Reflections on Image-Based Rendering

Richard Szeliski
The University of Washington

TUM AI Guest Lecture Series
January 28, 2021
Reflections on Image-Based Rendering

Richard Szeliski
The University of Washington

TUM AI Guest Lecture Series
January 28, 2021
Novel view synthesis is a long-standing problem at the intersection of computer graphics and computer vision. Seminal work in this field dates back to the 1990s, with early methods proposing to interpolate either between corresponding pixels from the input images, or between rays in space. Recent deep learning methods enabled tremendous improvements to the quality of the results, and brought renewed popularity to the field. The teaser above shows novel view synthesis from different recent methods. *From left to right: Yoon et al. [1], Mildenhall et al. [2], Wiles et al. [3], and Choi et al. [4]. Images and videos courtesy of the respective authors.*
New edition of my book – almost done


© 2021 Richard Szeliski, Facebook

https://szeliski.org/Book
New edition of my book

1 Introduction
- What is computer vision?
- A brief history
- Book overview
- Sample syllabus
- Notation

2 Image formation
- Geometric primitives and transformations
- Photometric image formation
- The digital camera

3 Image processing
- Point operators
- Linear filtering
- Non-linear filtering
- Fourier transforms
- Pyramids and wavelets
- Geometric transformations

4 Model fitting and optimization
- Scattered data interpolation
- Variational methods and regularization
- Markov random fields

5 Deep learning
- Supervised learning
- Unsupervised learning
- Deep neural networks
- Convolutional networks
- More complex models

6 Recognition
- Instance recognition
- Image classification
- Object detection
- Semantic segmentation
- Video understanding
- Vision and language

7 Feature detection and matching
- Points and patches
- Edges and contours
- Contour tracking
- Lines and vanishing points
- Segmentation

8 Image alignment and stitching
- Pairwise alignment
- Image stitching
- Global alignment
- Compositing

9 Motion estimation
- Translational alignment
- Parametric motion
- Optical flow
- Layered motion

10 Computational photography
- Photometric calibration
- High dynamic range imaging
- Super-resolution and blur removal
- Image mosaicing and compositing
- Texture analysis and synthesis

11 Structure from motion and SLAM
- Geometric intrinsic calibration
- Pose estimation
- Two-frame structure from motion
- Multi-frame structure from motion
- Simultaneous localization and mapping (SLAM)

12 Depth estimation
- Epipolar geometry
- Sparse correspondence
- Dense correspondence
- Local methods
- Global optimization
- Deep networks
- Multi-view stereo
- Monocular depth estimation

13 3D reconstruction
- Shape from X
- Surface representations
- 3D scanning
- Point-based representations
- Volumetric representations
- Model-based reconstruction
- Recovering texture maps and albedos

14 Image-based rendering
- View interpolation
- Layered depth images
- Light fields and Lumigraphs
- Environment maps
- Neural rendering
Outline

• Multi-view stereo
• Image-Based Rendering
  • Lumigraphs, Light Fields, Sprites with Depth, and Layers
• Virtual Viewpoint Video
• 360° and 3D Video
• 3D Photos
• Reflections and transparency
• Neural rendering
Multi-view Stereo
View Interpolation

• Given two images with correspondences, *morph* (warp and cross-dissolve) between them [Chen & Williams, SIGGRAPH’93]

[Matthies,Szeliski,Kanade’88]
View Morphing

• Morph between pair of images using epipolar geometry [Seitz & Dyer, SIGGRAPH’96]
Video view interpolation
Interactive 3D video scenarios

• Sports events, e.g., CMU’s 30-camera “EyeVision” system at SuperBowl XXXV) and 2016

• Concert performances, plays, circus acts

• Games

• Instructional video, e.g., golf, skating, martial arts

• Interactive (Internet) video
Plane Sweep Stereo

• Sweep family of planes through volume

input image

virtual camera

composite

← projective re-sampling of \((X,Y,Z)\)

