Computational Interaction for a Universally Accessible Metaverse

Ruofei Du (杜若飞)
Senior Research Scientist
Google Labs, San Francisco
www.ruofeidu.com
me@duruofei.com
Ruofei Du is a Senior Research Scientist at Google and works on creating novel interactive technologies for virtual and augmented reality. Du's research covers a wide range of topics in VR and AR, including depth-based interaction (DepthLab), mixed-reality social platforms (Geology and Social Street View), 4D video-based rendering (Montage4D and VideoFields), foveated rendering (KFR, EFR, Foveated360), and deep learning in graphics (HumanGPS and SketchColorization). Du served as a committee member in CHI, UIST, SIGGRAPH Asia XR, ICMI and an Associate Editor of Frontiers in Virtual Reality and IEEE TCSVT. Du holds a Ph.D. in Computer Science from University of Maryland, College Park. In their own words: I am passionate about inventing interactive technologies with computer graphics, 3D vision, and HCI. Feel free to visit my research, artsy projects, youtube, talks, github, and shader toy demos for fun!
Self Intro

Ruofei Du (杜若飞)

Human-Computer Interaction

- Geollery
  - CHI '19, Web3D '19, VR '19

- Kernel Foveated Rendering
  - I3D '18, VR '20, TVCG '20

- VideoFields
  - Web3D '16

- Social Street View
  - Web3D '16
  - Best Paper Award

- Montage4D
  - I3D '18
  - JCGT '19

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- LogRectilinear
  - IEEE VR '21 (TVCG)
  - TVCG Honorable Mention

- DepthLab
  - UIST '20
  - 13K Installs

- Ad hoc UI
  - CHIEA '22

- GazeChat
  - UIST '21

- ProtoSound
  - CHI '22

- CollaboVR
  - ISMAR '20

- HandSight
  - ECCVW '14
  - TACCESS '15

- Video Fields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- DepthLab
  - UIST '20
  - 13K Installs

- HumanGPS
  - CVPR' 21

- HumanGPS
  - CVPR' 21

- PRIF
  - ECCV' 22

- LogRectilinear
  - IEEE VR '21 (TVCG)
  - TVCG Honorable Mention

- VideoFields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- DepthLab
  - UIST '20
  - 13K Installs

- VideoFields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- DepthLab
  - UIST '20
  - 13K Installs

- VideoFields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- DepthLab
  - UIST '20
  - 13K Installs

- VideoFields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18

- DepthLab
  - UIST '20
  - 13K Installs

- VideoFields
  - Web3D '16

- SketchyScene
  - TOG (SIGGRAPH Asia) '19, ECCV '18
Self Intro
Ruofei Du (杜若飞)

Geollery
CHI '19, Web3D '19, VR '19

Social Street View
Web3D '16
Best Paper Award

OmniSyn
IEEE VR '22

LogRectilinear
IEEE VR '21 (TVCG)
TVCG Honorable Mention

VideoFields
Web3D '16

ProtoSound
CHI '22

GazeChat
UIST '21

HumanGPS
CVPR '21

SketchyScene
TOG (SIGGRAPH Asia) '19, ECCV '18

Montage4D
I3D '18
JCGT '19

CollaboVR
ISMAR '20

HandSight
ECCVW '14
TACCESS '15

Kernel Foveated Rendering
I3D '18, VR '20, TVCG '20

Ad hoc UI
CHIEA '22

DepthLab
UIST '20
13K Installs

SlurpAR
DIS '22

ISAMAR '20
元宇宙中的交互计算与包容普惠
Computational Interaction for a Universally Accessible Metaverse

Ruofei Du (杜若飞)
Senior Research Scientist
Google Labs, San Francisco
www.ruofeidu.com
me@duruofei.com
Metaverse


Snow Crash
Metaverse envisioned a **persistent** digital world where people are fully connected as **virtual representations**.

As a teenager, my dream was to live in a metaverse...

However, today I wish metaverse is only a tool to **make information more useful and accessible** and help people to **live a better physical life**.
元宇宙中的交互计算与包容普惠
Computational Interaction for a Universally Accessible Metaverse

第一章 · 镜像世界与实时渲染

Ruofei Du (杜若飞)
Senior Research Scientist
Google Labs, San Francisco
www.ruofeidu.com
me@duruofei.com
Project Geollery.com: Reconstructing a Live Mirrored World With Geotagged Social Media

Digital twin? Metaverse?

Greetings!

Hi, friends!

Hello!

