Algorithms for Image-Based Rendering with an Application to Driving Simulation

George Drettakis

GRAPHDECO/Inria Sophia-Antipolis, Université Côte d'Azur

http://team.inria.fr/graphdeco

informatics mathematics

GraphDeco

GRAPHics and Design with hEterogeneous COntent

UNIVERSITÉ CÔTE D'AZUR

Graphics for Driving Simulators

- Requirements for Computer Graphics (CG) in driving simulators
  - Realism
  - Large environments
  - Ease of Creation
  - Variety
- Current practice
  - Computer Graphics with Synthetic Content

Current Graphics for Simulators

- Getting better but:
  - Hard to create environments
  - Realism not quite there yet

CARLA – Intel

Gazebo

Computer Graphics (CG) with Synthetic Content

Can provide realistic content, but…

- Requires huge amount of manual labor, high cost
- Rendering quality can be good, but expensive

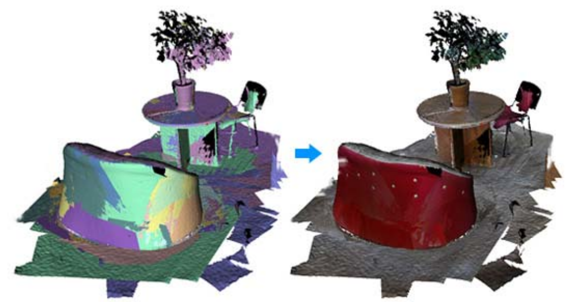
CG is about *creating* images

- Ultimately a *forward* process
- Often involves simulation or procedural methods

Captured Content

- Multi-view stereo, game changer

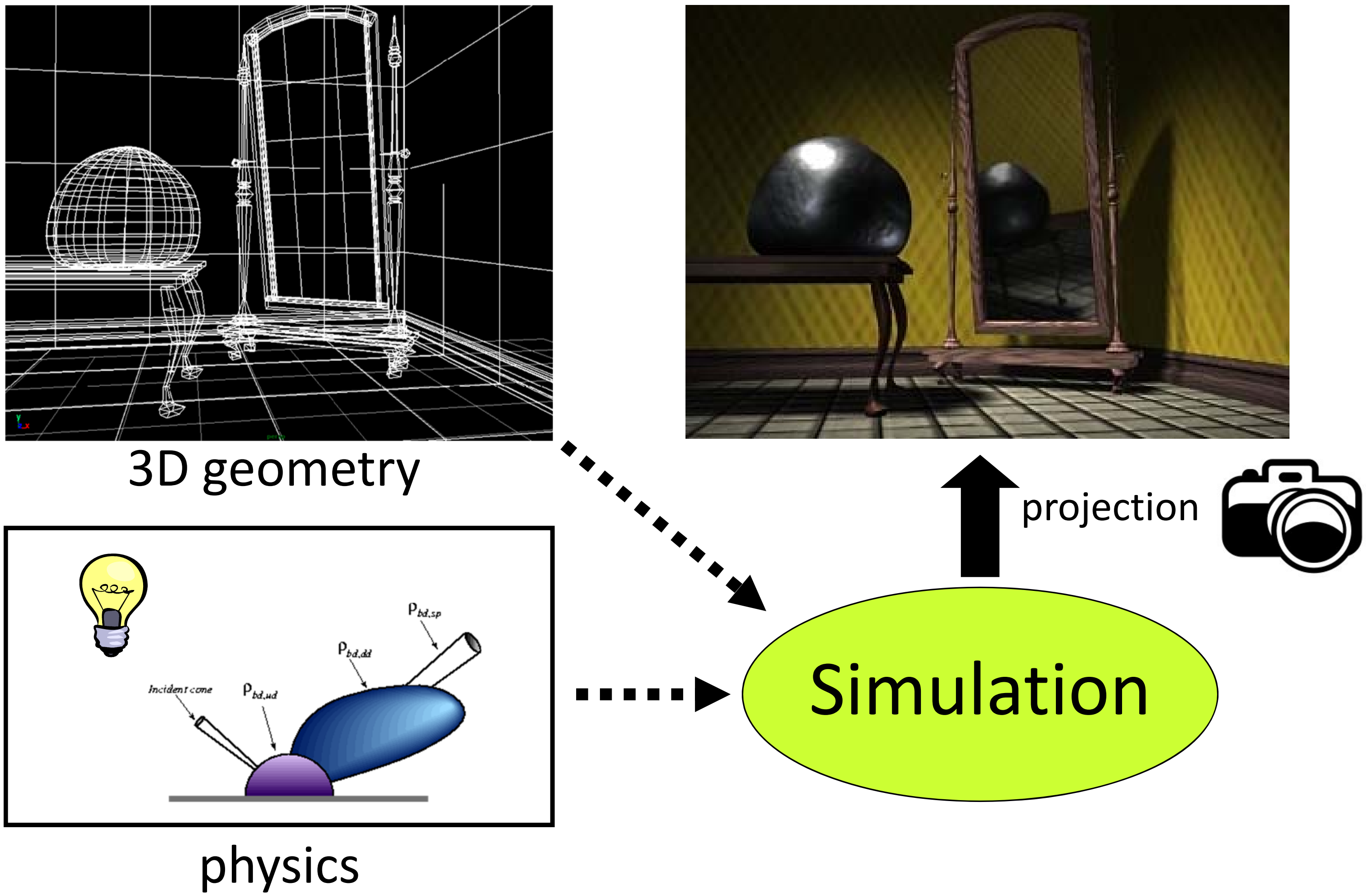
- Multiplication of sensors
  - Traditional expensive: LIDAR etc.
  - Cheaper, accessible: Kinect etc



Capture methods seen as *inverse* problem

- Frequently use optimization methods
- Image synthesis is often the ultimate goal

Traditional Computer Graphics



The richness of our everyday world

Photo by Svetlana Lazebnik

Creating Realistic Imagery

Computer Graphics

+ great creative possibilities
+ easy to manipulate objects / viewpoint / light
- Tremendous expertise and work for realism

Photography

+ instantly realistic
+ easy to acquire
- very hard to manipulate objects / viewpoint / light

Creating Realistic Imagery

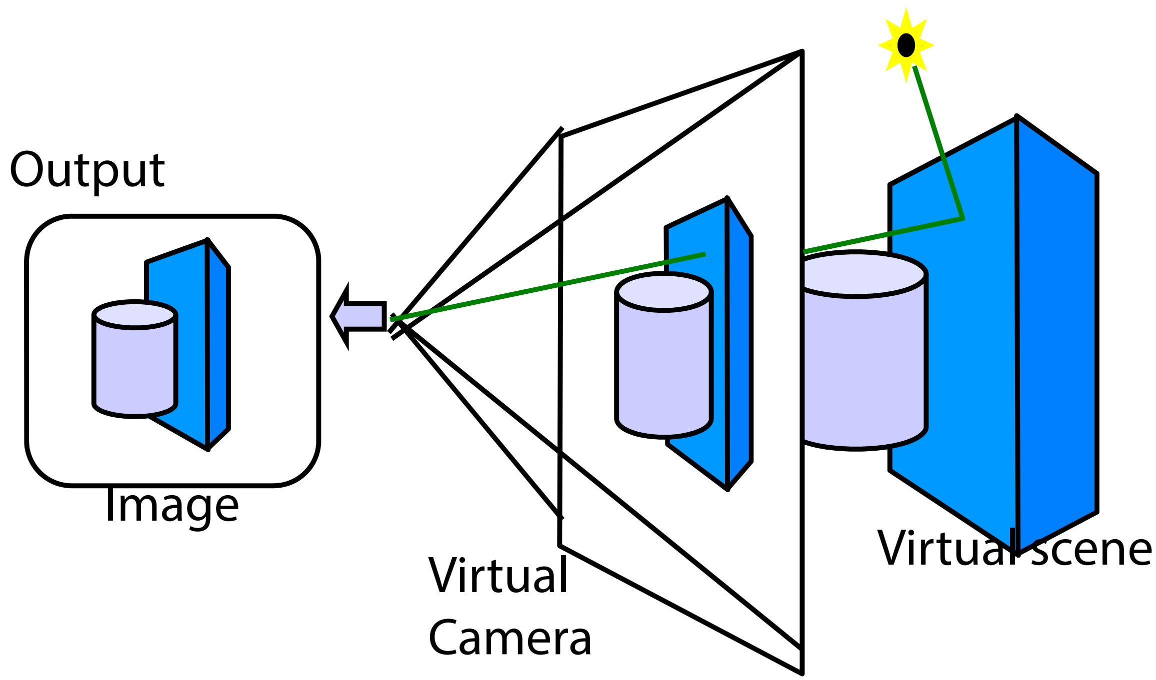
Computer Graphics	Image-Based Render (IBR)	Photography

