Computational 3D Photography

Extracting Shape, Motion and Appearance from Images

Marc Pollefeys

ETH Zurich

Qualcomm AR lecture

29 November 2011

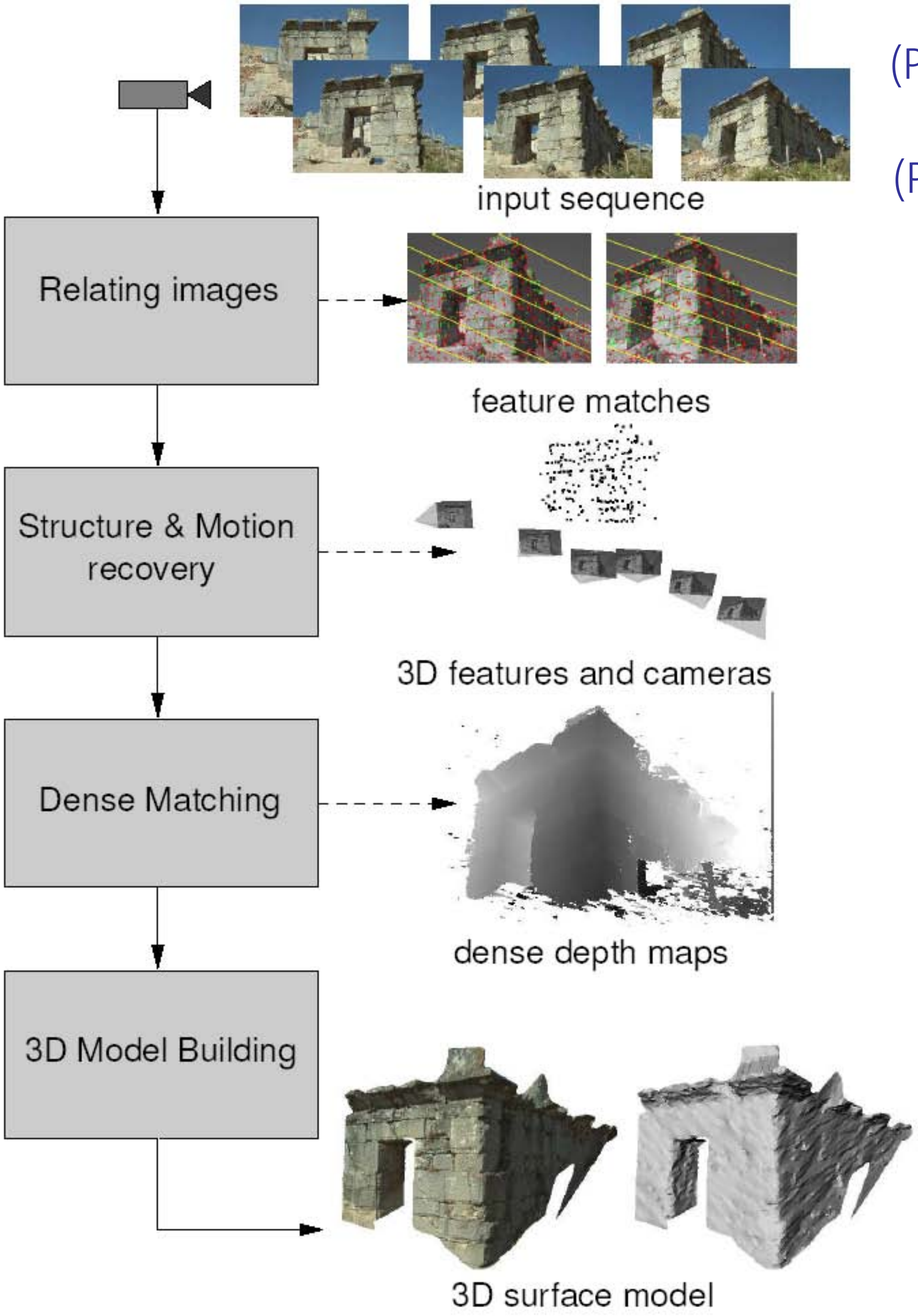
ETH Zürich

Computer Vision
and Geometry Lab

(Pollefeys et al. ICCV’98)

…

(Pollefeys et al. IJCV’04)



Relating images

Structure & Motion recovery

Dense Matching

3D Model Building

input sequence

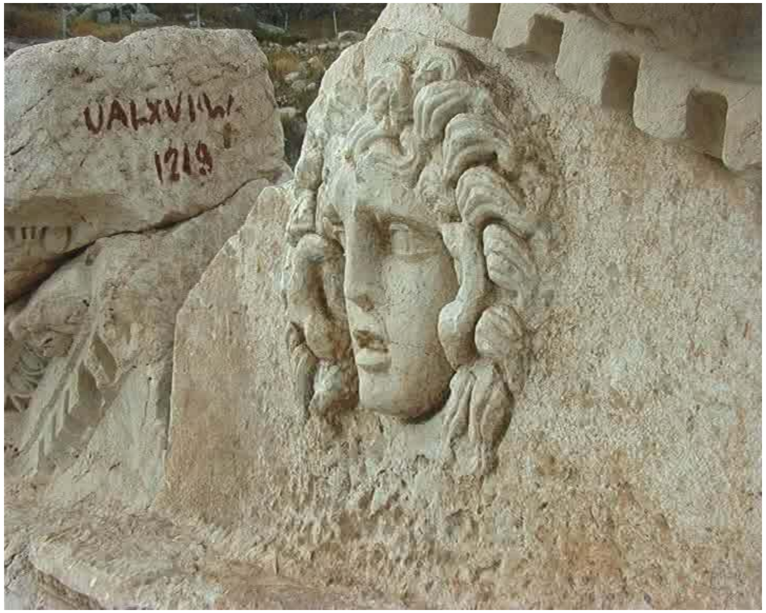
feature matches

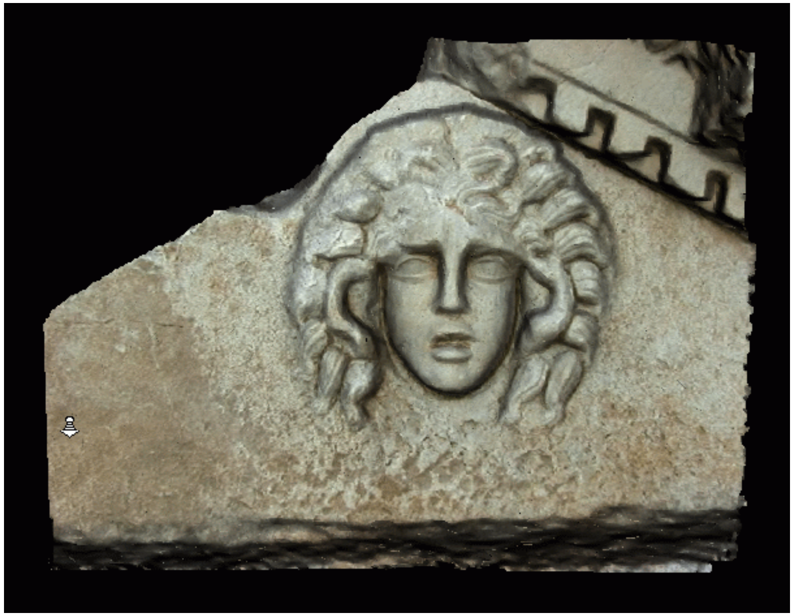
3D features and cameras

dense depth maps

3D surface model

Video → 3D model





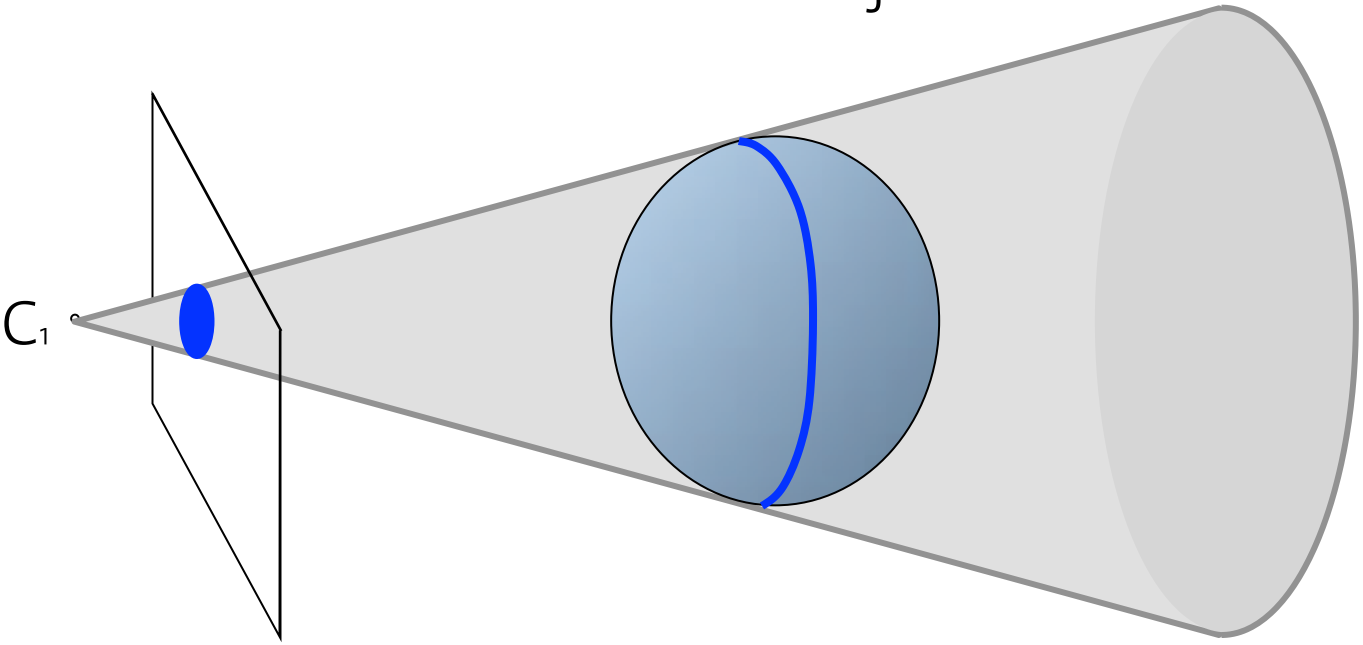
accuracy ~1/500 from DV video (i.e. 140kb jpegs 576x720)

Talk outline

- Introduction
- Object modeling
- Scene modeling
- People/event modeling
- Summary and conclusion

2D → 3D reconstruction: silhouette constraints

Additional constraint for closed objects

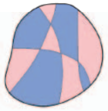


Silhouettes

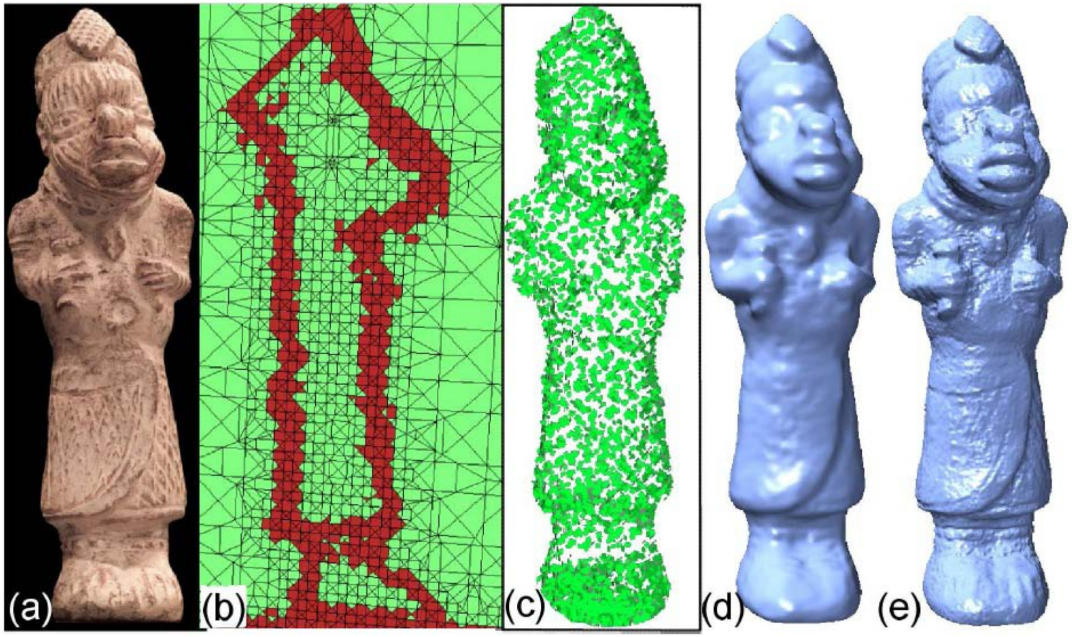
- object inside cone (visual hull)
- object tangent to cone (rim)

Multi-view 3D object reconstruction

- Combine dense matching with silhouette constraints (Compute graph min-cut to obtain watertight surface)
  - Exact silhouettes (Sinha & Pollefeys ICCV□05)
  - Photo-consistency adaptive tetrahedral mesh (Sinha et al. ICCV□07)



(two-colored rim-mesh)

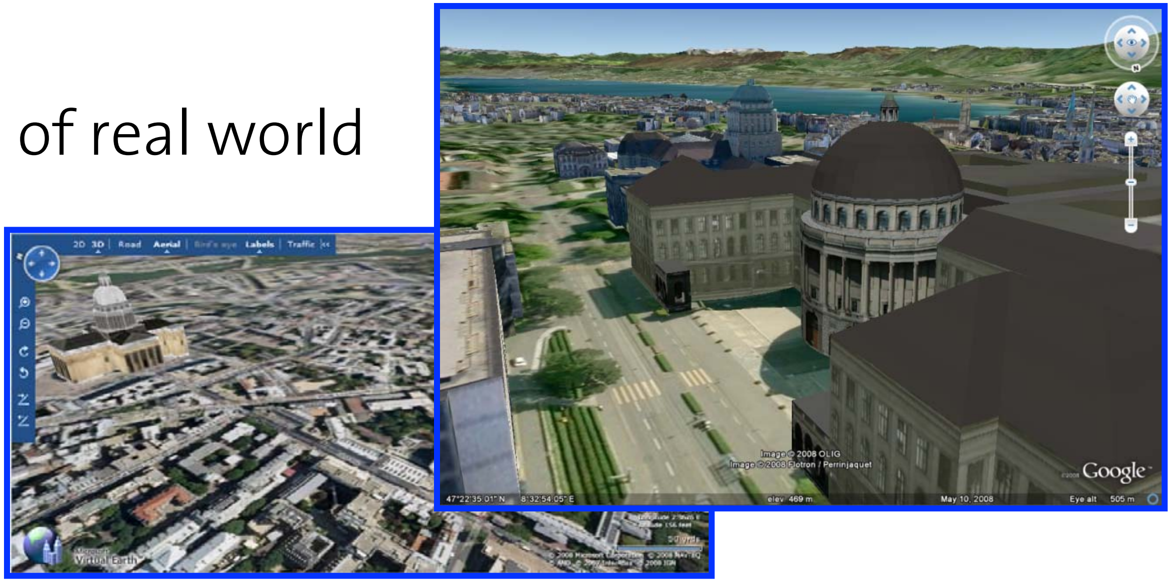


Talk outline

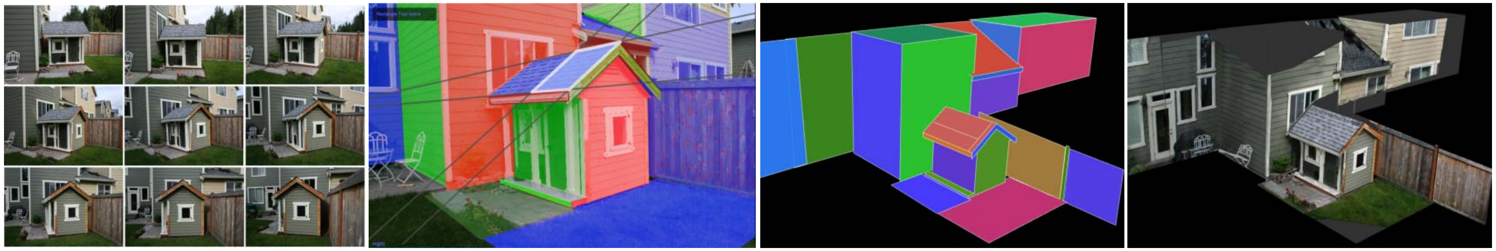
- Introduction
- Object modeling
- Scene modeling
- People/event modeling
- Summary and conclusion

Modeling the world

- Need for 3D models of real world

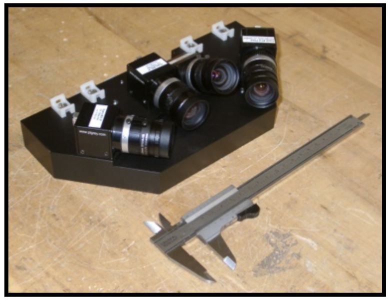

2D 3D | Road Aerial | Bird's eye Labels | Traffic
Virtual Earth
Image © 2008 OLIG
Image © 2008 Flotron / Perrinjaquet
Google
47°22'35.01" N 8°32'54.05" E
elev 469 m
May 10, 2008
Eye alt 505 m

e.g. interactive 3D modeling of architecture (Sinha et al. Siggraph Asia 08)