• each plane defines an image \(\Rightarrow\) composite homography
Plane Sweep Stereo

• For each depth plane
  • compute composite (mosaic) image — \textit{mean}
  • compute error image — \textit{variance}
  • convert to confidence and aggregate spatially

• Select winning depth at each pixel
Plane sweep stereo

- Re-order (pixel / disparity) evaluation loops

for every pixel, for every disparity compute cost
Image-Based Rendering
Computer Graphics

Output

Image

Synthetic Camera

Model

Richard Szeliski

Reflections on Image-Based Rendering
Computer Vision

Output

Model

Real Scene

Real Cameras
But, vision technology fails
...and so does graphics
Image-Based Rendering

Output

Image

Synthetic Camera

Images+Model

Real Scene

Real Cameras
-or-
Expensive Image Synthesis

Richard Szeliski
Lumigraph / Light Field [1996]

Outside convex space

Empty

4D

Stuff

Richard Szeliski

Reflections on Image-Based Rendering
Lumigraph – Capture

• Convert images into a solid 3D model

• Render from images and model
Lumigraph – Image Effects

Can model effects such as:

• parallax
• occlusion
• translucency
• refraction
• highlights
• reflections
Unstructured Lumigraph

• What if the images aren’t sampled on a regular 2D grid?
• Can still re-sample rays
• Ray weighting becomes more complex [Heigl et al., DAGM’99]
• Unstructured Lumigraph [Buehler et al., SIGGRAPH’2000]
• Deep blending [Hedman et al., SG Asia 2018]
• FVS [Riegler & Koltun, ECCV’2020]
Surface Light Fields

- [Wood et al, SIGGRAPH 2000]
- Turn 4D parameterization around:
  - image @ every surface pt.
- Leverage coherence:
  - compress radiance fn (BRDF * illumination) after rotation by $n$
Surface Light Fields

• [Wood et al, SIGGRAPH 2000]

• ...

• Implicit Differentiable Renderer [Yariv et al., NeurIPS 2020]
Environment Matting [2000]

Figure 1  Sample composite images constructed with the techniques of this paper: slow but accurate on the left, and a more restricted example acquired at video rates on the right.
Layered Depth Image

2.5 D?
Layered Depth Image

• Rendering from LDI
  [Shade et al., SIGGRAPH’98]

• Incremental in LDI X and Y
• Guaranteed to be in back-to-front order
Sprites with Depth

• Represent scene as collection of cutouts with depth (planes + parallax)
• Render back to front with fwd/inverse warping [Shade et al., SIGGRAPH’98]
• Basis of Virtual Viewpoint Video [Zitnick et al. 2004]
Multiplane images

Figure 14.7  Finely sliced fronto-parallel layers: (a) stack of acetates (Szeliski and Golland 1999) © 1999 Springer and (b) multiplane images (Zhou, Tucker, Flynn et al. 2018) © 2018 ACM.
Multiplane images

Input images

Inferred MPI Representation

A novel view synthesized from MPI
Multi-sphere and layered meshes

Immersive Light Field Video with a Layered Mesh Representation

MICHAEL BROXTON*, JOHN FLYNN*, RYAN OVERBECK*, DANIEL ERICKSON*, PETER HEDMAN, MATTHEW DUVALL, JASON DOURGARIAN, JAY BUSCH, MATT WHALEN, and PAUL DEBEVEC, Google

(a) Capture Rig  
(b) Multi-Sphere Image  
(c) Layered Mesh Representation

[SIGGRAPH’2020]
Virtual Viewpoint Video
Virtual Viewpoint Video [SIGGRAPH 2004]
Matting

Some pixels get influence for multiple surfaces.