Ruofei Du†, David Li†, and Amitabh Varshney
Introduction

Social Media
Motivation
Social Media + XR
Motivation
Social Media + XR

image courtesy:
instagram.com,
facebook.com,
twitter.com
Motivation

2D layout

image courtesy: pinterest.com
Motivation

Immersive Mixed Reality?

image courtesy:
viralized.com
Motivation

Pros and cons of the classic
Motivation

Pros and cons of the classic
Related Work

Social Street View, Du and Varshney
Web3D 2016 Best Paper Award
Related Work

Social Street View, Du and Varshney
Web3D 2016 Best Paper Award
Related Work

Social Street View, Du and Varshney
Web3D 2016 Best Paper Award
Related Work

3D Visual Popularity

Bulbul and Dahyot, 2017
Related Work

Virtual Oulu, Kukka et al.
CSCW 2017
Related Work

Immersive Trip Reports

Brejcha et al. UIST 2018
Related Work
Facebook Spaces, 2017
What's Next?
Research Question 1/3

What may a social media platform look like in mixed reality?
What's Next?
Research Question 2/3

What if we could allow social media sharing in a live mirrored world?
What use cases can we benefit from social media platform in XR?
Geollery.com
A Mixed-Reality Social Media Platform

- 3D buildings with 360° images
- Geotagged framed photos
- Geotagged street art
- Virtual avatars and live chats
- Geotagged virtual gifts
Conception, architecting & implementation

Geollery

A mixed reality system that can depict geotagged social media and online avatars with 3D textured buildings.
Extending the design space of

3D Social Media Platform

Progressive streaming, aggregation approaches, virtual representation of social media, co-presence with virtual avatars, and collaboration modes.
Conducting a user study of Geollery vs. Social Street View by discussing their benefits, limitations, and potential impacts to future 3D social media platforms.
System Overview
Geollery Workflow
System Overview

Geollery Workflow

2D polygons and metadata from OpenStreetMap

internal and external geotagged social media

virtual forms of social media: balloons, billboards, and gifts

shaded 3D buildings with 2D ground tiles

added avatars, clouds, trees, and day/night effects

Geollery fuses the mirrored world with geotagged data, street view 360° images, and virtual avatars.
Geollery.com
v2: a major leap

coarse detail
360° images
fine detail
building polygons
depth maps
System Overview

2D Map Data
System Overview
+Avatar +Trees +Clouds
System Overview

+Avatar +Trees +Clouds +Night
System Overview
Street View Panoramas
All data we used is publicly and widely available on the Internet.
Rendering Pipeline

Close-view Rendering

(a) initial spherical geometries  (b) depth correction  (c) intersection removal

(d) texturing individual geometry  (e) texturing with alpha blending  (f) rendering results in fine detail
Rendering Pipeline

Initial spherical geometries
Rendering Pipeline
Depth correction
Rendering Pipeline
Intersection removal
Texturing individual geometry
Rendering Pipeline
Texturing with alpha blending
Rendering Pipeline

Rendering result in the fine detail
Rendering Pipeline

Rendering result in the fine detail
User Study
Social Street View vs. Geollery
User Study
Quantitative Evaluation

Please compare the two systems and indicate the degree to which you agree with the following description. For example, for the first question, 4 is most immersive; 0 is most unengaging, 0 is neutral.

### Geology

<table>
<thead>
<tr>
<th></th>
<th>Unengaging</th>
<th>-4</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Straightforward</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Immersive</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

### Social Street View

<table>
<thead>
<tr>
<th></th>
<th>Unengaging</th>
<th>-4</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Immersive</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Straightforward</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Creative</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Creative</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Pleasant</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Practical</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Practical</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Simple</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Appealing</td>
<td>-4</td>
<td>-3</td>
<td>-2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
User Study
Quantitative Evaluation
I would like to use it for the food in different restaurants. I am always hesitating of different restaurants. It will be very easy to see all restaurants with street views. In Yelp, I can only see one restaurant at a time.
[I will use it for] exploring new places. If I am going on vacation somewhere, I could immerse myself into the location. If there are avatars around that area, I could ask questions.
I think it (Geollery) will be useful for families. I just taught my grandpa how to use Facetime last week and it would great if I could teleport to their house and meet with them, then we could chat and share photos with our avatars.
if there is a way to unify the interaction between them, there will be more realistic buildings [and] you could have more roof structures. Terrains will be interesting to add on.
What wonderful five years in Maryland!
Instant Panoramic Texture Mapping with Semantic Object Matching for Large-Scale Urban Scene Reproduction

TVCG 2021, Jinwoo Park, Ik-beom Jeon, Student Members, Sung-eui Yoon, and Woontack Woo

Fig. 2: Overview. In a pre-process, our system constructs 3D Scene Data, which contains five different input resources: street-view images, 3D models, estimated panoramic-segmentation images, synthetic panoramic-depth images, and refined extrinsic camera parameters. For sparse sampling of street-view images according to a user’s current position, 3D Scene Data is divided into smaller Districts. In a real-time process, pixels in a position buffer at a current view are projected onto sampled street-view images to get texture colors. After passing a depth test and a semantic object matching test, filtered candidate colors are properly blended as a final color. As a final step of complete scene rendering without visual holes, low-pass filtering and semantic 3D inpainting are applied to sky and non-sky areas respectively. Note that our main contributions lie on utilizing semantic information in the proper intermediate steps (marked in red boxes) to enhance both rendering quality and performance time.
OmniSyn: Intermediate View Synthesis Between Wide-Baseline Panoramas

