Lecture 26:

Intro to Virtual Reality (Cont)

Computer Graphics and Imaging
UC Berkeley CS184/284A
Display Requirements Derive From Human Perception

Example #4: Motion Parallax from Eye Motion
The 5D Plenoptic Function

\[ P(x, y, z, \theta, \phi) \]

3D Position

2D Direction

[Adelson, Bergen 1991]
Google Cardboard: Tracking Using Headset Camera

Tracking uses gyro / rear-facing camera to estimate user’s viewpoint

- 2D rotation tracking generally works well
- 3D positional tracking a challenge in general environments
Environment-Supported Vision-Based Tracking?

Early VR test room at Valve, with markers positioned throughout environment

Image credit: gizmodo.com
Oculus Rift IR LED Tracking System

Oculus Rift + IR LED sensor
Oculus Rift IR LED Tracking Hardware

https://www.ifixit.com/Teardown/Oculus+Rift+Constellation+Teardown/61128

Photo taken with IR-sensitive camera

https://www.ifixit.com/Teardown/Oculus+Rift+Constellation+Teardown/61128
Oculus Rift IR Camera

- IR filter
  - (blocks visible spectrum)
- Camera lens
- CMOS sensor
  - Note: silicon is sensitive to visible and IR wavelengths

https://www.ifixit.com/Teardown/Oculus+Rift+Constellation+Teardown/61128
Recall: Passive Optical Motion Capture

- Markers on subject
- Positions by triangulation from multiple cameras
- 8+ cameras, 240 Hz, occlusions are difficult

Slide credit: Steve Marschner
Active Optical Motion Capture

- Each LED marker emits unique blinking pattern (ID)
- Reduce marker ambiguities / unintended swapping
- Have some lag to acquire marker IDs
Oculus Rift Uses Active Marker Motion Capture

• Motion capture: unknown shape, multiple cameras
• VR head tracking: known shape, single camera

Credit: Oliver Kreylos, https://www.youtube.com/watch?v=O7Dt9Im34OI
6 DOF Head Pose Estimation

Head pose: 6 degrees of freedom (unknowns)

- 3D position and 3D rotation of headset (e.g. can represent as 4x4 matrix)

Inputs:

- Fixed: relative 3D position of markers on headset (e.g. can represent each marker offset as 4x4 matrix)
- Fixed: camera viewpoint (ignoring distortion, also a 4x4 projective mapping of 3D scene to 2D image)
- Each frame: 2D position of each headset marker in image

Pose calculation:

- Write down equations mapping each marker to image pixel location as a function of 6 degrees of freedom
- Solve for 6 degrees of freedom (e.g. least squares)
HTC Vive Tracking System ("Lighthouse")

Structured light transmitter

Photodiode arrays on headset and hand-held controllers
Vive Headset & Controllers Have Array of IR Photodiodes

(Prototype) Headset and controller are covered with IR photodiodes
HTC Vive Structured Light Emitter ("Lighthouse")

Light emitter contains array of LEDs (white) and two spinning wheels with lasers

Sequence of LED flash and laser sweeps provide structured lighting throughout room

Credit: Gizmodo: http://gizmodo.com/this-is-how-valve-s-amazing-lighthouse-tracking-technol-1705356768
HTC Vive Tracking System

For each frame, lighthouse does the following:

- LED pulse, followed by horizontal laser sweep
- LED pulse, followed by vertical laser sweep

Each photodiode on headset measures time offset between pulse and laser arrival

- Determines the x and y offset in the lighthouse’s field of view
- In effect, obtain an image containing the 2D location of each photodiode in the world
  - (Can think of the lighthouse as a virtual “camera”)
HTC Vive Tracking System ("Lighthouse")

Credit: rvdm88 / youtube. https://www.youtube.com/watch?v=J54dotTt7k0
Tracking Summary

Looked at three tracking methods

- Camera on headset + computer vision + gyro
- External camera + marker array on headset
- External structured light + sensor array on headset