Background Surface

Foreground Surface

Image

Camera

Close up of real image:

Multiple colors and depths at boundary pixels...
Find matting information:

1. Find boundary strips using depth.

2. Within boundary strips compute the colors and depths of the foreground and background object.
Why matting is important

No Matting

Matting
Virtual Viewpoint Video

Two-layer model with thin boundary strips
[Zitnick et al., SIGGRAPH’04]

Main Layer: Boundary Layer:

- Color
- Depth
- Color
- Depth
- Alpha

Richard Szeliski - Reflections on Image-Based Rendering
Massive Arabesque
360° Video
360 Video

Ladybug (six-camera head)

[Uyttendaele et al. 2004]
Acquisition platforms (today)
360 Video
360 Video

Richard Szeliski

Reflections on Image-Based Rendering
Google Jump [2015]
Facebook Surround 360 [2016]

Cameras
- Resolution 2048x2048 - 41 megapixel
- Frame rate 60fps max - Sensor format 1"
- Interface USB3.0

Fisheye Lens
- Fixed focus, focal length 2.7mm
- Manual iris, Iris range F/8 - F/16
- Angular FOV 180° - 195° (Ø 86 mm)

Wide Angle Lens
- Fixed focus, focal length 7mm
- Fixed iris F2.4
- Angular FOV 77°
Facebook Surround 360 [2017]

Facebook’s new Surround 360 video cameras let you move around inside live-action scenes

The freedom of VR with the fidelity of real life

By Nick Statt | @nickstatt | Apr 19, 2017, 1:15pm EDT

Facebook today announced the second generation of its Surround 360 video camera design, and this time the company is serious about helping potential customers purchase it as an actual product. The Surround 360, which Facebook unveiled last year as an open-source spec guide for others to build off of, has been upgraded as both a larger, more capable unit and a smaller, more portable version.
An Integrated 6DoF Video Camera and System Design

ALBERT PARRA POZO, MICHAEL TOKSVIG, TERRY FILIBA SCHRAGER, and JOYCE HSU, Facebook Inc.
UDAY MATHUR, RED Digital Cinema
ALEXANDER SORKINE-HORNUNG, RICK SZELESKI, and BRIAN CABRAL, Facebook Inc.

Fig. 1. The commercial 16 camera system, an equirectangular depth map, and final color rendering produced from our system.

[Video] [SIGGRAPH Asia 2019]
Hemispherical light field capture & playback

(a) Capture Rig
(b) Multi-Sphere Image
(c) Layered Mesh Representation

IMMERSIVE LIGHT FIELD VIDEO WITH A LAYERED MESH REPRESENTATION

SIGGRAPH 2020 Technical Paper
Download PDF

Michael Brockton*, John Rynn*, Ryan Overbeck*, Daniel Erickson*, Peter Hedman, Matthew DuVall, Jason Dourgarian, Jay Busch, Matt Whalen, Paul Debevec

Richard Szeliski
Reflections on Image-Based Rendering
Stereo from *two* 360 cameras

Immersive Video Stabilization
First-person Hyperlapse

Create buttery-smooth “fast forwards” from action videos

[Kopf, Cohen, Szeliski, SIGGRAPH 2014]
Proxy Geometry
(for a single video frame)
Spatio-Temporal MRF Stitch
Large-Scale Reconstruction
Photo Tourism

Internet images

Computed 3D structure

[Snavely, Seitz, Szeliski, SIGGRAPH 2006]
System overview

Input photographs

Scene reconstruction

Photo Explorer

Relative camera positions and orientations
Point cloud
Sparse correspondence
Navigation: Prague Old Town Square
Piecewise planar proxies

[60 images] → [Structure from motion] → [Reconstruct Lines + Detect Multiple Planes] → [Piecewise planar depth-map]

[Sinha, Steedly, Szeliski ICCV’09]
Photo Tours - 2012

[Trevi Fountain]