David Li, Yinda Zhang, Christian Häne, Danhang Tang, Amitabh Varshney, and Ruofei Du, VR 2022
OmniSyn: Intermediate View Synthesis Between Wide-Baseline Panoramas

David Li, Yinda Zhang, Christian Häne, Danhang Tang, Amitabh Varshney, and Ruofei Du, VR 2022
Kernel Foveated Rendering

Xiaoxu Meng, Ruofei Du, Matthias Zwicker and Amitabh Varshney
Augmentarium | UMIACS
University of Maryland, College Park
ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games 2018
Kernel Log-polar Mapping

\[ u = K^{-1} \left( \log \left( x^2 + y^2 \right) \right) \cdot w \]
\[ v = \frac{\arctan \left( \frac{y}{x} \right) + 1 [y < 0] \cdot \pi}{2} \cdot h \]
\[ x = e^{u \cdot K(\frac{w}{h})} \cos \left( v \cdot \frac{2\pi}{h} \right) \]
\[ y = e^{u \cdot K(\frac{w}{h})} \sin \left( v \cdot \frac{2\pi}{h} \right) \]

- \( W \): screen width
- \( H \): screen height
- \( w \): buffer width
- \( h \): buffer width

- \( 1 [y < 0] \) = \( 1 \) if \( y < 0 \), \( 0 \) otherwise
- \( \tilde{I} = \log(W^2 + H^2) \)
- \( K(x) = \sum_{i=0}^{n} \beta_i x^i \), where \( \sum_{i=0}^{n} \beta_i = 1 \)
Eye-dominance-guided Foveated Rendering

Xiaoxu Meng, Ruofei Du, and Amitabh Varshney

IEEE Transactions on Visualization and Computer Graphics (TVCG)
A Log-Rectilinear Transformation for Foveated 360-Degree Video Streaming

David Li†, Ruofei Du†, Adharsh Babu†, Camelia Brumar†, Amitabh Varshney†

† University of Maryland, College Park  ‡ Google
Computational Interaction for a Universally Accessible Metaverse

Ruofei Du (杜若飞)
Senior Research Scientist
Google Labs, San Francisco
www.ruofeidu.com
me@duruofei.com
DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality

Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google | ACM UIST 2020
Introduction
Mobile Augmented Reality
Introduction

Google’s ARCore
Introduction

Google’s ARCore
Introduction
Mobile Augmented Reality
Is direct placement and rendering of 3D objects sufficient for realistic AR experiences?
Not always!
Virtual content looks like it’s “pasted on the screen” rather than “in the world”!
How can we bring these advanced features to mobile AR experiences without relying on dedicated sensors or the need for computationally expensive surface reconstruction?
Introduction

Depth Map
Google
- Pixel 2, Pixel 2 XL, Pixel 3, Pixel 3 XL, Pixel 3a, Pixel 3a XL, Pixel 4, Pixel 4 XL

Huawei
- Honor 10, Honor V20, Mate 20 Lite, Mate 20, Mate 20 X, Nova 3, Nova 4, P20, P30, P30 Pro

LG
- G8X ThinQ, V35 ThinQ, V50S ThinQ, V60 ThinQ 5G

OnePlus
- OnePlus 6, OnePlus 6T, OnePlus 7, OnePlus 7 Pro, OnePlus 7 Pro 5G, OnePlus 7T, OnePlus 7T Pro

Oppo
- Reno Ace

Samsung
- Galaxy A80, Galaxy Note8, Galaxy Note9, Galaxy Note10, Galaxy Note10 5G, Galaxy Note10+, Galaxy Note10+ 5G, Galaxy S8, Galaxy S8+, Galaxy S9, Galaxy S9+, Galaxy S10e, Galaxy S10, Galaxy S10+, Galaxy S10 5G, Galaxy S20, Galaxy S20+ 5G, Galaxy S20 Ultra 5G

Sony
- Xperia XZ2, Xperia XZ2 Compact, Xperia XZ2 Premium, Xperia XZ3

Xiaomi
- Pocophone F1

And growing…

https://developers.google.com/ar/discover/supported-devices
Is there *more* to realism than occlusion?
Surface interaction?
Realistic Physics?
Path Planning?
DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality

Ruofei Du, Eric Turner, Max Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Crues, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google | ACM UIST 2020
Introduction