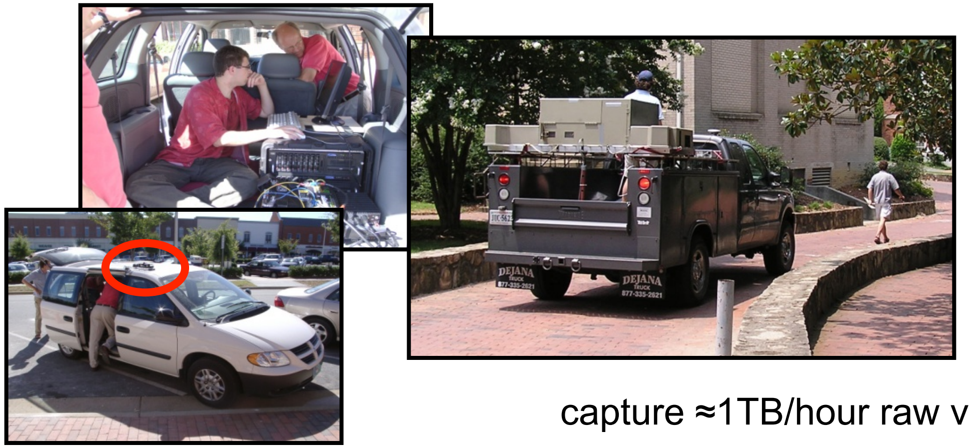


collaboration with Microsoft Research

Fast automated video-based modeling of cities

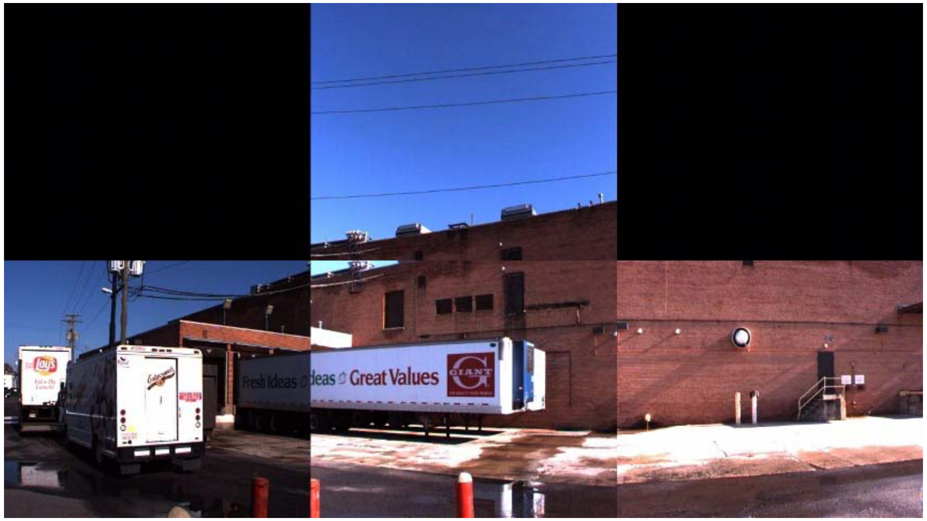


2x4 cameras
1024x768@30Hz



capture ≈1TB/hour raw video data

GPS/INS system



Fast video-based modeling of cities

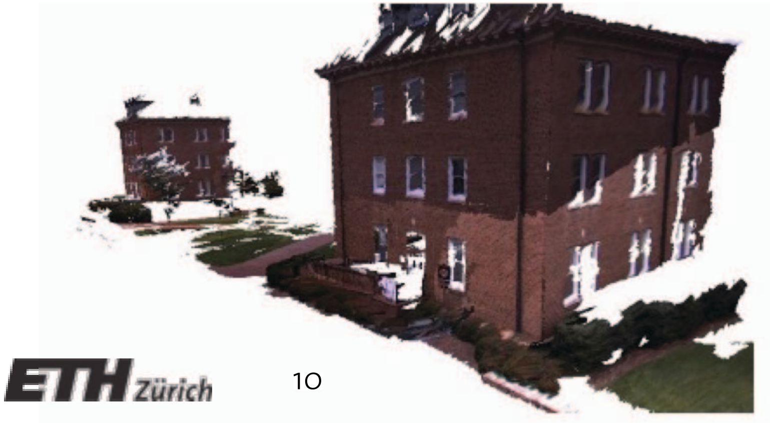
Fast video processing pipeline

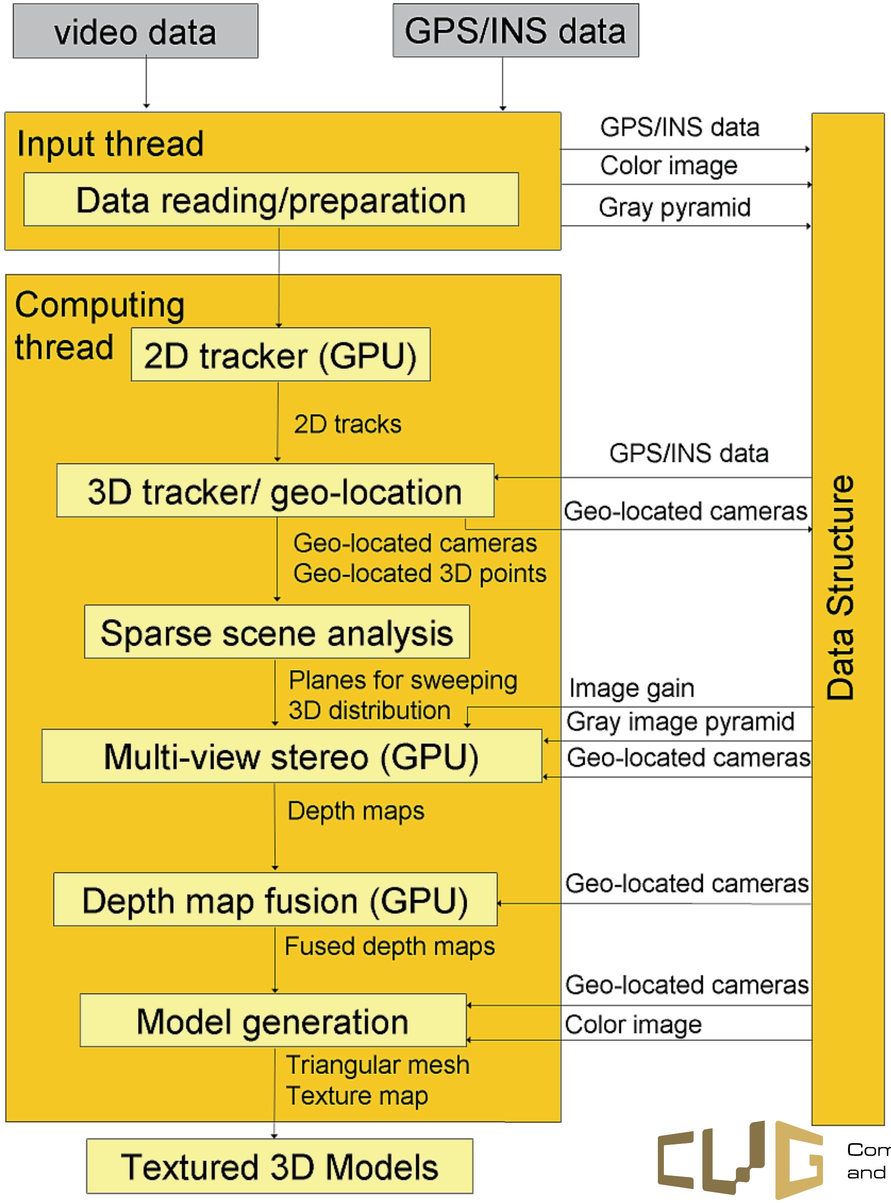
- up to 26Hz on single CPU/GPU

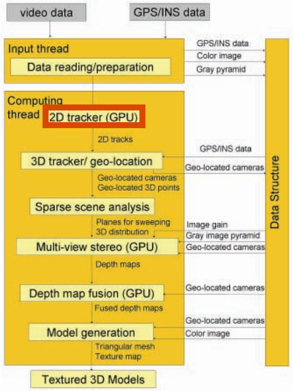
- Most image processing on GPU
  (x10-x100 faster)

- Exploits urban structure

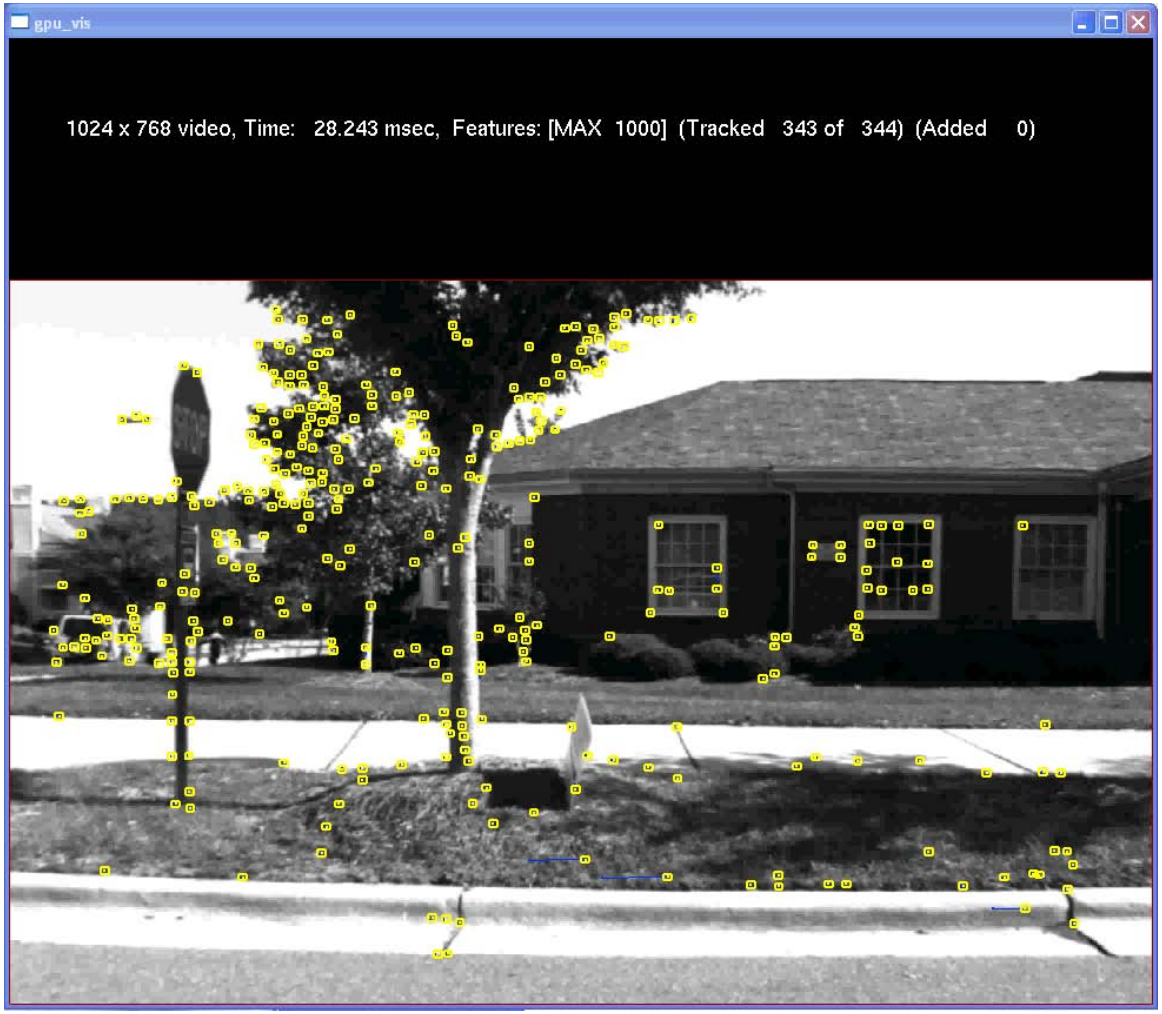
- Generates textured 3D mesh
(Pollefeys et al. IJCV, 2008)







2D Feature Tracker



1024 x 768 video, Time: 28.243 msec, Features: [MAX 1000] (Tracked 343 of 344) (Added 0)

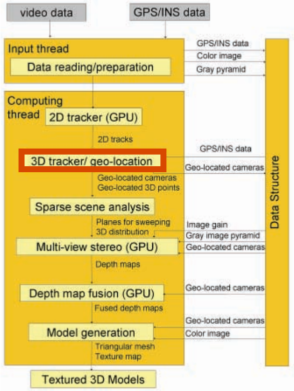
fast GPU-based feature tracking
(Sinha et al. MVA□07, Zach et al.08)

+ tracking of exposure changes
(Kim et al. ICCV07)

Graphics Processor Unit (GPU)
(e.g. 240 processing cores)

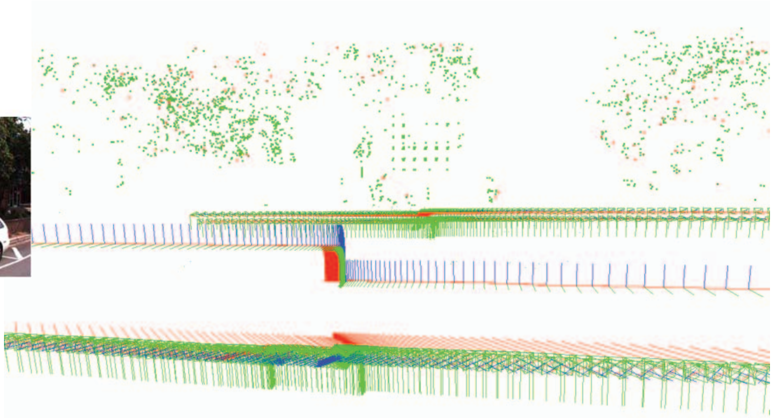
http://cs.unc.edu/~ssinha/Research/GPU_KLT/
http://cs.unc.edu/~cmzach/opensource.html

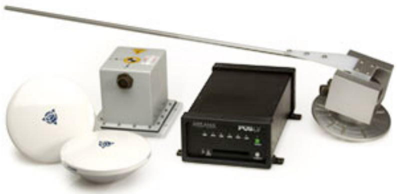
tracks 1000 features at 200Hz


video data
GPS/INS data
Input thread
Data reading/preparation
GPS/INS data
Color image
Gray pyramid
Computing thread
2D tracker (GPU)
2D tracks
3D tracker/ geo-location
GPS/INS data
Geo-located cameras
Geo-located cameras
Geo-located 3D points
Sparse scene analysis
Planes for sweeping
3D distribution
Image gain
Gray image pyramid
Geo-located cameras
Multi-view stereo (GPU)
Depth maps
Geo-located cameras
Depth map fusion (GPU)
Fused depth maps
Geo-located cameras
Color image
Model generation
Triangular mesh
Texture map
Textured 3D Models
Data Structure

3D Tracker / Geo-location

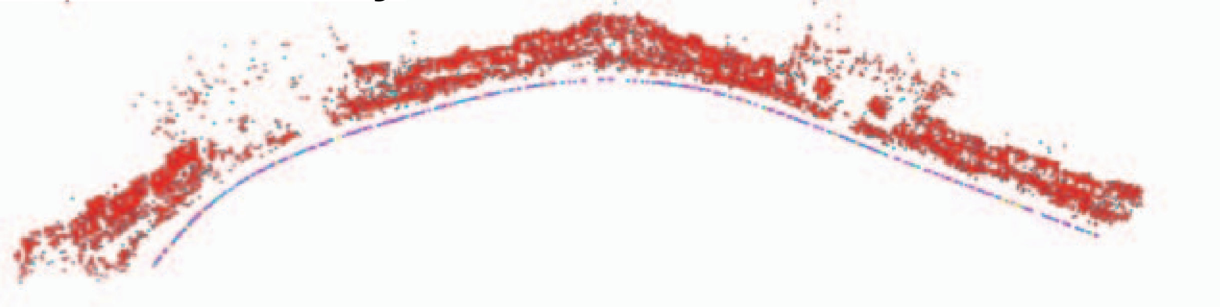
- Fusion of 2D video tracks and INS/GPS





Inertial Navigation System (INS)
Global Positioning System (GPS)

or use 2D video tracks only (need to deal with drift, see later)

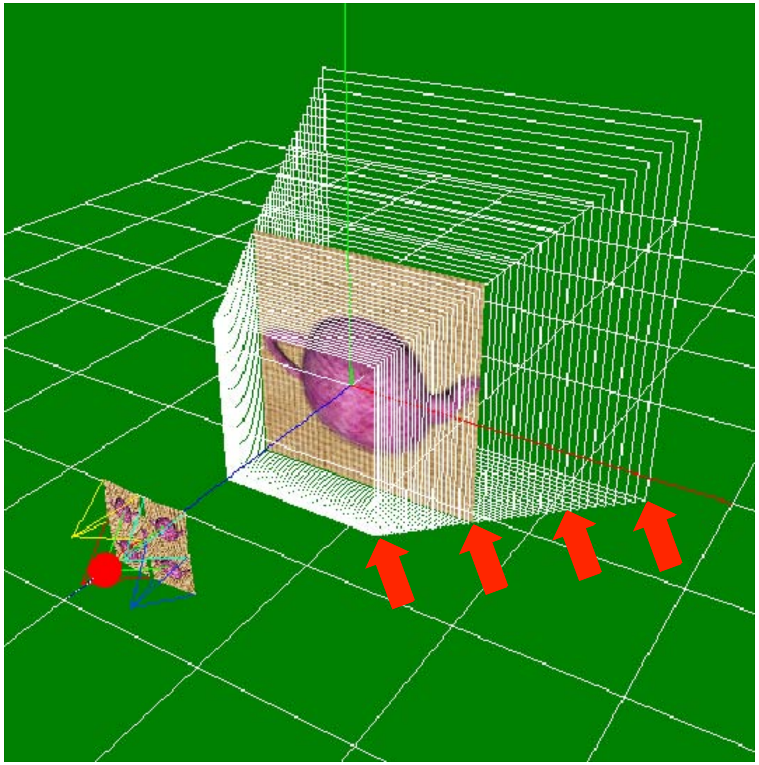


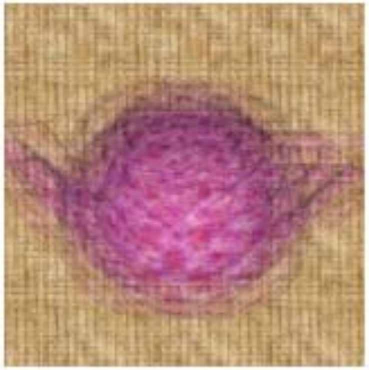
Interesting option to use vertical orientation (Fraundorfer et al. ECCV2010) or vehicle motion (Scaramuzza et al. ICCV2009) to facilitate motion estimation

Dense multi-view matching

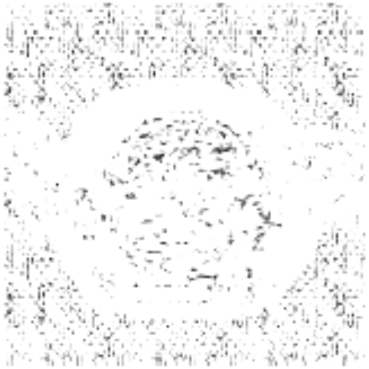
- Plane-sweep multi-view depth estimation on GPU

(Yang & Pollefeys, CVPR□03)

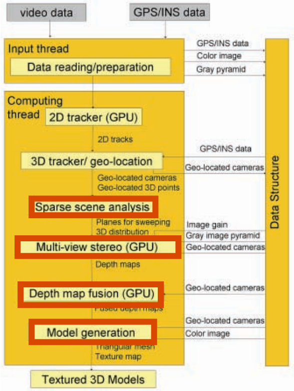




Blend:
$(I_0+I_1+I_2+I_3+I_4)/5$
(correct depth=in focus)

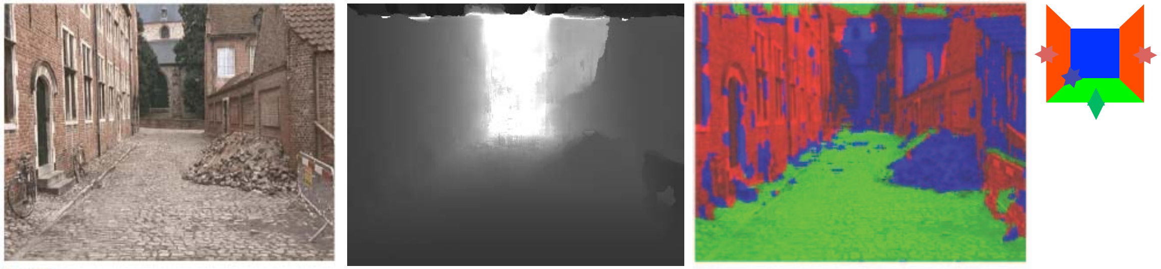


Sum of Absolute Differences:
$|I_1-I_0|+|I_2-I_0|+|I_3-I_0|+|I_4-I_0|$
(correct depth=small value
=dark)



Dense 3D surface reconstruction

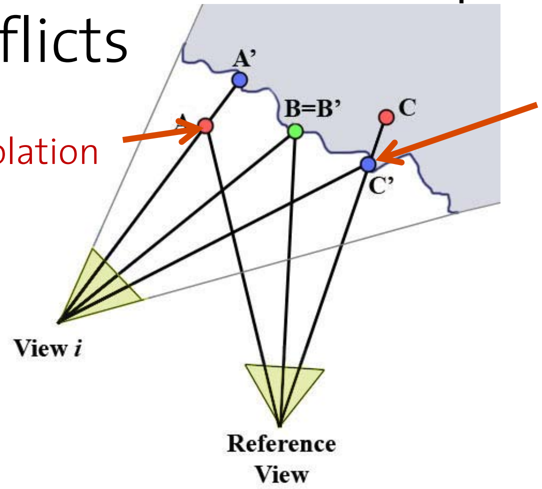
- Multi-Directional plane-sweeping stereo (Gallup et al., CVPR07)
  - Sweep along façade & ground-plane directions



choose best-cost solution over depth and orientation

3D model from 11 video frames (hand-held)

- Fuse depth-maps to obtain consensus depth map by minimizing visibility conflicts (Merrell et al., ICCV07)

