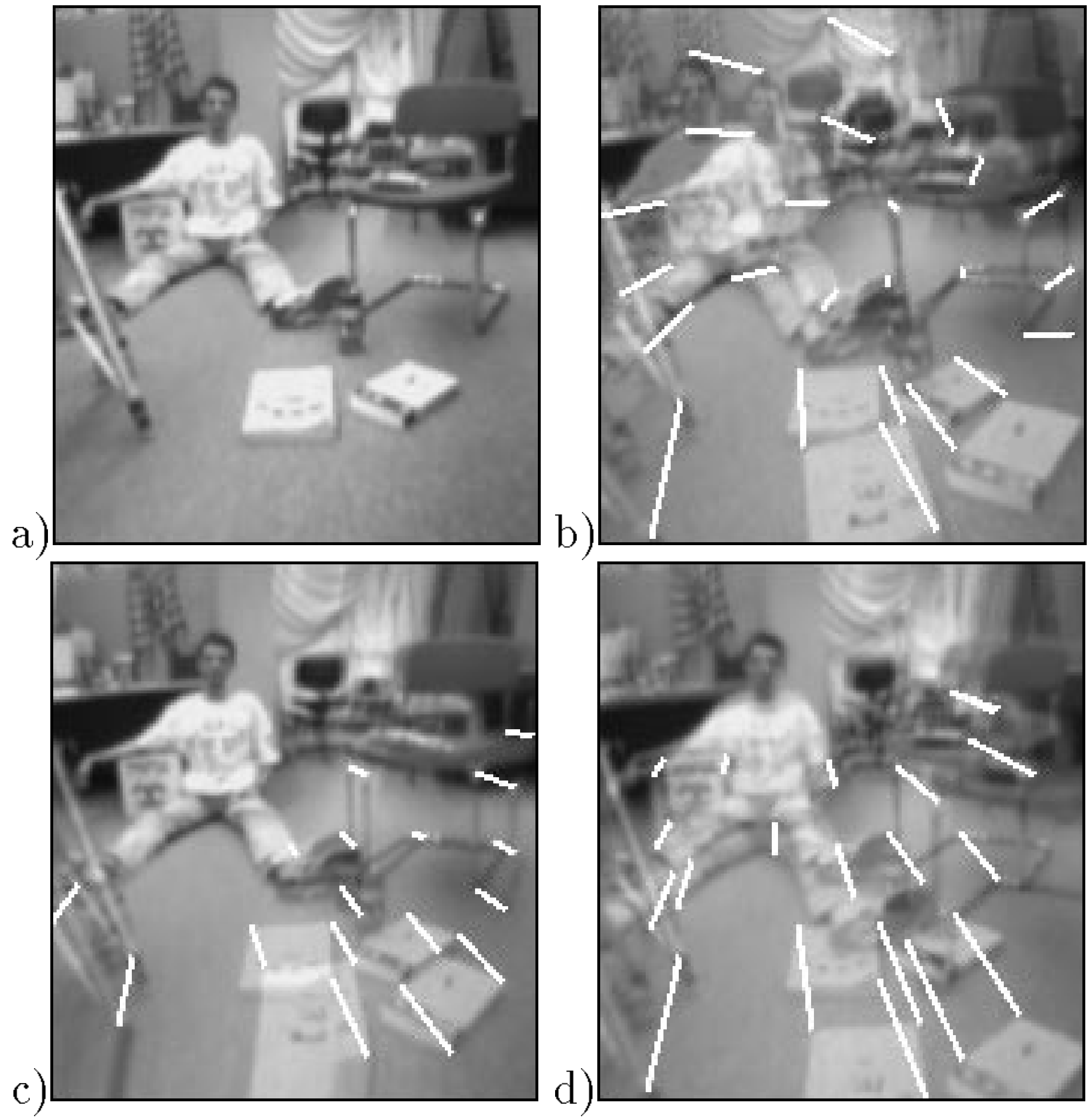
# Irani et al. 1994

- **Estimating “ego-motion” from motion in image**
- **Motion (6D) of camera induces motion (2D) in image**
- **Key fact #1: effect of rotation (about center of projection) merely rearranges the image**

  motion of points is independent of depth

  single motion describes the whole image
- **Key fact #2: effect of translation, once rotation is factored out, moves points along lines through a point**

  center of expansion = projection of translation dir. into image




[Irani et al. 1994]

Figure 3: Camera Stabilization.
a) One of the frames in the sequence.
b) The average of two frames, having both rotation and translation. The white lines display the image motion.
c) The average of the two frames after registration of the shirt. Only effects of camera translation remain.
d) The average of the two frames after recovering the ego-motion, and canceling the camera rotation. This results in a stabilized pair of images.

# Litvin et al. 2003

- **Based on 2D registration**

  find affine warp that best matches adjacent frames

$$\boldsymbol{x}_{n+m} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \mathbf{A}_n^m \boldsymbol{x}_n + \boldsymbol{b}_n^m$$

← motion model between frames $n$ and $n + m$

$$E(\mathrm{I}^n, \mathrm{I}^{n+m}, \mathbf{A}_n^m, \boldsymbol{b}_n^m) = \sum_{\boldsymbol{x} \in \chi} \varphi\left(\mathrm{I}^n(\boldsymbol{x}) - \mathrm{I}^{n+m}(\mathbf{A}_n^m \boldsymbol{x} + \boldsymbol{b}_n^m)\right)$$

← minimize this to fit motion

- **Describe "camera motion" by accumulating transforms**

$$\tilde{\mathbf{T}}_n = \prod_{k=1}^{n} \mathbf{T}_n \qquad \mathbf{T}_n(\mathbf{x}) = \mathbf{A}_n^1 \mathbf{x} + \mathbf{b}_n^1$$