Realism
Ease of creation
Manipulation

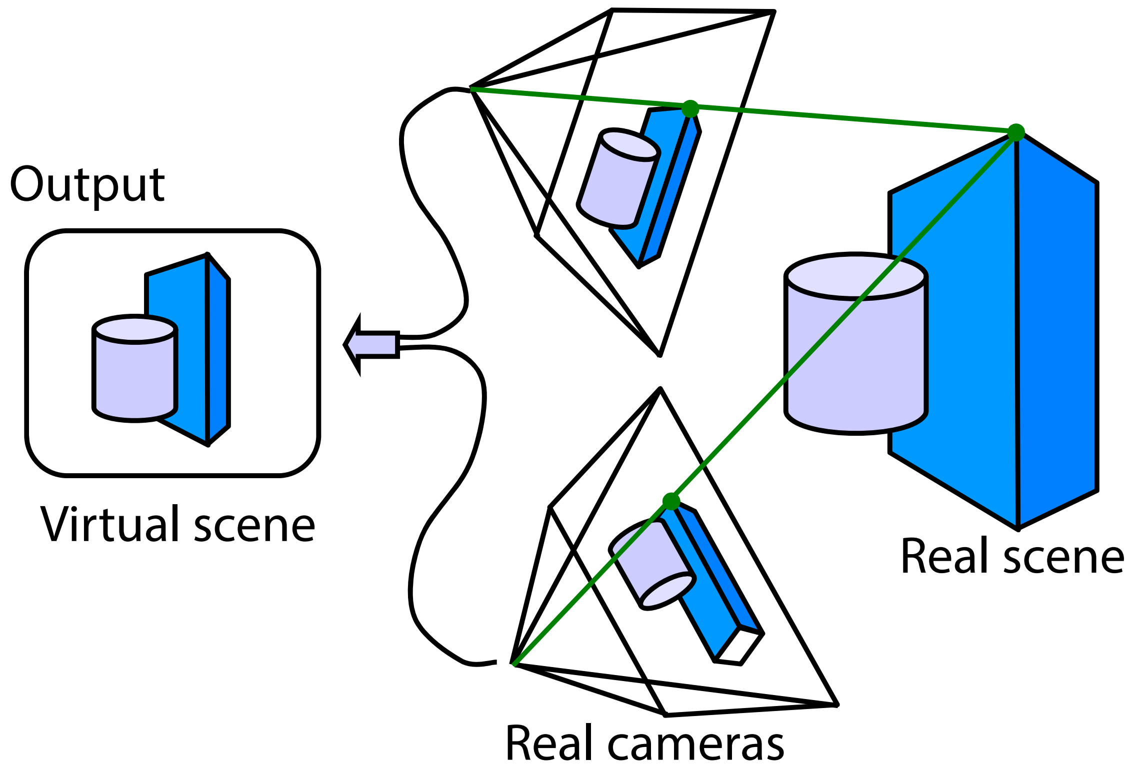
- \+ great creative possibilities
- \+ easy to manipulate objects / viewpoint / light
- \- Tremendous expertise and work for realism

- \+ instantly realistic
- \+ easy to acquire
- \- very hard to manipulate objects / viewpoint / light

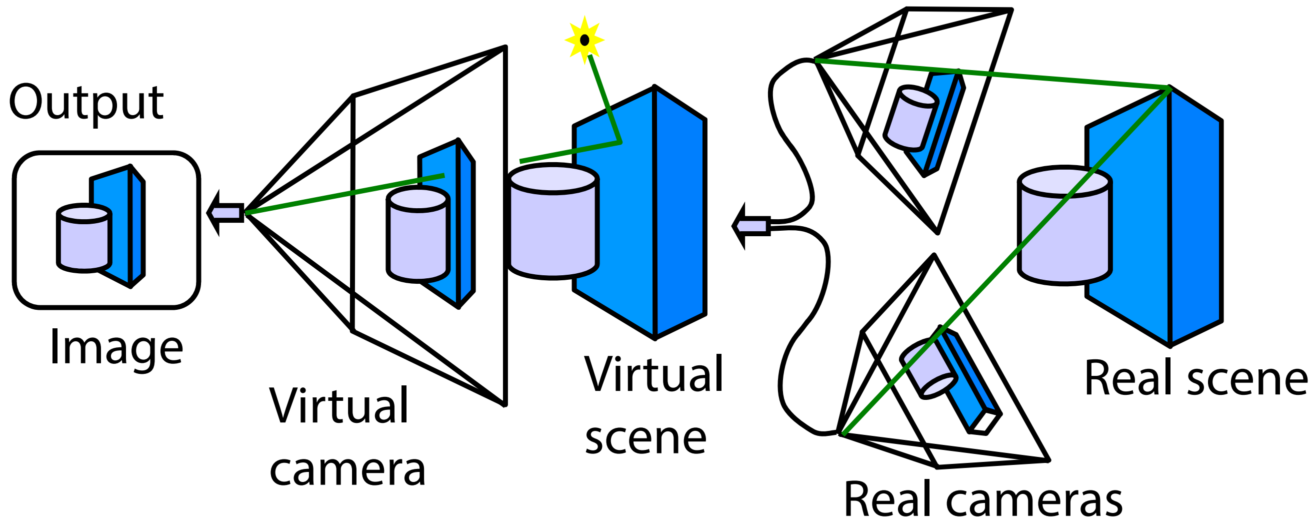
Image synthesis



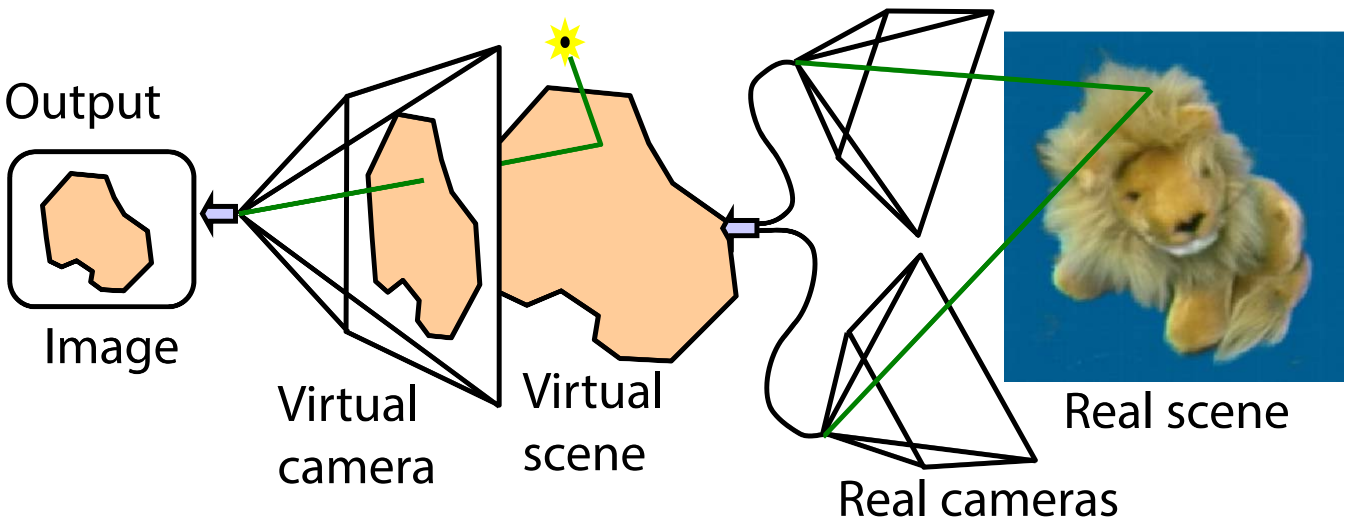
Computer vision



Computer vision and graphics



Imperfect computer vision



Imperfect computer graphics

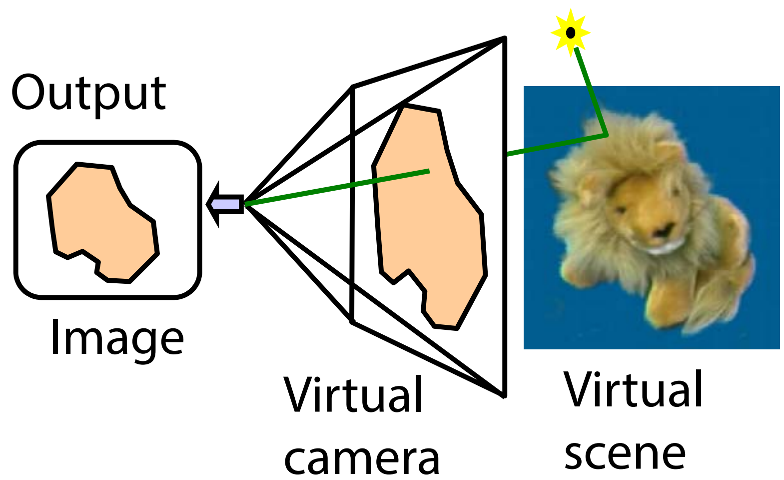


Image-based rendering (IBR)