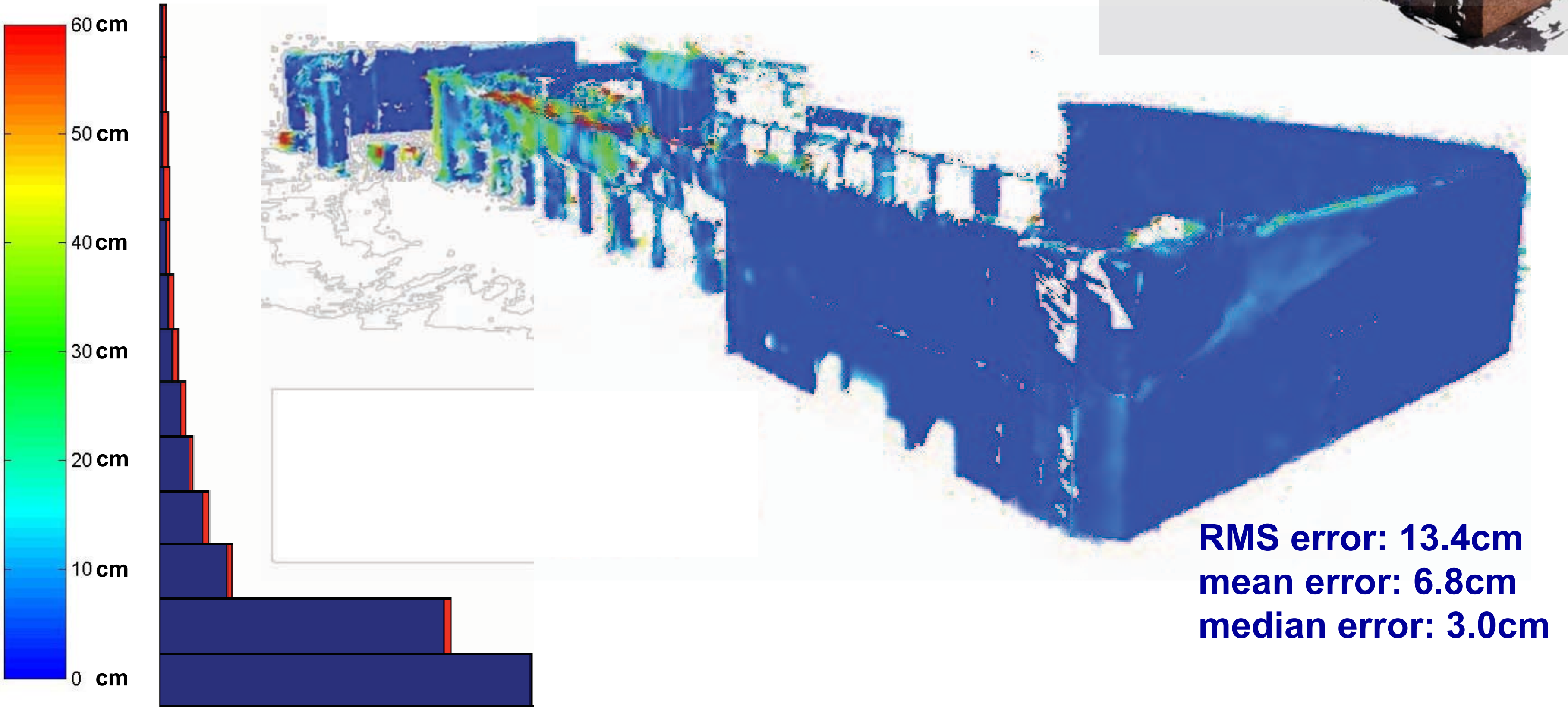


free-space violation

occlusion conflict

3D-from-video evaluation: Firestone building

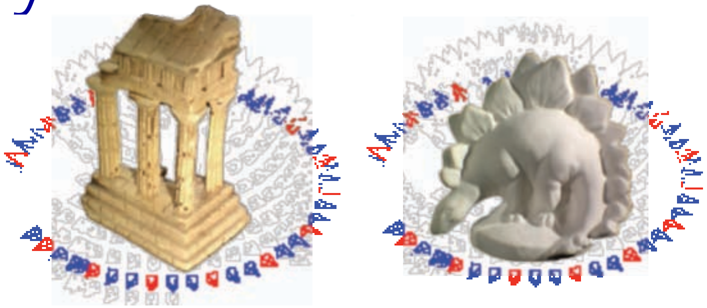
building surveyed to 6mm



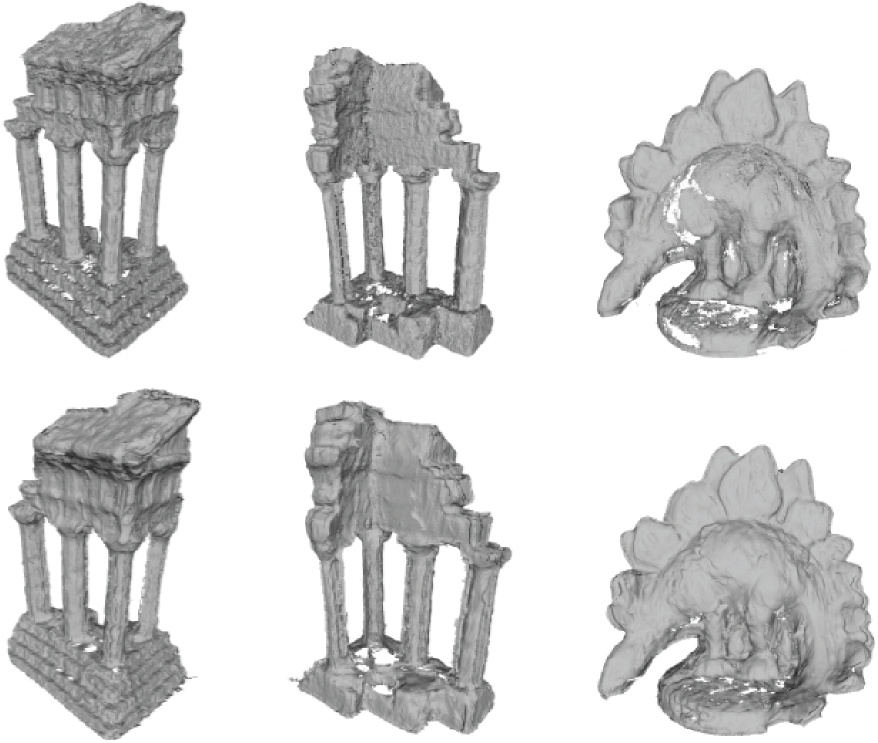
RMS error: 13.4cm
mean error: 6.8cm
median error: 3.0cm

error histogram

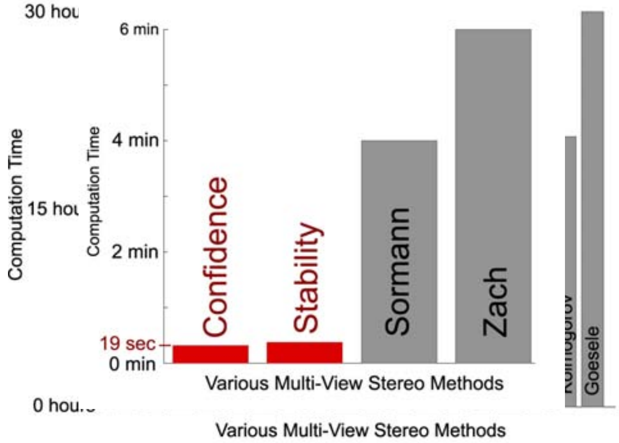
3D-from-video evaluation:
Middlebury Multi-View Stereo Evaluation Benchmark



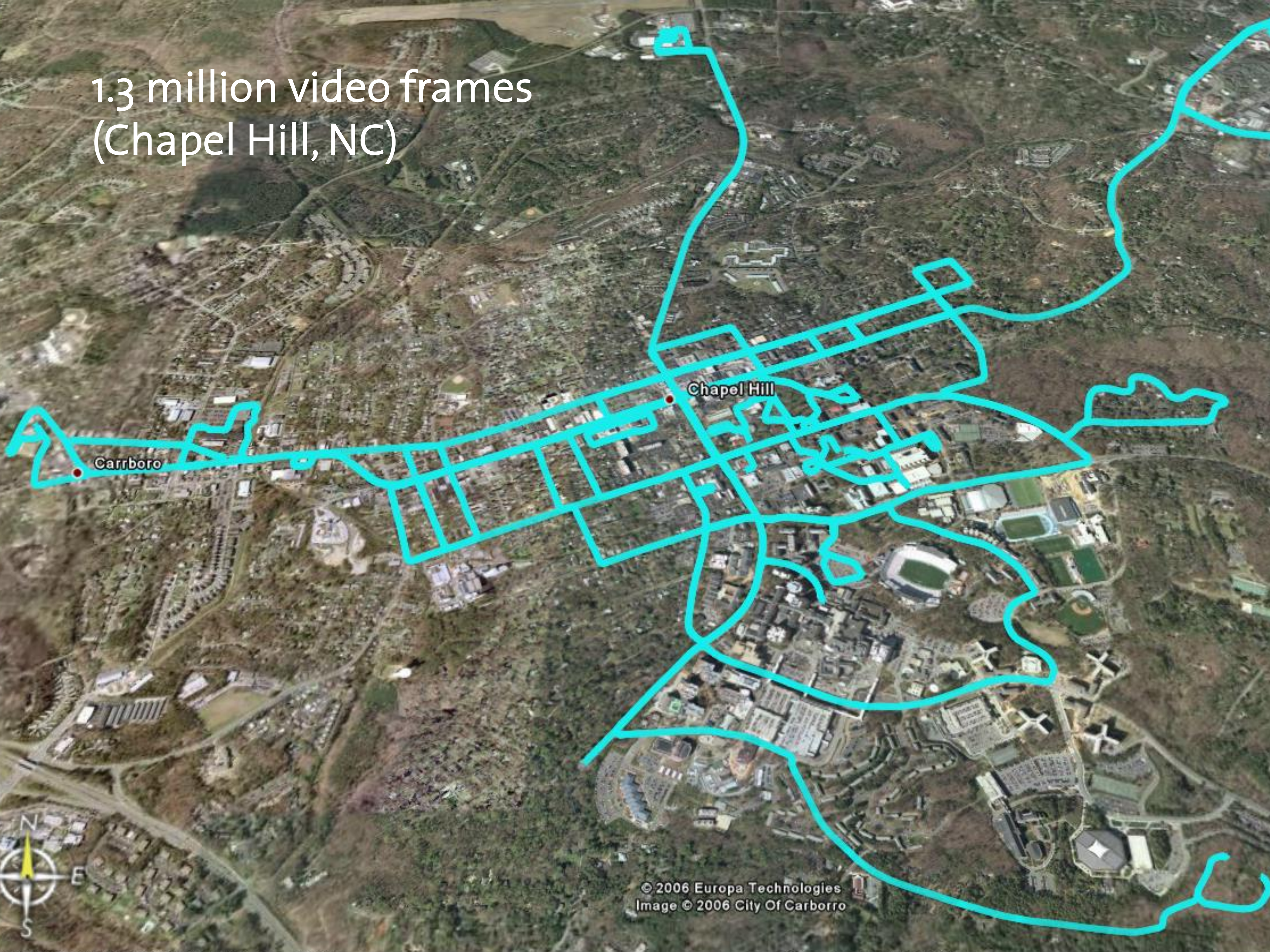
Ring datasets: 47 images



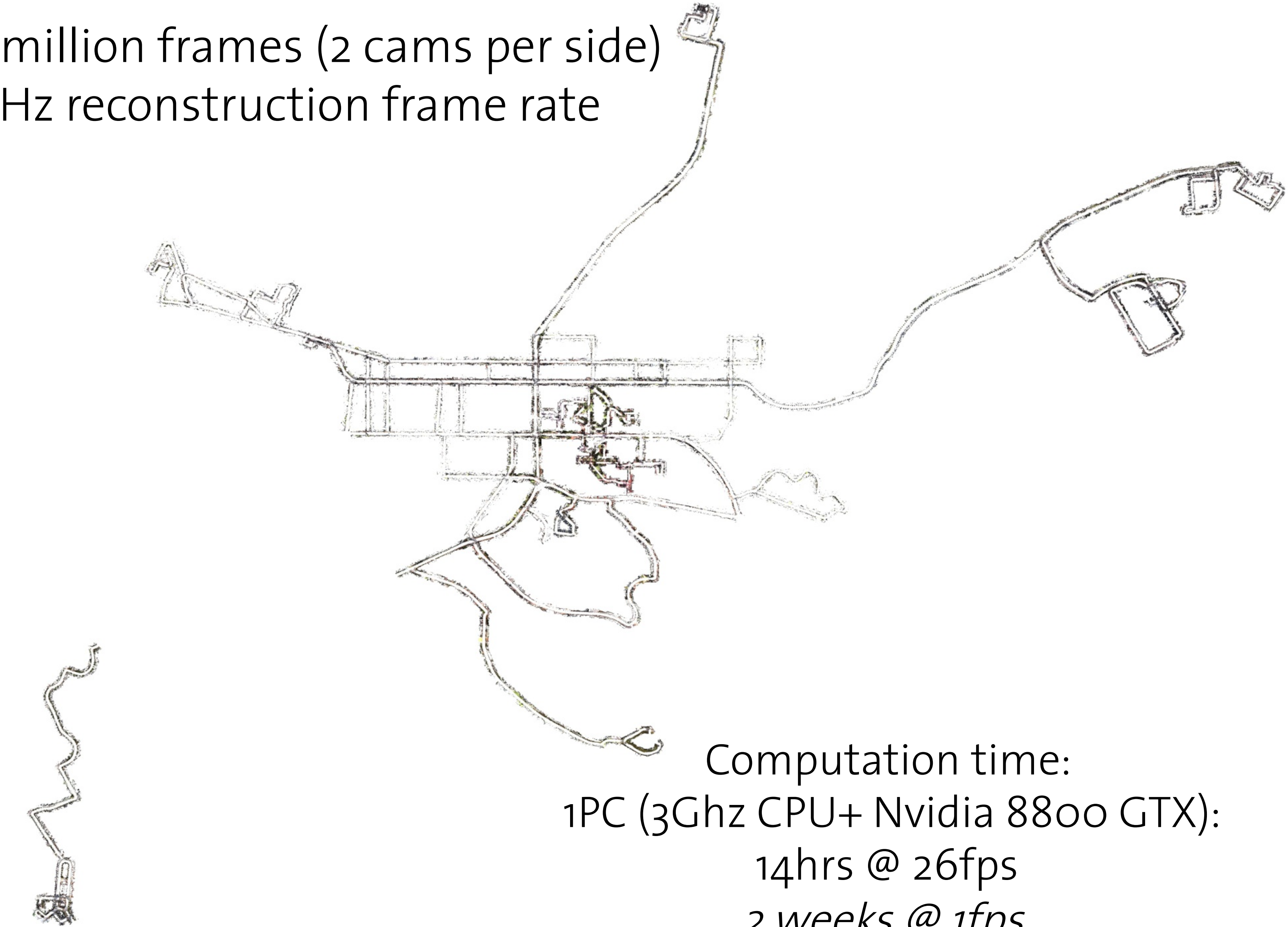
Results competitive
but much, much faster
(30 minutes → 30 seconds)



1.3 million video frames
(Chapel Hill, NC)