[Kushal et al., 3DIMPVT 2012]
The Visual Turing Test - 2013

[Shan et al., 3DV 2013]
Casual 3D Photography

Peter Hedman, Suhib Alsisan, Richard Szeliski, Johannes Kopf
SIGGRAPH Asia 2017
Figure 2: A breakdown of the 3D photo reconstruction algorithm into its six stages, with corresponding inputs and outputs: (a) Capture and pre-processing, Sec. 4.1; (b) Sparse reconstruction, Sec. 4.2; (c) Dense reconstruction, Sec. 4.3; (d) Warping into a central panorama, Sec. 4.4.1; (e) Parallax-tolerant Stitching, Sec. 4.4.2; (f) Two-layer fusion, Sec. 4.4.3.
Casual 3D Photography

(a) Front color-and-depth panorama
(b) Front detail
(c) Back detail
Casual 3D Photography

- Forest Rock
- Creepy Attic
- Gymnasium
- Gas Works Park
- Boat Shed
- Church
- Jakobstad Museum
- Water Tower
- Library
- Pike Place
- Gum Wall
- British Museum

360° × 180° scenes captured with DSLR cameras

- Sofa
- Cafe
- Troll
- Gravity
- Kitchen
- Clowns
- Kerry Park

Partial scenes captured with DSLR cameras

Partial scenes captured with cell phone cameras
Instant 3D Photography

Peter Hedman  
University College London

Johannes Kopf  
Facebook

* This work was done while Peter was working as a contractor for Facebook.

Our work enables practical and casual 3D capture with regular dual camera cell phones. Left: A burst of input color-and-depth image pairs that we captured with a dual camera cell phone at a rate of one image per second. Right: 3D panorama generated with our algorithm in about the same time it took to capture. The geometry is highly detailed and enables viewing with binocular and motion parallax in VR, as well as applying 3D effects that interact with the scene, e.g., through occlusions (right).
Practical 3D Photography

Johannes Kopf  Ocean Quigley
Suhib Alsisan  Josh Patterson
Francis Ge  Jossie Tirado
Yangming Chong  Shu Wu
Kevin Matzen  Michael F. Cohen

Figure 1. 3D Photo Creation. Runtime measured on iPhone X.
3D Photos on Facebook

Estimate depth map from photo to create an interactive animation
3D Photos on Facebook

Estimate depth map from photo to create an interactive animation
3D Photos blog post

Powered by AI: Turning any 2D photo into 3D using convolutional neural nets

February 28, 2020 Written by Kevin Matzen, Matthew Yu, Jonathan Lehman, Peizhao Zhang, Jan-Michael Frahm, Peter Vajda, Johannes Kopf, Matt Uyttendaele

One Shot 3D Photography

JOHANNES KOPF, KEVIN MATZEN, SUHIB ALSISAN, OCEAN QUIGLEY, FRANCIS GE, YANGMING CHONG, JOSH PATTERTSON, JAN-MICHAEL FRAHM, SHU WU, MATTHEW YU, PEIZHAO ZHANG, ZIJIAN HE, PETER VAJDA, AYUSH SARAF, and MICHAEL COHEN, Facebook

(a) Input
(b) Depth estimation (230 ms)
(c) Layer generation (94 ms)
(d) Color inpainting (540 ms)
(e) Meshing (234 ms)
(f) Novel view (real-time)

Processing: 1,098ms on a mobile phone (iPhone 11 Pro)

[SIGGRAPH 2020]
3D Photography using Context-aware Layered Depth Inpainting
CVPR’2020
Google Photos cinematic effect

Jamie Aspinall
Product Manager, Google Photos

Published Dec 15, 2020

Relive the moment with Cinematic photos

Cinematic photos help you relive your memories in a way that feels more vivid and realistic—so you feel like you’re transported back to that moment. To do this, we use machine learning to predict an image’s depth and produce a 3D representation of the scene—even if the original image doesn’t include depth information from the camera. Then we animate a virtual camera for a smooth panning effect—just like out of the movies.

