Microsoft HoloLens 2 and Azure Kinect as tools for computer vision research

Marc Pollefeys, Federica Bogo, Dorin Ungureanu,
Pooja Sama, Silvano Galliani | August 23, 2020

Tutorial overview

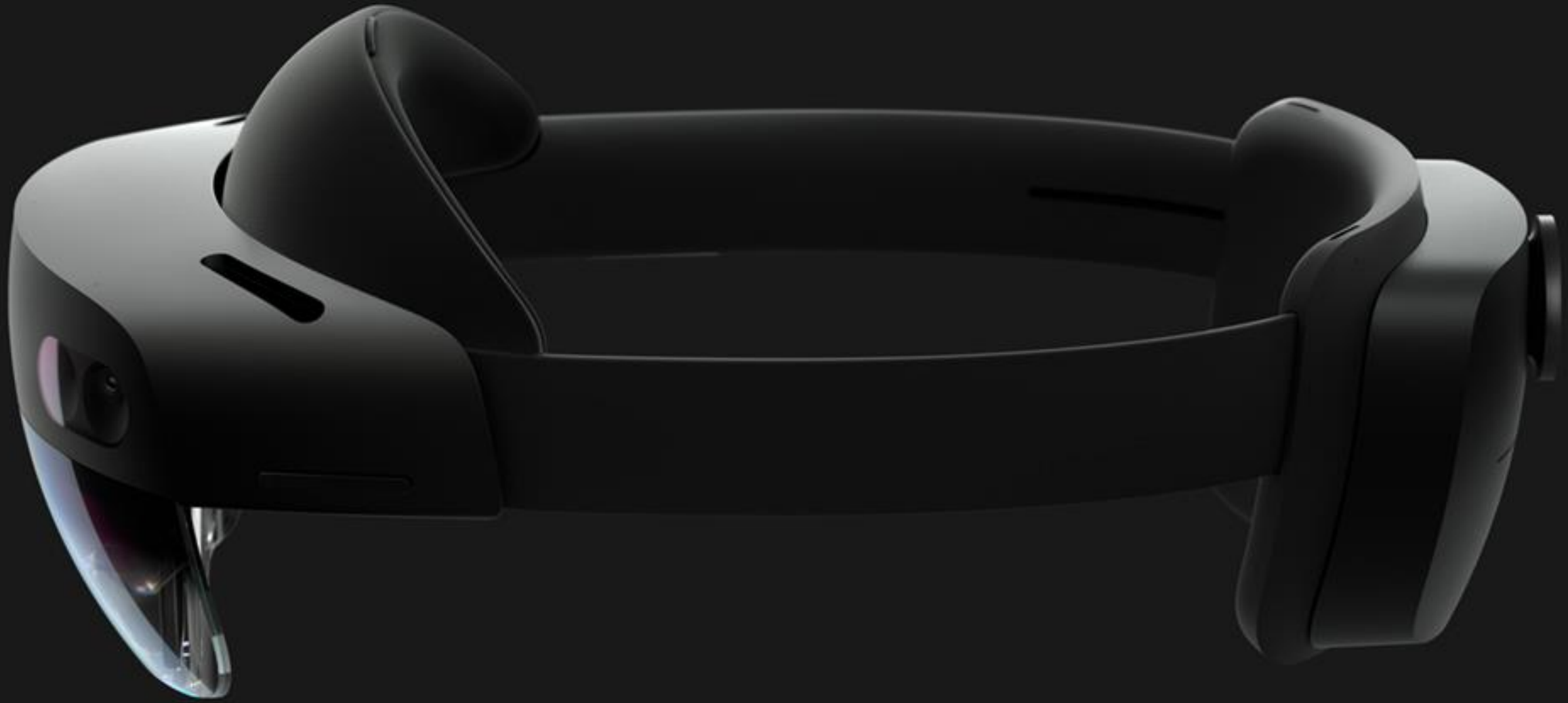