3D tracking + depth sensing an active research area

- SLAM, PTAM, DTAM…
- Microsoft Hololens, Magic Leap, Google Tango, Intel Realsense, …
Overview of VR Topics

• VR Displays

• VR Rendering

• VR Imaging
Rendering Latency in VR
Resolution Requirements in VR Are Very High

Future “retina” VR display:
57 ppd covering 200°
= 11K x 11K display per eye
= 220 MPixel

Human: ~160° view of field per eye (~200° overall)
(Note: does not account for eye’s ability to rotate in socket)

iPhone 6: 4.7 in “retina” display:
1.3 MPixel
326 ppi → 57 ppd
Latency Requirements in VR Are Very Low

The goal of a VR graphics system is to achieve “presence”, tricking the brain into thinking what it is seeing is real.

Achieving presence requires an exceptional low-latency system:

- What you see must change when you move your head!
- End-to-end latency: time from moving your head to the time new photons hit your eyes
  - Measure user’s head movement
  - Update scene/camera position
  - Render new image
  - Transfer image to headset, then transfer to display in headset
  - Actually emit light from display (photons hit user’s eyes)
- Latency goal of VR: 10-25 ms
  - Requires exceptionally low-latency head tracking
  - Requires exceptionally low-latency rendering and display
Thought Experiment: Effect of Latency

Consider 1,000 x 1,000 display spanning 100° field of view
• 10 pixels per degree

Assume:
• You move your head 90° in 1 second (only modest speed)
• End-to-end latency of system is a slow 50 ms (1/20 sec)

Result:
• Displayed pixels are off by 4.5° ~ 45 pixels from where they would be in an ideal system with 0 latency

Example credit: Michael Abrash
Challenge:
Low Latency and High Resolution Require High Rendering Speed
Recall: Retinal Resolution Falls Away from Fovea

- Highest density of cones in fovea (and no rods there)
- "Blind spot" at the optic disc, where optic nerve exits eye

[Roorda 1999]
Foveated Rendering

Idea: track user’s gaze, render with increasingly lower resolution farther away from gaze point

Three images blended into one for display
Foveated Rendering - Perceptual Effects

Patney et al., Towards Foveated Rendering for Gaze-Tracked Virtual Reality
SIGGRAPH Asisa 2016.
Foveated Rendering - Perceptual Effects

CONTEMPORARY FOVEATED RENDERING

Improves Performance by Reducing Peripheral Image Resolution
Effective for moderate foveation; artifacts at higher levels
(Foveation exaggerated for demonstration)

Patney et al., Towards Foveated Rendering for Gaze-Tracker Virtual Reality
SIGGRAPH Asia 2016.
Foveated Rendering - Perceptual Effects

TEMPORALLY-STABLE FOVEATED RENDERING

Smooth Peripheral Blur
Eliminates flicker in periphery, but creates "tunnel-vision" effect
(Foveation exaggerated for demonstration)

Patney et al., Towards Foveated Rendering for Gaze-Tracked Virtual Reality
SIGGRAPH Asisa 2016.
Foveated Rendering - Perceptual Effects

Contrast-Preserving vs Foveated Rendering

Avoids "tunnel-vision" in periphery
(Foveation exaggerated for demonstration)

Patney et al., Towards Foveated Rendering for Gaze-Tracker Virtual Reality
SIGGRAPH Asisa 2016.
Foveated Rendering

Perceptual considerations:

• If we render low resolution in periphery, have to be careful of aliasing / flickering

• If we render with a smooth image blur in the periphery, users experience a “tunnel vision” effect

• Research indicates that we should boost the contrast of low-frequency content in the periphery
Challenge: Distortion in VR Rendering
Requirement: Wide Field of View

View of checkerboard through Oculus Rift (DK2) lens

Lens introduces distortion

- Pincushion distortion
- Chromatic aberration (different wavelengths of light refract by different amount)