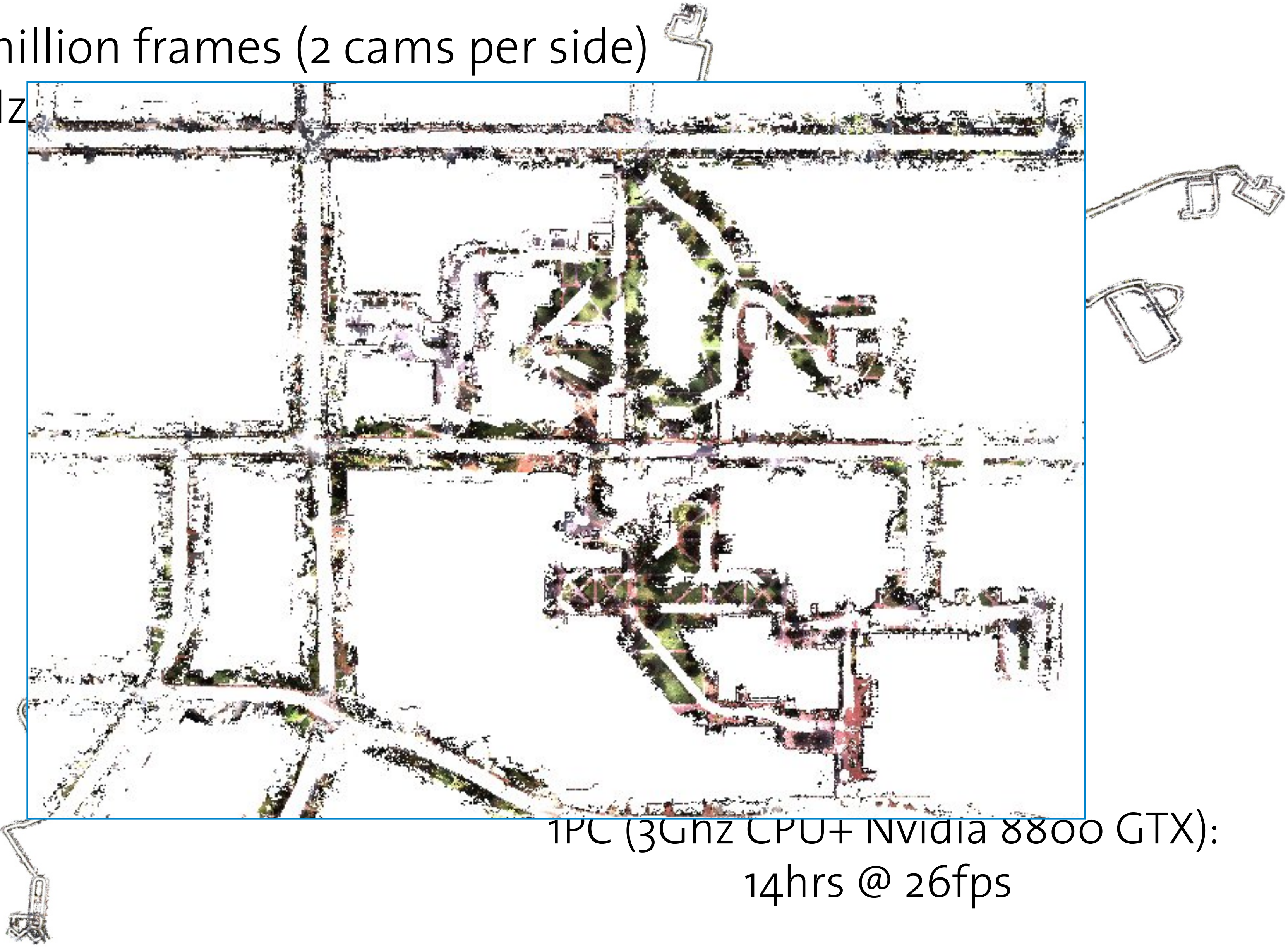


- 1.3 million frames (2 cams per side)
- 26 Hz reconstruction frame rate



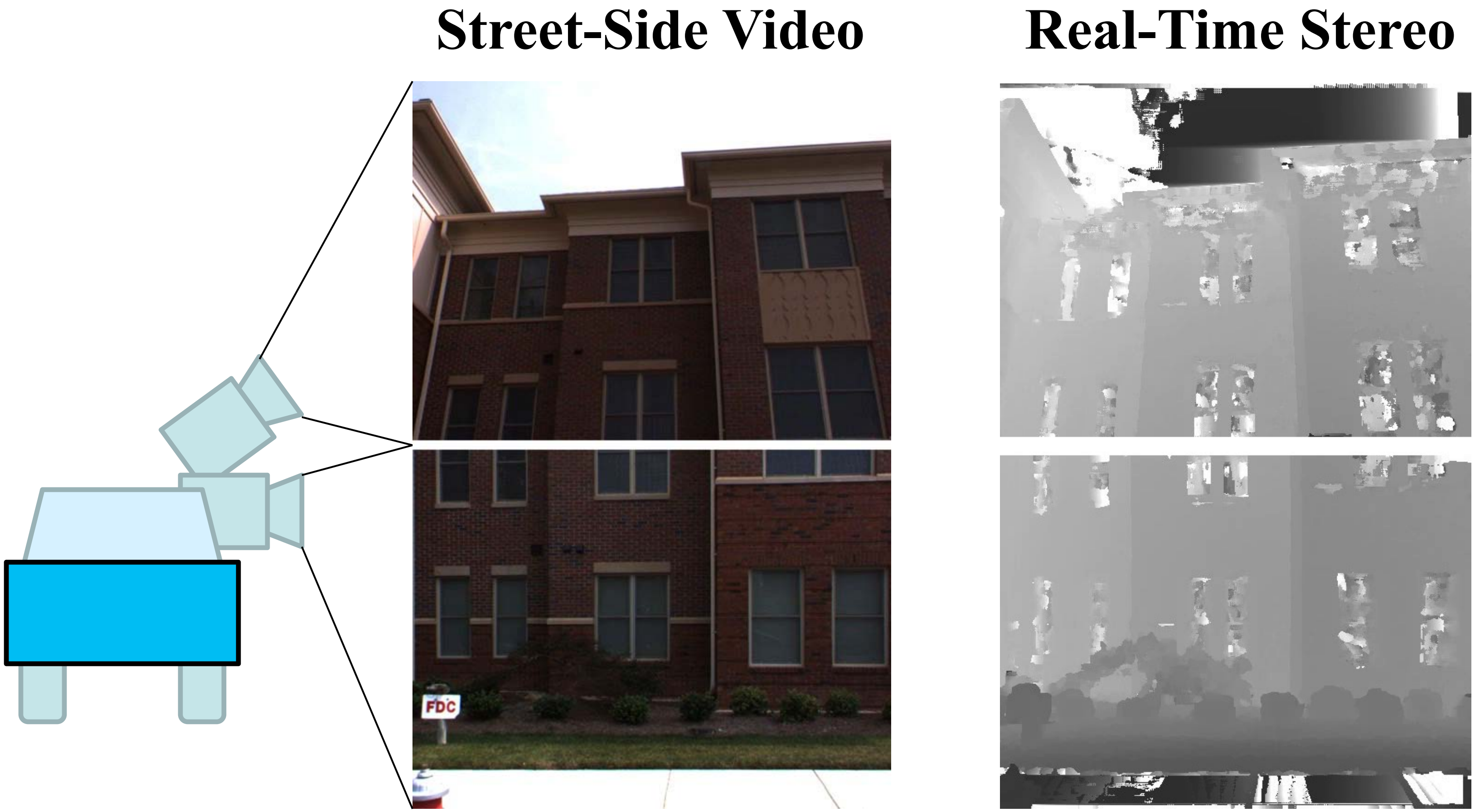
Computation time:
1PC (3Ghz CPU+ Nvidia 8800 GTX):
14hrs @ 26fps
2 weeks @ 1fps
2.5 years @ 1fpm

- 1.3 million frames (2 cams per side)
- 26 Hz



1PC (3Ghz CPU+ Nvidia 8800 GTX):
14hrs @ 26fps

Real-time stereo limitations


Street-Side Video

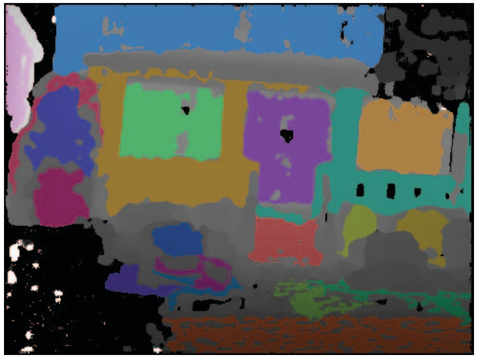
Real-Time Stereo

Notice problems at windows and homogeneous areas

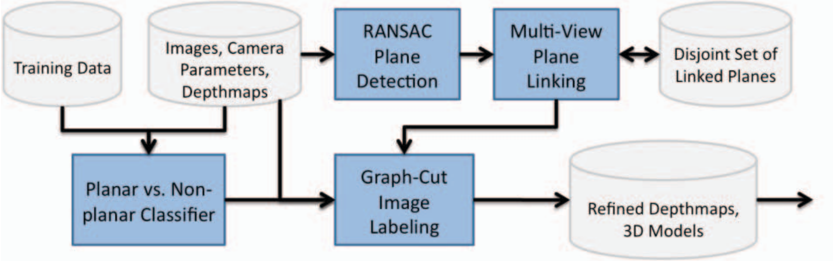
Including planar prior for urban scenes

(Gallup et al. CVPR10)

Video Frame

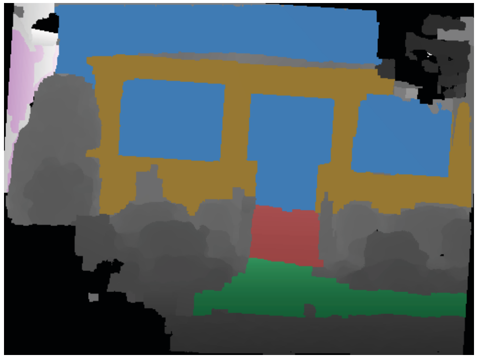


Depthmap with RANSAC planes

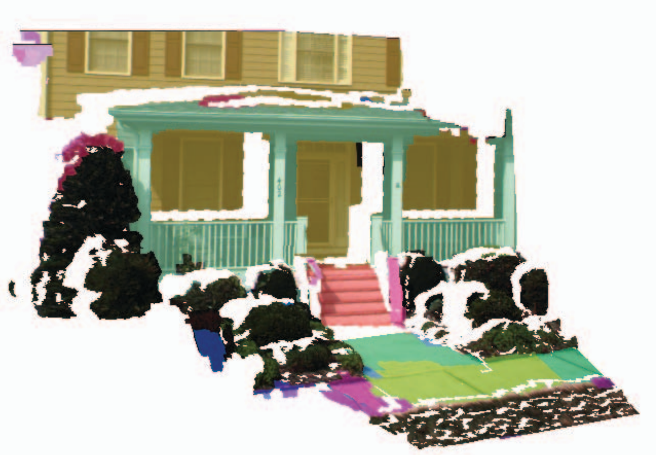

Training Data
Images, Camera Parameters, Depthmaps
RANSAC Plane Detection
Multi-View Plane Linking
Disjoint Set of Linked Planes
Planar vs. Non-planar Classifier
Graph-Cut Image Labeling
Refined Depthmaps, 3D Models

Flowchart

Planar Class Probability Map



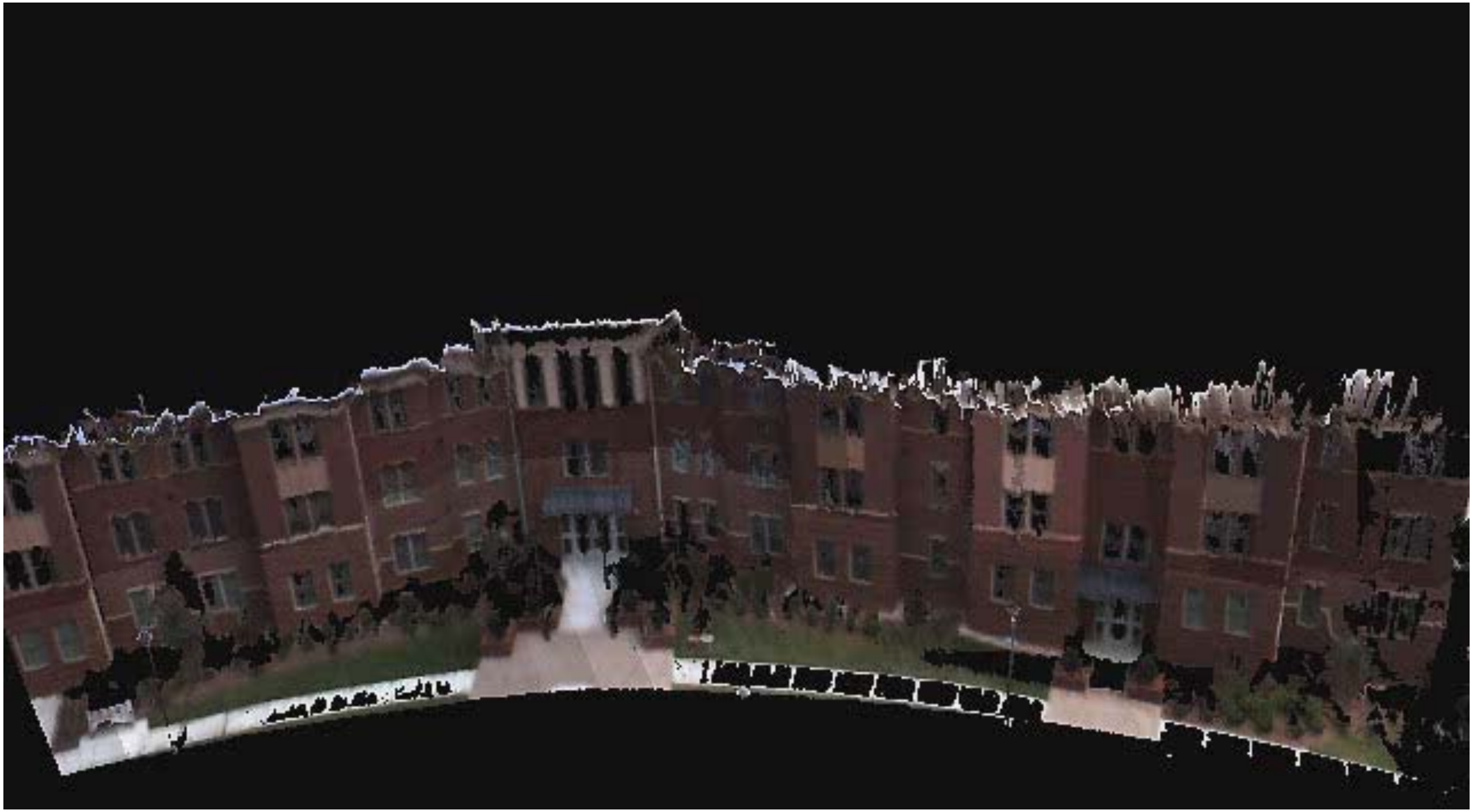
Graph-Cut Labeling



3D Model

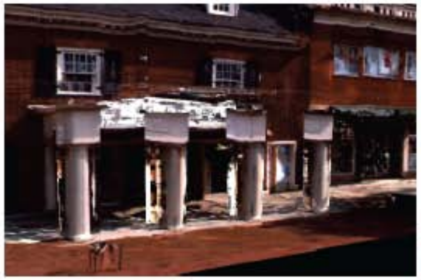
Including planar prior for urban scenes

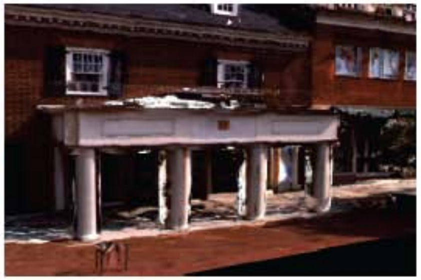
(Gallup et al. CVPR10)

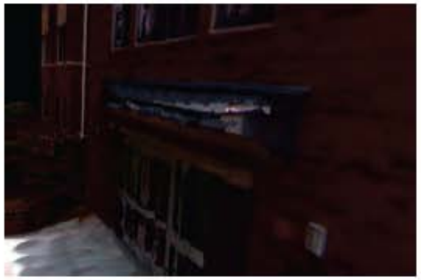


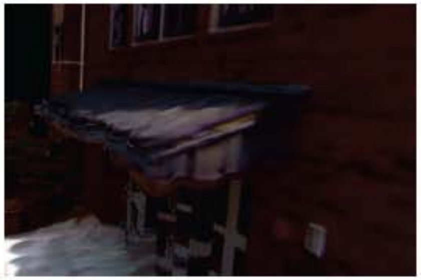
n-layer heightmap fusion

(Gallup et al. DAGM10)


1 Layer


3 Layer


1 Layer


3 Layer

From 2D StreetView to 3D models

(Gallup et al. DAGM10)

Real-time processing of video (30fps on PC, leveraging GPU)

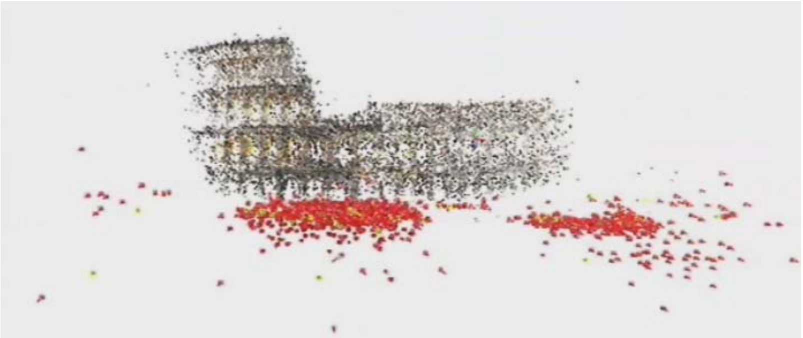
Building Rome on a cloudless day

(Frahm et al. ECCV 2010)

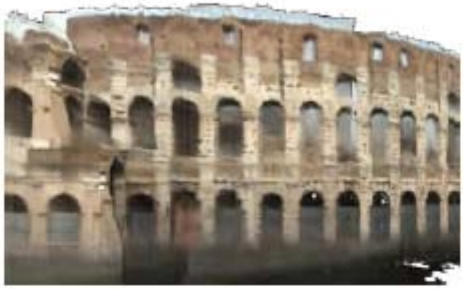
GIST & clustering (1h35)

SIFT & Geometric verification (11h36)

SfM & Bundle (8h35)



Dense Reconstruction (1h58)



Some numbers

- 1PC
- 2.88M images (650GB)
- 100k clusters (GIST: 4GB/176MB)
- 22k SfM with 307k images
- 63k 3D models
- Largest model 5700 images
- Total time 23h53

for comparison: Argawal'09 only 150k images/64PC/24h

Computer Vision and Geometry Lab

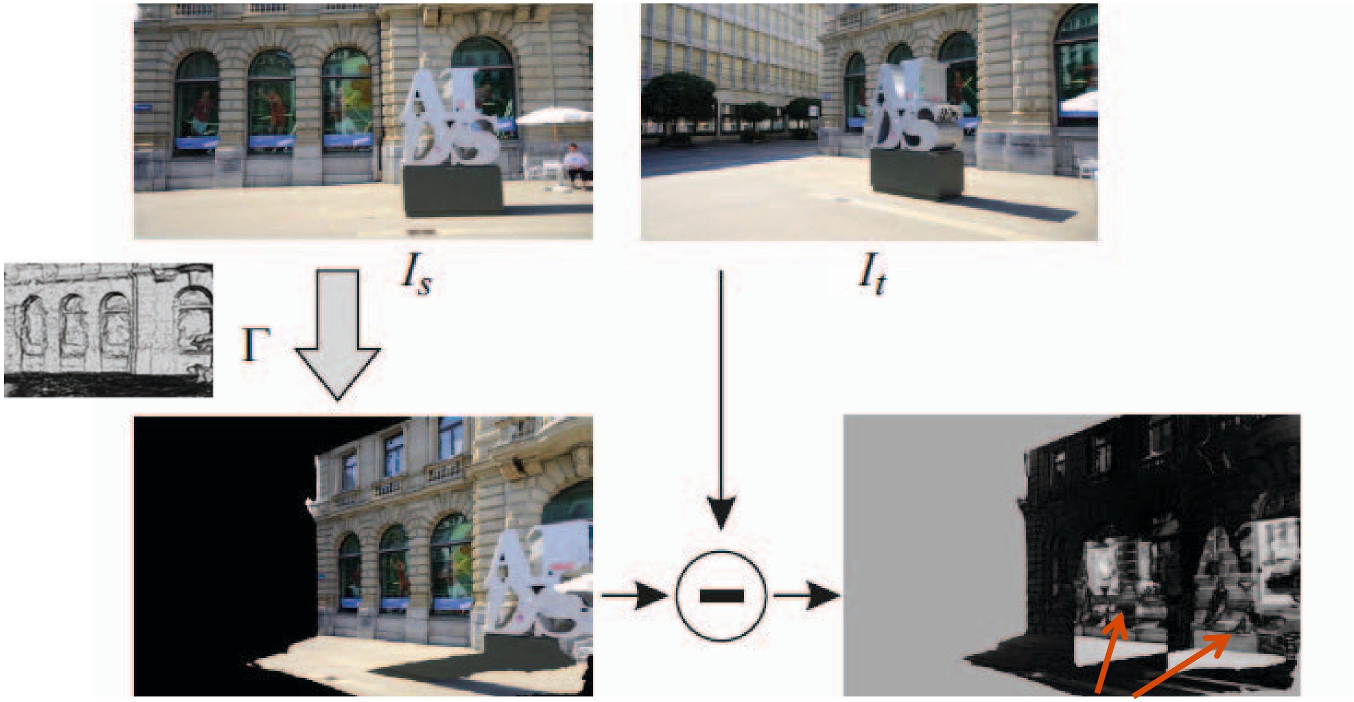
Building Rome on a cloudless day

(Frahm et al. ECCV 2010)

Appearance-invariant change detection

(Taneja et al. ICCV2011)

- Estimate pose between “old” model and “new” images
- Transfer and compare “new” images by warping according to “old” model



notice double shadows

Appearance-invariant change detection

(Taneja et al. ICCV2011)

“old” image

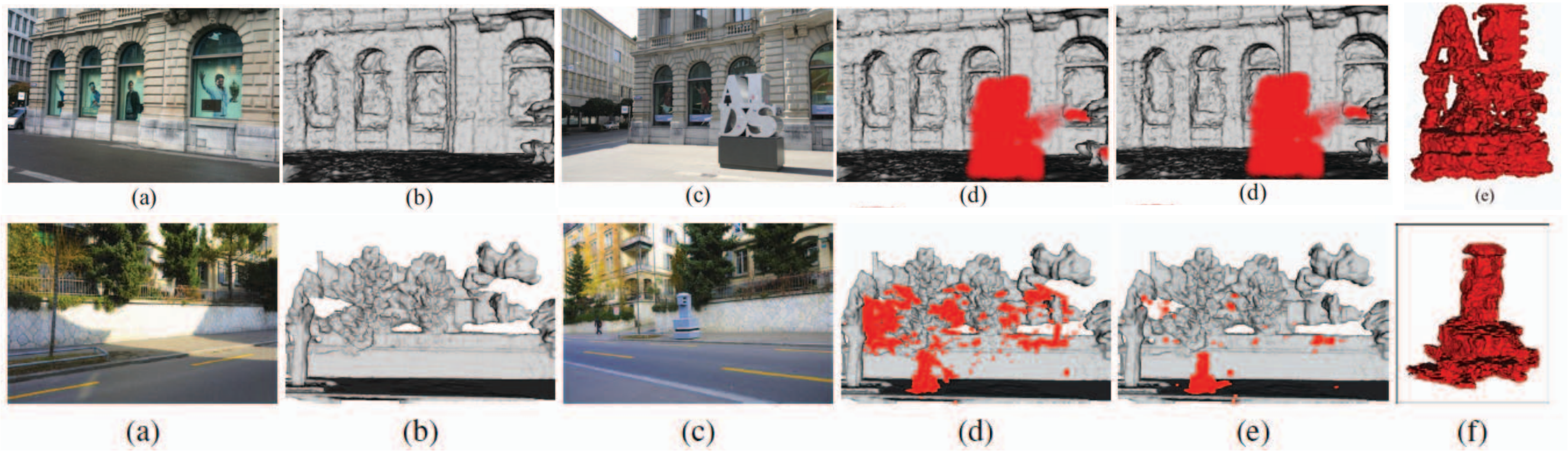
“old” model

“new” image

difference

salient Δ

Δ model



discard:

- people
- cars
- vegetation

probabilistic visual hull

Video-only large-scale reconstruction?