Depth Lab
Related Work

Valentin et al.
Depth Maps
Depth from Motion
Best Practices

Use depth-certified ARCore devices

Minimal movement in the scene

Encourage users to move the device

Depth from 0 to 8 meters

Best accuracy 0.5 to 5 meters
Optimized to give you the best depth

Depth from Motion is fused with state-of-the-art Machine Learning

Depth leverages specialized hardware like a Time-of-Flight sensor when available
Introduction

Depth Lab
Introduction

Depth Generation
Introduction
Depth Lab
Related Work
Valentin et al.
Introduction

Depth Lab
Up to 8 meters, with the best within 0.5m to 5m
Motivation
Gap from raw depth to applications
Design Process
3 Brainstorming Sessions

- 3 brainstorming sessions
- 18 participants
- 39 aggregated ideas
Supplementary Material for DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality

Design Process
3 Brainstorming Sessions
System Architecture overview

Input
- Camera Image
- Phone Orientation
- Camera Parameters
  - focal length
  - intrinsic matrix
  - extrinsic matrix
  - projection matrix

DepthLab Data Structures and Utilities
- Conversion Utilities
  - screen uv/xy → depth
  - screen uv/xy ↔ world vertex
  - screen uv/xy → local normal
  - screen uv/xy → world normal
  - depth uv ↔ depth xy
  - screen uv ↔ screen xy

Depth Array
Depth Mesh
Depth Texture

DepthLab Algorithms
- Localized Depth
  - orientation
  - hit test
  - reflection
- Surface Depth
  - physics
  - texture decal
  - depth mesh
- Dense Depth
  - relighting
  - occlusion
  - aperture
2D array (160x120 and above) of 16-bit integers
Data Structure

Depth Mesh

(a) input depth map

(b) template mesh

(c) real-time depth mesh

winding order of the template mesh
Data Structure

Depth Texture
<table>
<thead>
<tr>
<th></th>
<th>Localized Depth</th>
<th>Surface Depth</th>
<th>Dense Depth</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU</strong></td>
<td>✓</td>
<td>✓</td>
<td>× (non-real-time)</td>
</tr>
<tr>
<td><strong>GPU</strong></td>
<td>N/A</td>
<td>✓ (compute shader)</td>
<td>✓ (fragment shader)</td>
</tr>
<tr>
<td><strong>Prerequisite</strong></td>
<td>point projection</td>
<td>depth mesh</td>
<td>anti-aliasing</td>
</tr>
<tr>
<td></td>
<td>normal estimation</td>
<td>triplanar mapping</td>
<td>multi-pass rendering</td>
</tr>
<tr>
<td><strong>Data Structure</strong></td>
<td>depth array</td>
<td>depth mesh</td>
<td>depth texture</td>
</tr>
<tr>
<td><strong>Example Use Cases</strong></td>
<td>physical measure</td>
<td>collision &amp; physics</td>
<td>scene relighting</td>
</tr>
<tr>
<td></td>
<td>oriented 3D cursor</td>
<td>virtual shadows</td>
<td>aperture effects</td>
</tr>
<tr>
<td></td>
<td>path planning</td>
<td>texture decals</td>
<td>occluded objects</td>
</tr>
</tbody>
</table>
Localized Depth
Coordinate System Conversion

Conversion Utilities

screen uv/xy → depth
screen uv/xy ↔ world vertex
screen uv/xy → local normal
screen uv/xy → world normal
depth uv ↔ depth xy
screen uv ↔ screen xy
\[ n_p = (v_p - v_{p+(1,0)}) \times (v_p - v_{p+(0,1)}) \]
Localized Depth
Normal Estimation

Algorithm 1: Estimation of the Normal Vector of a Screen Point in DepthLab.

Input : A screen point \( p \leftarrow (x, y) \) and focal length \( f \).
Output : The estimated normal vector \( n \).

1. Set the sample radius: \( r \leftarrow 2 \) pixels.
2. Initialize the counts along two axes: \( c_X \leftarrow 0, c_Y \leftarrow 0 \).
3. Initialize the correlation along two axes: \( \rho_X \leftarrow 0, \rho_Y \leftarrow 0 \).
4. for \( \Delta x \in [-r, r] \) do
   for \( \Delta y \in [-r, r] \) do
     Continue if \( \Delta x = 0 \) and \( \Delta y = 0 \).
     Set neighbor's coordinates: \( q \leftarrow [x + \Delta x, y + \Delta y] \).
     Set \( q \)'s distance in depth: \( d_{pq} \leftarrow \|D(p), D(q)\| \).
     Continue if \( d_{pq} = 0 \).
     if \( \Delta x \neq 0 \) then
       \( c_X \leftarrow c_X + 1 \).
       \( \rho_X \leftarrow \rho_X + d_{pq} / \Delta x \).
     end
     if \( \Delta y \neq 0 \) then
       \( c_Y \leftarrow c_Y + 1 \).
       \( \rho_Y \leftarrow \rho_Y + d_{pq} / \Delta y \).
     end
   end
20. Set pixel size: \( \lambda \leftarrow \frac{D(p)}{f} \).
21. return the normal vector \( n : \left( -\frac{\rho_Y}{\lambda c_Y}, -\frac{\rho_X}{\lambda c_X}, -1 \right) \).
Localized Depth
Normal Estimation
Localized Depth
Avatar Path Planning
Localized Depth
Rain and Snow
Surface Depth

Use Cases

(a) input depth map

(b) template mesh

(c) real-time depth mesh

winding order of the template mesh
Physics with depth mesh.
Texture decals with depth mesh.
Projection mapping with depth mesh.
Dense Depth
Depth Texture - Antialiasing
Dense Depth
Why normal map does not work?

