Enabling Architecture Research for Augmented and Virtual Reality

Muhammad Huzaifa, PhD Candidate
Advisor: Sarita Adve
University of Illinois at Urbana-Champaign

AR/VR's Crazy Requirements

SAMSUNG ODYSSEY

2.3 Mpixels/eye
90 Hz
110° FoV
Latency?
250 W
500 mm2
700 grams

200x perf!
2500x power!
5x area!
10x weight!

IDEAL HEADSET

100 Mpixels/eye
144 Hz
175° FoV
20 ms or less
100s mW
100 mm2
10s grams

Why AR/VR?

Up and coming killer application

Challenges span entire system

Great driver for hardware specialization
- Several AR/VR kernels are shared across domains

Challenges

State-of-the-art closely guarded by industry

No open-source benchmarks, no models, no simulators

Where do we start?

VR Pipeline

Developed aspirational VR pipeline to capture key components and key system interactions

Input

Head tracking
Eye tracking
Hand tracking
Depth
Networking

Rendering

Stereoscopic rendering
Foveated rendering
3D audio
3D video
SLAM

Post-processing

Contrast preservation
Chromatic aberration correction
Lens distortion correction
Temporal anti-aliasing

Optimizations

Time warp
Space warp

Output

Adaptive display
Ambisonics

VR Pipeline

Really just an assortment of asynchronous pipelines!

Input
Head tracking
Eye tracking
Hand tracking
Depth
Networking

→

Rendering
Stereoscopic rendering
Foveated rendering
3D audio
3D video
SLAM

→

Post-processing
Contrast preservation
Chromatic aberration correction
Lens distortion correction
Temporal anti-aliasing

→

Optimizations
Time warp
Space warp

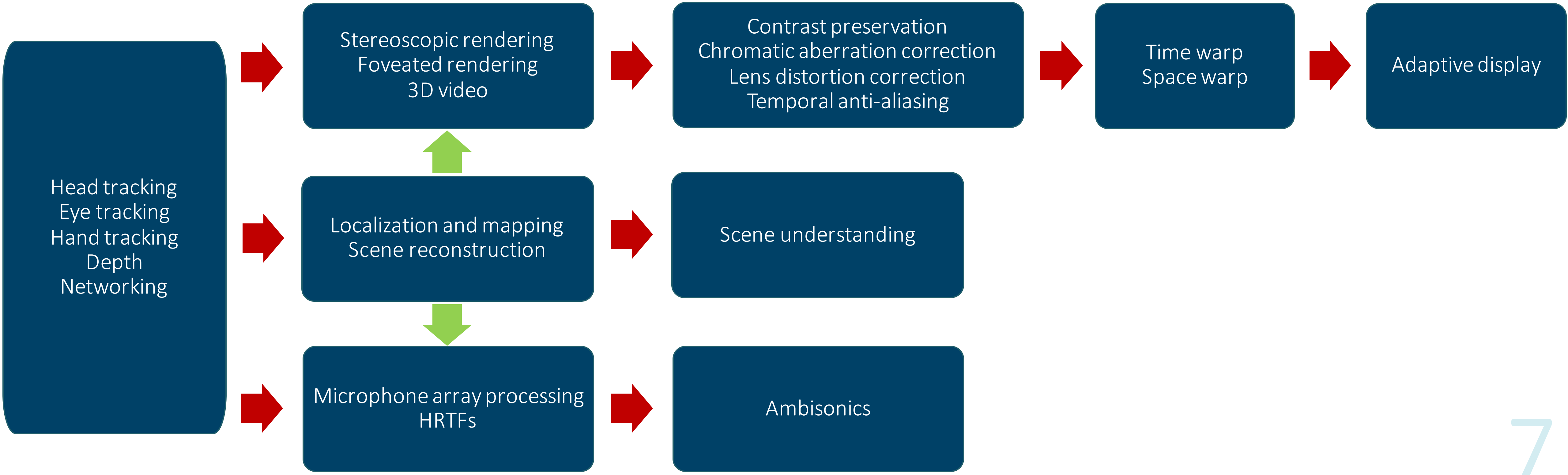
→

Output
Adaptive display
Ambisonics

VR Pipeline

Really just an assortment of asynchronous pipelines!