Introduction

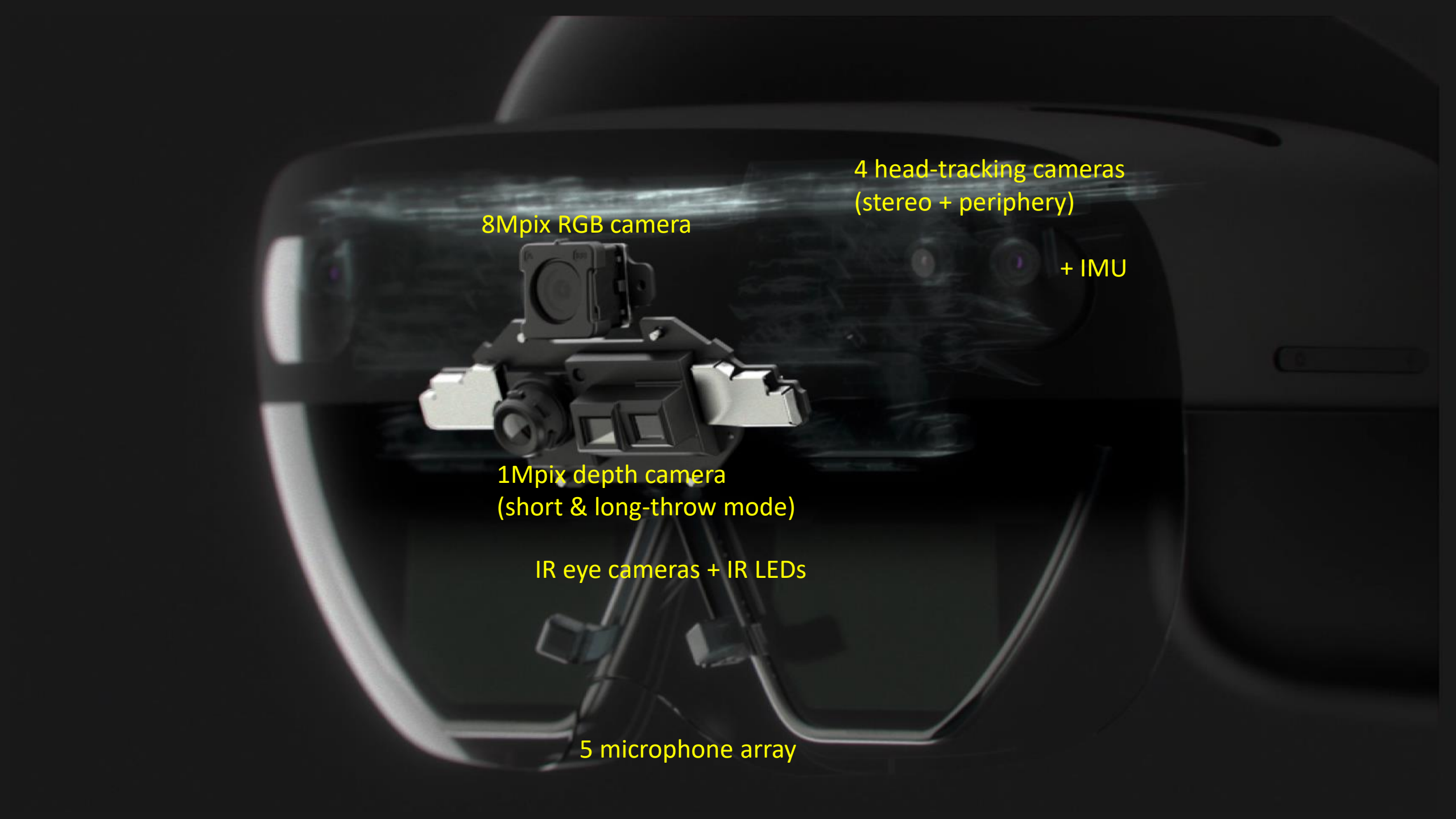
HoloLens2 Research Mode

- API overview
- HoloLens2forCV repository

Azure Kinect

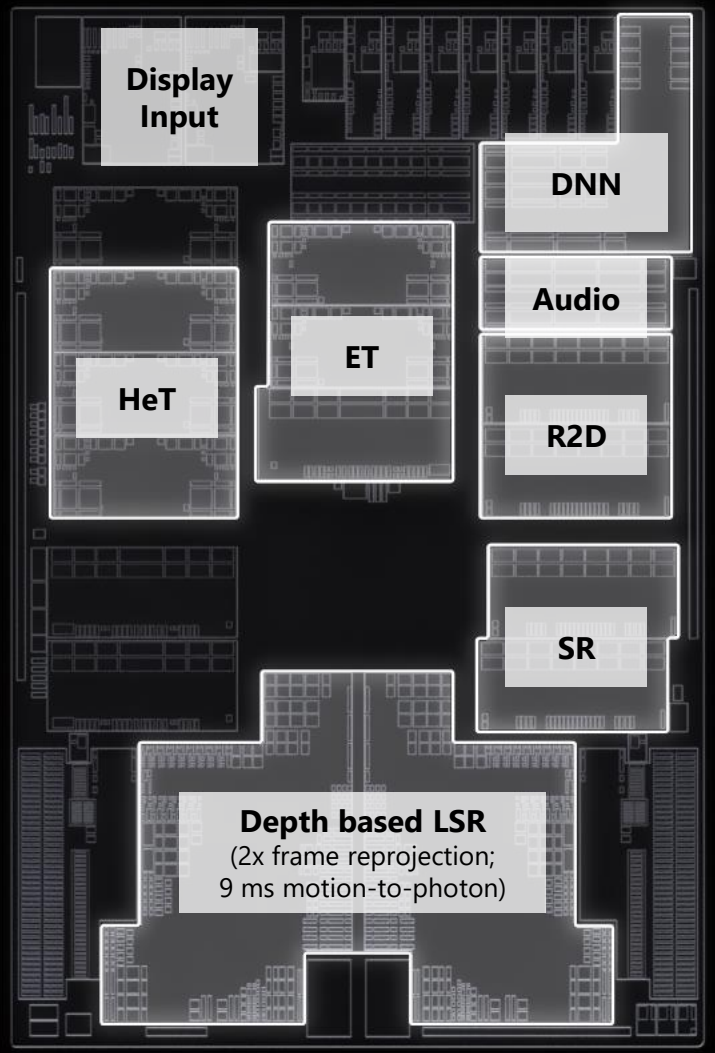
- Open Source Sensor SDK
- Body Tracking SDK
- Demos