Input: Depth map $D$, the camera image $I$, camera intrinsic matrix $K$, $L$ light sources $L = \{L_i|, i \in L\}$ with each light's location $v_{L_i}$ and intensity in RGB channels $\phi_{L_i}$.

Output: Relighted image $O$.

1. for each image pixel $p \in$ depth map $D$ in parallel do
   2. Sample $p$’s depth value $d \leftarrow D(p)$.
   3. Compute the corresponding 3D vertex $v_p$ of the screen point $p$ using the camera intrinsic matrix $v_p$ with $K$:
      \[ v_p = D(p) \cdot K^{-1} [p, 1] \]
   4. Initialize relighting coefficients of $v_p$ in RGB: $\phi_p \leftarrow 0$.
   5. for each light $L_i \in$ light sources $L$ do
      6. Set the current photon coordinates $v_o \leftarrow v_p$.
      7. Set the current photon energy $E_o \leftarrow 1$.
      8. while $v_o \neq v_{L_i}$ do
          9. Compute the weighted distance between the photon to the physical environment
             \[ \Delta d \leftarrow \alpha |v_o - v_{L_i}| + (1 - \alpha) |v_o - v_{L_i}|, \alpha = 0.5. \]
          10. Decay the photon energy: $E_o \leftarrow 95\% E_o$
          11. Accumulate the relighting coefficients:
              \[ \phi_o \leftarrow \phi_o + \Delta d \phi_{L_i} \]
          12. March the photon towards the light source:
              \[ v_o \leftarrow v_o + (v_{L_i} - v_o)/S, \text{ here } S = 10, \text{ depending on the mobile computing budget.} \]
      end
   end
13. Sample pixel’s original color: $\Phi_p \leftarrow I(p)$.
14. Apply relighting effect:
    \[ O(p) \leftarrow \gamma \cdot |0.5 - \phi_p| \cdot \Phi_p \cdot 1.5 - \Phi - \Phi_p, \text{ here } \gamma \leftarrow 3. \]
15. end
Dense Depth
Real-time relighting
Dense Depth
Real-time relighting

go/realtime-relighting, go/relit
Dense Depth
Wide-aperture effect
Dense Depth
Occlusion-based rendering
Experiments
DepthLab minimum viable application
<table>
<thead>
<tr>
<th>Procedure</th>
<th>Timings (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DepthLab's overall processing and rendering in Unity</td>
<td>8.32</td>
</tr>
<tr>
<td>DepthLab's data structure update and GPU uploading</td>
<td>1.63</td>
</tr>
<tr>
<td>Point Depth: normal estimation algorithm</td>
<td>&lt; 0.01</td>
</tr>
<tr>
<td>Surface Depth: depth mesh update algorithm</td>
<td>2.41</td>
</tr>
<tr>
<td>Per-pixel Depth: visualization with single texture fetch</td>
<td>0.32</td>
</tr>
</tbody>
</table>
Experiments

Relighting

- Input color
- Output with #samples=8
- Input depth
- Output with #samples=128

Graph showing rendering time per frame (ms) versus number of samples per ray.
Experiments
Aperture effects

(a) examples of aperture effects   (b) performance benchmark
Discussion
Deployment with partners
Discussion
Deployment with partners
Discussion
Deployment with partners
AR Realism

In TikTok
AR Realism
Built into Lens Studio for Snapchat Lenses

Snap
Dancing Hotdog

Kevaid
Saving Chelon

Quixotical
The Seed: World of Anthrotopia
Raw Depth API

Provides a more detailed representation of the geometry of the objects in the scene.

Camera Image

3D Point Cloud
New depth capabilities

**Raw Depth API**

Provides a more detailed representation of the geometry of the objects in the scene.
Depth Hit Test

Try it yourself!

New depth capabilities

TeamViewer
LifeAR App

ARCore
Depth Lab App
Limitations
Design space of dynamic depth

Dynamic Depth? HoloDesk, HyperDepth, Digits, Holoportation for mobile AR?
Envision
Design space of dynamic
depth
GitHub
Please feel free to fork!

**ARCore Depth Lab - Depth API Samples for Unity**

Copyright 2020 Google LLC. All rights reserved.