- **Define desired motion by smoothing with Kalman filter**

  estimate smooth trajectory assuming shake is noise

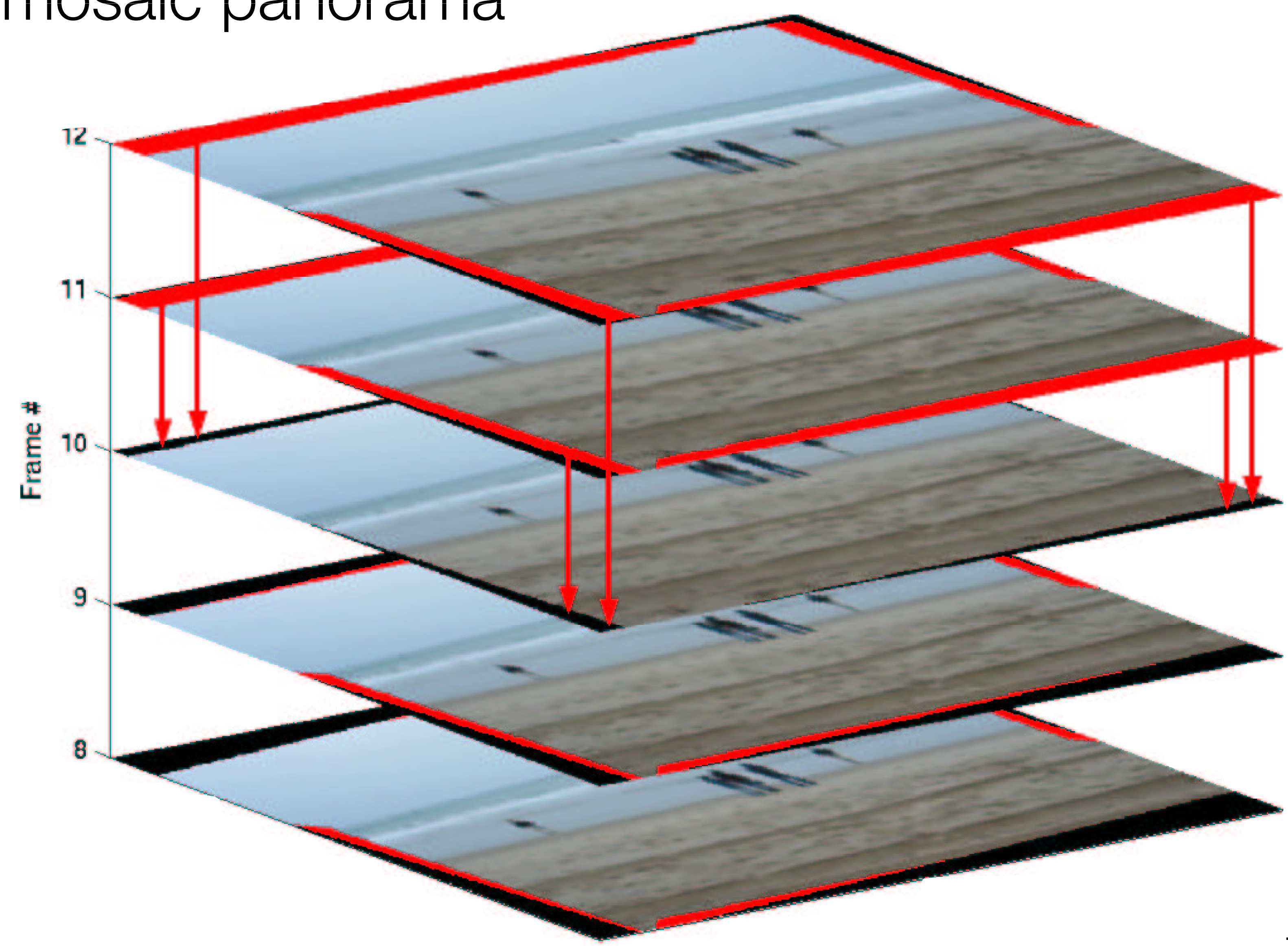
$$\{\mathbf{T}_n\} \rightsquigarrow \{\hat{\mathbf{T}}_n\}$$

# Litvin et al. 2003

- **Warp each frame to match smoothed motion**

$$\bar{\mathbf{T}}_n = \hat{\mathbf{T}}_n \tilde{\mathbf{T}}_n^{-1}$$

- **Fill in missing bits from nearby frames**
  this is akin to building a mosaic panorama



# Litvin et al. 2003

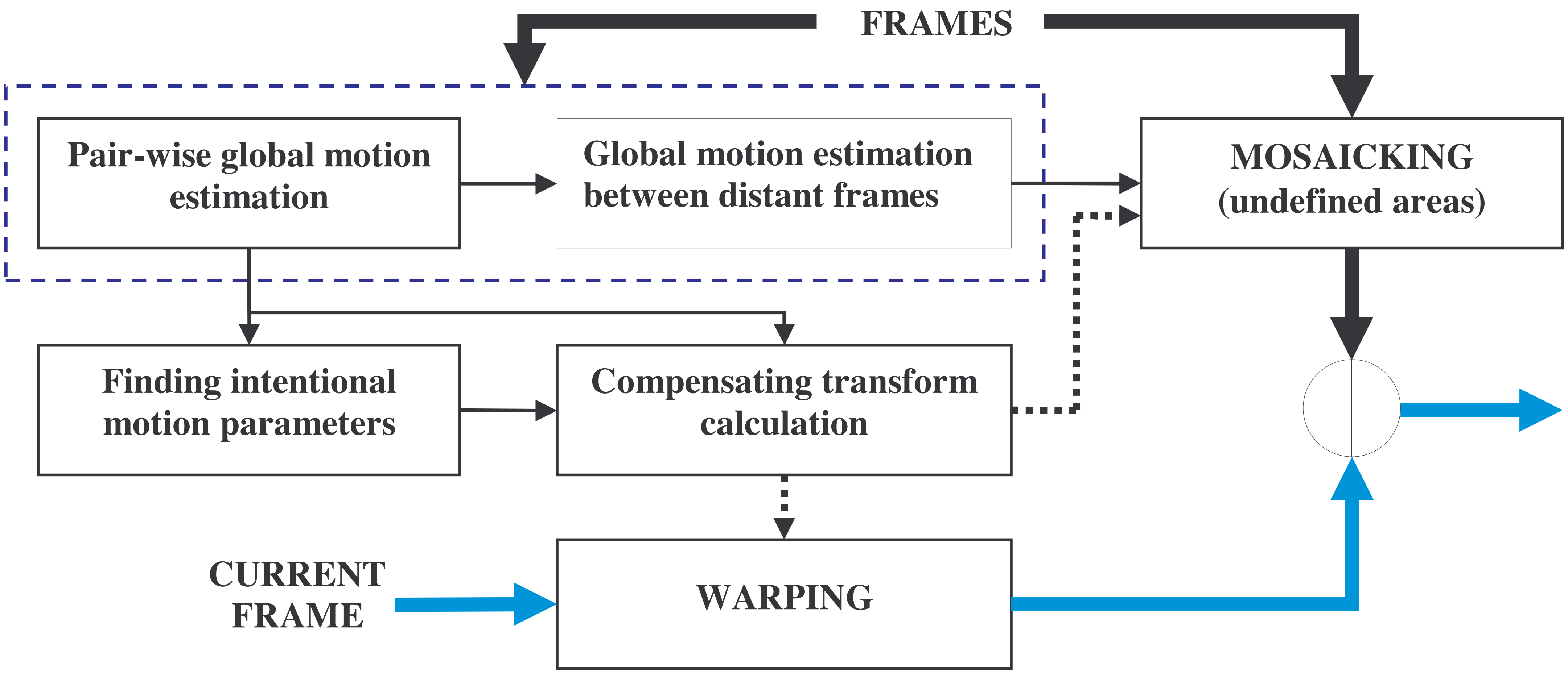



Figure 1: Video stabilization algorithm. Flow of frames (intensities) is shown by thick arrows.