Icon credit: Eyes designed by SuperAtic LABS from the thenounproject.com
Image credit: Cass Everitt
Recall Software Correction of Lens Distortion in Photography

Pincushion distortion

Barrel distortion

Rectilinear

m43photo.blogspot.com

Credit: The Photoshop Creative Team
http://blog.photoshopcreative.co.uk
Software Compensation of Lens Distortion in VR Rendering

Step 1: Render scene using traditional graphics pipeline at full resolution for each eye
Step 2: Warp images in manner that scene appears correct after physical lens distortion
   (Can use separate distortions to R, G, B to approximately correct chromatic aberration)

Image credit: Oculus VR developer guide
Related Challenge: Rendering via Planar Projection

Recall: rasterization-based graphics is based on perspective projection to plane
• Distorts image under high FOV, as needed in VR rendering
• Recall: VR rendering spans wide FOV

Potential solution space: curved displays, ray casting to achieve uniform angular resolution, rendering with piecewise linear projection plane (different plane per tile of screen)

Image credit: Cass Everitt
Challenge: Eye Motion And Finite Rendering Rate
Consider Finite VR Display Refresh Rate

Reality (continuous)

VR (discrete display refresh)

- Red object fixed;
- Eye gaze fixed

- Light from display
  (light updates every frame)

Spacetime diagrams adopted from presentations by Michael Abrash
Eyes designed by SuperAtic LABS from the thenounproject.com
Case 2: Object Moving Relative to Eye

Reality (continuous)

VR (discrete display refresh)

- Red object moving left to right;
- Eye gaze fixed

- Effect: time discretization
- OK: same perceptual effect as on regular 2D displays

Spacetime diagrams adopted from presentations by Michael Abrash
Eyes designed by SuperAtic LABS from the thenounproject.com
Case 3: Eye Moving to Track Moving Object

- Red object moving left to right;
- Eye gaze moving left to right to track object
- Eye is moving continuously relative to display
- During each frame, image of object falls behind eye
- Result: smearing/strobing effect ("judder")

Spacetime diagrams adopted from presentations by Michael Abrash
Eyes designed by SuperAtic LABS from the thenounproject.com
Reducing Judder: Increase Frame Rate

- Higher frame rate (right-most diagram)
  - Closer approximation of ground truth

Spacetime diagrams adopted from presentations by Michael Abrash
Reducing Judder: Low Persistence Display

Continuous ground truth

Original judder

Low-persistence display

Low-persistence display: pixels emit light for small fraction of frame

• Oculus DK2 OLED low-persistence display:
  • 75 Hz frame rate = ~13 ms per frame
  • Pixel persistence = 2-3 ms

Spacetime diagrams adopted from presentations by Michael Abrash
Near-Future VR Rendering System Components

- Low-latency image processing for subject tracking
- Massive parallel computation for high-resolution rendering
- Exceptionally high bandwidth connection between renderer and display: e.g., 4K x 4K per eye at 90 fps!
- High-resolution, high-frame rate, wide-field of view display
- In headset motion/accel sensors + eye tracker
- On headset graphics processor for sensor processing and re-projection
Overview of VR Topics

Areas we will discuss over next few lectures

- VR Displays
- VR Rendering
- VR Imaging
Attendance Time

If you are seated in class, go to this form and sign in:

- https://tinyurl.com/184lecture

Notes:

- Time-stamp will be taken when you submit form. Do it now, won’t count later.
- Don’t tell friends outside class to fill it out now, because we will audit at some point in semester.
- Failing audit will have large negative consequence. You don’t need to, because you have an alternative!
Spherical Imaging
(Monocular 360)
Dual Fisheye
Stitching Challenges

Want this ray sampling

Get this ray sampling
# Spherical Array of Cameras

![Panono 360 degree Camera](image)