Depth Lab is a set of ARCore Depth API samples that provides assets using depth for advanced geometry-aware features in AR interaction and rendering. Some of these features have been used in this Depth API overview video.

**ARCore Depth API** is enabled on a subset of ARCore-certified Android devices. iOS devices (iPhone, iPad) are not supported. Find the list of devices with Depth API support (marked with Supports Depth API) here: https://developers.google.com/ar/discover/supported-devices. See the ARCore developer documentation for more information.

Download the pre-built ARCore Depth Lab app on Google Play Store today.

**Sample features**

The sample scenes demonstrate three different ways to access depth:

1. **Localized depth**: Sample single depth values at certain texture coordinates (CPU).
   - Character locomotion on uneven terrain
   - Collision checking for AR object placement
   - Laser beam reflections
   - Oriented 3D reticles
ARCore Depth Lab

Google Samples  Tools

Everyone

⚠️ You don't have any devices.

Installed
KEY QUOTES

“The result is a more believable scene, because the depth detection going on under the hood means your smartphone better understands every object in a scene and how far apart each object is from one another. Google says it’s able to do this through optimizing existing software, so you won’t need a phone with a specific sensor or type of processor. It’s also all happening on the device itself, and not relying on any help from the cloud.” - The Verge

“Occlusion is arguably as important to AR as positional tracking is to VR. Without it, the AR view will often “break the illusion” through depth conflicts.” - UploadVR

“Alone, that feature (creating a depth map with one camera) would be impressive, but Google’s intended use of the API is even better: occlusion, a trick by which digital objects can appear to be overlapped by real-world objects, blending the augmented and real worlds more seamlessly than with mere AR overlays.” - VentureBeat

“Along with the Environmental HDR feature that blends natural light into AR scenes, ARCore now rivals ARKit with its own exclusive feature. While ARKit 3 offers People Occlusion and Body Tracking on compatible iPhones, the Depth API gives ARCore apps a level of environmental understanding that ARKit can’t touch as of yet.” - Next Reality

"More sophisticated implementations make use of multiple cameras...That's what makes this new Depth API almost magical. With just one camera, ARCore is able to create 3D depth maps ... in real-time as you move your phone around." - Slash Gear
Impact

Significant Media Coverage

COVERAGE LINKS

- Google is improving its augmented reality tool so virtual cats can hide behind your sofa. ZDNet. December 10, 2019.
- Google’s ARCore Is Getting Full Occlusion For More Real AR. UploadVR. December 9, 2019.
- Google ARCore Depth API Now Available, Letting Devs Make AR More Realistic. RoadToVR. December 9, 2019.
- ARCore phones can now detect depth with a single camera. 9to5Google. December 9, 2019.
- ARCore Depth API lets you hide cats behind sofas even with one camera. SlashGear. December 9, 2019.
- Blending Realities with the ARCore Depth API. Google Developers. December 9, 2019.
WebXR + ARCore Depth:

Hugging Face Depth:
https://huggingface.co/spaces/Detomo/Depth-Estimation

ARCore Depth Lab Play Store App:
DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality

Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google | ACM UIST 2020
Thank you!

DepthLab | UIST 2020

DEPTHLAB: REAL-TIME 3D INTERACTION WITH DEPTH MAPS FOR MOBILE AUGMENTED REALITY

ACM UIST 2020

Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google LLC
DEPHTLAB: REAL-TIME 3D INTERACTION WITH DEPTH MAPS FOR MOBILE AUGMENTED REALITY

ACM UIST 2020

Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google LLC
DepthLab: Real-time 3D Interaction with Depth Maps for Mobile Augmented Reality

Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, David Kim

Google | ACM UIST 2020
Ad hoc UI: On-the-fly Transformation of Everyday Objects into Tangible 6DOF Interfaces for AR

Ruofei Du, Alex Olwal, Mathieu Le Goc, Shengzhi Wu, Danhang Tang, Yinda Zhang, Jun Zhang, David Joseph Tan, Federico Tombari, David Kim

Google | CHI 2022 Interactivity
Opportunistic Interfaces for Augmented Reality: Transforming Everyday Objects into Tangible 6DoF Interfaces Using Ad hoc UI

Ruofei Du, Alex Olwal, Mathieu Le Goc, Shengzhi Wu, Danhang Tang, Yinda Zhang, Jun Zhang, David Joseph Tan, Federico Tombari, David Kim

Google | ACM CHI 2022
How can we allow users to instantly transform arbitrary everyday objects into Tangible User Interfaces?
Representations
Physical objects - shape - 2D
Representations
Physical objects - shape - 3D
Representations
Physical objects - portability
Representations
Physical objects - Deformability
Representations

Physical objects
Representations

Physical objects
Representations

Physical objects
Representations

Physical objects
“Slurp” Revisited: Using Software Reconstruction to Reflect on Spatial Interactivity and Locative Media