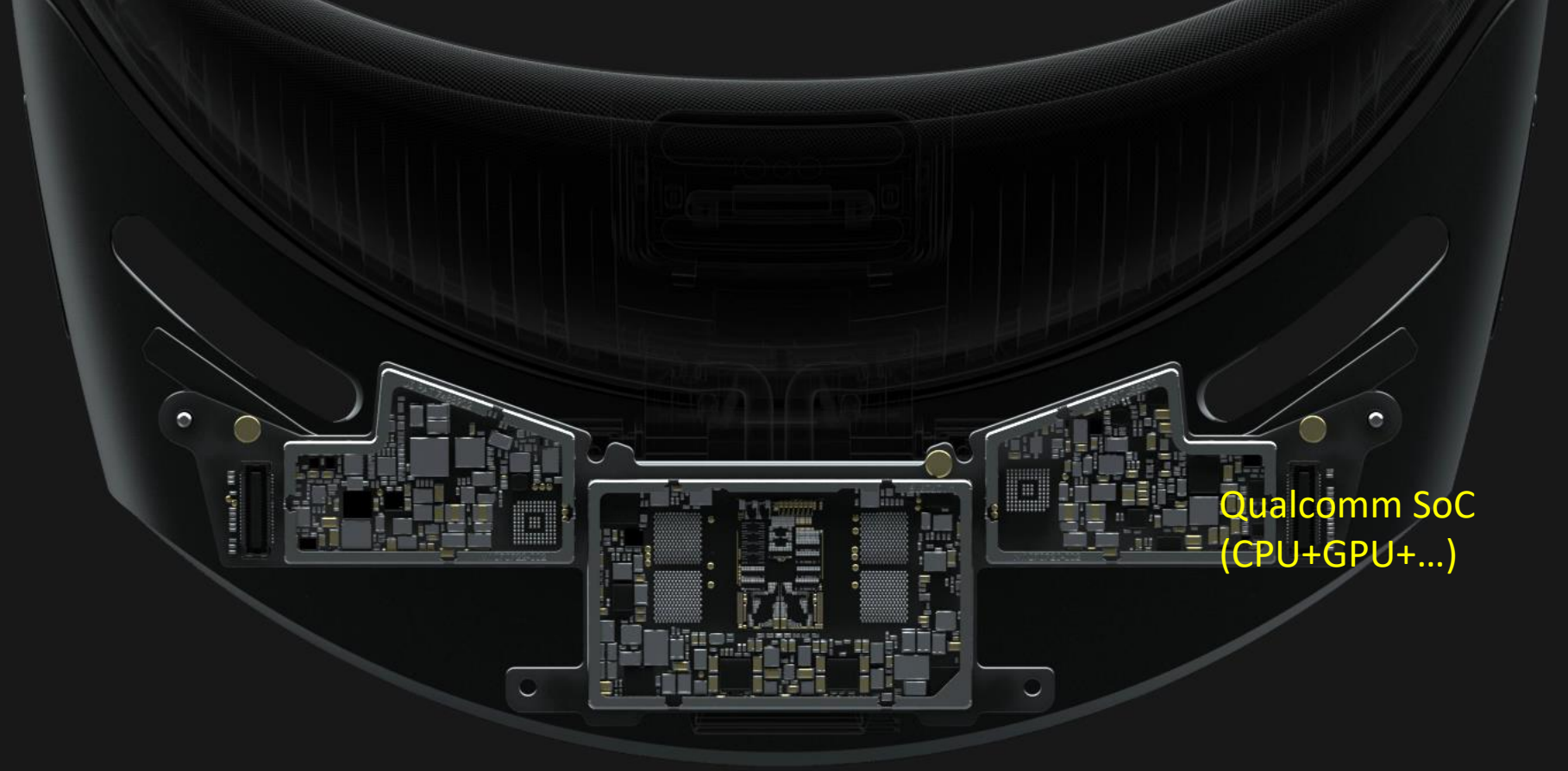



4 head-tracking cameras
(stereo + periphery)
8Mpix RGB camera
+ IMU
1Mpix depth camera
(short & long-throw mode)
IR eye cameras + IR LEDs
5 microphone array


HPU (DSPs, DNN AI core, LSR)

HPU


Display Input
DNN
Audio
ET
HeT
R2D
SR