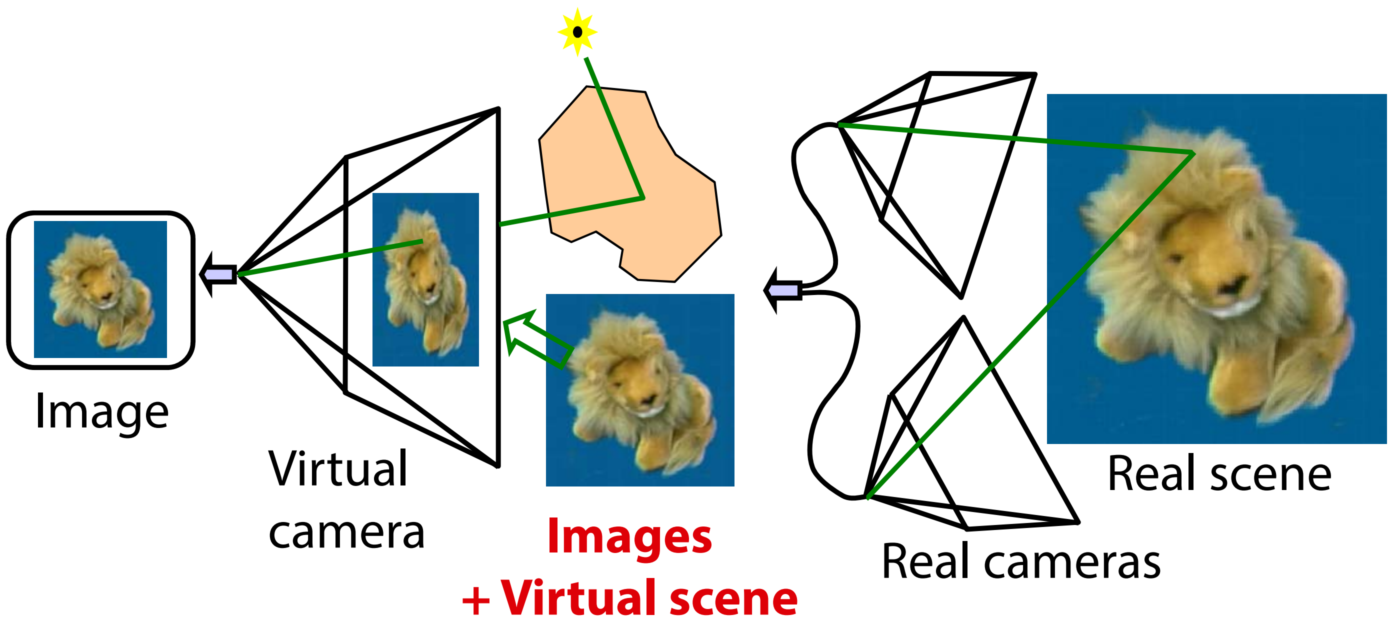
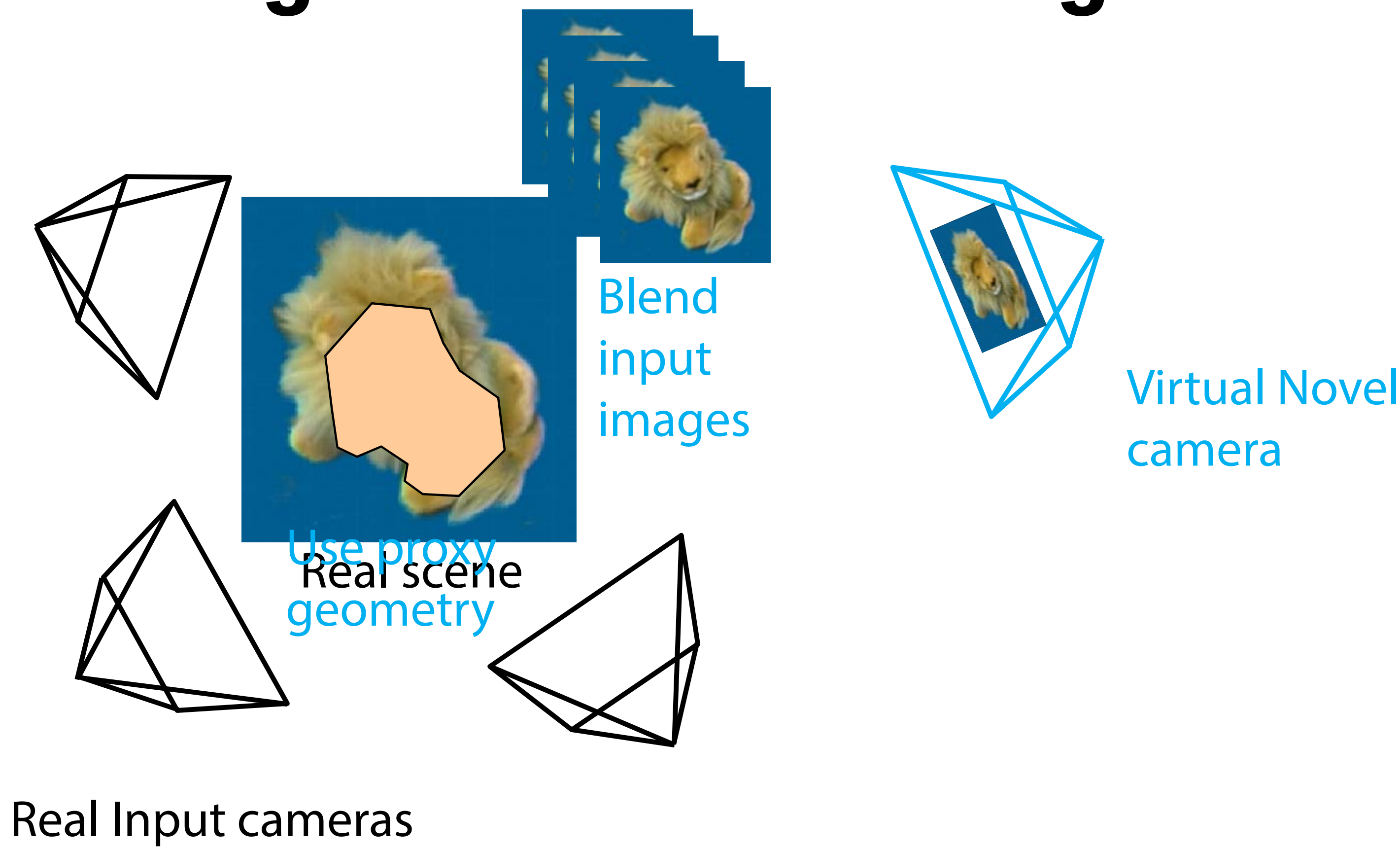