Shengzhi Wu, Daragh Byrne, Ruofei Du, and Molly Steenson
ACM DIS 2022
Overview of our design process and artifacts
RetroSphere: Self-Contained Passive 3D Controller Tracking for Augmented Reality

Ananta Narayanan Balaji, Clayton Kimber, David Li, Shengzhi Wu, Ruofei Du, David Kim
ACM Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT) 2022
元宇宙中的交互计算与包容普惠
Computational Interaction for a Universally Accessible Metaverse

第三章 · 数字人与增强交互

Ruofei Du (杜若飞)
Senior Research Scientist
Google Labs, San Francisco
www.ruofeidu.com
me@duruofei.com
Montage4D: Interactive Seamless Fusion of Multiview Video Textures

Ruofei Du\textsuperscript{†‡}, Ming Chuang\textsuperscript{*†}, Wayne Chang\textsuperscript{‡}, Hugues Hoppe\textsuperscript{‡§}, and Amitabh Varshney\textsuperscript{†}

\textsuperscript{†}Augmentarium | UMIACS | University of Maryland, College Park
\textsuperscript{‡}Microsoft Research, Redmond
\textsuperscript{*§}PerceptIn Inc.
\textsuperscript{§}Google Inc.
Fusing multiview video textures onto dynamic task with real-time constraint is a challenging task.

30% of the users does not believe the 3D reconstructed person looks real.
Motivation

Visual Quality Matters
Workflow
Identify and diffuse the seams

Input triangle meshes from Fusion4D

Rasterized depth maps

Seams caused by misregistration and occlusion

Discrete geodesic distance fields to diffuse texture fields from the seams

Texture maps with foreground segmentation

Update temporal texture fields

Montage4D Results
Seams

Causes
Geodesic is the **shortest** route between two points on the surface.

On triangle meshes, this is challenging because of the computation of **tangent directions**. And shortest paths are defined on **edges** instead of the vertices.
Figure 6: Examples of the initial seam triangles and the propagation process for updating the geodesics.
Temporal Texture Fields
Reduce temporal flickering

Color Scheme for the Texture Fields
### Table 1: Comparison between Holoportation and Montage4D in cross-validation experiments

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Frames</th>
<th>#vertices / frame</th>
<th>#triangles / frame</th>
<th><strong>Holoportation</strong></th>
<th><strong>Montage4D</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>RMSE</td>
<td>SSIM</td>
</tr>
<tr>
<td>Timo</td>
<td>837</td>
<td>131K</td>
<td>251K</td>
<td>5.63%</td>
<td>0.9805</td>
</tr>
<tr>
<td>Yury</td>
<td>803</td>
<td>132K</td>
<td>312K</td>
<td>5.44%</td>
<td>0.9695</td>
</tr>
<tr>
<td>Sergio</td>
<td>837</td>
<td>215K</td>
<td>404K</td>
<td>7.74%</td>
<td>0.9704</td>
</tr>
<tr>
<td>Girl</td>
<td>1192</td>
<td>173K</td>
<td>367K</td>
<td>7.16%</td>
<td>0.9691</td>
</tr>
<tr>
<td>Julien</td>
<td>526</td>
<td>157K</td>
<td>339K</td>
<td>12.63%</td>
<td>0.9511</td>
</tr>
</tbody>
</table>

Montage4D achieves better quality with over 90 FPS
- Root mean square error (RMSE) ↓
- Structural similarity (SSIM) ↑
- Signal-to-noise ratio (PSNR) ↑
In conclusion, Montage4D provides a practical texturing solution for real-time 3D reconstructions. In the future, we envision that Montage4D is useful for fusing the massive multi-view video data into VR applications like remote business meeting, remote training, and broadcasting industries.
Total Relighting: Learning to Relight Portraits for Background Replacement

ROHIT PANDEY*, SERGIO ORTS ESCOLANO*, CHLOE LEGENDE*, CHRISTIAN HÄNE, SOFIEN BOUAZIZ, CHRISTOPH RHEMANN, PAUL DEBEVEC, and SEAN FANELLO, Google Research

Fig. 1. Given a portrait and an arbitrary high dynamic range lighting environment, our framework uses machine learning to composite the subject into a new scene, while accurately modeling their appearance in the target illumination condition. We estimate a high quality alpha matte, foreground element, albedo map, and surface normals, and we propose a novel, per-pixel lighting representation within a deep learning framework.
HumanGPS: Geodesic PreServing Feature for Dense Human Correspondences

CVPR 2021

Feitong Tan\textsuperscript{1,2} Danhang Tang\textsuperscript{1} Mingsong Dou\textsuperscript{1} Kaiwen Guo\textsuperscript{1} Rohit Pandey\textsuperscript{1} Cem Keskin\textsuperscript{1}
Ruofei Du\textsuperscript{1} Deqing Sun\textsuperscript{1} Sofien Bouaziz\textsuperscript{1} Sean Fanello\textsuperscript{1} Ping Tan\textsuperscript{2} Yinda Zhang\textsuperscript{1}