Challenges: Many domains, difficult to integrate multiple code bases, modeling graphics and system interactions



Approach

Input	Rendering	Post-processing	Optimizations	Output
Head tracking Eye tracking Hand tracking Depth Networking	Stereoscopic rendering Foveated rendering 3D audio 3D video SLAM	Contrast preservation Chromatic aberration correction Lens distortion correction Temporal anti-aliasing	Time warp Space warp	Adaptive display Ambisonics

Step 1: Collect state-of-the-art codes for each component

Approach

Input	Rendering	Post-processing	Optimizations	Output
Head tracking Eye tracking Hand tracking Depth Networking	Stereoscopic rendering Foveated rendering 3D audio 3D video SLAM	Contrast preservation Chromatic aberration correction Lens distortion correction Temporal anti-aliasing	Time warp Space warp	Adaptive display Ambisonics

Step 1: Collect state-of-the-art codes for each component

Step 2: Analyze each component in isolation

Step 3: Create intermediate mini VR pipelines to understand interactions

This talk: [SLAM + Renderer] ➔ Adaptive Display

Step 4: Use analysis to drive *scalable hardware specialization*

SLAM

Simultaneous Localization and Mapping: where am I in the world and what the world looks like

ELASTICFUSION

Dense

Directly uses pixel intensities and depths for tracking

Dense reconstruction

Computationally expensive

Pose estimation + optimization + deformation graph

50% reductions, 30% memcpys

ORB

Spa

Per
tra

No

Computationally less expensive

Computer vision + RANSAC + BA

Vision dominates – FAST, matrix ops

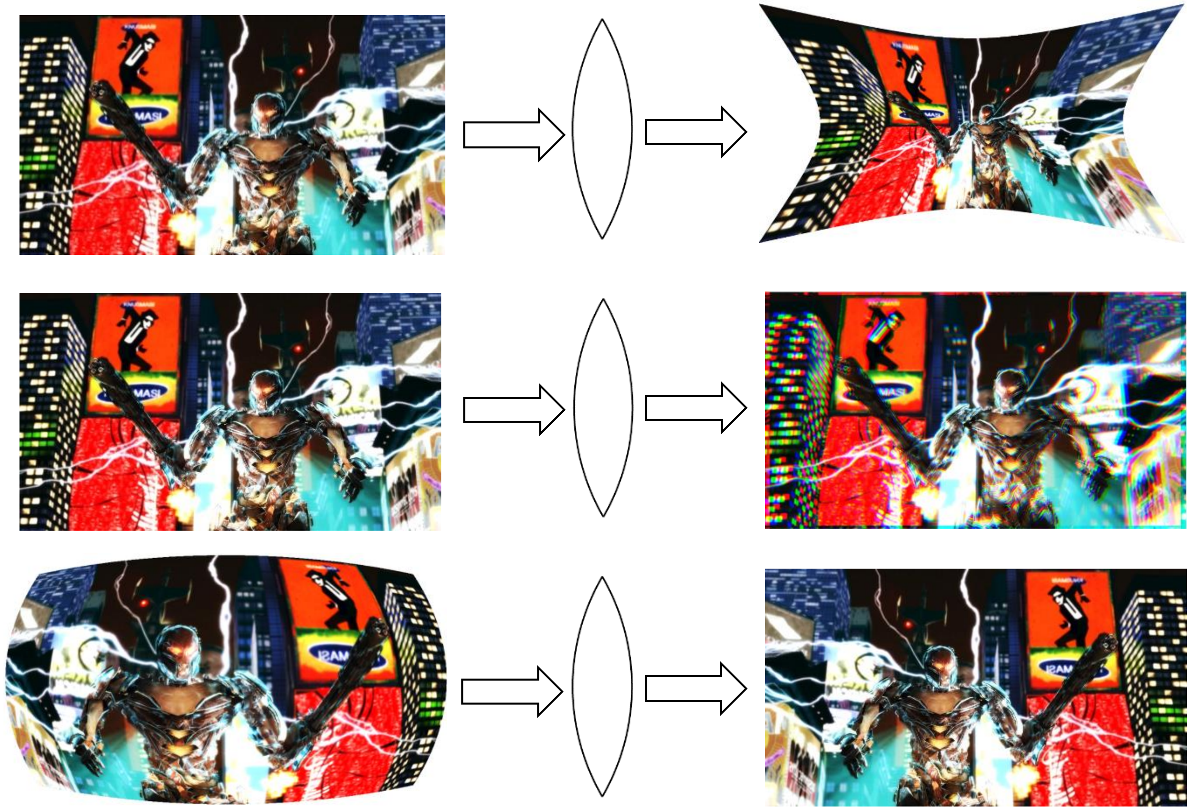
Lens & Chromatic Distortion Correction

Pincushion distortion and chromatic aberrations caused by curved lenses

Six different mathematical models implemented in OpenGL

All models perform similarly

Bound by irregular texture accesses



Asynchronous Time Warp

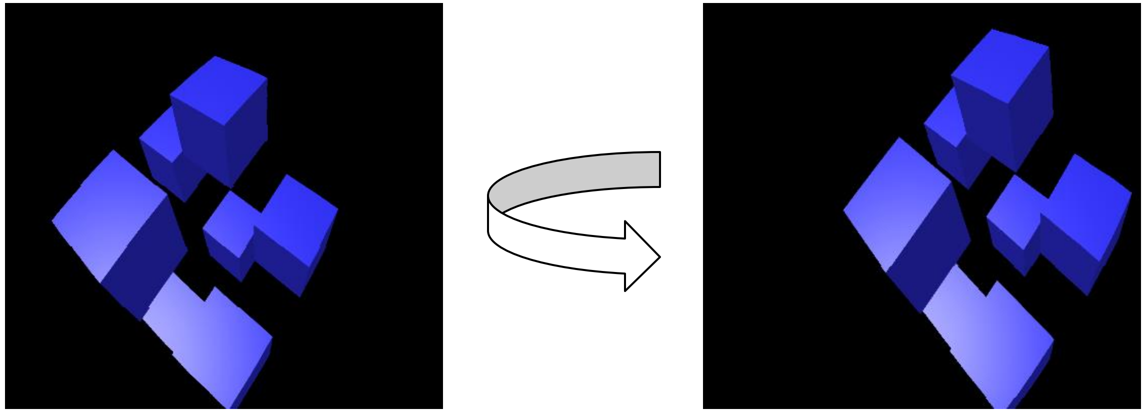
Warping frame to account for head movement

Still under development

- pthreads + OpenGL

Insights so far

- Matrix multiplication heavy
- Locking on framebuffer and eye textures



Hologram

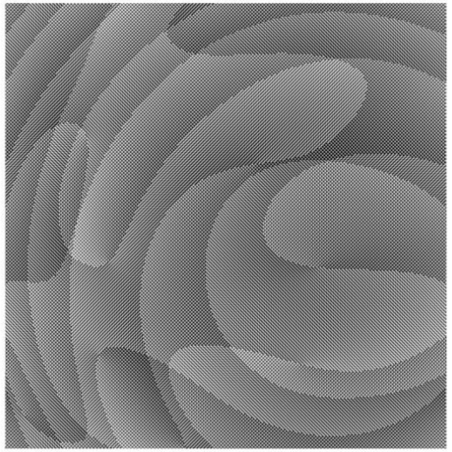
Mapping images to multi-focal displays

Two different algorithms implemented in CUDA

- Algorithm choice based on number of depth planes

~1.8 ms/iteration & ~10 iterations – not real time for 144 Hz!