Image-based rendering

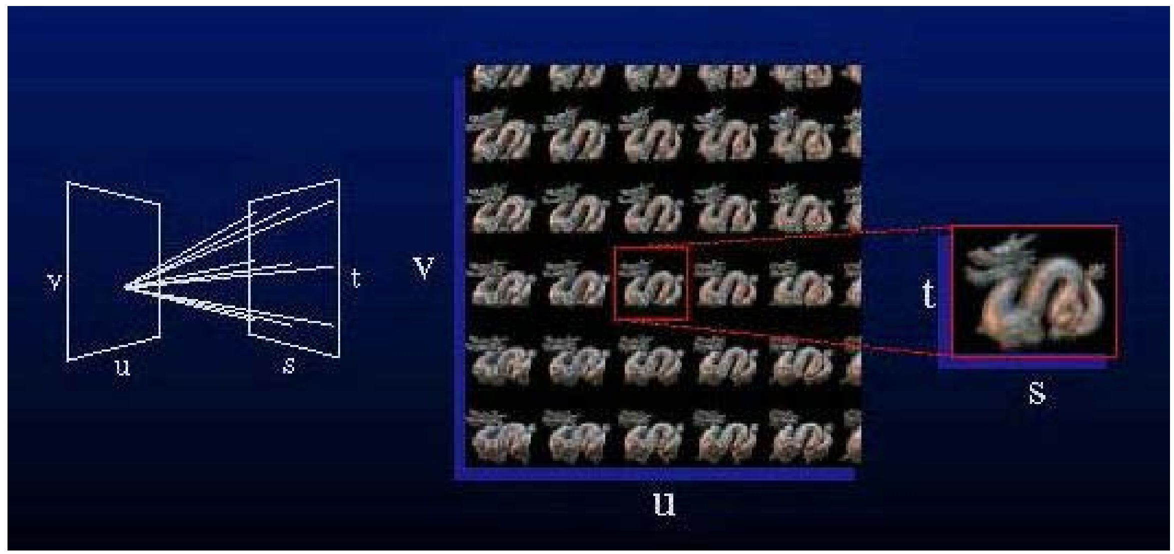


Background on IBR

Capturing multiple viewpoints

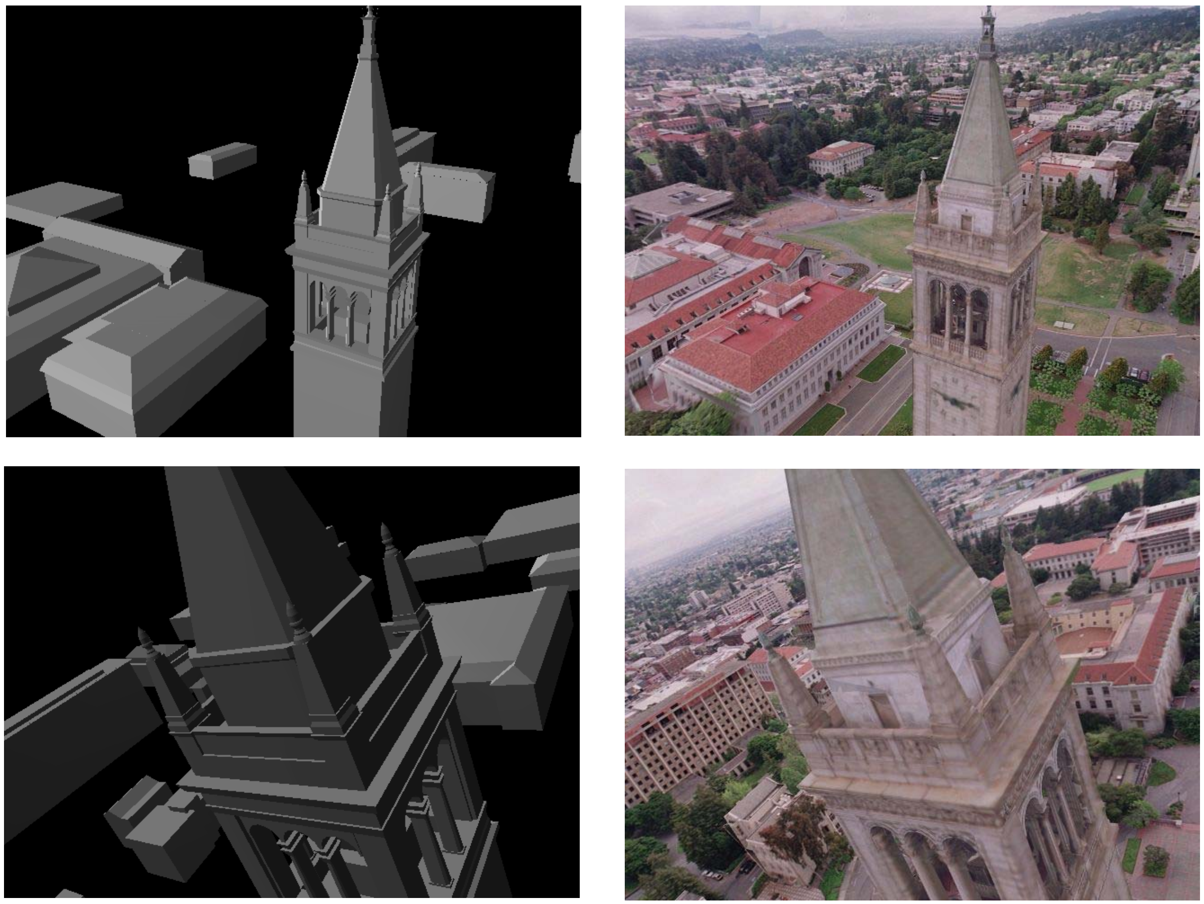
Capture the ***Lightfield*** [Levoy and Hanrahan, SIGGRAPH 1996]

- Every light ray from camera plane (u,v) in direction of the image plane (s,t)