[Litvin et al. 2003]

[Litvin et al. 2003]

translation only, no mosaicking

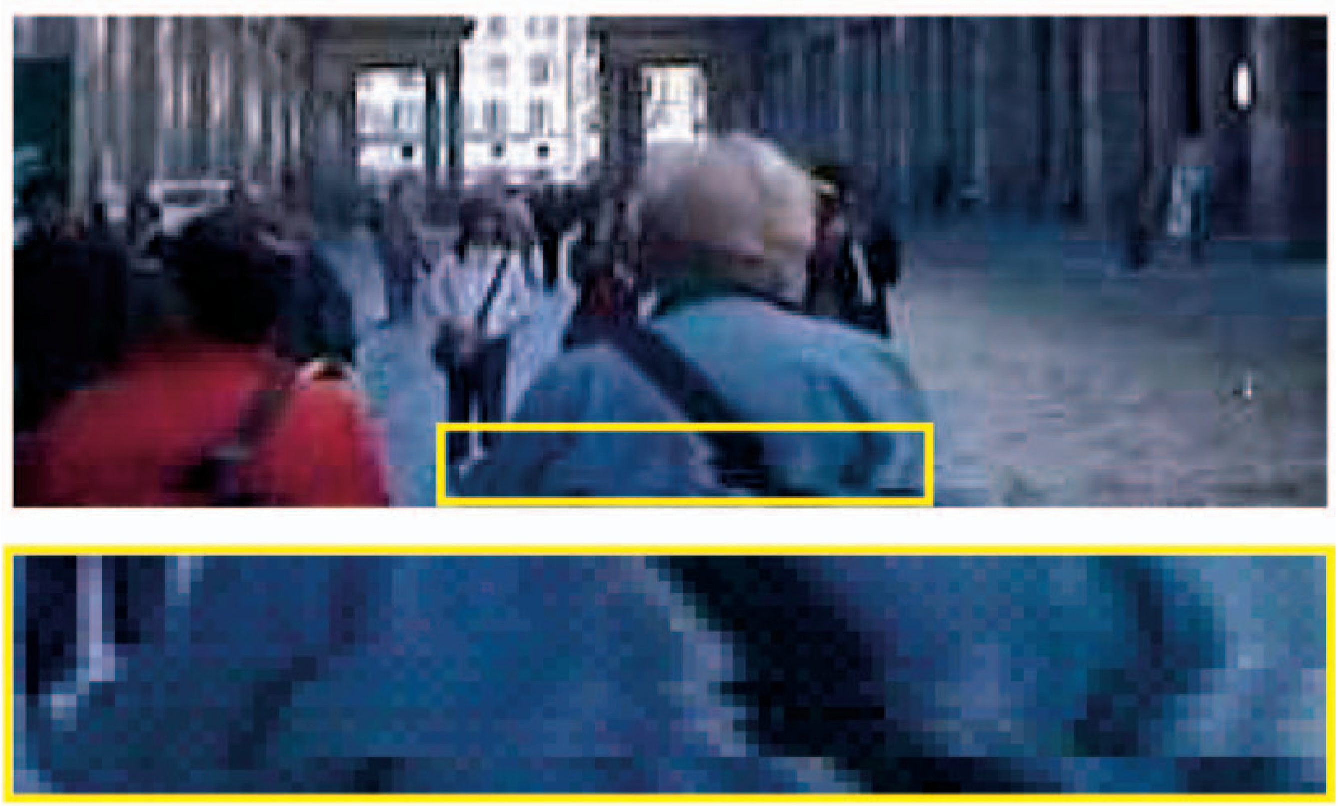
[Litvin et al. 2003]

[Litvin et al. 2003]

[Litvin et al. 2003]

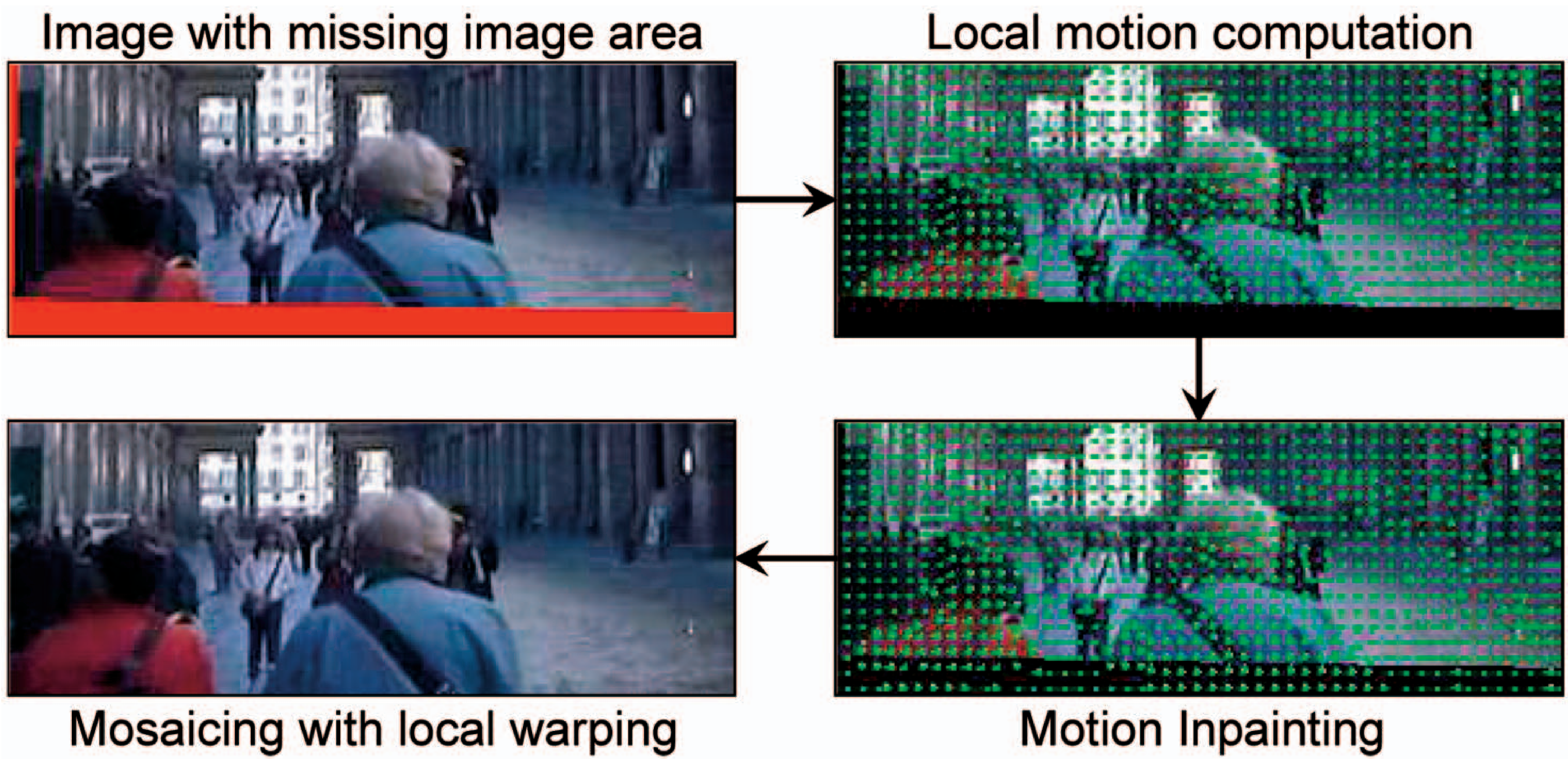
# Matsushita et al. 2006

- **Also uses affine motion model + smoothing**
- **Problem with mosaicking to fill in: global motion model does not fit locally**