Compute bound: reductions + transcendentals



Ambisonics

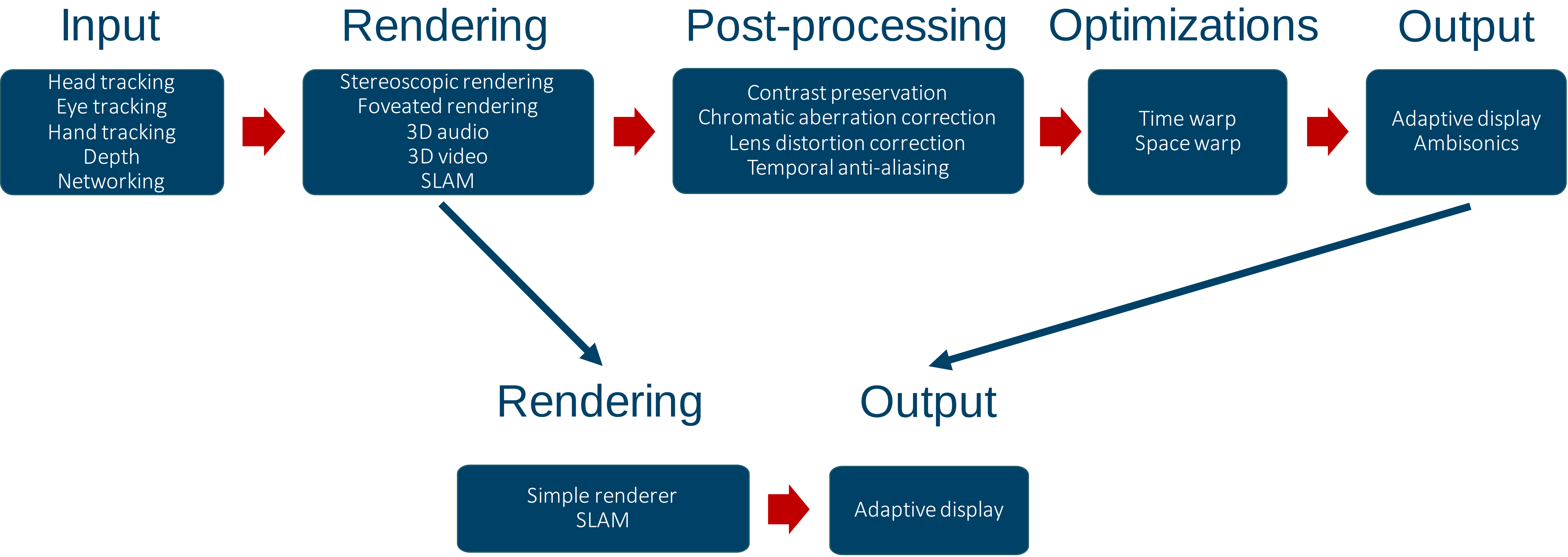
Mapping sound from virtual channels to a given speaker configuration

Five different decoders
- Implemented in CUDA
- Simple and low fidelity to complex and high fidelity