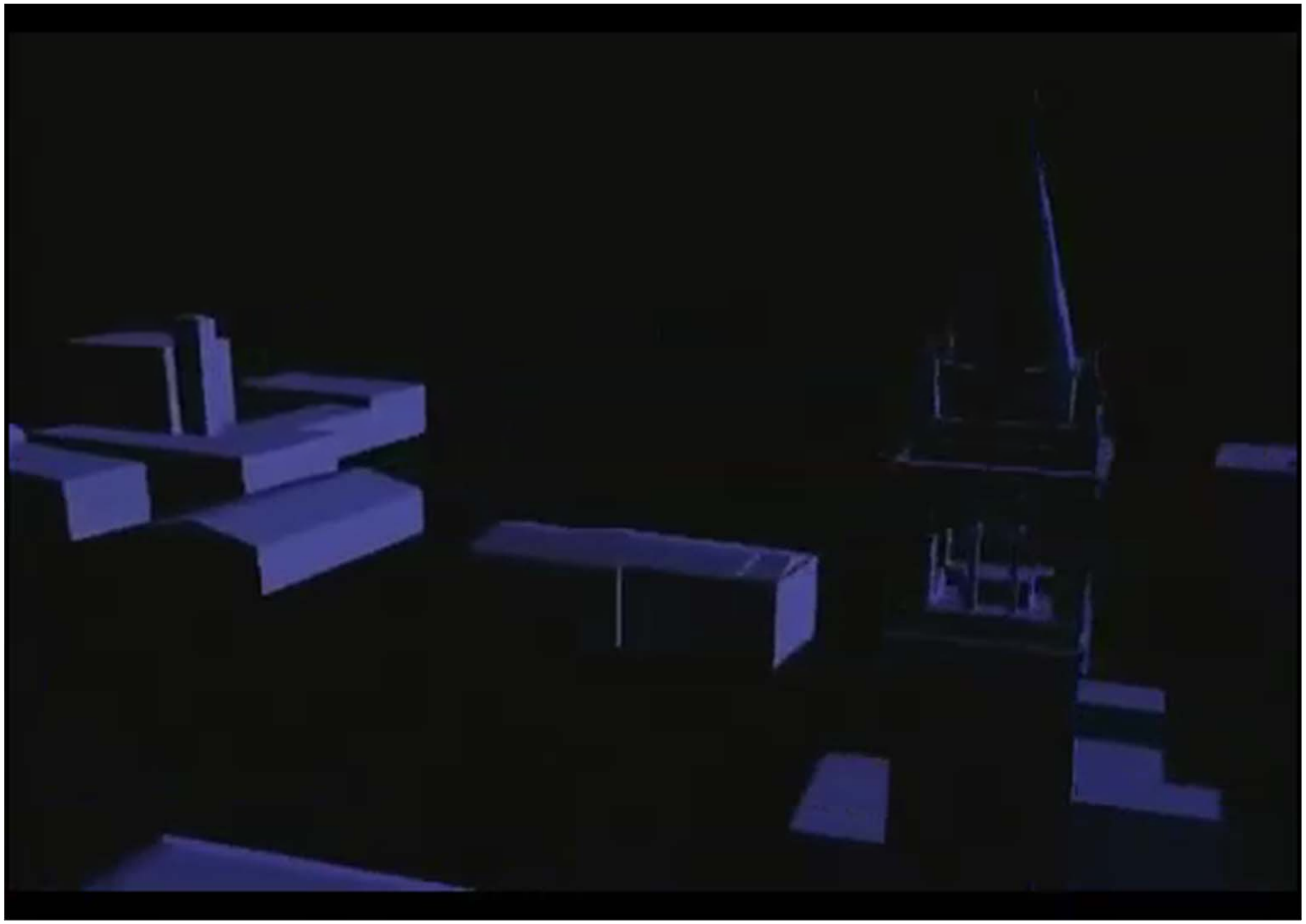


Lots of data! High redundancy, compression

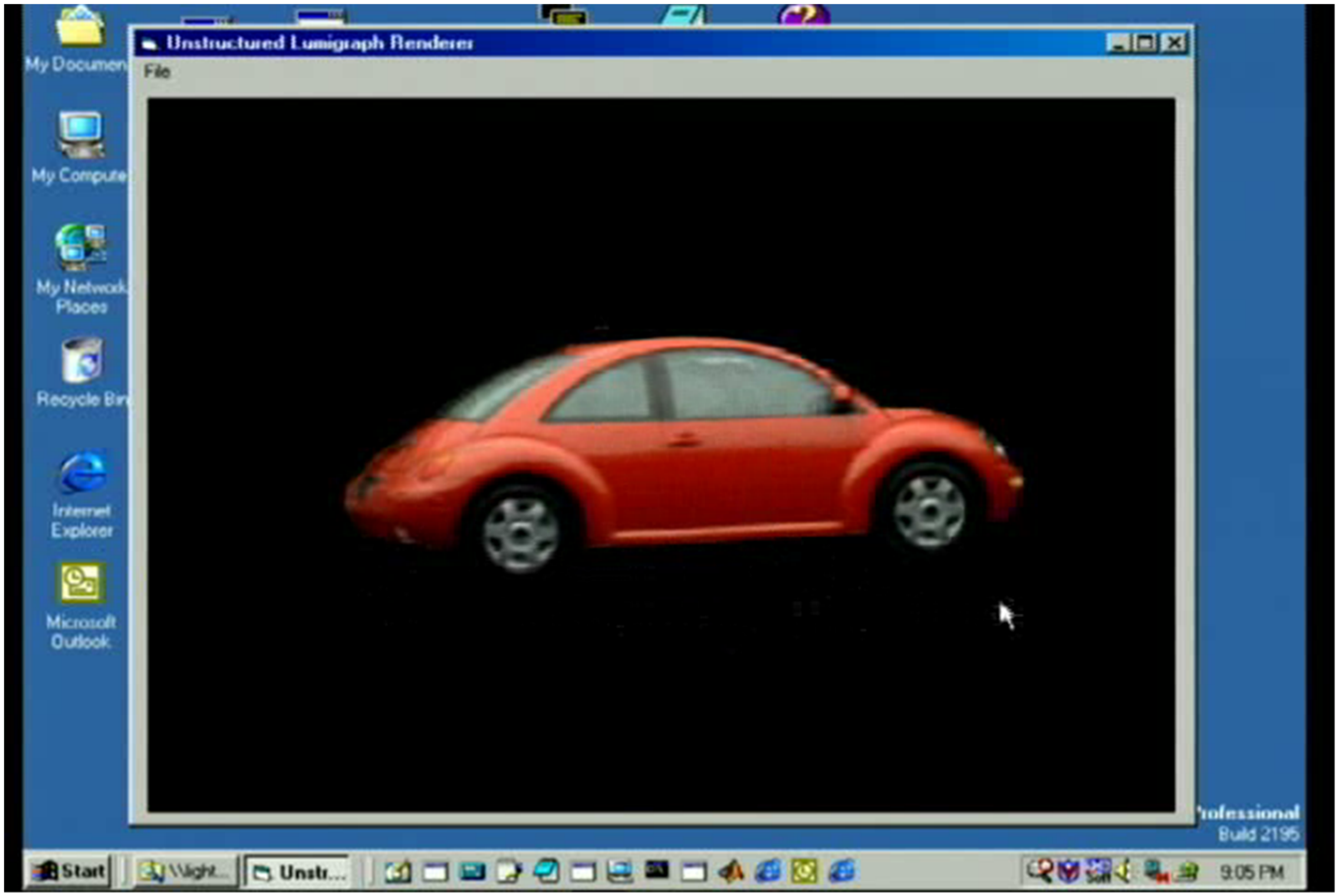
View-dependent texture mapping
[Debevec & Malik SIGGRAPH 1997]



View-dependent texture mapping [Debevec & Malik SIGGRAPH 1997]



Unstructured Lumigraph (ULR)
[Buelher et al. SIGGRAPH 2001]

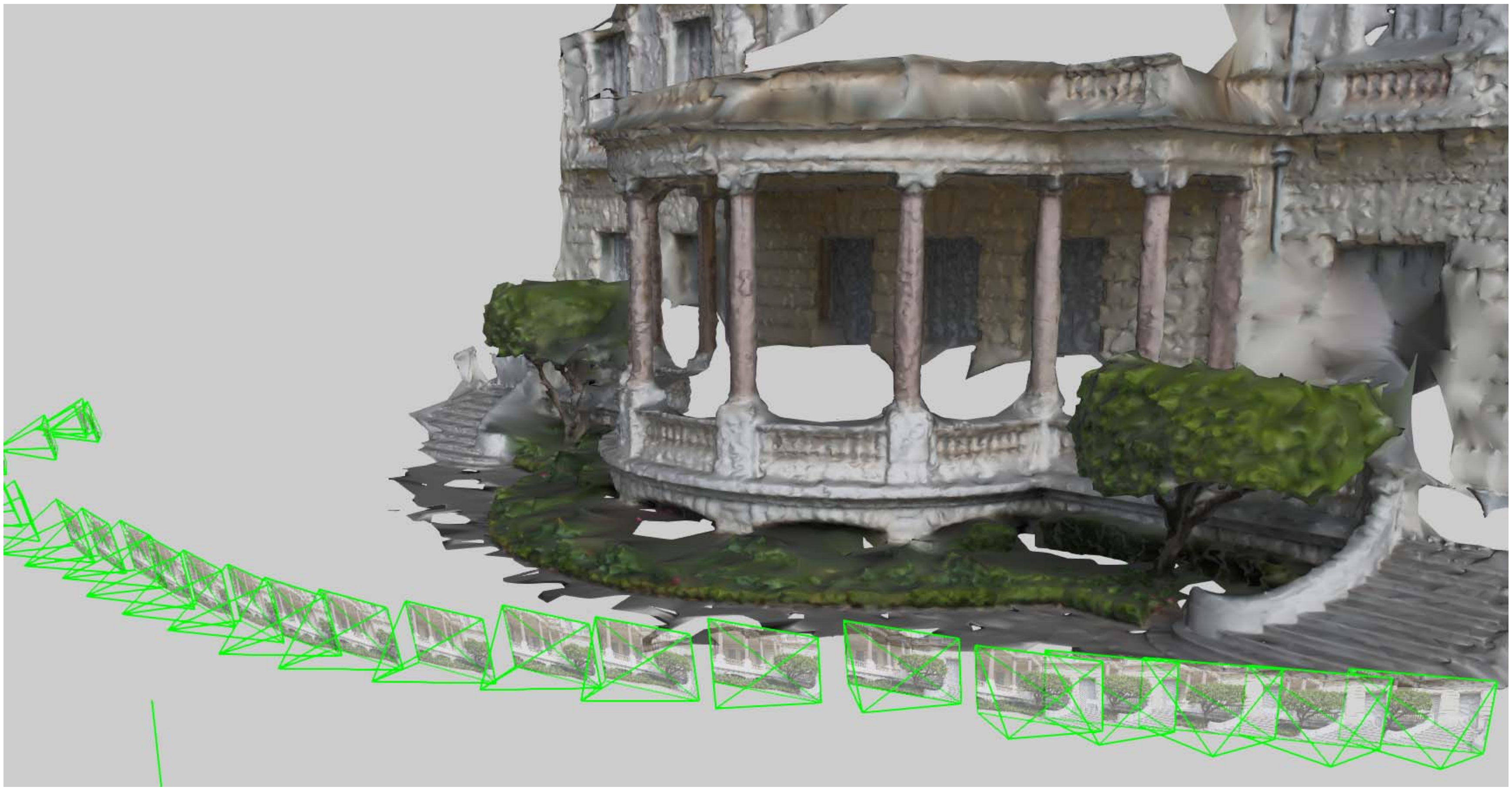


Recent Image-Based Rendering

- Approximate multi-view capture of a scene
  - Camera calibration (Structure from Motion, SfM)
  - Multi-view Stereo reconstruction (MVS)
- Now a mature technology, commercially available solutions
  - Bentley/Acute3D ContextCapture
  - RealityCapture
  - PhotoScan
  - … etc