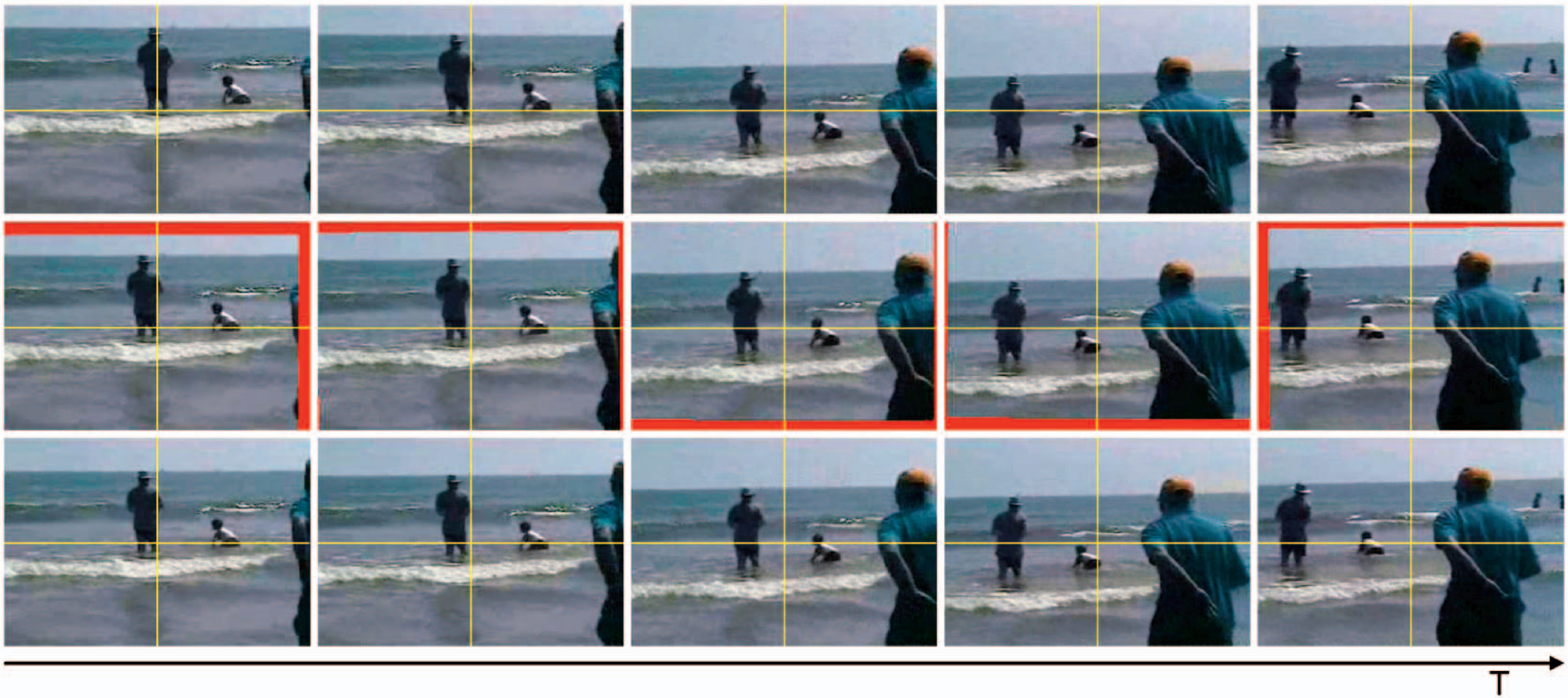


- **Solution: use local motion (optical flow)**
- **With flow in hand, also de-blur**

# Matsushita et al. 2006



# Matsushita et al. 2006



# Matsushita deblurring

- **Registration does not remove blur**

  Follow optical flow looking for sharper pixels to copy

registration only

with deblurring

# 3D methods

- **2D stabilization cannot model large camera *translations***
  - depth-dependent motion of points
  - occlusions and disocclusions
- **Warping of images to change viewpoint is Image-Based Rendering**
  - long-studied graphics/vision topic (since mid 90s)
  - construct weak model of scene, use that to predict motion
- **3D stabilization approach**
  1. reconstruct 3D geometry using Structure from Motion
  2. filter 6D camera path to get smooth path
  3. compute warps and apply them to frames

# 3D stabilization

[Liu et al. 2009]

unstabilized

# 3D stabilization

[Liu et al. 2009]