\textsuperscript{1}Google \quad \textsuperscript{2}Simon Fraser University
Live Demo

1. Click Choose File to upload a human image and a mask image (recommended w : h = 256 : 384) or use the example images: 1 2 3
2. Click ‘Process’ button to run the model. You may use Pen, Brush, Eraser, and Clear to doodle on the mask image.
Note that the first time ‘Process’ takes longer time for initialization. The model may take a few seconds to load in some region.
GazeChat
Enhancing Virtual Conferences With Gaze-Aware 3D Photos

Zhenyi He†, Keru Wang†, Brandon Yushan Feng‡, Ruofei Du†, Ken Perlin†

† New York University
‡ University of Maryland, College Park
† Google
Introduction
VR headset & video streaming
Related Work


half-silvered mirror
45°
camera 1
camera 2
camera 3
eye tracker
lcd panel
Related Work

MultiView (2005)
Related Work

MMSpace (2016)
Our Work
GazeChat (UIST 2021)
Gaze awareness, defined here as knowing what someone is looking at.
Gaze Awareness

Definition

raw input image

gaze correction

gaze redirection

GazeChat
Gaze Correction

**Definition**
Gaze Rediction

Definition

eye contact

who is looking at whom
Pipeline

System

(a) Input Data
- a profile photo
- webcam video

(b) Intermediate Results
- a depth map
- a 3D mesh
- an eye mask
- gaze directions
- 20 synthesized images with gaze redirection

(c) Rendering of GazeChat
- Zhenyi
- Bob
- Charlie
- Dave
Eye Tracking
WebGazer.js
Neural Rendering

Eye movement
Eye Movement Synthesis
First Order Motion Model
3D Photo Rendering

3D photos

(a) Original Profile
(b) Depth Image
(c) 3D Photo
(d) Synthesized Gaze Images
(e) Eye Mask
(f) Gaze Redirection
3D Photo Rendering

3D photos
Layouts

UI

Third Person

Eye Contact

[Images of people]

[Images of people]
CollaboVR: A Reconfigurable Framework for Creative Collaboration in Virtual Reality

Zhenyi He*
Ruofei Du†
Ken Perlin*

*Future Reality Lab, New York University  †Google LLC
SketchyScene: Richly-Annotated Scene Sketches
Changqing Zou, Qian Yu, Ruofei Du, Haoran Mo, Yi-Zhe Song, Tao Xiang, Chengying Gao, Baoquan Chen, and Hao Zhang (ECCV 2022)

It is a sunny day. It is a family picnic. There are four people, a basket, two apples, one cup, and two bananas on a picnic rug. There are two trees in the distance.

Fig. 1. A scene sketch from our dataset SKETCHYSCENE that is user-generated based on the reference image shown, a segmentation result (middle) obtained by a method trained on SKETCHYSCENE, and a typical application: sketch captioning.
Language-based Colorization of Scene Sketches
Changqing Zou, Haoran Mo, Chengying Gao, Ruofei Du, and Hongbo Fu (ACM Transaction on Graphics, SIGGRAPH Asia 2019)

Scene sketch

"the car is red with black windows"

"the road is black"/
"colorize the road with black"/
"black road"

"all the trees are green"
"the sun in the sky is yellow"
"the grasses are dark green"

"the sky is blue and the ground is green"
ProtoSound: A Personalized and Scalable Sound Recognition System for Deaf and Hard-of-Hearing Users

Figure 1: ProtoSound is a technique to customize a sound recognition model using very few recordings, enabling the model to scale across contextual variations of sound (e.g., water flowing on a stainless steel vs. a porcelain sink) and support new user-specific sound classes (e.g., a piano). Images show some example sound categories that were trained and recognized during our field evaluation using an experimental mobile app built off ProtoSound. See our supplementary video for details.
Wearable Subtitles
Augmenting Spoken Communication with Lightweight Eyewear for All-day Captioning

Figure 1. Our Wearable Subtitles proof-of-concept shows how eyewear could benefit people who are deaf or hard of hearing. We explore hands-free access to spoken communication, situational and speaker awareness, and improved understanding while engaged in a primary task. Our lightweight (54 g) 3D-printed eyewear prototype augments the user’s perception of speech and sounds in a socially acceptable form factor with an architecture that could enable up to 15 hours of continuous transcription.
Future Directions
The Ultimate XR Platform
Future Directions

Fuses Past Events
Future Directions
With the present
Future Directions

And look into the future
Future Directions

Change the way we communicate in 3D and consume the information.
Future Directions

Consume the information throughout the world
Computational Interaction for a Universally Accessible Metaverse