https://blog.google/products/photos/new-cinematic-photos-and-more-ways-relive-your-memories/
What’s missing?
Reflections and Transparency
Image-Based Rendering with Reflections

• Reflections, gloss, and highlights are everywhere

• How do these affect image-based modeling / rendering?
  [Sinha et al., SIGGRAPH 2012]
Standard IBR with Reflections
Our New Rendering System
Front Depth

Rear Depth

Input
Input

Front Layer

Rear Layer
Image-Based Rendering in the Gradient Domain

• Wrong depth for textureless or transparent areas

• Solve by reconstructing depth at gradients and re-integrating

[Kopf et al. SIGGRAPH Asia 2013]
Overview

Input

Preprocessing

Gradient domain rendering

Integration
Gradient Domain
Our Method

Standard IBR  Our IBR
A Computational Approach for Obstruction-Free Photography

Tianfan Xue\textsuperscript{1}\textsuperscript{*}, Michael Rubinstein\textsuperscript{2}\textsuperscript{*}, Ce Liu\textsuperscript{2}\textsuperscript{*}, William T. Freeman\textsuperscript{1,2}

\textsuperscript{1}MIT CSAIL  \hspace{1cm} \textsuperscript{2}Google Research

\textsuperscript{*} Part of this work was done while Michael Rubinstein and Ce Liu were at Microsoft Research, and when Tianfan Xue was an intern at Microsoft Research New England.
Video Reflection Removal Through Spatio-Temporal Optimization

Ajay Nandoriya\textsuperscript{1}, Mohamed Elgarib\textsuperscript{1}, Changil Kim\textsuperscript{2}, Mohamed Hefeeda\textsuperscript{3}, and Wojciech Matusik\textsuperscript{2}

\textsuperscript{1}Qatar Computing Research Institute, HBKU  \textsuperscript{2}MIT CSAIL  \textsuperscript{3}Simon Fraser University

[ICCV 2017]
Reflection Removal Using a Dual-Pixel Sensor

Abhijith Punnapurath  
York University  
pabhijith@eecs.yorku.ca

Michael S. Brown  
York University  
mbrown@eecs.yorku.ca

Abstract

Reflection removal is the challenging problem of removing unwanted reflections that occur when imaging a scene that is behind a pane of glass. In this paper, we show that most cameras have an overlooked mechanism that can greatly simplify this task. Specifically, modern DSLR and smartphone cameras use dual pixel (DP) sensors that have two photodiodes per pixel to provide two sub-aperture views of the scene from a single captured image. “Defocus-disparity” cues, which are natural by-products of the DP sensor encoded within these two sub-aperture views, can be used to distinguish between image gradients belonging to the in-focus background and those caused by reflection interference. This gradient information can then be incorporated into an optimization framework to recover the background layer with higher accuracy than currently possible from the single captured image. As part of this work, we provide the first image dataset for reflection removal consisting of the sub-aperture views from the DP sensor.

[CVPR 2019]
Open issues

• Improve stereo matching
  • Plane + parallax representation
• Reflectivity (β) estimation
  • Iterative Refinement
• Handle distorted reflections
  • [ See next slide ]
• Model real-valued reflectivity
  • Fresnel reflection
Real-World Normal Map Capture for Nearly Flat Reflective Surfaces

Bastien Jacquet¹, Christian Häne¹, Kevin Köser¹2*, Marc Pollefeys¹

ETH Zürich¹
Zürich, Switzerland

GEOMAR Helmholtz Centre for Ocean Research²
Kiel, Germany

Abstract

Although specular objects have gained interest in recent years, virtually no approaches exist for markerless reconstruction of reflective scenes in the wild. In this work, we present a practical approach to capturing normal maps in real-world scenes using video only. We focus on nearly planar surfaces such as windows, facades from glass or metal, or frames, screens and other indoor objects and show how normal maps of these can be obtained without the use of an artificial calibration object. Rather, we track the reflections of real-world straight lines, while moving with a hand-held