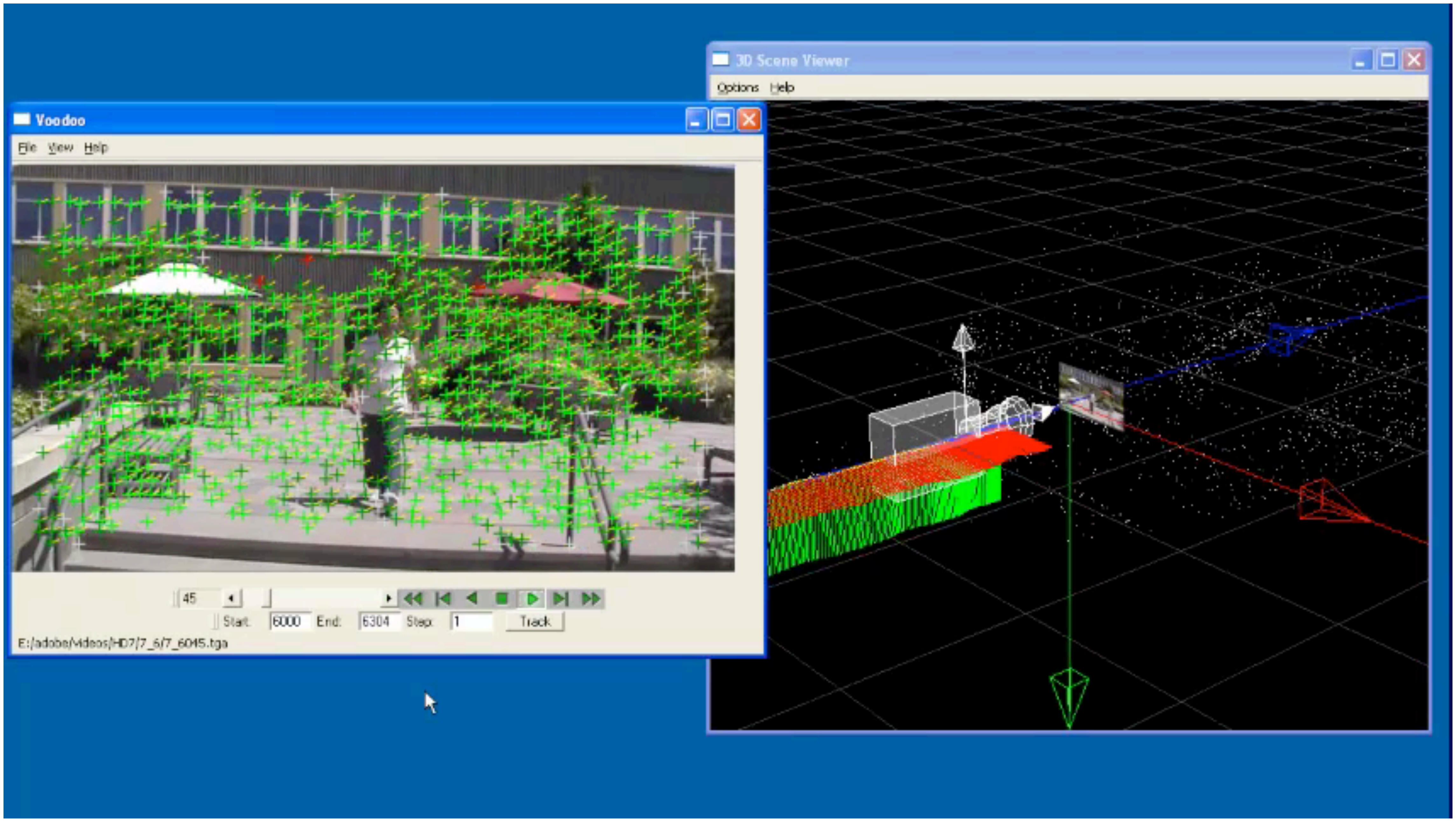
2D stabilization

# 3D stabilization

[Liu et al. 2009]

naïve stabilization with SFM

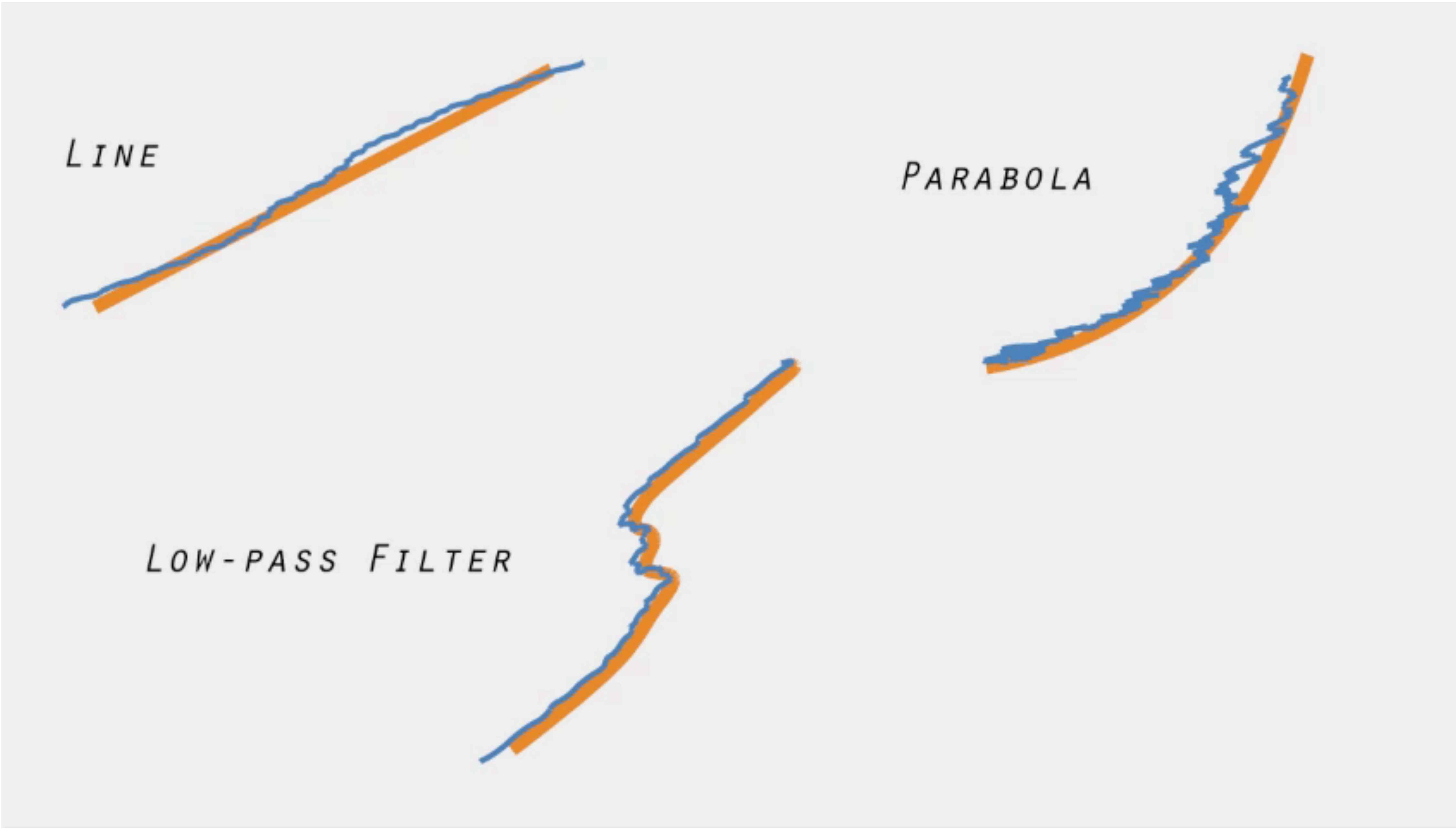
# Liu et al. 2009




[Liu et al. 2009]

running Voodoo SFM software

# Defining 3D camera paths




[Liu et al. 2009]

# Liu et al. 2009

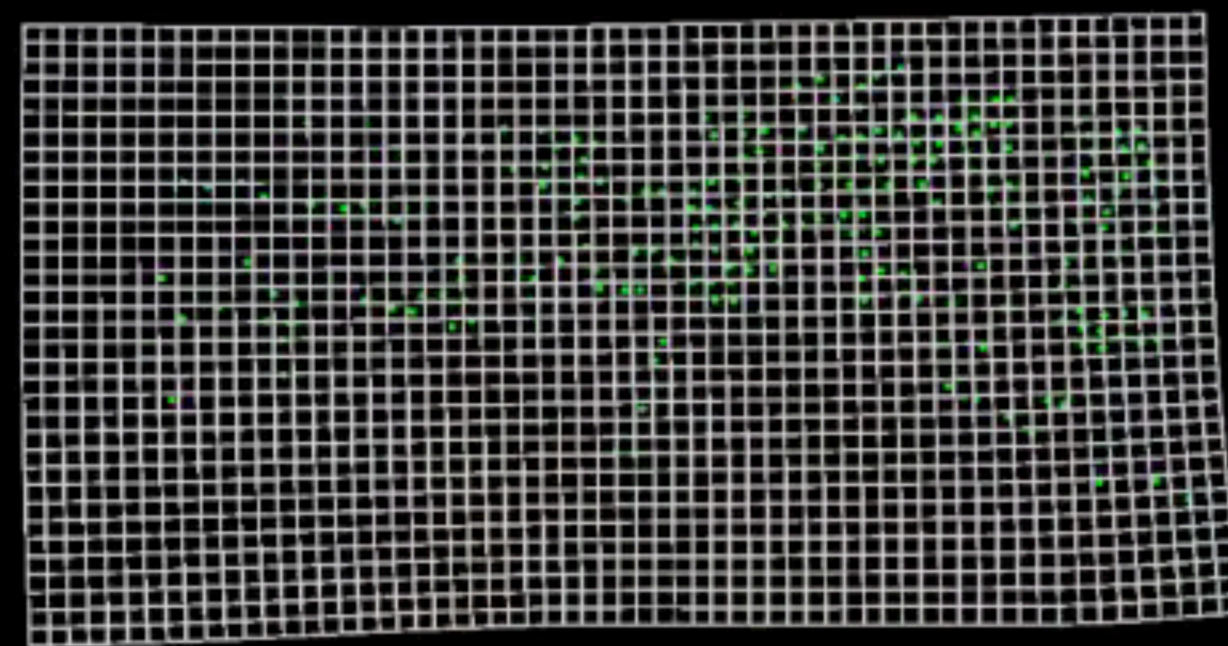
- **Key idea: define 2D warp based on 3D model**
- **Use projected 3D features to define a smooth warp**