Most complex decoder takes ~180 ms on a Pascal GPU

Bound by irregular accesses + atomics

Mini VR Pipeline



Mini VR Pipeline Findings

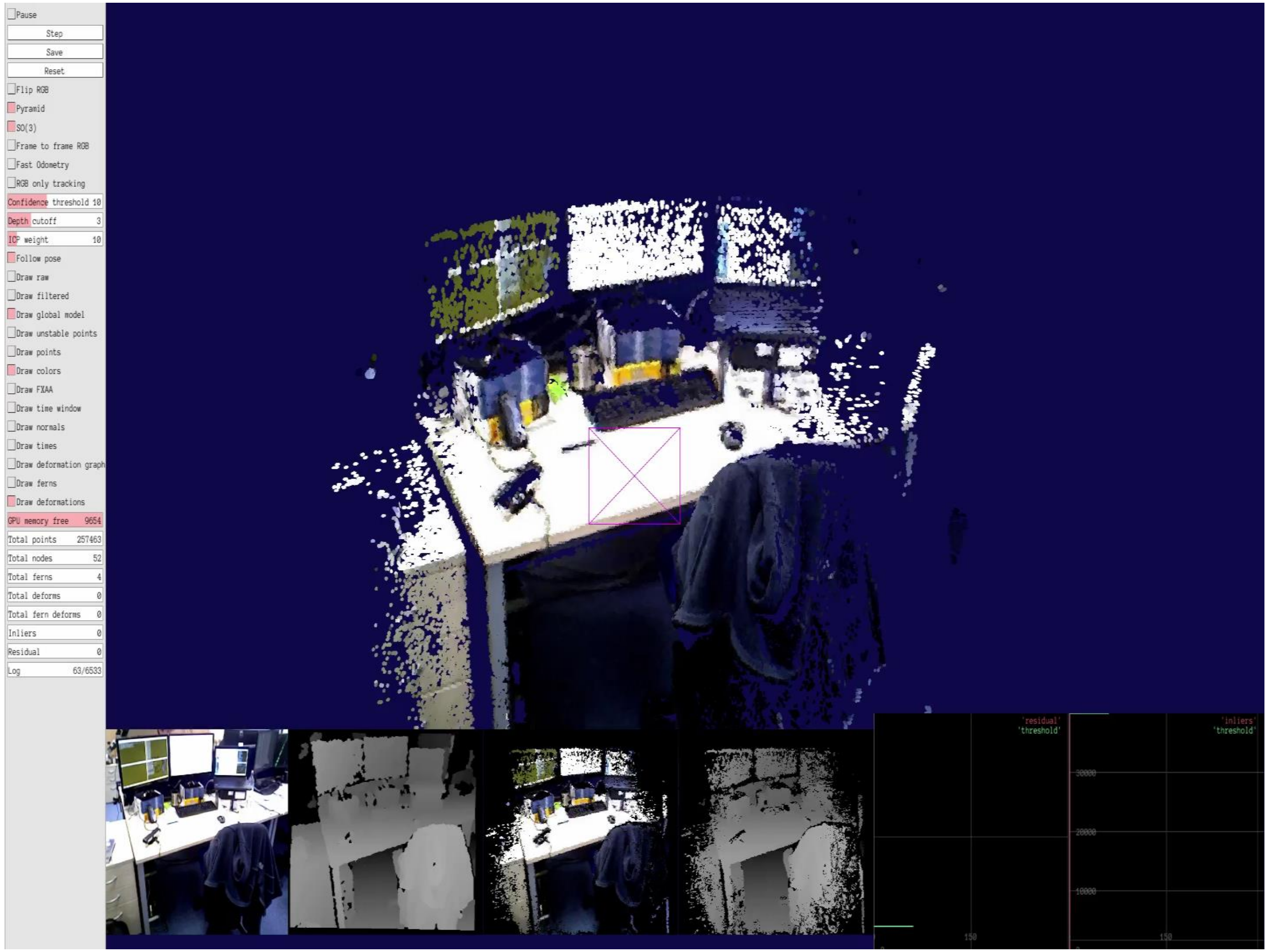
No longer real time on Titan Xp at Vive's resolution (2160x1200) at 30 fps
- ElasticFusion (20 ms) + Renderer (5 ms) + Hologram (18 ms) = 43 ms
- Deadline: 33 ms

Common compute pattern: reductions
- ~50% of CUDA execution time in ElasticFusion; ~63% in Hologram
- Reductions are on custom data-structures!