Figure 1. Real-world glass reflection. Notice that reflection in different windows on the same facade can appear very different due to minor deformations and normal variations. Our goal is to capture normal maps of real windows to faithfully reproduce this effect.
Neural Rendering
Richard Szeliski

Reflections on Image-Based Rendering
Richard Szeliski

Reflections on Image-Based Rendering
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
<th>Speaker(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00-09:15</td>
<td>Welcome and Introduction</td>
<td>Michael Zollhofer</td>
</tr>
<tr>
<td>09:15-09:30</td>
<td>Fundamentals, Taxonomy, Neural Rendering</td>
<td>Ayush Tevani</td>
</tr>
<tr>
<td></td>
<td>Semantic Photo Synthesis and Manipulation</td>
<td></td>
</tr>
<tr>
<td>09:30-09:40</td>
<td>Overview</td>
<td>Jun-Yan Zhu</td>
</tr>
<tr>
<td>09:40-10:00</td>
<td>Semantic Image Synthesis with Spatially-Adaptive Normalization</td>
<td>Taesung Park</td>
</tr>
<tr>
<td>10:00-10:30</td>
<td>Coffee Break</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Facial Reenactment &amp; Body Reenactment</td>
<td></td>
</tr>
<tr>
<td>10:25-10:35</td>
<td>Overview</td>
<td>Justus Thies</td>
</tr>
<tr>
<td>10:35-11:00</td>
<td>Neural Rendering for High-Quality Synthesis of Human Portrait Video and Images</td>
<td>Christian Theebald</td>
</tr>
<tr>
<td>11:00-11:20</td>
<td>Neural Rendering for Virtual Avatars</td>
<td>Alaksandra Shyshyka</td>
</tr>
<tr>
<td>11:20-11:35</td>
<td>Overview</td>
<td>Vincent Sitzmann</td>
</tr>
<tr>
<td>11:30-11:50</td>
<td>Neural Rendering in the Wild</td>
<td>Moustafa Meshry</td>
</tr>
<tr>
<td>11:50-12:10</td>
<td>NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis</td>
<td>Ben Mildenhall</td>
</tr>
<tr>
<td>12:10-13:20</td>
<td>Lunch Break</td>
<td></td>
</tr>
<tr>
<td>13:20-13:30</td>
<td>Overview</td>
<td>Zexiang Xu</td>
</tr>
<tr>
<td>13:30-13:50</td>
<td>Multi-view Relighting Using a Geometry-Aware Network</td>
<td>Julien Philip</td>
</tr>
<tr>
<td>13:50-14:10</td>
<td>Neural Inverse Rendering</td>
<td>Abhimitra Meka</td>
</tr>
<tr>
<td></td>
<td>Free Viewpoint Videos</td>
<td></td>
</tr>
<tr>
<td>14:10-14:20</td>
<td>Overview</td>
<td>Sean Fanello</td>
</tr>
<tr>
<td>14:20-14:40</td>
<td>Neural Rendering for Performance Capture</td>
<td>Rohit K. Pandey</td>
</tr>
<tr>
<td>14:40-15:00</td>
<td>Neural Volumes: Learning Dynamic Renderable Volumes from Images</td>
<td>Stephen Lombardi</td>
</tr>
<tr>
<td>15:00-15:30</td>
<td>Coffee Break</td>
<td></td>
</tr>
<tr>
<td>15:30-15:45</td>
<td>Social Implications, Open Challenges, Conclusion</td>
<td>Ohad Fried</td>
</tr>
<tr>
<td>15:45-16:15</td>
<td>Followup Discussion</td>
<td></td>
</tr>
</tbody>
</table>
3D representations for neural rendering

- 3D models & textures
- Depth images and layers
- Voxels
- Implicit functions (MLPs)

Richard Szeliski
Reflections on Image-Based Rendering
SynSin: view synthesis from a single image