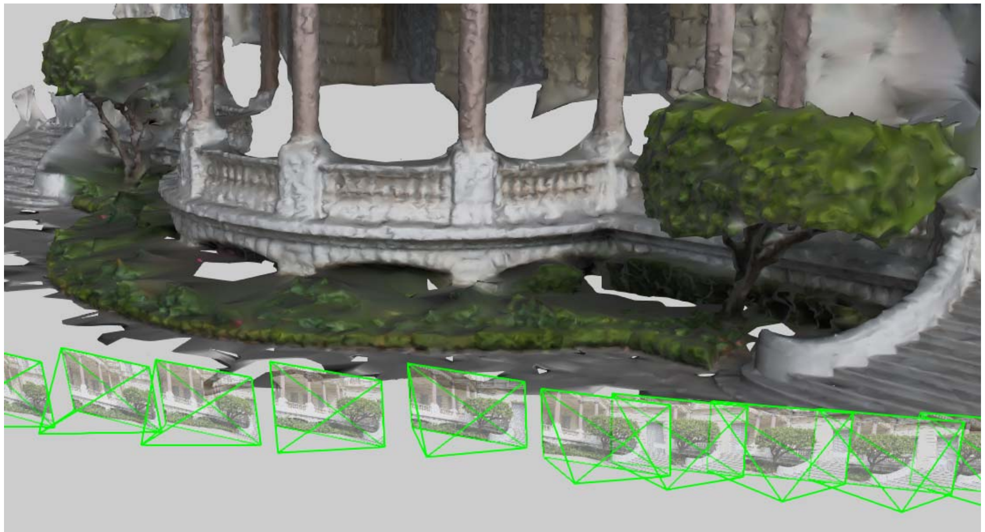
Multi-View Stereo

Multi-View Stereo



SfM and MVS

- We have the positions and orientations of the camera for each photo



- We have an approximate geometric model
  - Missing parts and details
  - Overestimated geometry
  - Bad normals/surface reconstruction

How to render ?

Option 1: Textured Mesh

Textured Mesh

Option 2: Traditional IBR

Basic Idea

- Use MVS *geometry* and *blend* images using a set of “clever” *weights*
- Image blending:
  - Corrects some of the geometry errors
  - Provides some view-dependent effects

Option 2: ULR [Buehler 2001]

Prior work: ULR? [Buehler2001]

Moving highlights

Novel Solutions for IBR at Inria/GRAPHDECO

Address two main shortcomings:

- Inaccurate proxy geometry
- Finding good blending weights

How ?

- Create a representation *per-view* overcoming limitations on geometry
- New algorithms to improve blending of different views

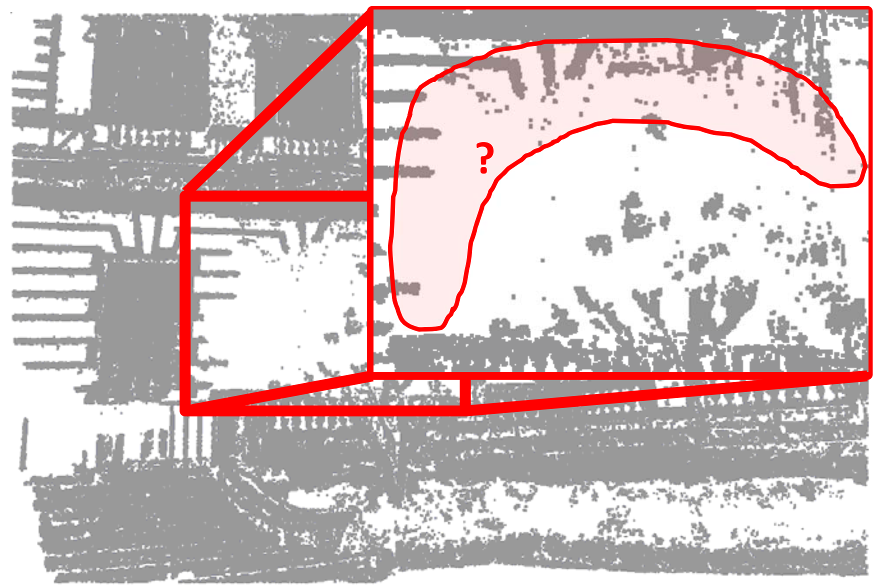
Depth synthesis and local warps for plausible image-based navigation

G. Chaurasia, S. Duchêne,
O. Sorkine-Hornung, G. Drettakis

ACM Trans. On Graphics 2013

Main Problems

- Sparse depth
- Occlusion boundaries



Compensate using 2D image-based approximations

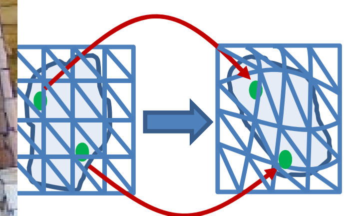


Image warping	Adaptive blending

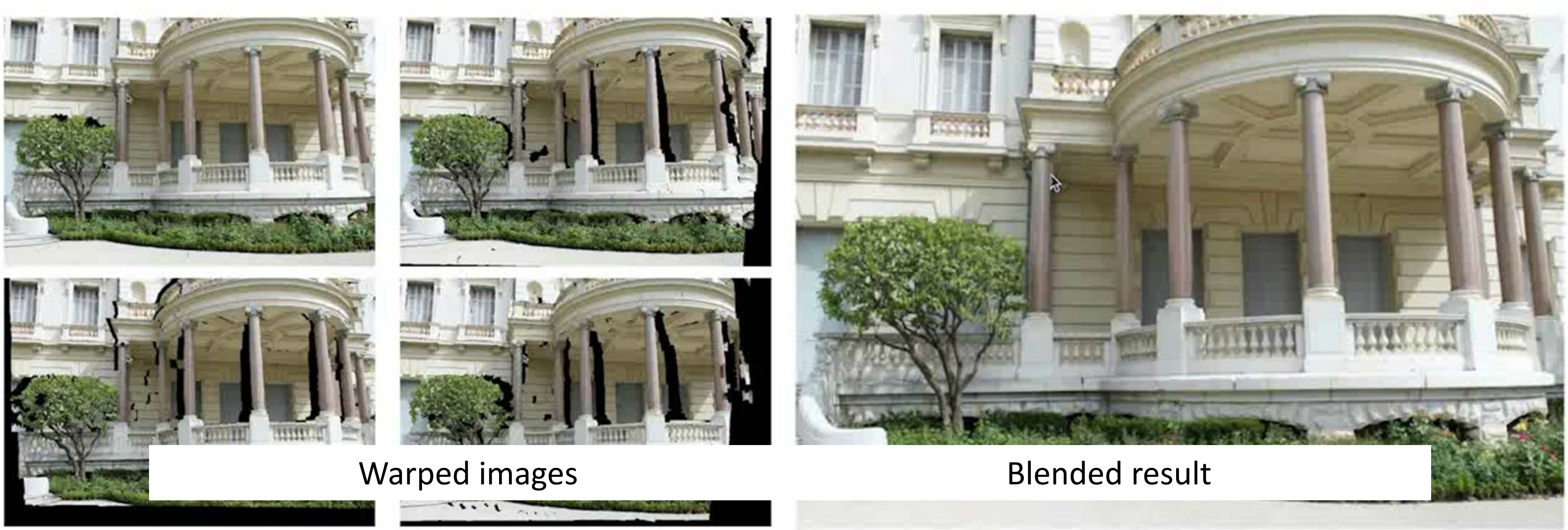
Fill depth in [illegible] image warps

Local Shape Preserving Warp

Image reprojected into novel view using depth samples

Image warped into novel view using shape-preserving warp

Warp and Blend



Results

Scalable Inside-Out Image-Based Rendering

Peter Hedman [1], Tobias Ritschel [1], George Drettakis [2], *Gabriel Brostow* [1]