Challenge:

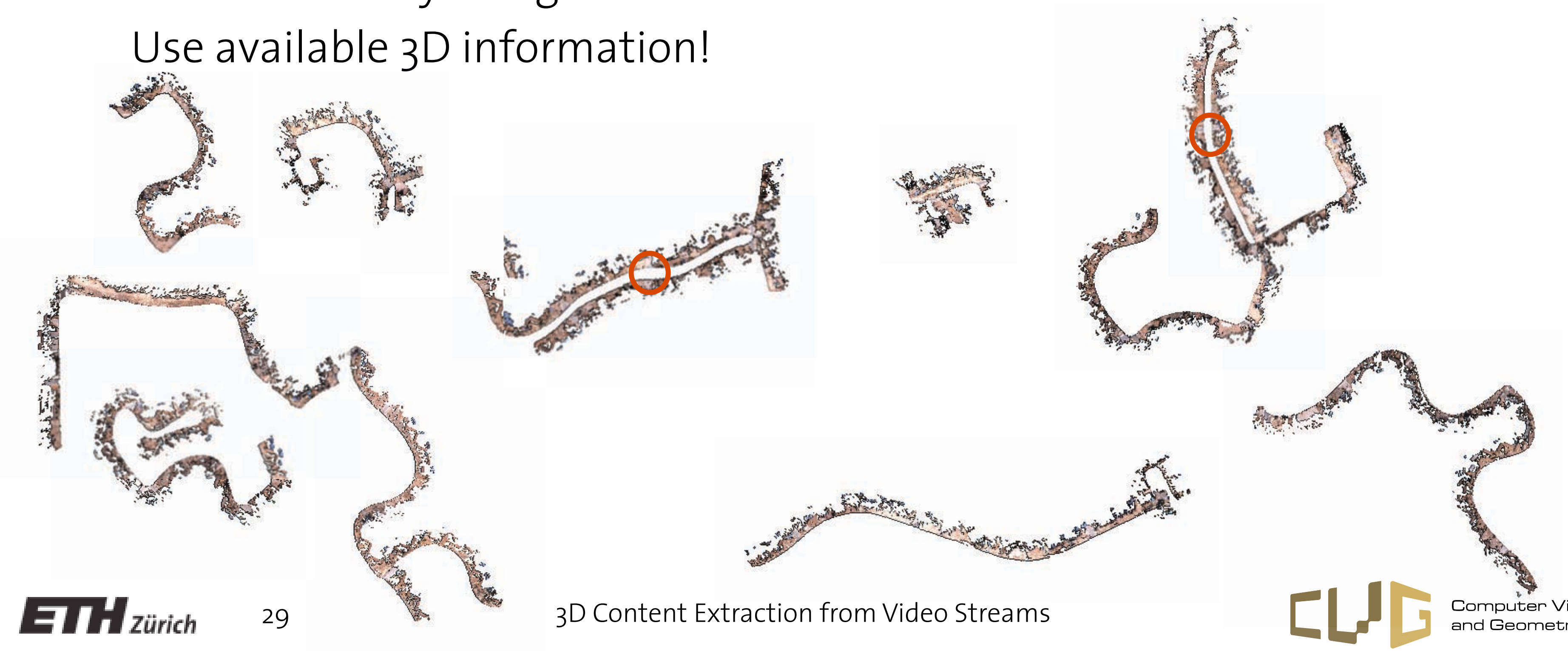
Error accumulation yields drift of relative scale, orientation and position

Solution:

Cancel drift by closing loops (e.g. at intersections)

Need to visually recognize locations

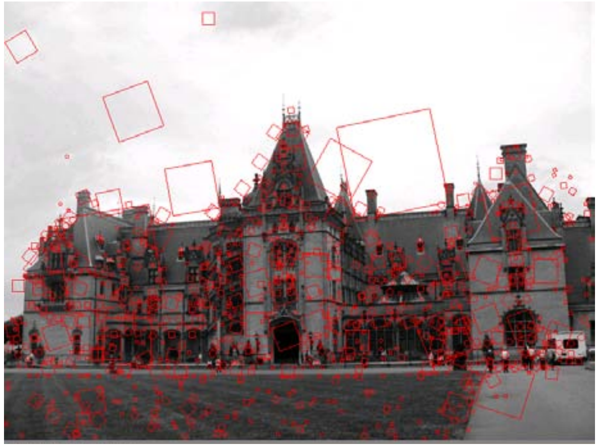
Use available 3D information!



Matching video segments/3D models

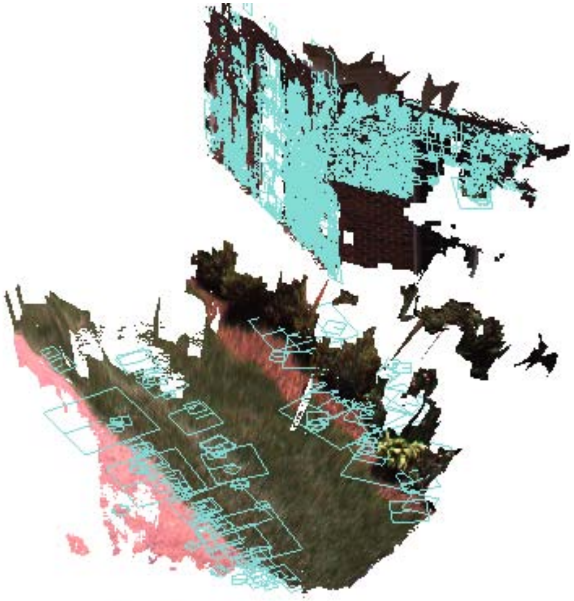
SIFT features

- Extracted from 2D images
- Variation due to viewpoint

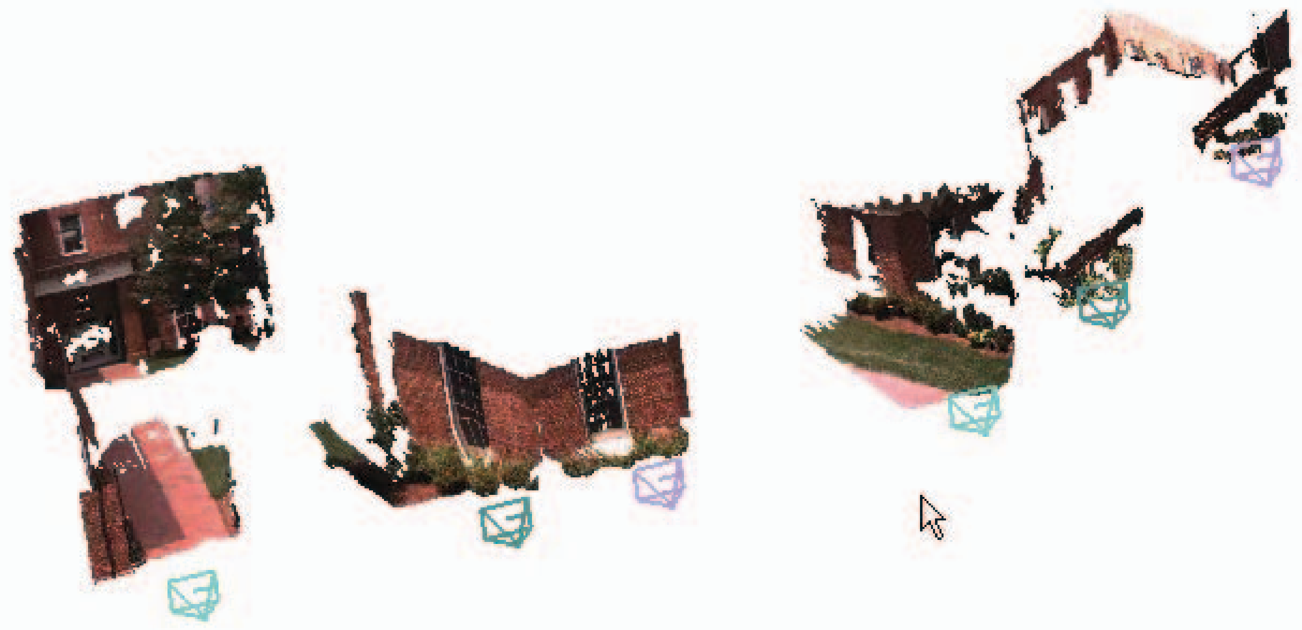


VIP features (Wu et al., CVPR08)

- Extracted from 3D model
- Viewpoint invariant



3D Models with VIPs



Geo-location from images

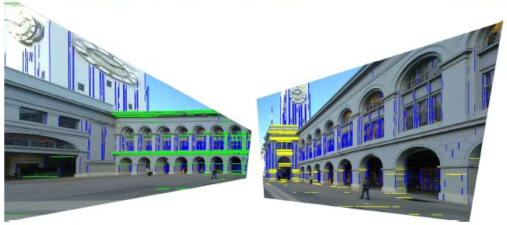
(Baatz et al., ECCV2010;
Chen et al. CVPR 2011)

Images + 3D Database

Building ortho-textures

descriptor database

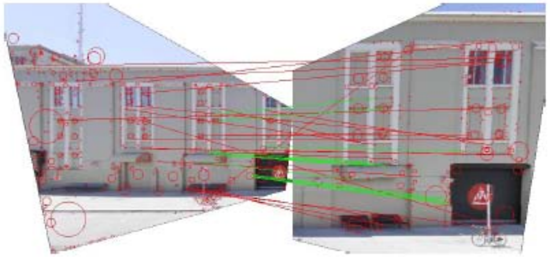
Rectification of query image

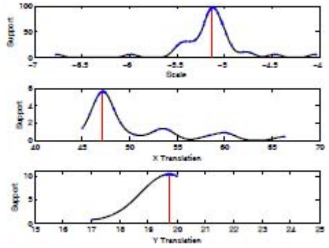


rectified features

promising candidates

Geometric verification





scale

x translation

y translation

Collaboration with NOKIA Connecting People

Minimal relative pose with know vertical

(Fraundorfer et al., ECCV2010)

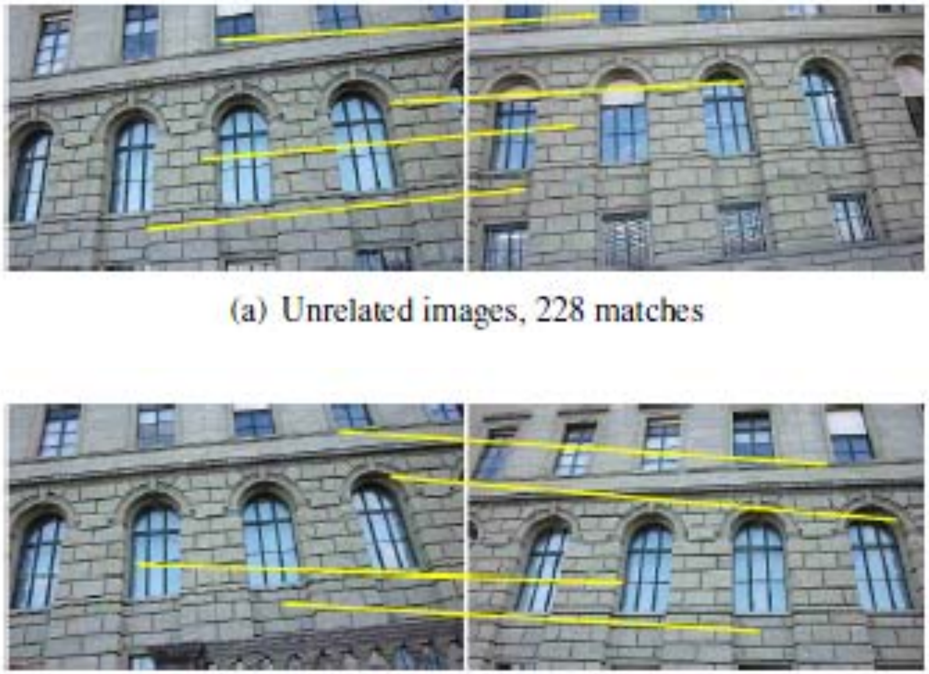
Vertical direction can often be estimated

- inertial sensor
- vanishing point

$$E = \begin{bmatrix} t_z \sin(y) & -t_z \cos(y) & t_y \\ t_z \cos(y) & t_z \sin(y) & -t_x \\ -t_y \cos(y) - t_x \sin(y) & t_x \cos(y) - t_y \sin(y) & 0 \end{bmatrix}$$

5 linear unknowns → linear 5 point algorithm
3 unknowns → quartic 3 point algorithm

Challenge: repetition ambiguity

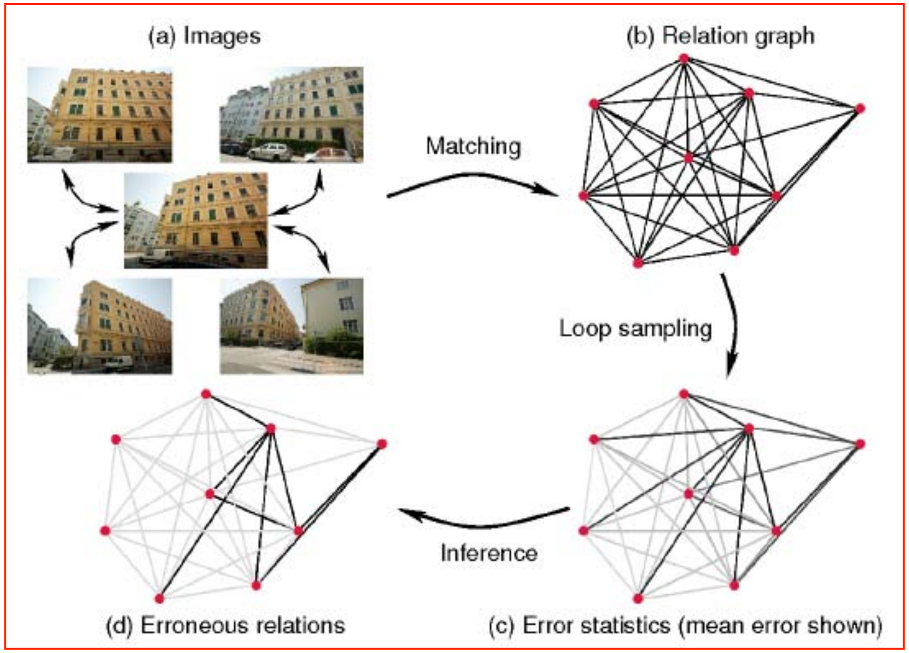


(a) Unrelated images, 228 matches

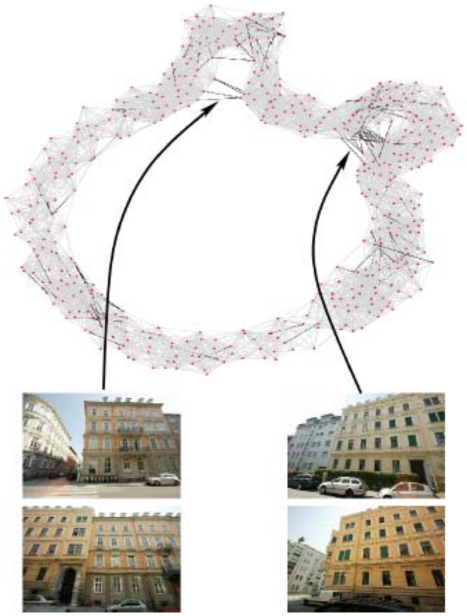
(b) Snapped to the wrong repetition, 331 matches

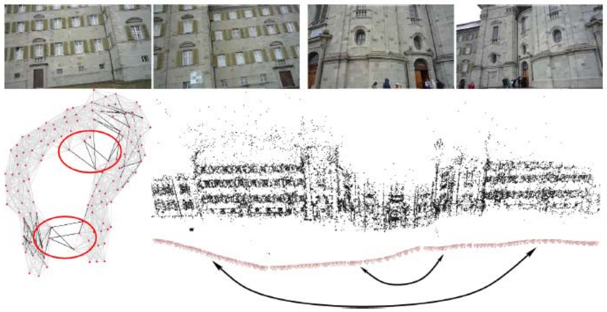
→ result in incorrect correspondences !

Disambiguating visual relations using loop constraints

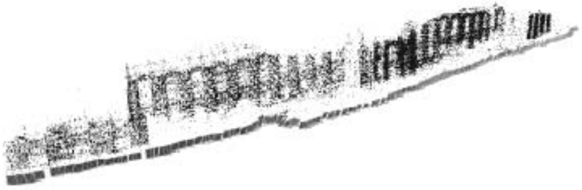


(Zach et al CVPR‘10)





(a) W/o edge filtering (143 views registered)

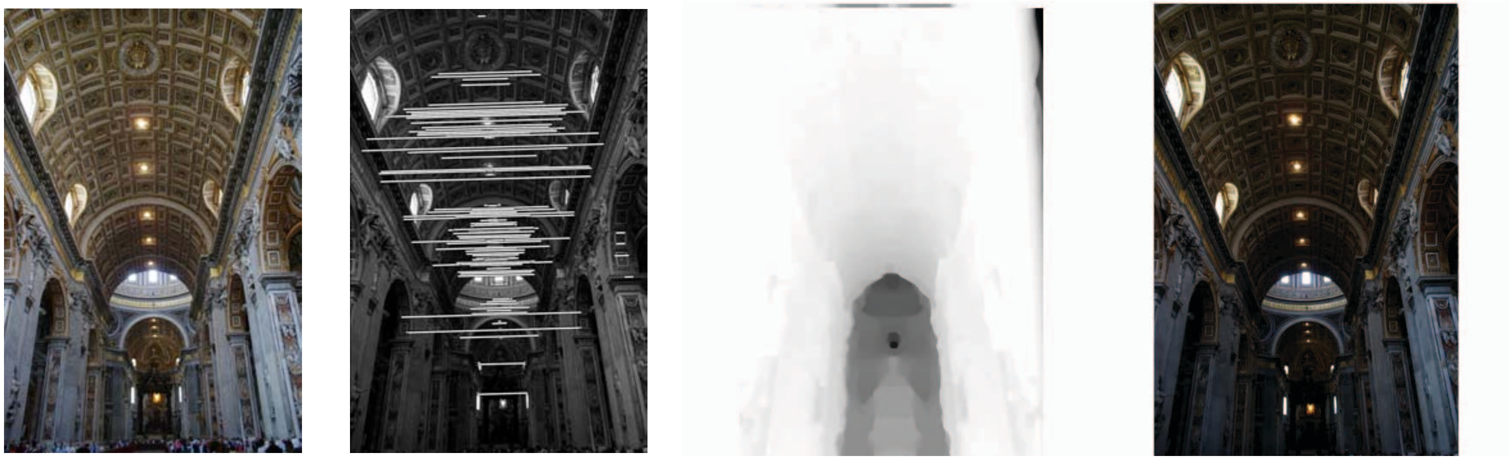


(b) With edge filtering (all 189 views registered)

Dense reconstruction from symmetry

(Koeser et al DAGM'11)
recipient DAGM main prize)