SynSin: End-to-end View Synthesis from a Single Image

Olivia Wiles\textsuperscript{1*}  Georgia Gkioxari\textsuperscript{2}  Richard Szeliski\textsuperscript{3}  Justin Johnson\textsuperscript{2,4}

\textsuperscript{1}University of Oxford  \textsuperscript{2}Facebook AI Research  \textsuperscript{3}Facebook  \textsuperscript{4}University of Michigan

Figure 1: End-to-end view synthesis. Given a single RGB image (red), SynSin generates images of the scene at new viewpoints (blue). SynSin predicts a 3D point cloud, which is projected onto new views using our differentiable renderer; the rendered point cloud is passed to a GAN to synthesise the output image. SynSin is trained end-to-end, without 3D supervision.
SynSin: view synthesis from a single image

Figure 2: **Our end-to-end system.** The system takes as input an image $I$ of a scene and change in pose $T$. The *spatial feature predictor* ($f$) learns a set of features $F$ (visualised by projecting features using PCA to RGB) and the *depth regressor* ($d$) a depth map $D$. $F$ are projected into 3D (the diagram shows RGB for clarity) to give a point cloud $\mathcal{P}$ of features. $\mathcal{P}$ is transformed according to $T$ and rendered. The rendered features $\tilde{F}$ are passed through the *refinement network* ($g$) to generate the final image $I_G$. $I_G$ should match the target image, which we enforce using a set of discriminators and photometric losses.
SynSin: view synthesis from a single image
Animating Pictures

Animating Pictures with Eulerian Motion Fields

Aleksander Holynski¹, Brian Curless¹, Steven M. Seitz¹, Richard Szeliski²
¹University of Washington, ²Facebook

https://eulerian.cs.washington.edu/

(a) Input image  (b) Output looping video

https://eulerian.cs.washington.edu/
Figure 2: **Overview:** Given an input image $I_0$, our motion estimation network predicts a motion field $M$. Through Euler integration, $M$ is used to generate future and past displacement fields $F_{0\rightarrow t}$ and $F_{0\rightarrow t-N}$, which define the source pixel locations in all other frames $t$. To animate the input image using our estimated motion, we first use a feature encoder network to encode the image as a feature map $D_0$. This feature map is warped by the displacement fields (using a novel symmetric splatting technique) to produce the corresponding warped feature map $D_t$. The warped features are provided to the decoder network to create the output video frame $I_t$. 

"Richard Szeliski  
Reflections on Image-Based Rendering"
Animating Pictures

Animating Pictures with Eulerian Motion Fields

Aleksander Holynski  
University of Washington

Brian Curless  
University of Washington

Steven M. Seitz  
University of Washington

Richard Szeliski  
Facebook
Figure 2: **Overview:** Given an input image $I_0$, our motion estimation network predicts a motion field $M$ that is used to generate future and past displacement fields $F_{k-1}$ and $F_{k+1}$, which define the source pixels. To animate the input image using our estimated motion, we first use a feature encoder network to encode the feature map $D_0$. This feature map is warped by the displacement fields (using a novel symmetric splatting technique) to obtain the feature map $D_t$. The warped features are provided to the decoder network to create the output video frame $I_t$.

Figure 5: **Training:** As described in Section 5.1, each frame in our generated looping video is composed of textures from two warped frames. To supervise this process during training, i.e., to have a real frame to compare against, we perform our symmetric splatting using the features from two different frames, $I_0$ and $I_N$ (instead of $I_0$ twice, as in inference). We enforce the motion field $M$ to match...
… wrapping up …
Outline

• Multi-view stereo
• Image-Based Rendering
  • Lumigraphs, Light Fields, Sprites with Depth, and Layers
• Virtual Viewpoint Video
• 360° and 3D Video
• 3D Photos
• Reflections and transparency
• Neural rendering
Thank you