Depth based LSR
(2x frame reprojection;
9 ms motion-to-photon)


Qualcomm SoC
(CPU+GPU+…)

Where are you?

- Head tracking
  - Location in the world
  - Head orientation

HoloLens Head tracking

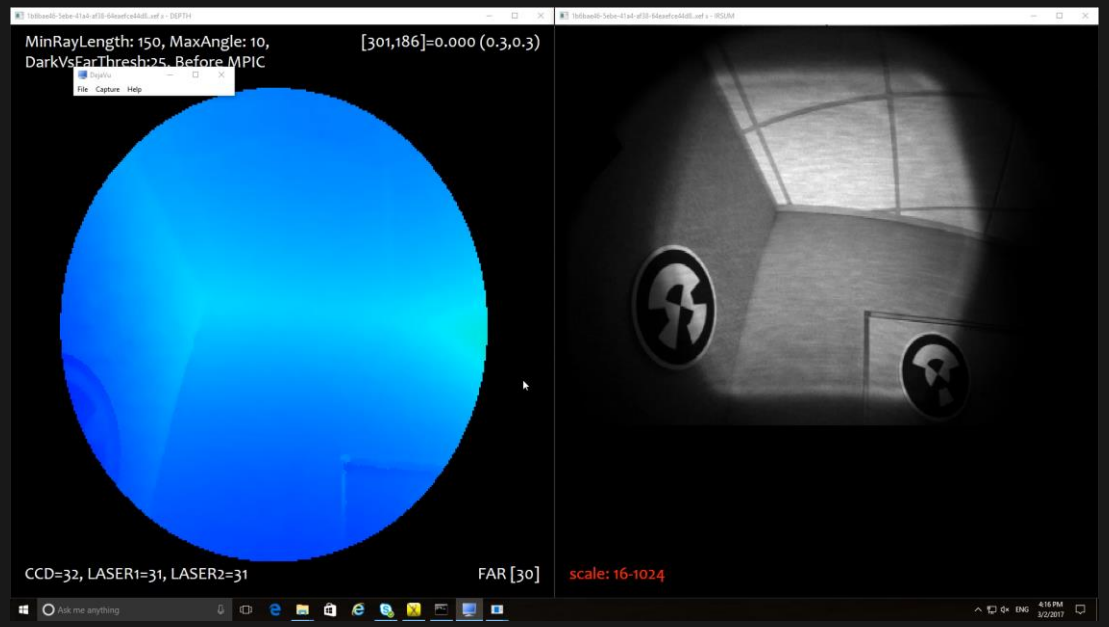
highly accurate visual-inertial odometry