- Detect symmetry and perform dense matching

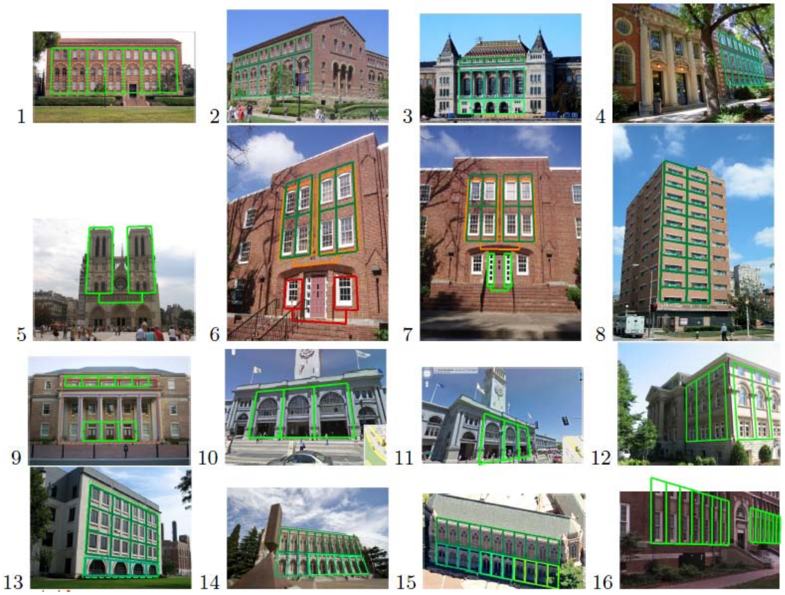


more examples:

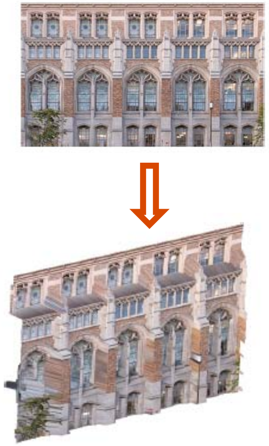
http://tinyurl.com/depthfromsymmetry

Towards Parsing Urban Scenes

- Detecting symmetries and repetitions (Wu et al ECCV'10)



- Applications:
  - Extracting architectural grammars
  - Matching repeating structures
  - Shape from symmetry and repetition (Wu et al CVPR11)



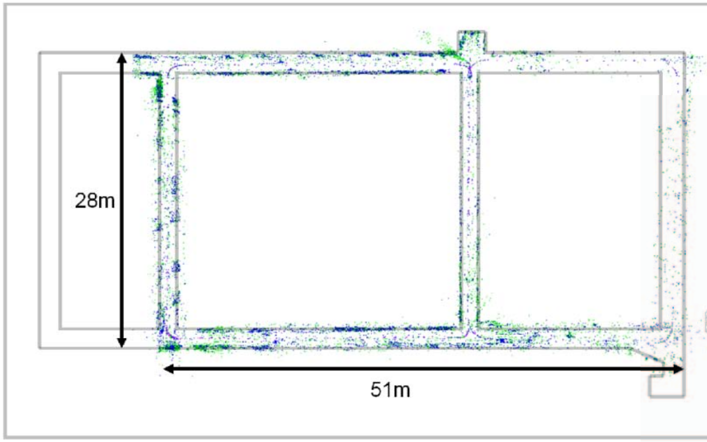
Real-Time Stereo Visual SLAM

(Clipp et al., IROS2010;
Lim et al., CVPR2011)

- Stereo KLT for local motion estimation
- SIFT for feature redetection and loop closure
- Local and global bundle adjustment

Parallel, Real–Time VSLAM

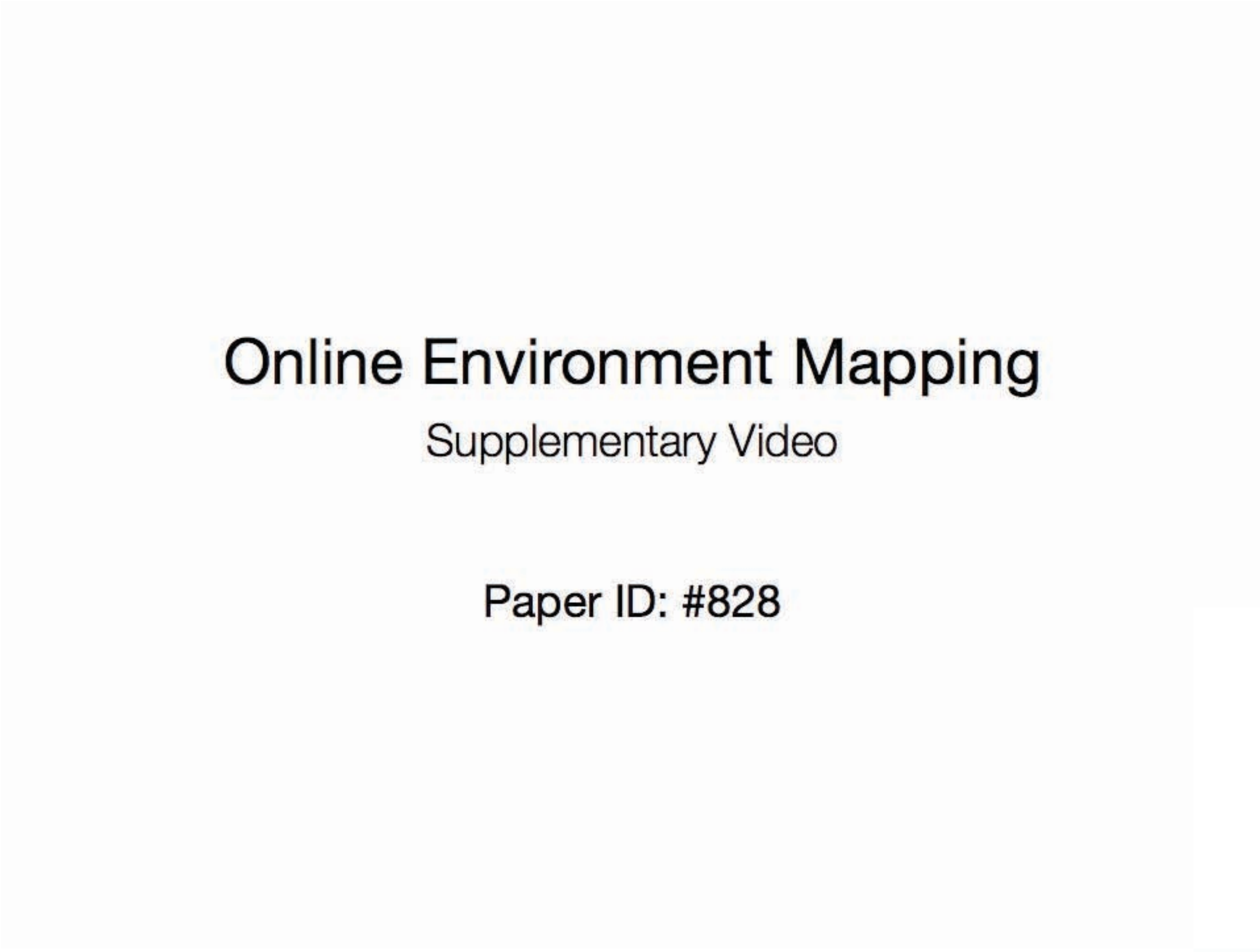
IROS 2010



Collaboration with HONDA The Power of Dreams

Real-Time Stereo Visual SLAM

(Lim et al., CVPR2011)



Online Environment Mapping

Supplementary Video

Paper ID: #828

Collaboration with HONDA The Power of Dreams

Computer Vision and Geometry Lab

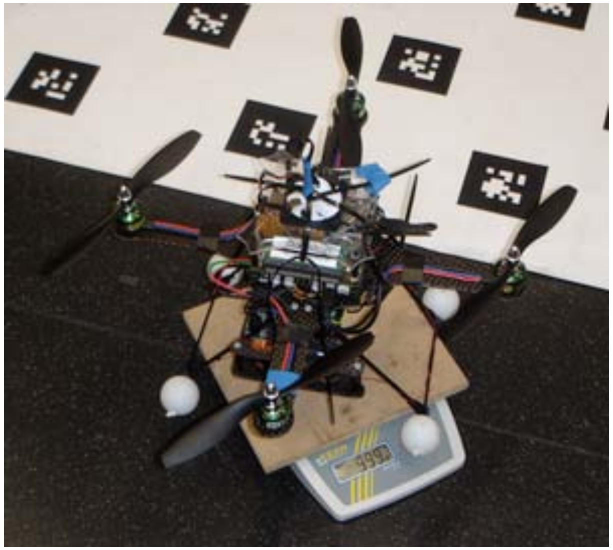
More applications of SLAM

OmniTour

(Saurer et al., 3DPVT2010)

Funded with Google™ward

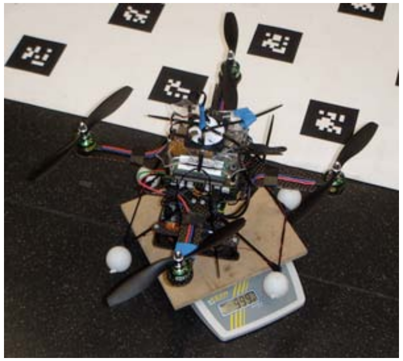
MAVs



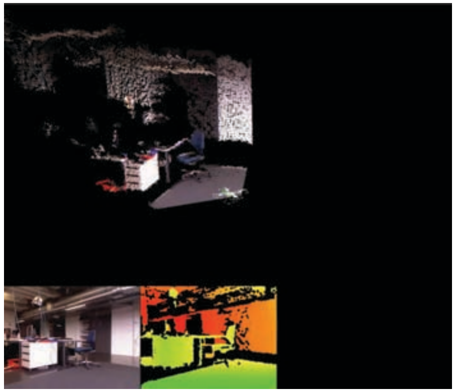
PixHawk student team
1st place **autonomy** EMAV09
(http://pixhawk.ethz.ch/)

Autonomous micro-helicopter navigation

(Meyer et al. ICRA11; Heng et al. ICRA11; Lee et al. ICRA11; Heng IROS11,...)



Student build MAV platform
developed for vision-based control



More on PixHawk: http://pixhawk.ethz.ch

Computer Vision
and Geometry Lab

OmniTour

(Saurer et al., 3DPVT2010)

Immersive tour building tool

- Omnidirectional video
- Approximate SfM
- Interactive map allignment

OmniTour

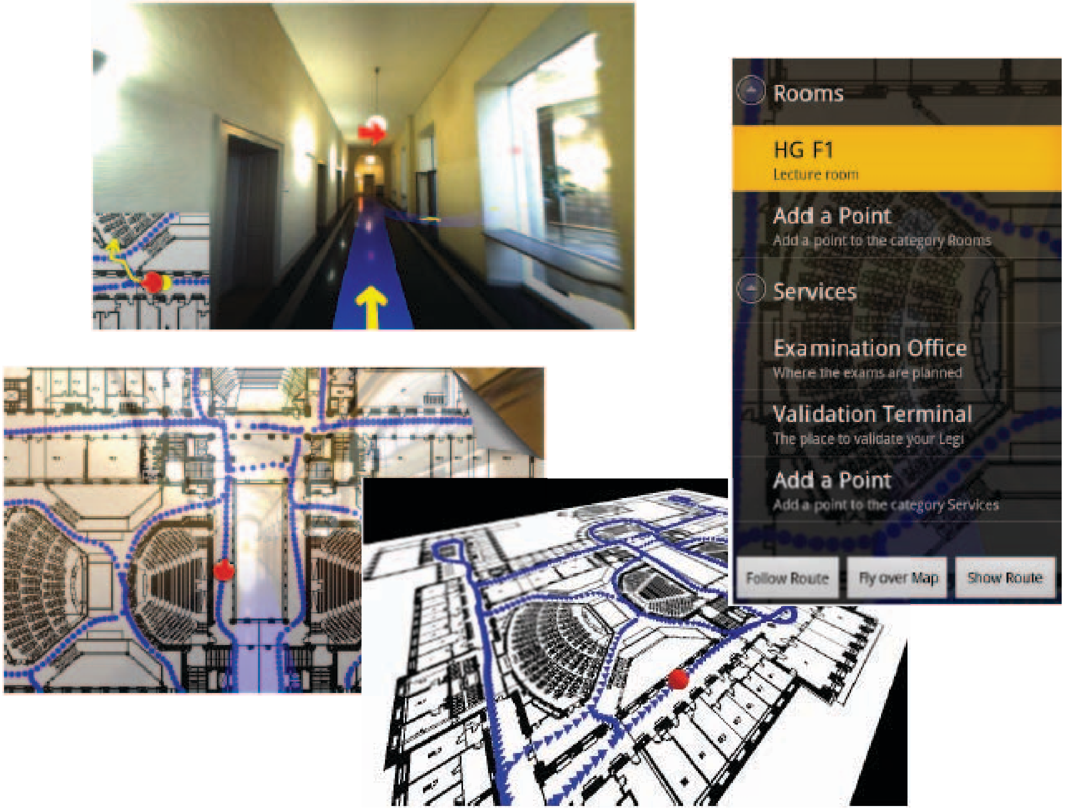
(Saurer et al., 3DPVT2010)

Authoring Tool

MobileTour

Schmid'11 (BS thesis)

Android application for exploration, navigation, editing POI



Rooms

HG F1
Lecture room

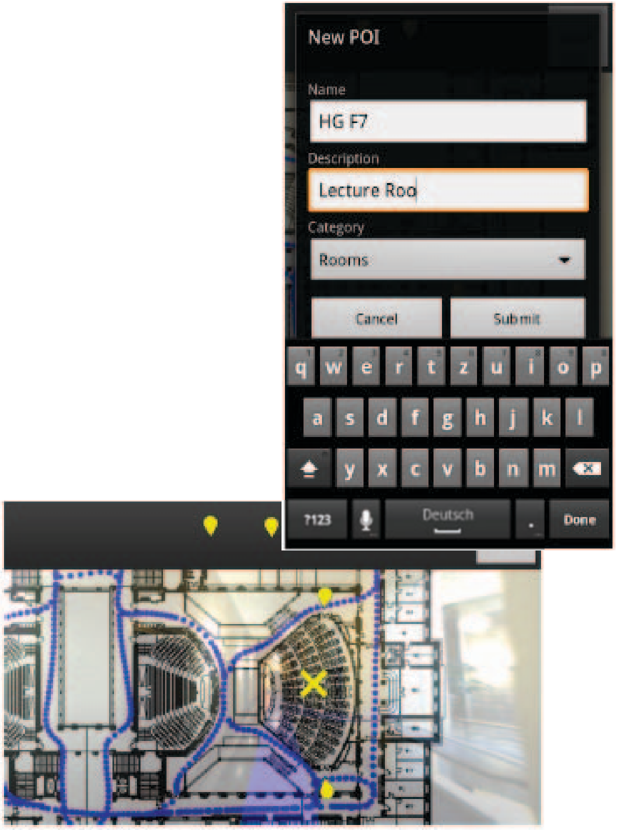
Add a Point
Add a point to the category Rooms

Services

Examination Office
Where the exams are planned

Validation Terminal
The place to validate your Legi

Add a Point
Add a point to the category Services



New POI

Name: HG F7

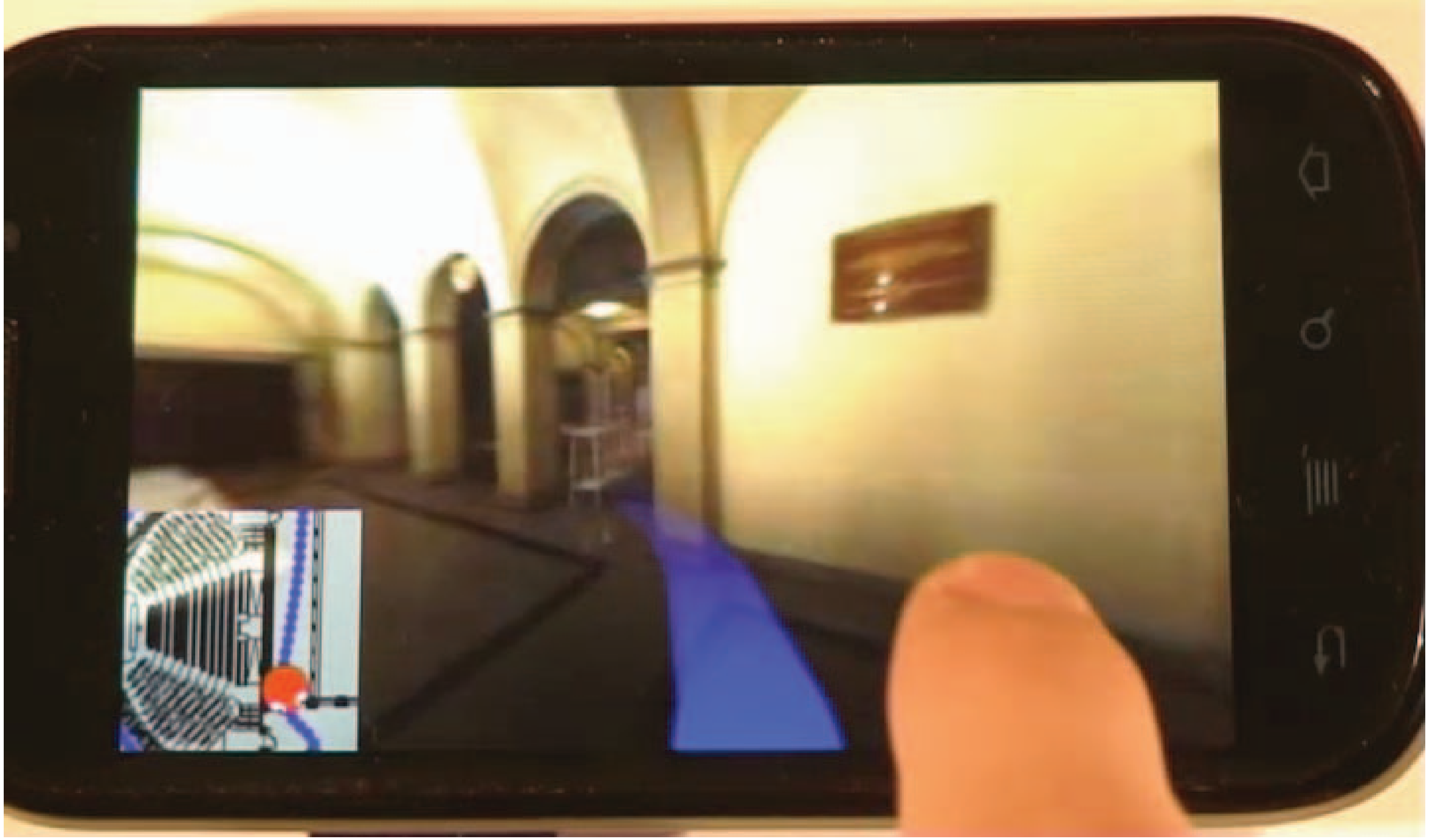
Description: Lecture Roo

Category: Rooms

Also work on indoor mobile localization   Waldin'11 (BS thesis)

MobileTour

Schmid'11 (BS thesis)



Also work on indoor mobile localization Waldin'11 (BS thesis)

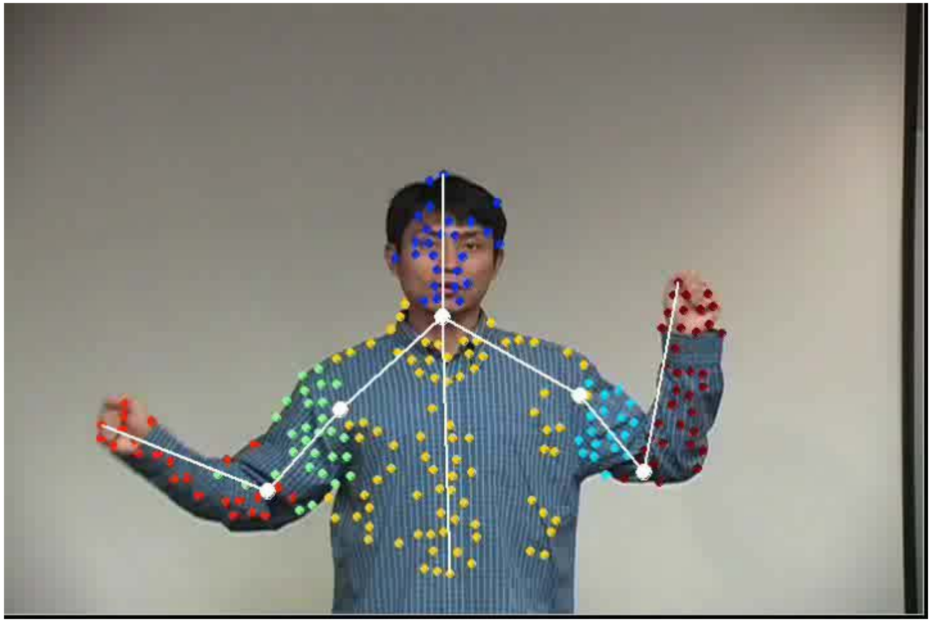
Talk outline

- Introduction
- Object modeling
- Scene modeling
- People/event modeling
- Summary and conclusion