  uses “as-rigid-as-possible formulation”

  weight distortion according to salience (a la Seam Carving)

  also enforce temporal coherence

Grid Mesh & Points



Output

[Liu et al. 2009]

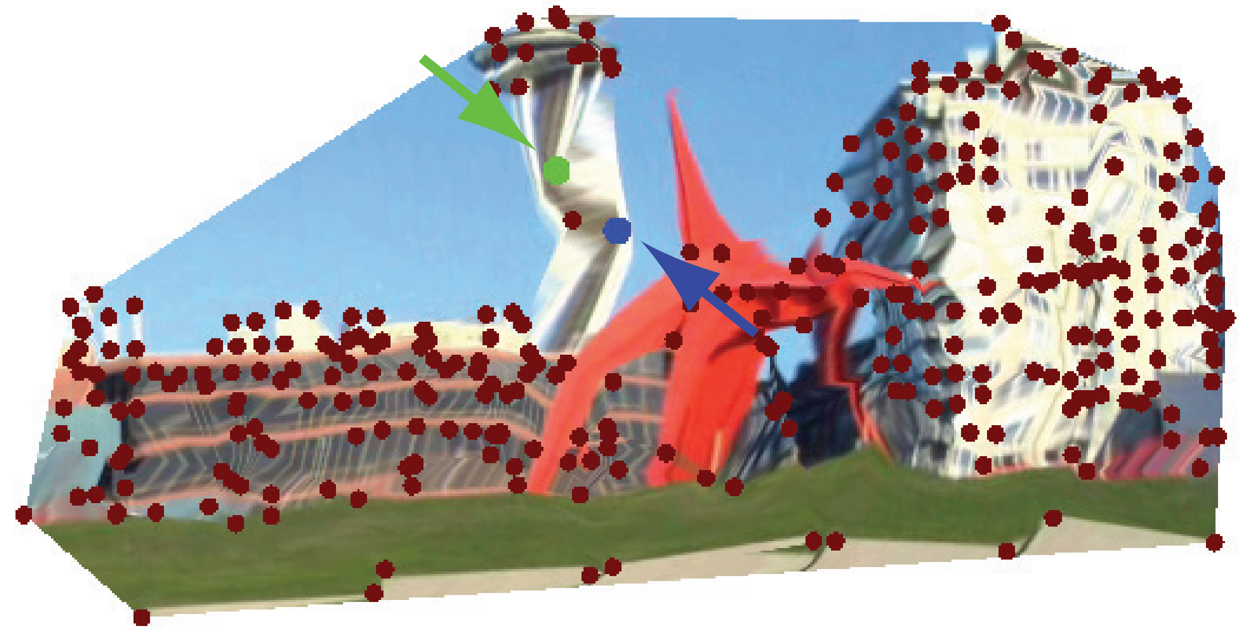

[Liu et al. 2009]

# Hybrid 2D/3D methods

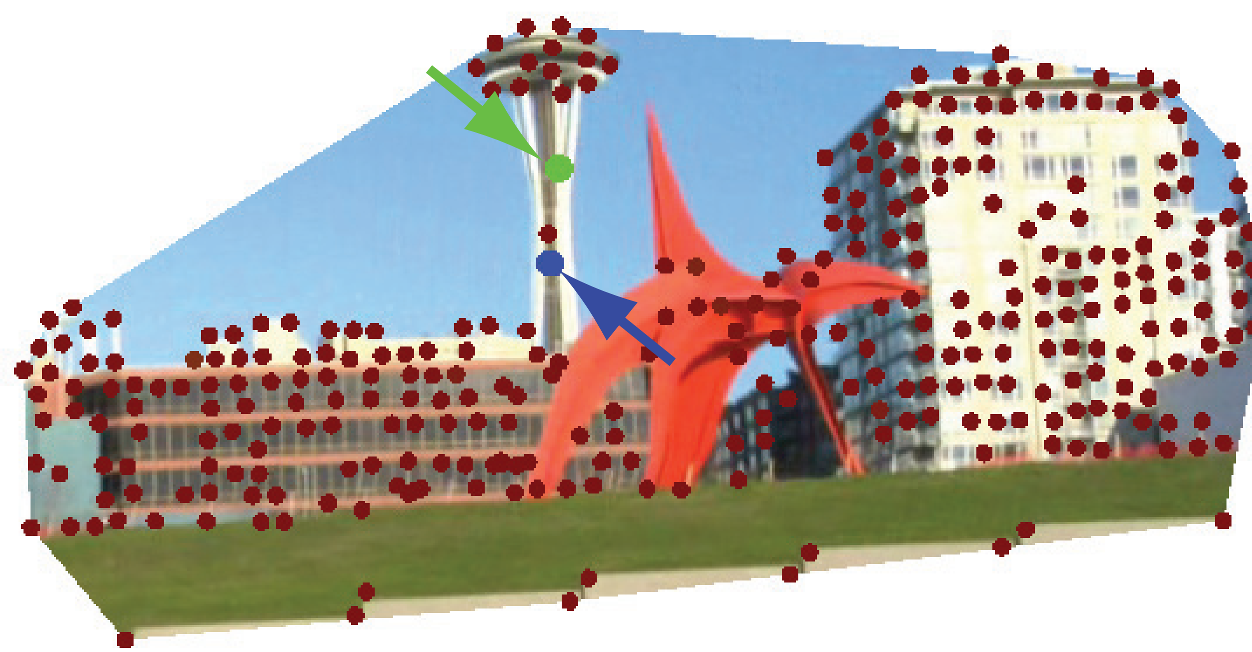
- **Problems with 3D**
  - big slow computation
  - can be fragile
- **Recall back to Irani 94**
  - contribution of translation to motion is constrained
  - has to be set of vectors pointing to FOE
- **More practical/robust: compute in 2D but use constraints derived from 3D considerations**