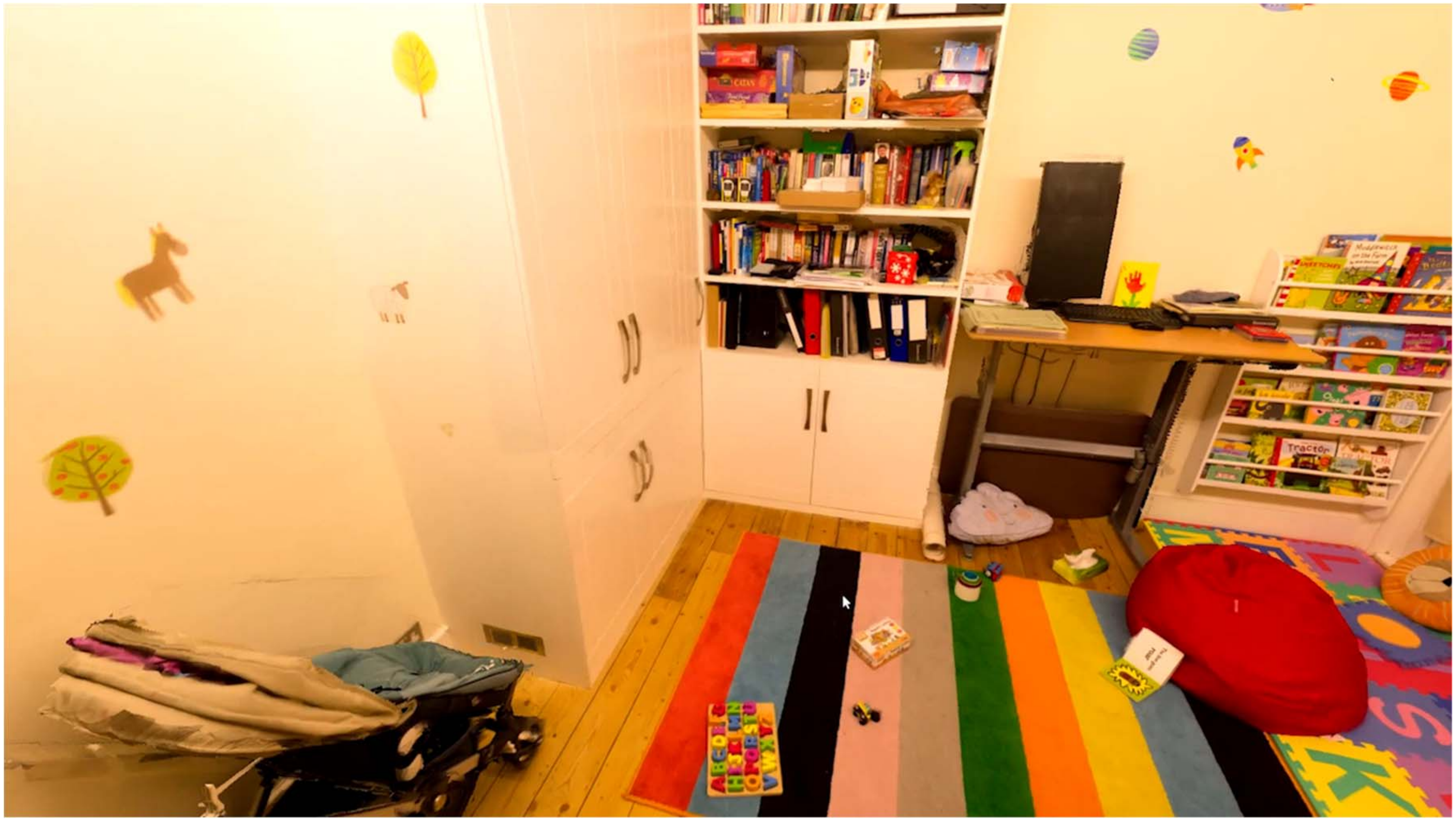
UCL [1], *Inria* [2]

ACM Transactions on Graphics (SIGGRAPH Asia 2016)

Improving Per-View Representation

- Improve per-view representation
  - Per-view *meshes*
- Start with Indoors Scenes
  - Initial version using depth sensors

Indoors Scenes



Indoors Scene

- Use Kinect to get reasonable *global* geometry
- Needs a large number of photos to cover the space
  - Larger overlap than outdoors

Prior work: Unstructured Lumigraph (ULR)?

[Devec1998, **Buehler2001**, Eisemann2008]

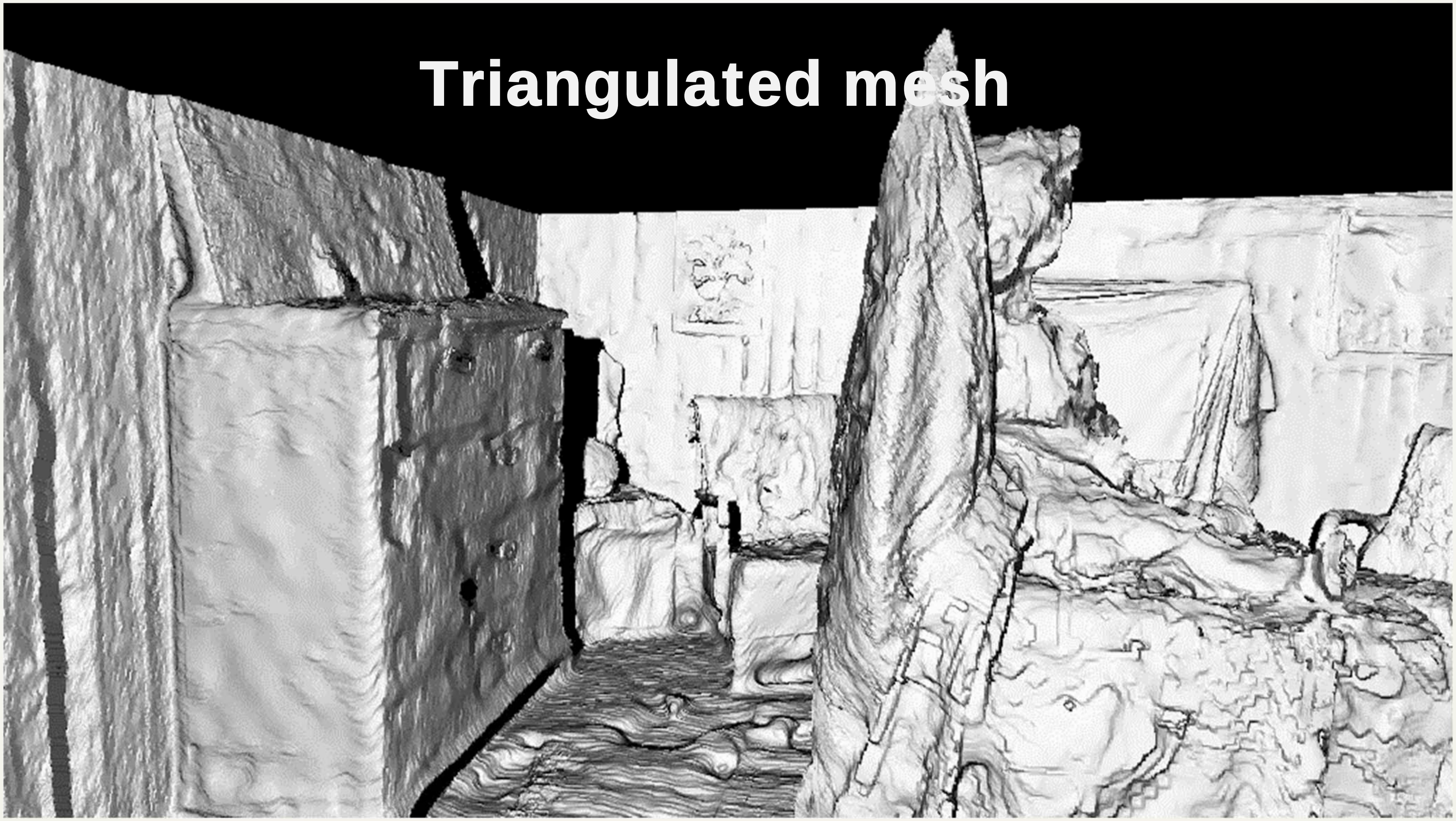
Inaccurate geometry

Prior work: Per-photograph geometry!