Monocular Articulated Motion and Shape Recovery

(Yan & Pollefeys, CVPR05/ECCV06/CVPR06 & PAMI08)

- Feature tracks of articulated bodies span multiple intersecting 4D linear subspaces (under affine imaging conditions)
- Motion segmentation using local subspace affinity
  - Best in recent comparison (Tron & Vidal, CVPR07)
- Kinematic chain recovery
- Articulated 3D motion and shape recovery



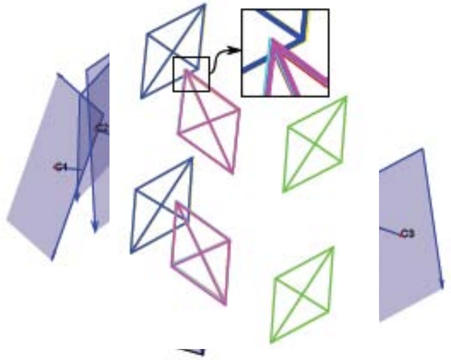
Multi-Camera Factorizations

(Angst & Pollefeys ICCV09/ECCV10)

- (Static) affine cameras
- Rigidly moving object
- Camera calibration using rigid motion
  - 2D feature point trajectories as input
  - No feature point correspondences between different camera views required

juxtapose x, y coordinates
for all points and cameras

$$\text{for all frames}\left\{\underbrace{\begin{bmatrix} x^1_{11} & y^1_{11} & x^1_{12} & y^1_{12} & x^1_{13} & y^1_{13} & \cdots \\ x^1_{21} & y^1_{21} & x^1_{22} & y^1_{22} & x^1_{23} & y^1_{23} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x^1_{m1} & y^1_{m1} & x^1_{m1} & y^1_{m1} & x^1_{m1} & y^1_{m1} & \cdots \end{bmatrix}}_{\text{rank} \leq 13}\right.$$



all tracks of all affine cameras form rank 13 subspace!
(for planar motion only rank 5)

Multi-Camera Factorizations

(Angst & Pollefeys ICCV09/ECCV10)

- Image coordinate
  - affine projection onto a camera axis (trilinear)

$$\mathbf{x}_{t,k,n} = \mathbf{C}_k \mathbf{M}_t \mathbf{S}_n$$

e.g. $$\left[\, u \,\right] = \underbrace{\left[\, r_{11} r_{12} r_{13} \;\middle|\; t_X \,\right]}_{\text{camera pose}} \underbrace{\begin{bmatrix} \mathbf{R} & \mathrm{t} \\ 0_3 & 1 \end{bmatrix}}_{\text{object motion}} \underbrace{\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}}_{\text{object shape}}$$

  - Stack observations in matrix

$$\begin{aligned} \mathbf{W} &= [\Downarrow_t \Rightarrow_k \Rightarrow_n \mathbf{x}_{t,k,n}] = [\Downarrow_t \Rightarrow_{n,k} (\mathrm{vec}(\mathbf{M}_t))^T (\mathbf{S}_n \otimes \mathbf{C}_k^T)] \\ &= \underbrace{[\Downarrow_t (\mathrm{vec}(\mathbf{M}_t)^T]}_{T \times 16} \underbrace{[[\Rightarrow_n \mathbf{S}_n] \otimes [\Rightarrow_k \mathbf{C}_k^T]]}_{16 \times 2KN} = \mathbf{AB} \end{aligned}$$

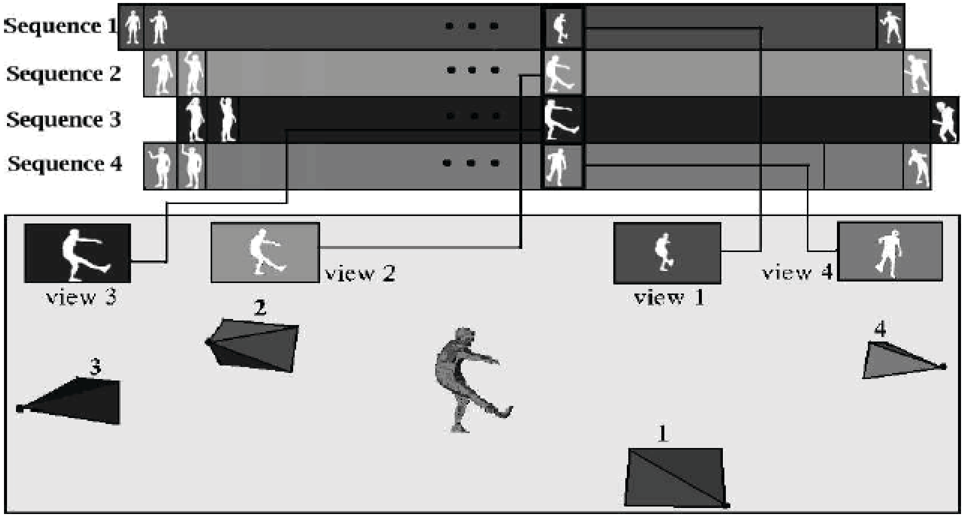
general motion: rank 13
planar motion: rank 5

$$T_{planar} = \begin{vmatrix} \cos\theta & \sin\theta & 0 & a \\ -\sin\theta & \cos\theta & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Camera network calibration from silhouettes

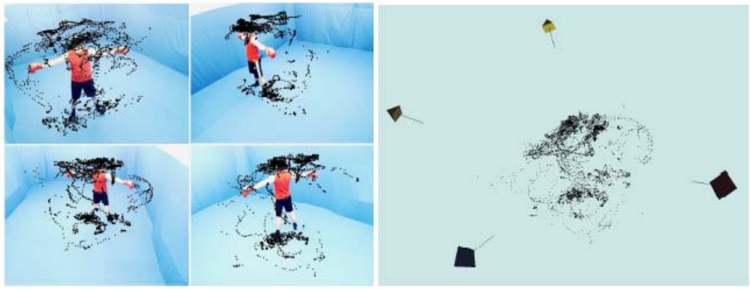
(Sinha et al., CVPR04; Sinha and Pollefeys ICPR04/IJCV10)

4 minutes of video from 4 camcorders (recorded at MIT)



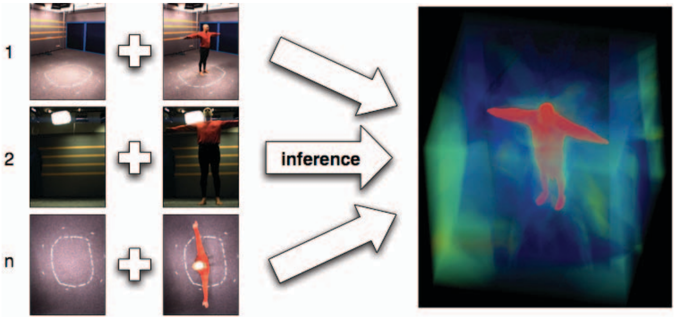
calibrate –and synchronize– camera network without requiring specific calibration data

Our approach is robust and efficient



http://cs.unc.edu/~ssinha/Research/silcalib/

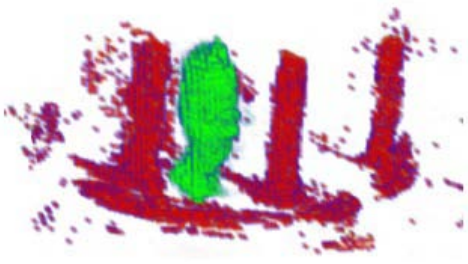
Probabilistic occupancy from silhouettes



(Franco and Boyer, ICCV05)

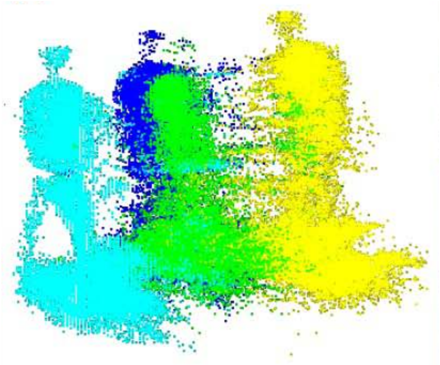
Occluder modeling

(Guan et al. CVPR07)



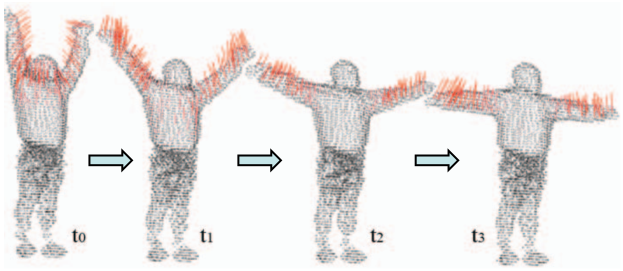
multi-person

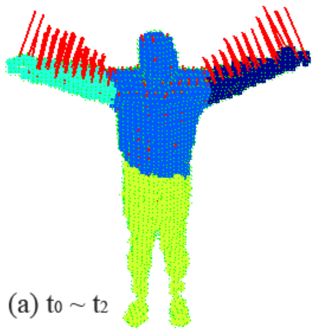
(Guan et al. CVPR08)



Occupancy flow

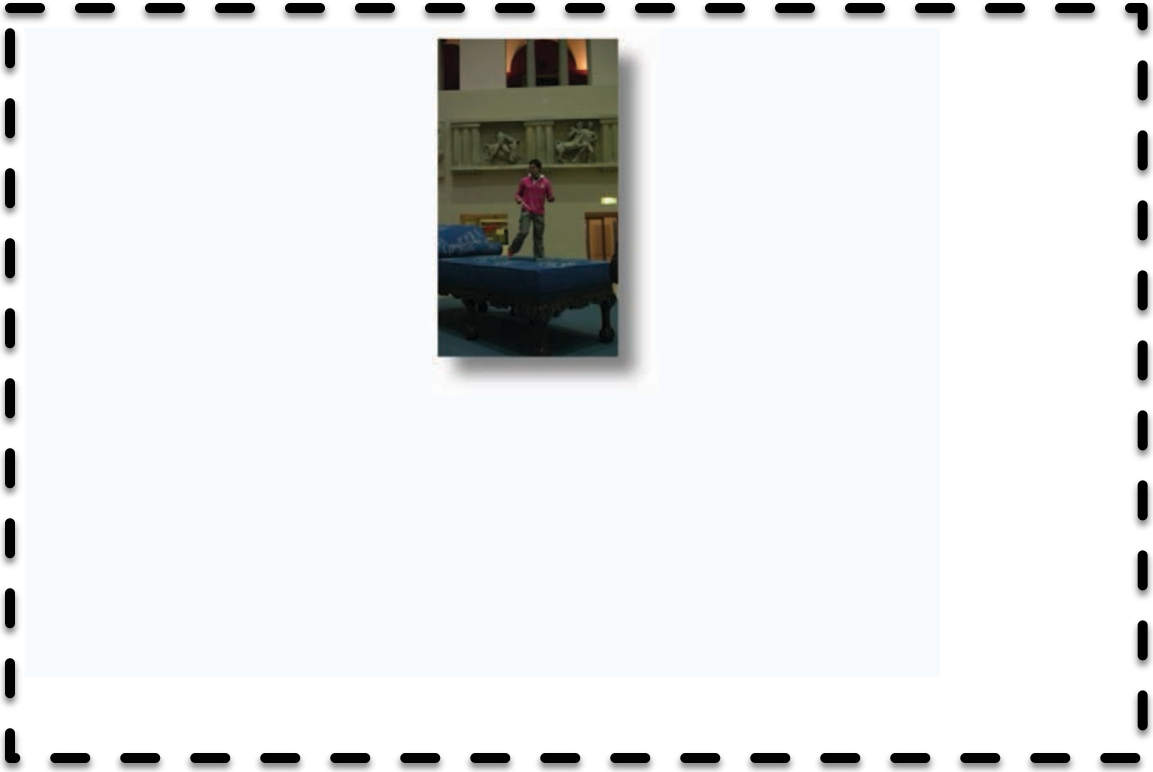
(Guan et al. CVPR10)





(a) $t_0 \sim t_2$

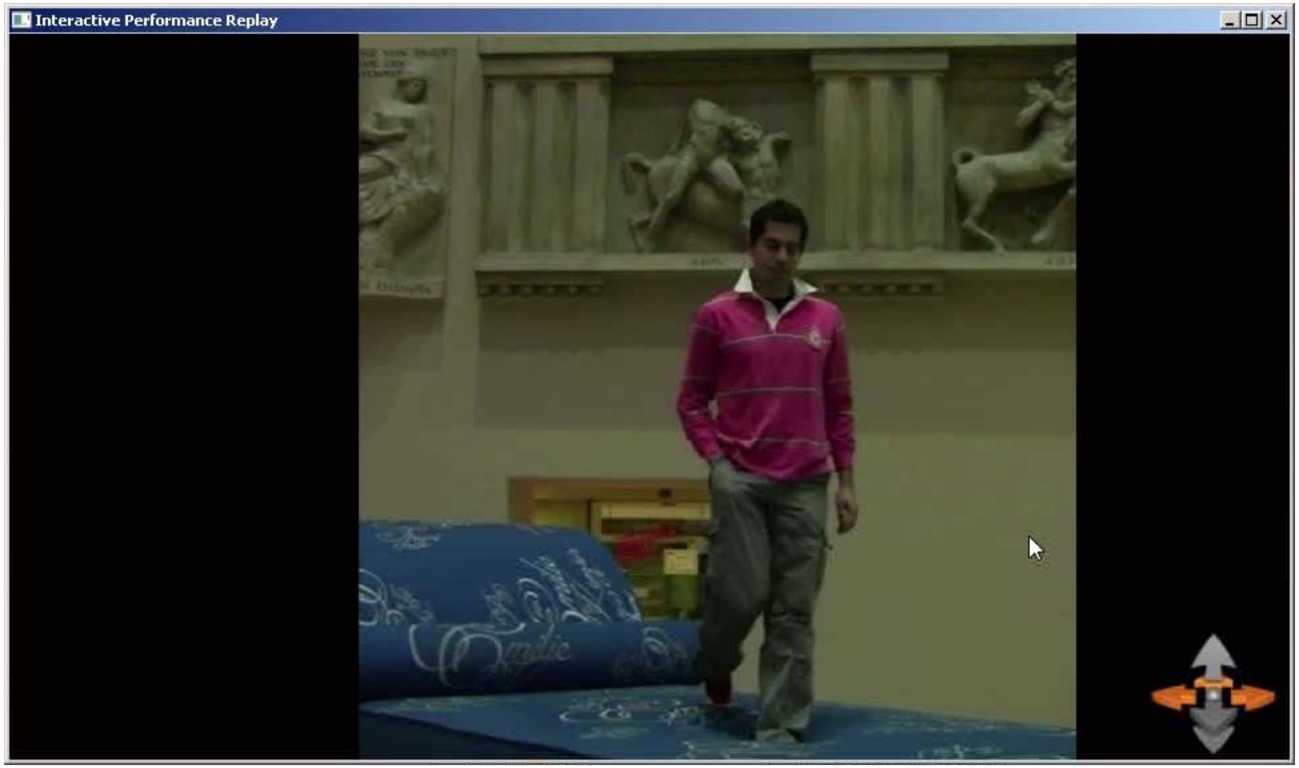
Interactive Navigation of casually captured videos



Collection of videos
of the same event
from different angles

Crowd of people
(with cameras)

Interactive Navigation of casually captured videos



Navigation in space and time

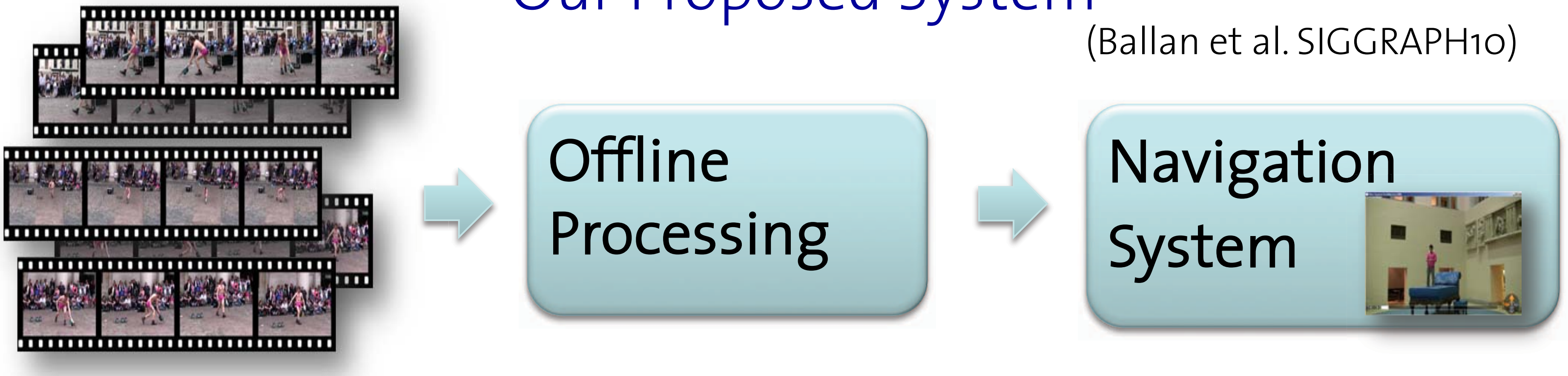
Casually Captured Videos

- Only few assumptions on the scene
- Large uncontrolled environments
- Filmed by nonprofessional people

How can we perform VBR in such a scenario?