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DIAMETER</td>
<td>11 cm</td>
</tr>
<tr>
<td>WEIGHT</td>
<td>approximately 480 g</td>
</tr>
<tr>
<td>CAMERAS</td>
<td>36 fixed-focus cameras</td>
</tr>
<tr>
<td>RESOLUTION</td>
<td>108 megapixels</td>
</tr>
<tr>
<td>PANONO APP</td>
<td>iOS 7+ and Android 4.2+</td>
</tr>
<tr>
<td>CHARGING</td>
<td>via USB cable</td>
</tr>
<tr>
<td>STORAGE CAPACITY</td>
<td>16 GB, approximately 600 Panono shots</td>
</tr>
<tr>
<td>CONNECTION</td>
<td>WiFi</td>
</tr>
<tr>
<td>SECURITY FEATURES</td>
<td>Theft protect mode</td>
</tr>
</tbody>
</table>

**Panono 360 degree Camera**
Stitching Challenges

Want this ray sampling

Get this ray sampling
High Quality Stitching Solution Uses Computer Vision

Use computer vision techniques:

• Detect image features (like SIFT features)
• Correlate features across frames (transform)
• Warp to align frames and blend
Spherical Stereo Imaging
What Pairs of Viewpoint Positions Do We Want To Sample?
Idea: Spin a Pair of Cameras About Midpoint

Store a set of movie pairs (one per angle)
But that’s a lot of data

Image Credit: Google Inc.
Omni-Directional Stereo Approximation

full-frame left and right eyes

ODS-approximated left and right eyes

Image Credit: Google Inc.
Omni-Directional Stereo Approximation

Extended to be omnidirectional

Image Credit: Google Inc.
Spinning Camera

Concentric Mosaics
Shum and He, SIGGRAPH 1999
Omni-Directional Stereo Representation

Encode left/right views as just two spherical images

- Render left and right views for each angular view independently, with regular viewing software
- Efficient and compact, but this is an approximation
  - Straight lines may appear slightly curved
  - Vertical disparity for close objects incorrect
Example (Rendered)

Left Eye

Right Eye
Two Eyes — Two Spherical Cameras?

Matzen et al. SIGGRAPH 2017
Low-Cost 360 Stereo Photography and Video Capture
Problem: Stereo Baseline Fluctuates With View Angle

Apparent stereo baseline decreases by $\cos(\theta)$ if rays are mapped directly

Matzen et al. SIGGRAPH 2017
Problem: Both Horizontal and Vertical Disparities Fluctuate

Matzen et al. SIGGRAPH 2017
Problems

• Disparity: incorrect baseline as view angle changes
• Occlusion: each camera blocks the other’s view!
Solution: Computational Photography

3D reconstruction
- Computer vision on stereo views

Disparity correction
- Use 3D model to correct stereo disparities
  - e.g. amplify horizontal disparities by $1/\cos(\theta)$
- Flip views when facing backwards

Hole filling
- Cut out view of other camera, and fill hole with pixels from other camera, as best possible

Matzen et al. SIGGRAPH 2017
Spherical Stereo Result

Matzen et al. SIGGRAPH 2017
Moving-Viewpoint Imaging
(Full Plenoptic Function?)
The 5D Plenoptic Function

\[ P(\theta, \phi, V_x, V_y, V_z) \]
4D Light Field

In a region of free-space, 5D plenoptic function simplifies to 4D because light is constant along a ray.

\[ P(\theta, \phi, V_x, V_y) = P(u, v, s, t) \]
Light Field Capture Robot

Original light field rendering paper
Take photographs of an object from all points on an enclosing sphere
Captures all light leaving an object – like a hologram

\[ L(x, y, \theta, \phi) \]

[Levoy & Hanrahan 1996] [Gortler et al. 1996]
Multi-Camera Array ⇒ 4D Light Field

[Wilburn et al. SIGGRAPH 2005]
[Wilburn et al. SIGGRAPH 2005]
Handheld 4D Light Field Camera (Plenoptic Camera)

Lytro Gen-2 Light Field Camera
Handheld Light Field Camera vs Camera Array

Camera array: e.g. 10x10 views distributed across large planar support

Plenoptic camera: e.g. 14x14 views distributed across small lens pupil
Note: antialiased across views, unlike camera array
The Intimacy of VR Graphics

Google’s Tilt Brush on HTC Vive
A Challenge: Intimate Proximity in VR Imaging
How Dense Are Camera Views Today?