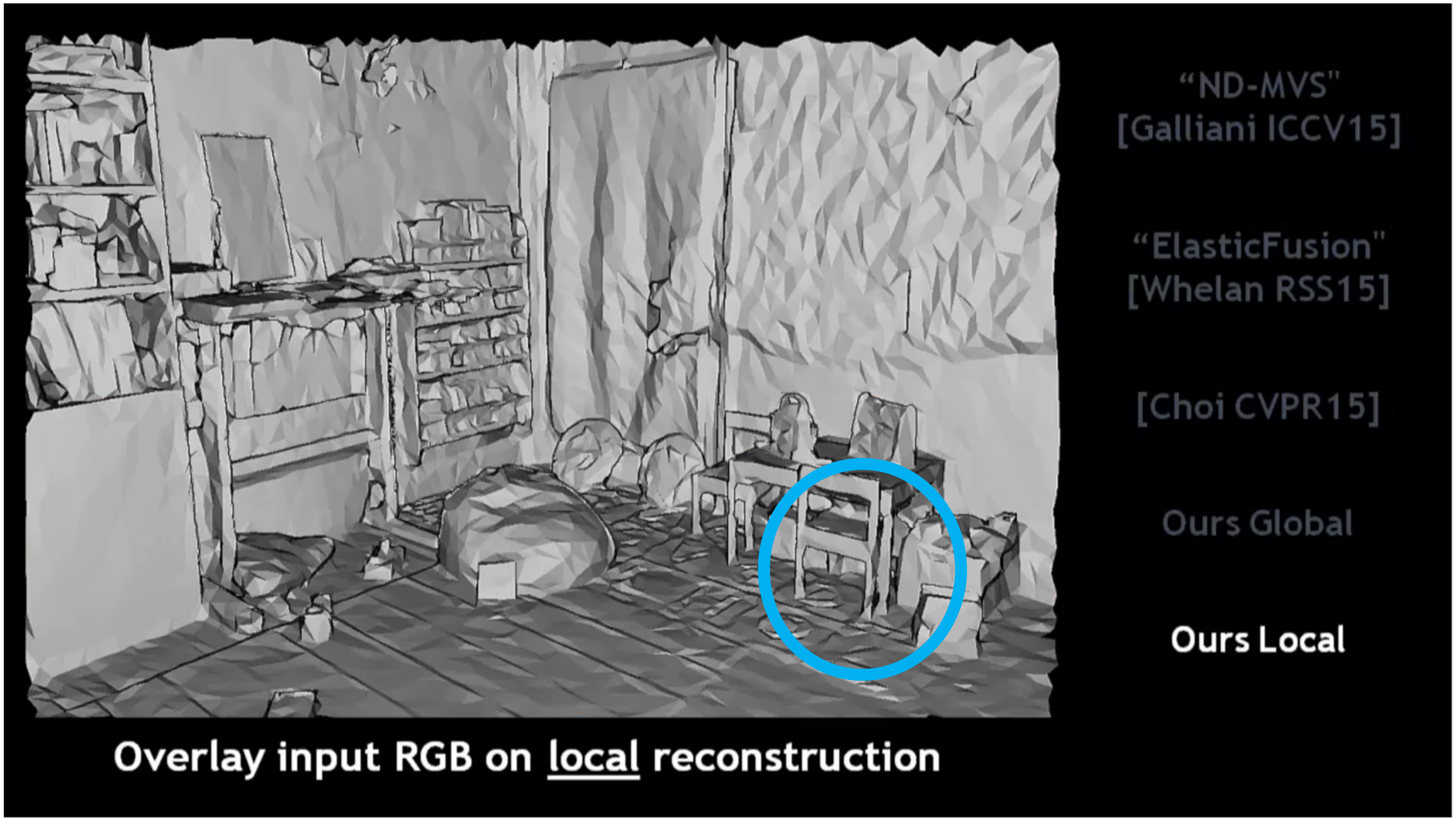
[**Ours**, Zitnick2007, Chaurasia2013, Kopf2013, Ortiz-Cayon2015]

Refined geometry

Refining per-photograph geometry



Indoors scenes



Indoors scenes

Museum scene: 298 input images

Deep Blending for Image-Based Rendering

Peter Hedman [1], Julien Philip [3], True Price [2] , Jan-Michael Frahm [2], George Drettakis [3], *Gabriel Brostow* [1]

UCL [1]*, UNC* [2] *, Inria* [3]

ACM Transactions on Graphics (SIGGRAPH Asia 2018 – conditionally accepted)

Deep Blending

- Further improve per-view meshes
  - Better refinement
  - Edge-aware mesh simplification
  - Can treat **outdoors** scenes !
- Learn blend weights by training in a leave-one-out manner
  - Train on 19 datasets, total of 2600 images
  - Learn blending weights

Deep Blending Results

Outdoors Scene: texture mapping

Deep Blending Results

Additional Challenges

- Changing the scene is hard in IBR
- Two advances:
  - Removing / moving objects
  - Changing the lighting (“relighting”)

Plane-Based Multi-View Inpainting for Image-Based Rendering

Julien Philip
Inria, Université Côte d'Azur

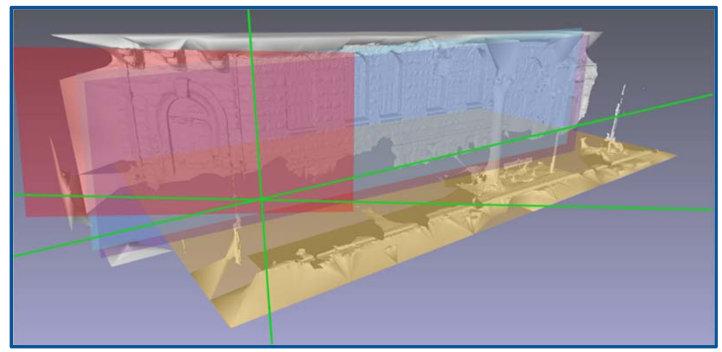
George Drettakis
Inria, Université Côte d'Azur

ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games
Montreal, Quebec, Canada: 17 May 2018

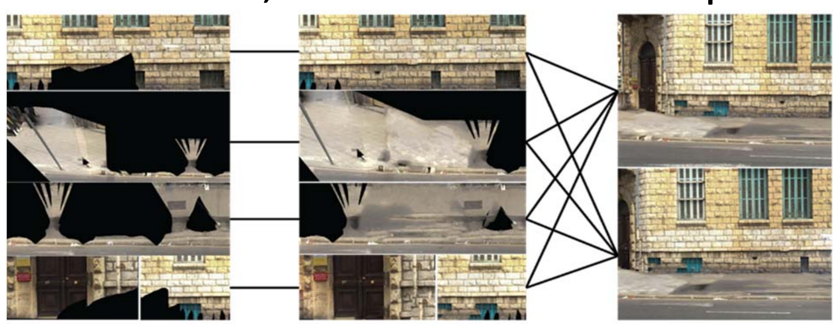
DEMO

Remove Objects

- Perform multi-view *inpainting*
  - Estimate local planar approximation of surface



  - Perform efficient multi-scale, resolution-aware inpainting using PatchMatch



Inpaint planar regions   Inpaint input images

Some of the Input Images

Our result

Our result

Multi-View Relighting for Outdoors Scenes

S. Duchêne, C. Riant, G. Chaurasia, J. Lopez-Moreno, P-Y. Laffont, S. Popov, A. Bousseau, G. Drettakis

ACM Transactions on Graphics 2015

Intrinsic Decomposition

Guide decomposition combining
3D information, rendering
and image-processing

Reflectance

Shading

Input Image

Results


Input
Reflectance
Sun visibility
Shading
Reflectance Layer
Environment Layer

Image-Based Relighting

Application to Image-Based Rendering with Relighting

Driving Simulation Application

- First challenge: capture of large scenes
  - Previous examples 20-200 images
  - Need 1000-2000 images for a city block
- Solution:
  - “Inria-bike” capture:

Driving Simulation Application

- Second challenge: Rendering large scenes with thousands of pictures
  - CPU memory; loading from disk
  - GPU memory; loading from CPU
- (Initial) Solution:
  - Streaming system

Driving Simulation Application

- Demo !
- Note: Semi-automatic cleanup for mesh outliers and moving cars (in the process of being automated)


Views
Capture
LRStream
Top view
STRAMIGIOLI
AZ-077-EV
Frame 843 - onStreamGPULoading Camera 645 [success=1]
Frame 841 - onStreamGPULoading Camera 1143 [success=1]
Frame 839 - onStreamGPULoading Camera 1140 [success=1]
Frame 837 - onStreamGPULoading Camera 446 [success=1]
Frame 835 - onStreamGPULoading Camera 1359 [success=1]

Future Work

- Correct remaining artifacts
  - Solution for reconstruction & rendering of cars
  - Hard cases in general:
    - Thin structures,
    - Reflections, transparency
  - Ad-hoc, specific solutions
  - Semantics & Learning to the rescue ?

Future Work

- Allow more editing
  - Better relighting solution
  - More geometry editing
- Seamless mix with synthetic content
  - Combine rendering modalities; develop solutions that are not mutually exclusive (ERC FUNGRAPH)

Conclusions

- Very promising technology to provide highly realistic CG for driving simulators
  - But technology must mature before “prime time”
- Could be a “game-changer” for realism in driving simulation
  - (Human) Training in assisted driving
  - (Machine) Training for autonomous driving

Funding

- EU FP7 VERVE *https://gv2.scss.tcd.ie/VERVE/*
- EU FP7 CR-PLAY *http://www.cr-play.eu/*
- EU H2020 EMOTIVE *https://www.emotiveproject.eu/*
- ANR SEMAPOLIS *https://project.inria.fr/semapolis/*

Thank you

Questions ?