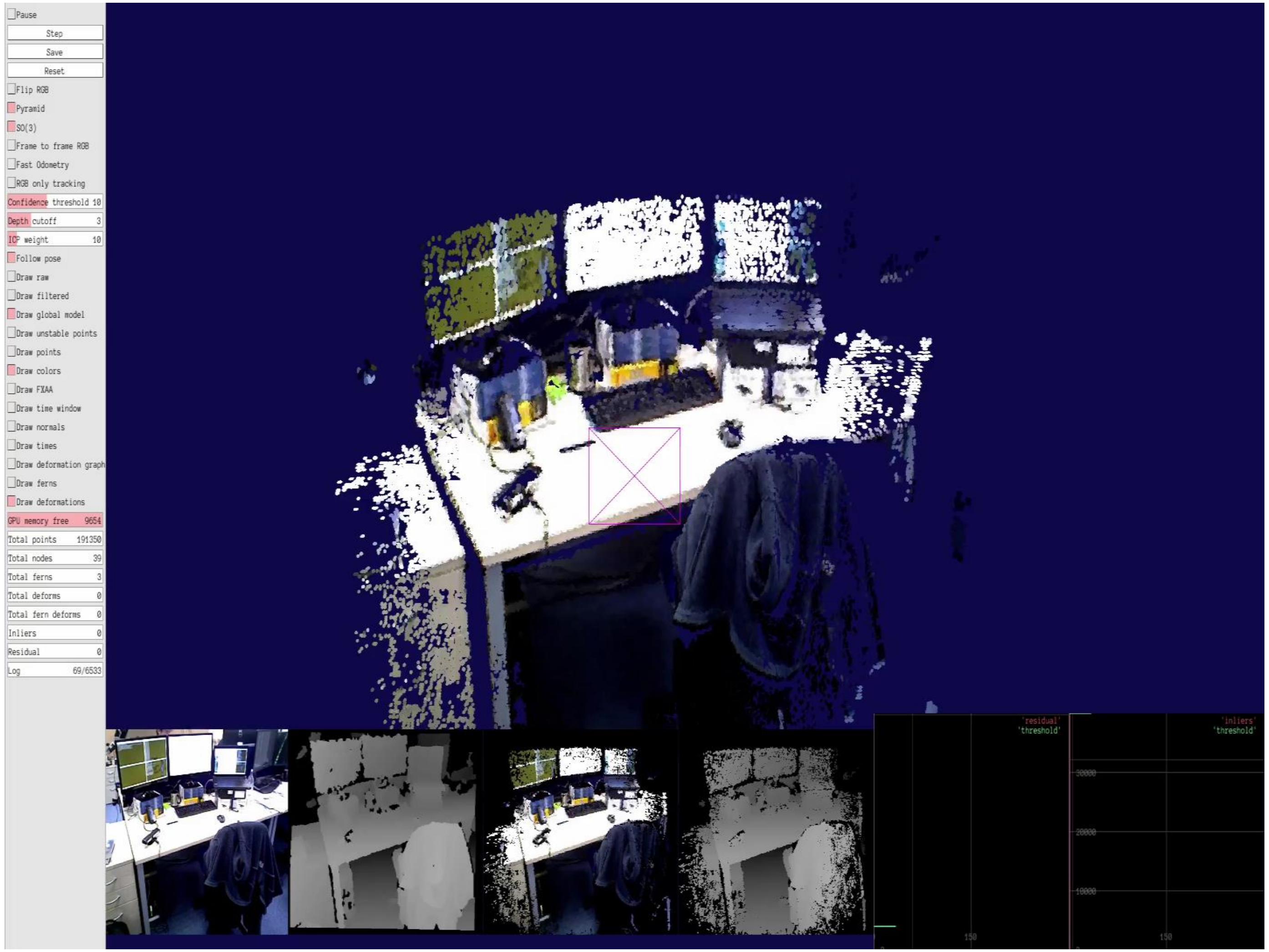
But different data layouts for each kernel

Need communication specialization techniques such as Spandex!

Vanilla ElasticFusion



ElasticFusion + Hologram



Architectural Challenges

Ever changing algorithms => programmable hardware

Limited die area => shared hardware

Memory, the destroyer of worlds => dynamic partitioning, allocation, scheduling

Tons of sensors => on-sensor computing

Distribution of work between glasses, phone, and cloud

Human brain is not perfect => approximate computing

We hope that our benchmark suite can help answer these questions!

What’s Next?

Finish application
- Aberration shaders between ElasticFusion and hologram ~~(easy~~ medium)
- Head & eye tracking + time & space warping combo (medium)
- Integrate audio pipeline (medium)
- Replace simple renderer with stereoscopic & foveated renderers (hard)

Repeat experiments on embedded platform; e.g., NVIDIA Jetson TX2

What's Next?

Finish application

- Aberration shaders between ElasticFusion and hologram ~~(easy~~ medium)
- Head & eye tracking + time & space warping combo (medium)
- Integrate audio pipeline (medium)
- Replace simple renderer with stereoscopic & foveated renderers (hard)

Repeat experiments on embedded platform; e.g., ~~NVIDIA Jetson TX2~~

What’s Next?

Finish application
- Aberration shaders between ElasticFusion and hologram ~~(easy~~ medium)
- Head & eye tracking + time & space warping combo (medium)
- Integrate audio pipeline (medium)
- Replace simple renderer with stereoscopic & foveated renderers (hard)

Repeat experiments on embedded platform; e.g., [redacted] Arm dev board

What's Next?

Finish application
- Aberration shaders between ElasticFusion and hologram ~~(easy~~ medium)
- Head & eye tracking + time & space warping combo (medium)
- Integrate audio pipeline (medium)
- Replace simple renderer with stereoscopic & foveated renderers (hard)

Repeat experiments on embedded platform; e.g., [redacted] Arm dev board

Use analysis to guide accelerator and memory system development
- **Spandex** is a promising platform: unified shared memory, flexible coherence and communication

Release application!

Thank you for your time!

Questions?

Muhammad Huzaifa | Sarita Adve | UIUC

JUMP

Joint University Microelectronics Program

www.src.org/program/jump

Semiconductor Research Corporation

@srcJUMP