4 cameras + IMU
highly optimized for power

What's around you

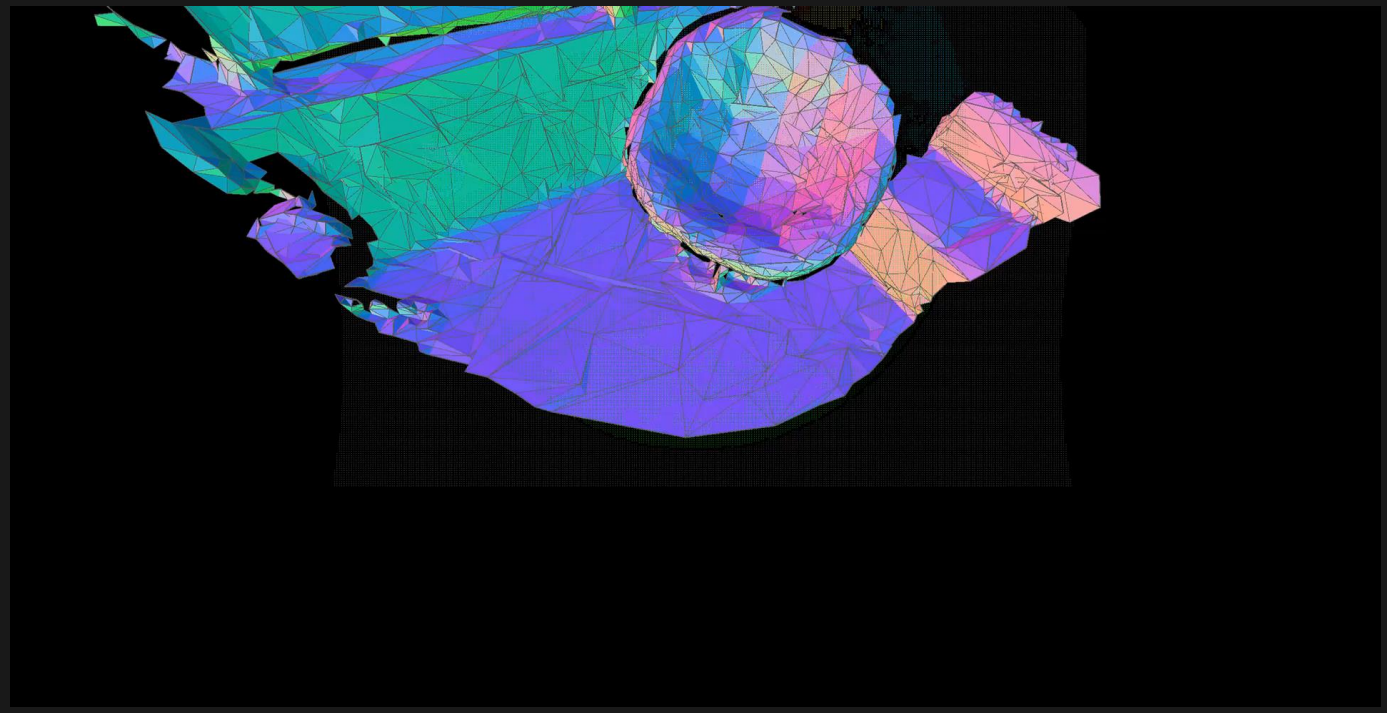
- Depth perception
  - TOF depth camera
  - Mesh generation