Multi-camera arrays: 50 - 100 views
Plenoptic cameras: 100 - 200 views
How Dense Must Cameras Views Be?
How Dense Must Camera Views Be?

\[ z_f = 0.6 \text{m} \]

Child in lap, front to back of head

\[ z_n = 0.3 \text{m} \]
How Dense Must Camera Views Be?

20/20 vision: \( \theta \approx (1/60)^\circ \)

Current HMDs: \( \theta \approx (1/10)^\circ \)
How Dense Must Camera Views Be?

Solving for minimum lateral motion:

\[
\Delta x = \frac{(z_f - z_n) - \sqrt{(z_f - z_n)^2 - 4 \tan^2 \theta z_n z_f}}{2 \tan \theta}
\]

- **20/20 vision:** \( \theta \approx \left(\frac{1}{60}\right)^\circ \) \( \Rightarrow \) \( \Delta x \approx \left(\frac{1}{1719}\right) \text{ ft} \)
- **Current HMDs:** \( \theta \approx \left(\frac{1}{10}\right)^\circ \) \( \Rightarrow \) \( \Delta x \approx \left(\frac{1}{286}\right) / \text{ ft} \)

20/20 vision: millions of views per square foot
Current HMDs: a hundred thousand views per square foot
How Dense Are Camera Views Today?

Multi-camera arrays:  50 - 100 views
Plenoptic cameras:    100 - 200 views
Google VR Camera Rig

Paul Debevec, Google
Open Problem: Capturing Dense Light Fields of Environments

Multi-Photo Scanning: Slow, Missing Views
Camera Array: Unwieldy, Missing Views
Handheld Plenoptic: Limited View Volume
Missing Views — Interpolate to Fill?
Missing Views — Interpolate to Fill?
Missing Views — Interpolate to Fill?
Missing Views — Interpolate to Fill?
Synthesize a 4D Light Field from A 2D Photo?
Learning to Synthesize
A 4D RGBD Light Field From a Single Image

ICCV 2017

Pratul Srinivasan, Tongzhou Wang,
Ashwinlal Sreelal, Ravi Ramamoorthi, Ren Ng
Database of 3000+ Light Field Shots of Flowers
Database of 3000+ Light Field Shots of Flowers
Training a Convolutional Neural Network

Figure 1. The convolutional neural network to estimate 4D ray depths from the input 2D image consists of 10 convolutional layers. We use dilated convolutions to enable each predicted ray depth to have access to the entire input image without the resolution loss caused by spatial downsampling or pooling. All filters are 3x3, and we use batch normalization and exponential linear unit activation functions for each layer except the last layer. The last layer is followed by a scaled tanh activation function to constrain the possible disparities.

- Depth estimation per ray
- Render views from depth + colors

Input 2D Image  Estimated 4D ray depths  Synthesized 4D light field
Input 2D Image
Rose: Estimated 4D Ray Depths
Rose: Synthesized 4D Light Field
Rose: True 4D Light Field
Rose

Input 2D Image  Synthesized Light Field  True Light Field
Things to Remember

VR presents many new graphics challenges!

Displays

• Head-pose tracking with high accuracy and low latency

Rendering

• Low-latency, high resolution & frame-rate, wide field of view, ...

Imaging

• 360 spherical, stereo, light field
Acknowledgments

Thanks to Kayvon Fatahalian, Alyosha Efros, Brian Wandell and Pratul Srinivasan for lecture resources and slides!