# Liu et al. 2010

- **Key idea: motion of 2D features is low rank**
- **Project feature motion into low-rank subspace, then fit warp**

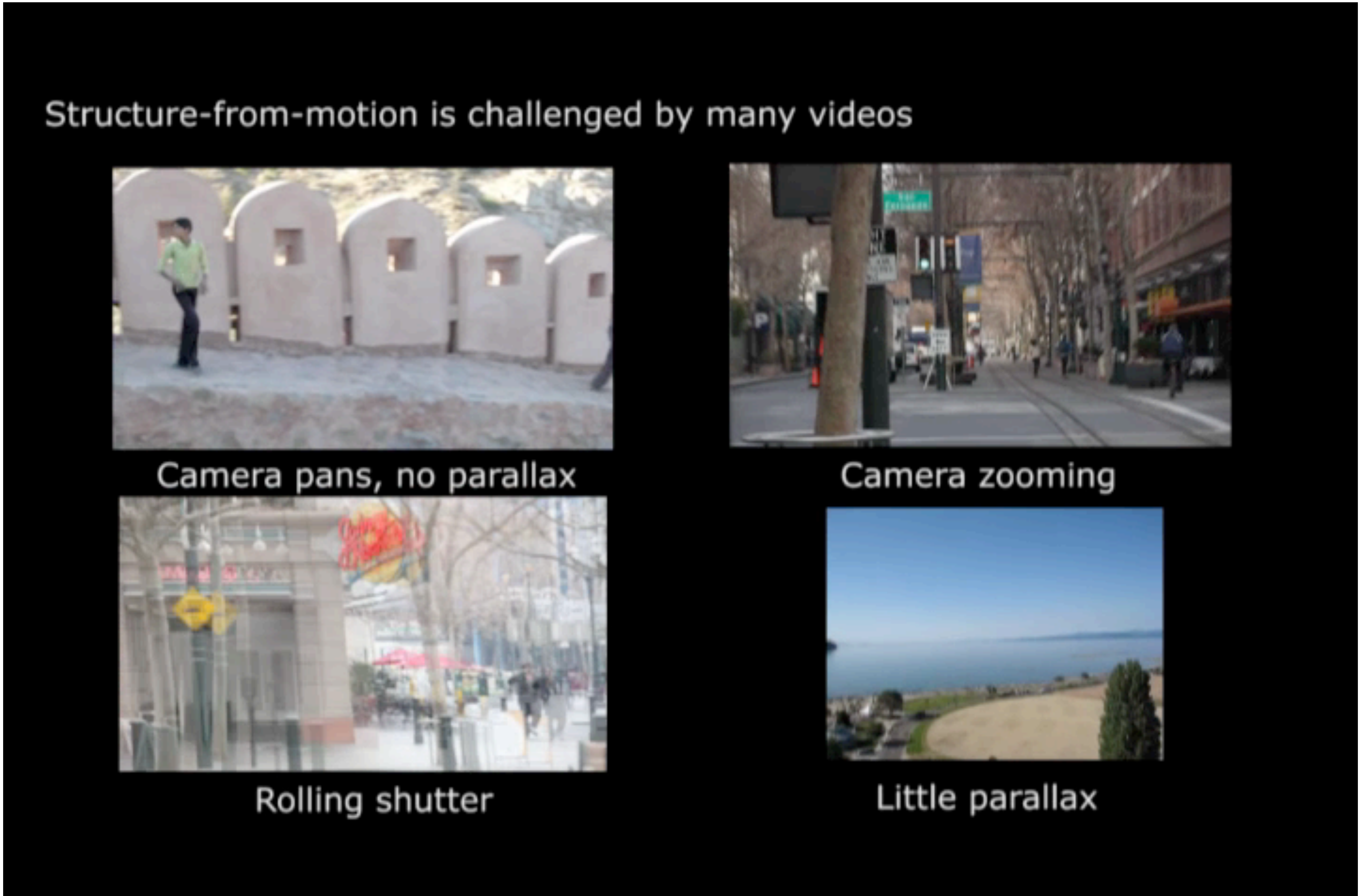


(a) Filter each trajectory independently



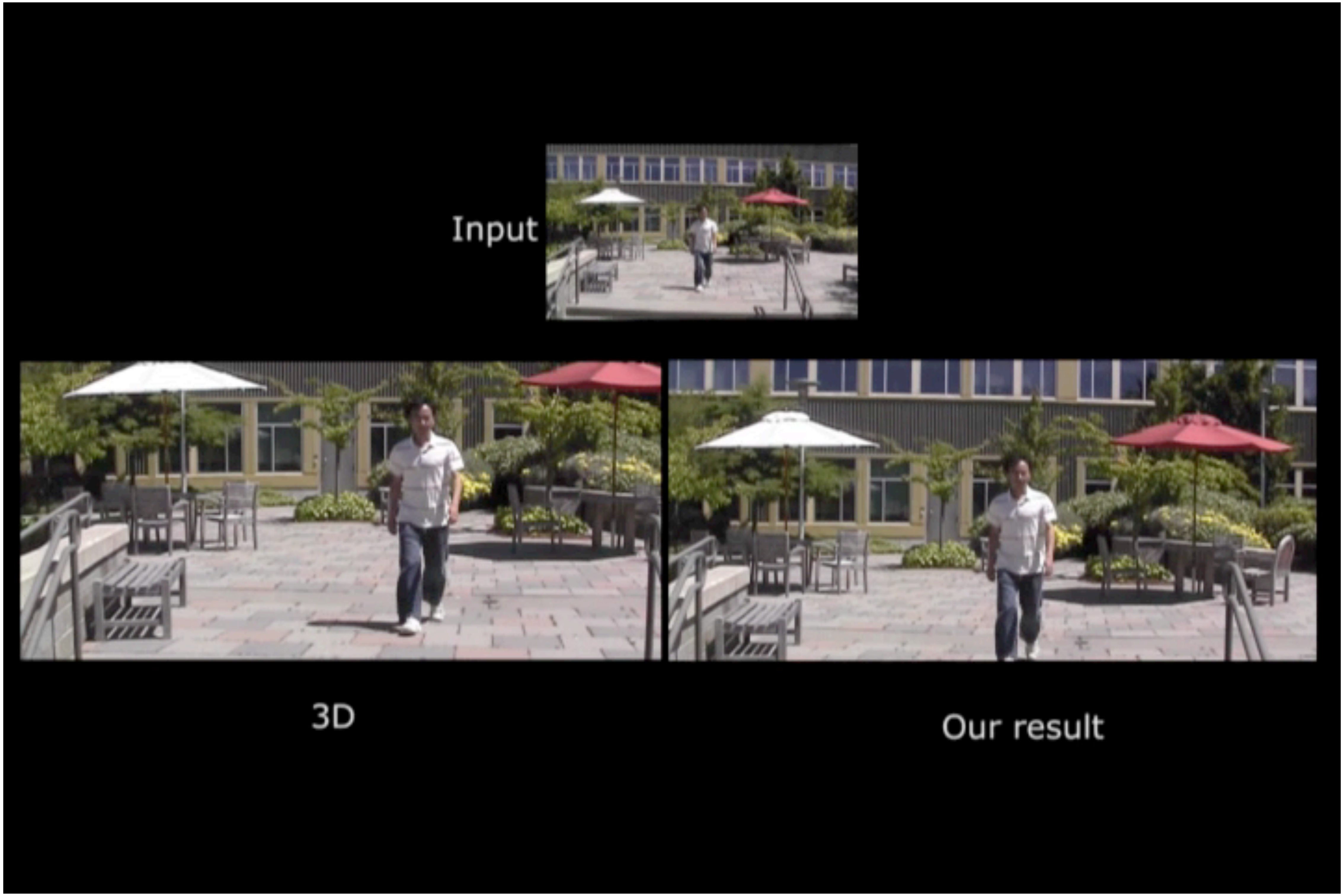
(b) Filter the eigen-trajectories

# Problem cases for 3D




[Liu et al. 2010]