Depth sensing



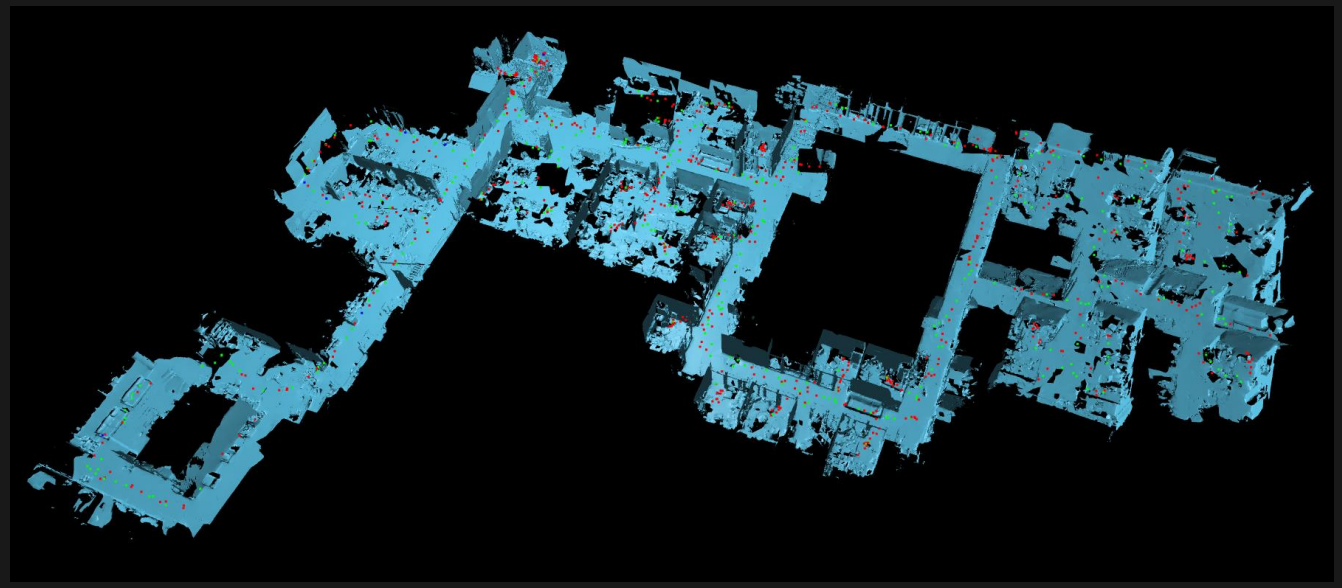
Depth + (near) IR image

Surface reconstruction

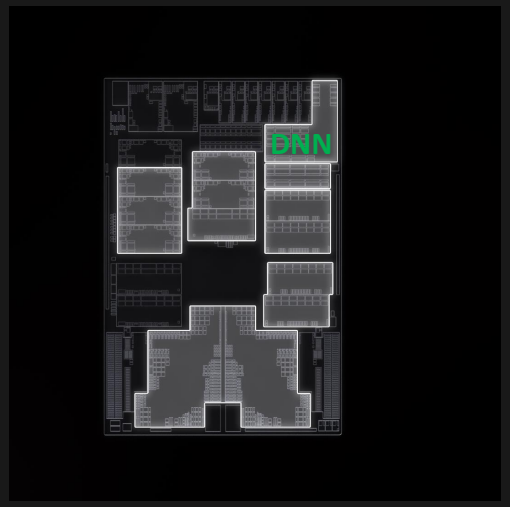


Fusion of depth measurements and mesh extraction

Spatial Mapping



What is around you?