Our Proposed System

(Ballan et al. SIGGRAPH10)



Video collection

Offline Processing

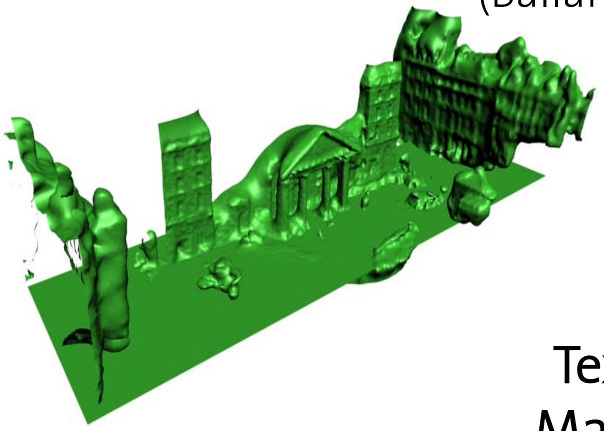
Navigation System

Offline Processing

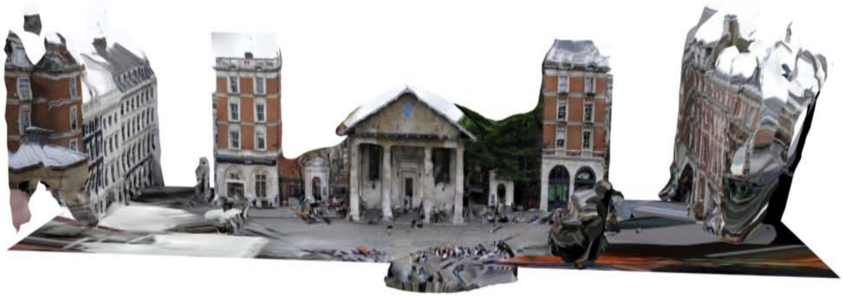
(Ballan et al. SIGGRAPH10)

Collection of images of the filming location

3D modeling



Texture Mapping



Offline Processing

(Ballan et al. SIGGRAPH10)

Video collection

Offline Processing

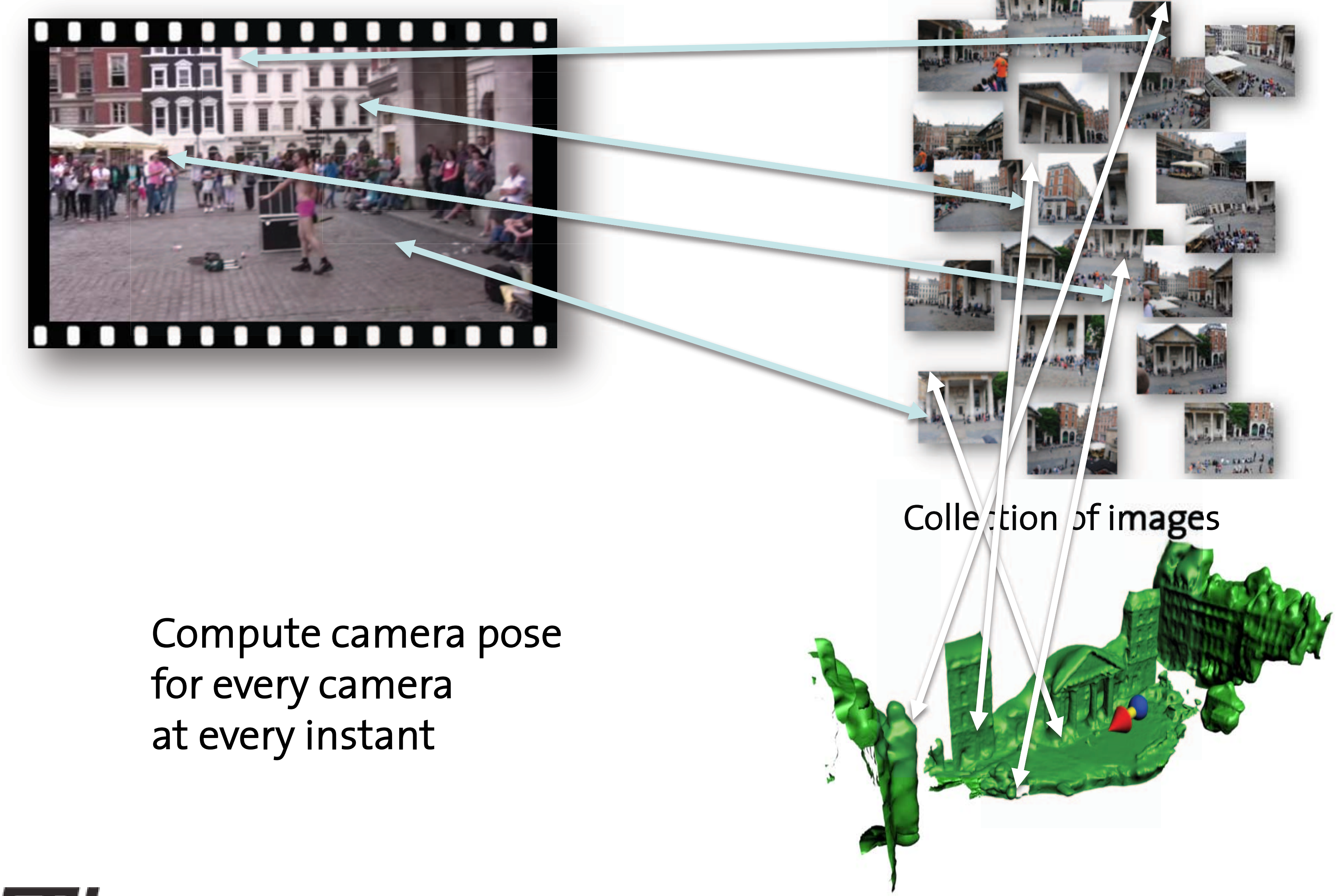
(Ballan et al. SIGGRAPH10)

Color calibration

Video collection

Time Synchronization

Spatial Calibration of the Videos

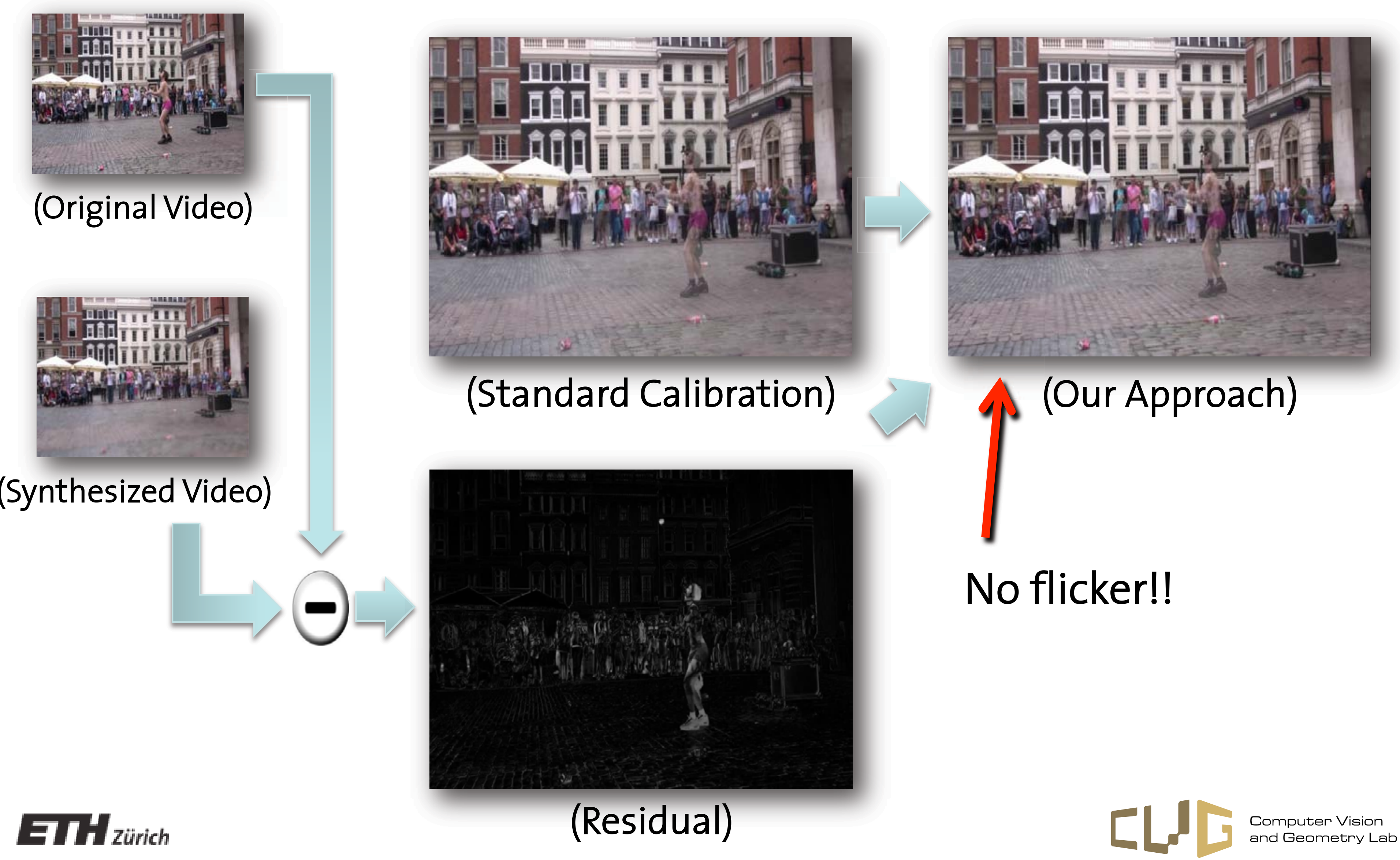


(Ballan et al. SIGGRAPH10)

Collection of images

Compute camera pose
for every camera
at every instant

Spatial Calibration of the Videos



(Original Video)

(Synthesized Video)

(Standard Calibration)

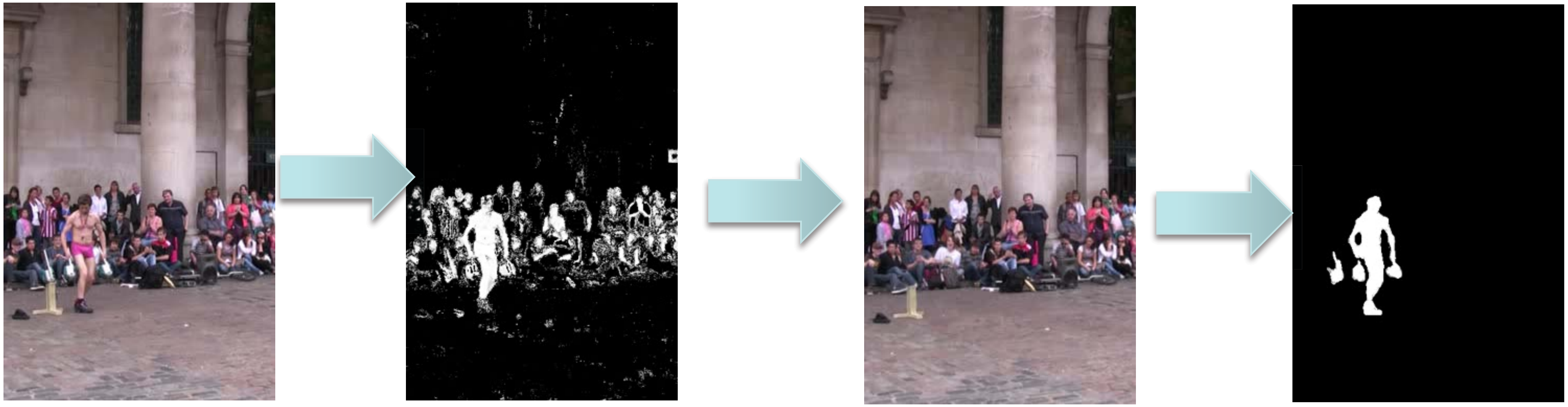
(Our Approach)

(Residual)

No flicker!!

Performer Segmentation

(Ballan et al. SIGGRAPH10)



Input video

Color based segmentation

Per-pixel color model of the background

Foreground-background segmentation

Rendering (interactive, on-line)

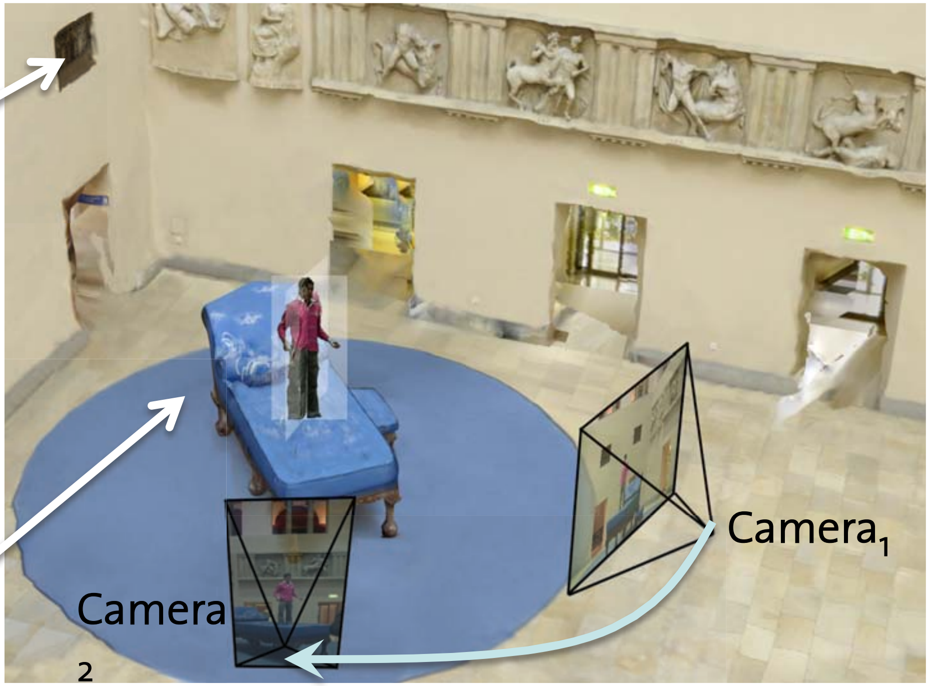
Request for a transition

Background

Pre-computed background geometry

Foreground

Billboards



Generate intermediate views along transition path

The Background

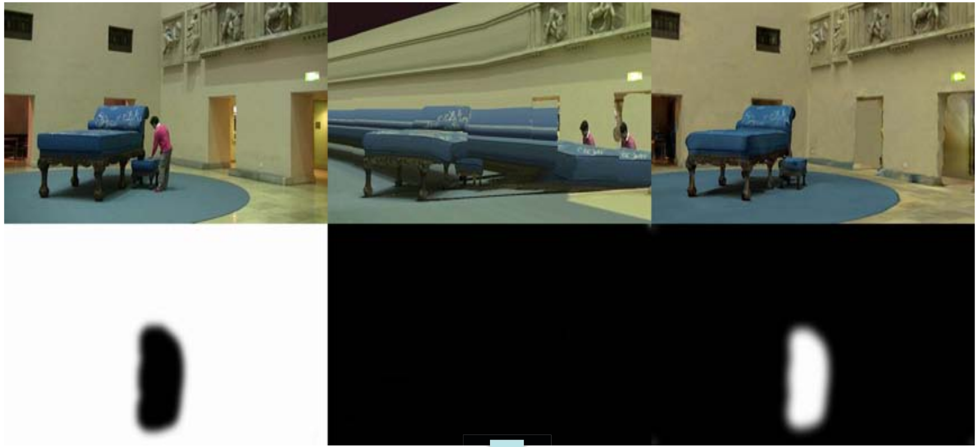
(Pre-computed texture)

ff

(Video stream)

adapt Unstructured Lumigraph (Buehler et al. '01)

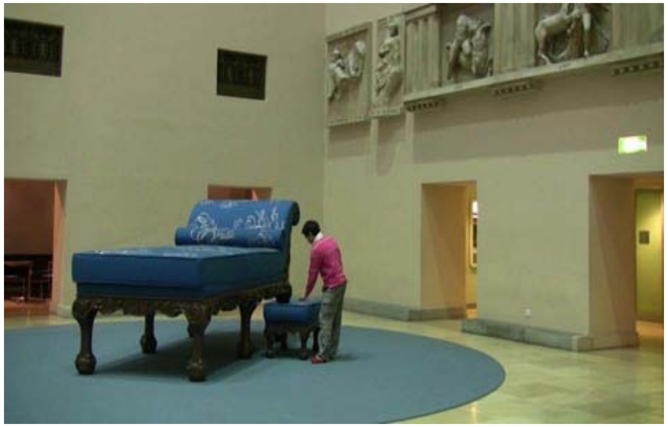
- Moving cameras
- Mask out the foreground
- Limit to only three sources to maintain real-time



(Camera 1) (Camera 2) (Background geometry)

Sources

Weights



Final rendering

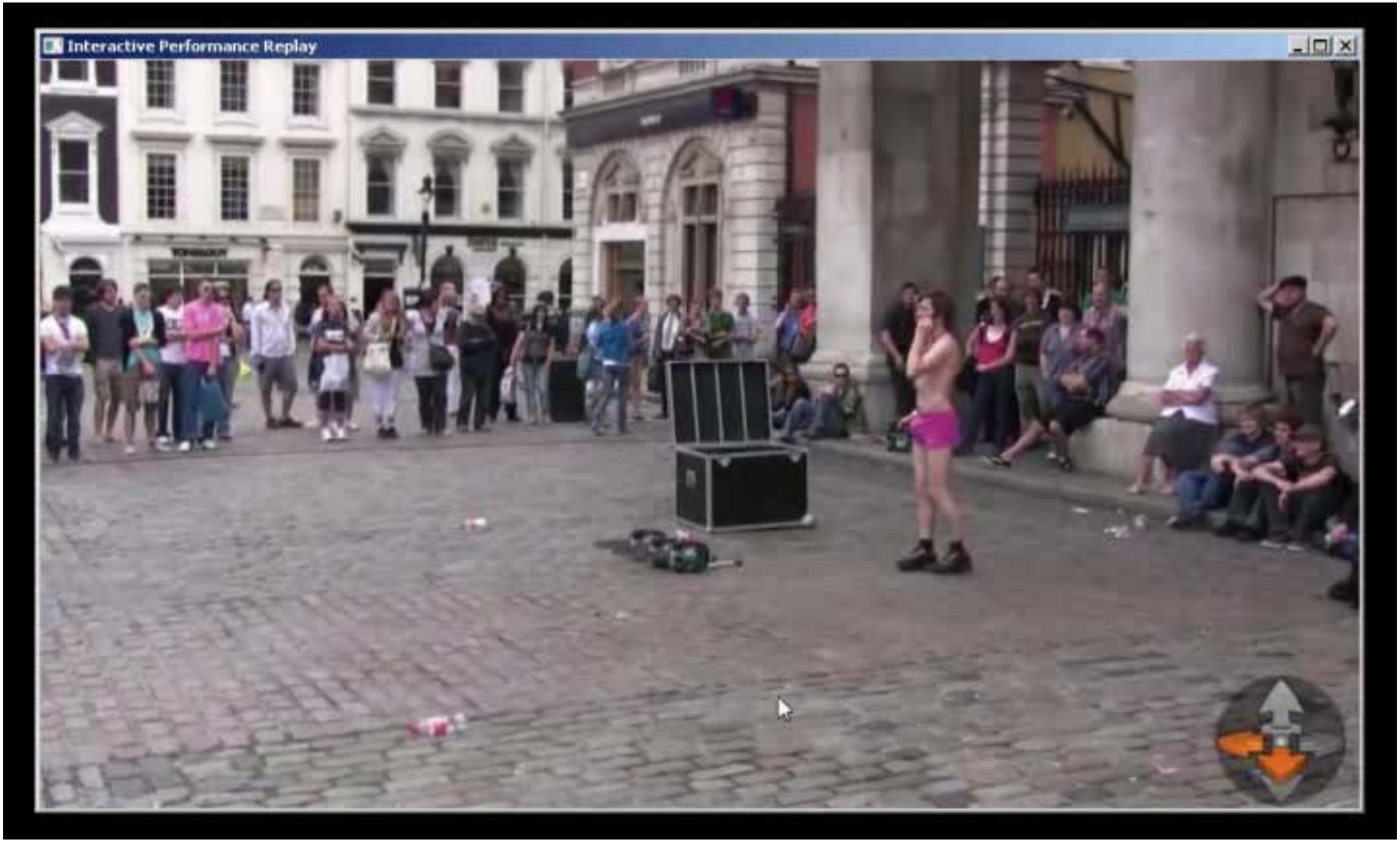
The Inter-Billboard distance

Unoptimized transition
(Naïve approach)

Optimized transition
(Our approach)

Interactive Navigation Tool: UI

(Ballan et al. SIGGRAPH10)



Interactive viewer, more results & datasets availal at: http://cvg.ethz.ch/research/unstructured-v

Starting Grant 4D Video

Conclusion

- Possibility to compute shape, motion and appearance from video, as well as camera system calibration
- Challenges:
  - Large-scale scenes
  - Dynamic objects, people in particular, in cluttered scenes
- Opportunities:
  - Advances in camera, processing, network and storage technologies
  - Lots of interesting applications in many different areas

Thank you for your attention!

Questions?