# Liu et al. 10 results




[Liu et al. 2010]

# Ringaby & Forssén 2012

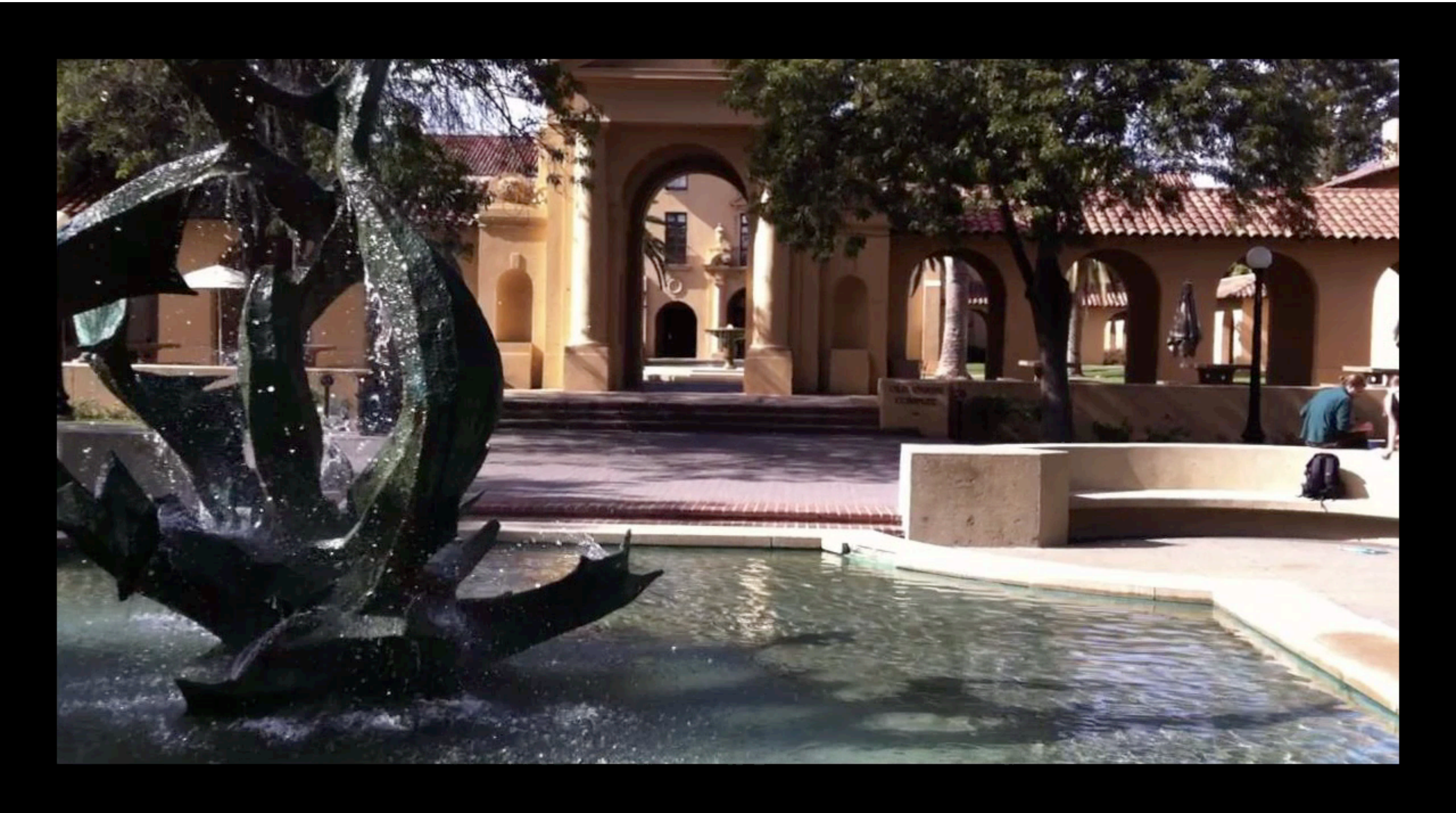
- **Simple answer to estimation: measure it!**
- **iPhone 4 came with gyros; use them**
- **Kalman filter to smooth out noise**
- **Note this makes 2D homographies that correspond to 3D *rotations* only**

# Hanning et al. 2011, Karpenko et al. 2011

- **Simple answer to estimation: measure it!**
- **iPhone 4 came with gyros; use them**
- **Kalman filter to smooth out noise**
- **Note this makes 2D homographies that correspond to 3D *rotations* only**
- **After iPhone 4 this was low-hanging fruit!**

  two groups did it

# Karpenko et al. 2011 results



# Bibliography


C. Buehler, M. Bosse, & L. McMillan, **Non-metric image-based rendering for video stabilization**, *CVPR 2001*.

A. Fitzgibbon, Y. Wexler, & A. Zisserman, **Image-based rendering using image-based priors**, *ICCV 2003*.

G. Hanning, N. Forslow, P. Forssen, E. Ringaby, D. Tornqvist, and J. Callmer, **Stabilizing cell phone video using inertial measurement sensors**, *Second IEEE International Workshop on Mobile Vision* (2011).

M. Irani, B. Rousso, and S. Peleg, **Recovery of ego-motion using image stabilization**, *CVPR 1994*.

A. Karpenko, D. Jacobs, J. Baek, and M. Levoy, **Digital video stabilization and rolling shutter correction using gyroscopes**, *Stanford University Tech. Rep. CTSR 2011-03.*

A. Litvin, J. Konrad, and W.C. Karl, **Probabilistic video stabilization using Kalman filtering and mosaicing**, *SPIE Conference on Electronic Imaging 2003.*

F. Liu, M. Gleicher, H. Jin, and A. Agarwala. **Content-preserving warps for 3D video stabilization**, *SIGGRAPH 2009*.

F. Liu, M. Gleicher, J. Wang, H. Jin, and A. Agarwala, **Subspace video stabilization**, *TOG* 30:1 (2011).

Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.Y. Shum, **Full-frame video stabilization with motion inpainting**, *PAMI* 28:7 (2006).

E. Ringaby & P.E. Forssén, **Efficient Video Rectification and Stabilisation for Cell-Phones**, *IJCV* 96:3 (2011).