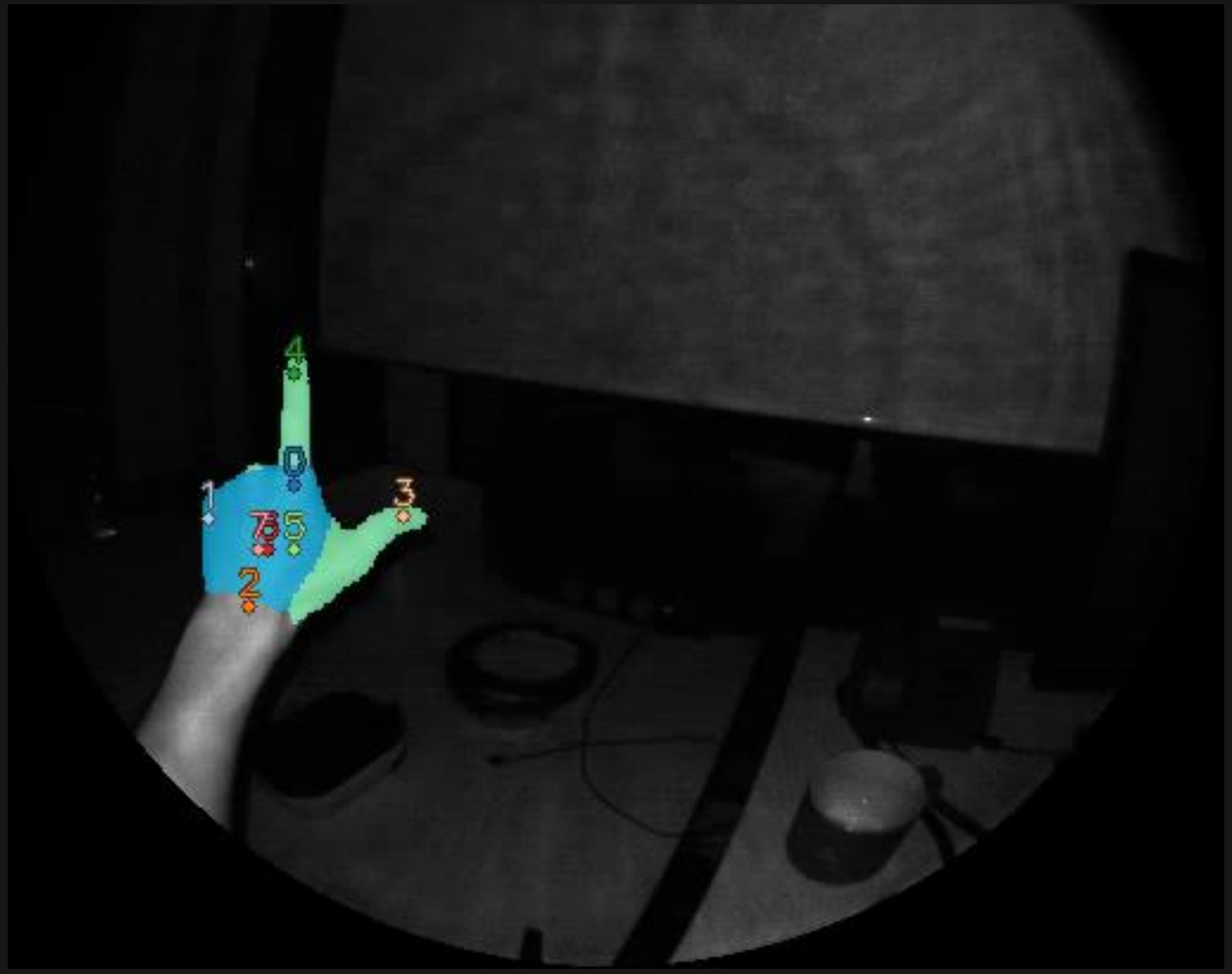


Semantic segmentation demo

What are you doing?

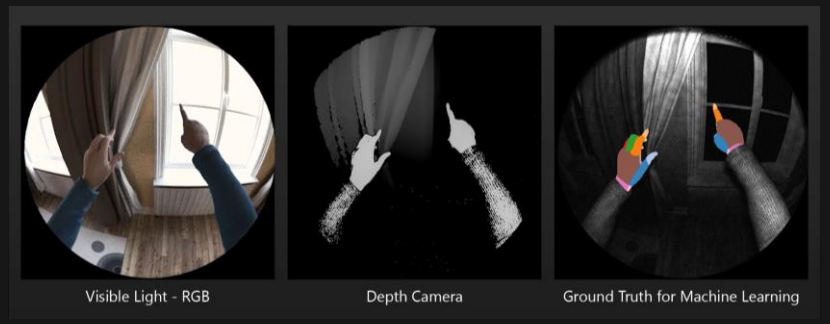
- Gestures/Hand tracking
- Gaze direction
- Speech

HoloLens2 Hand-tracking

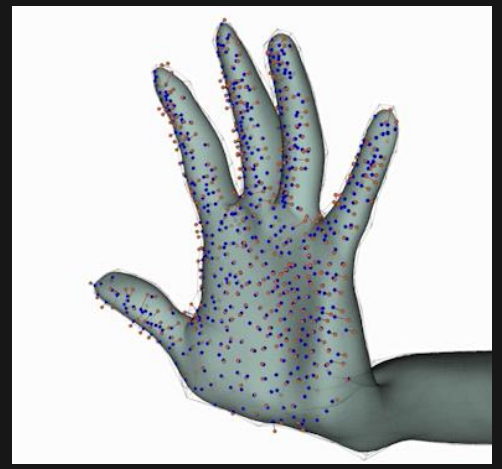


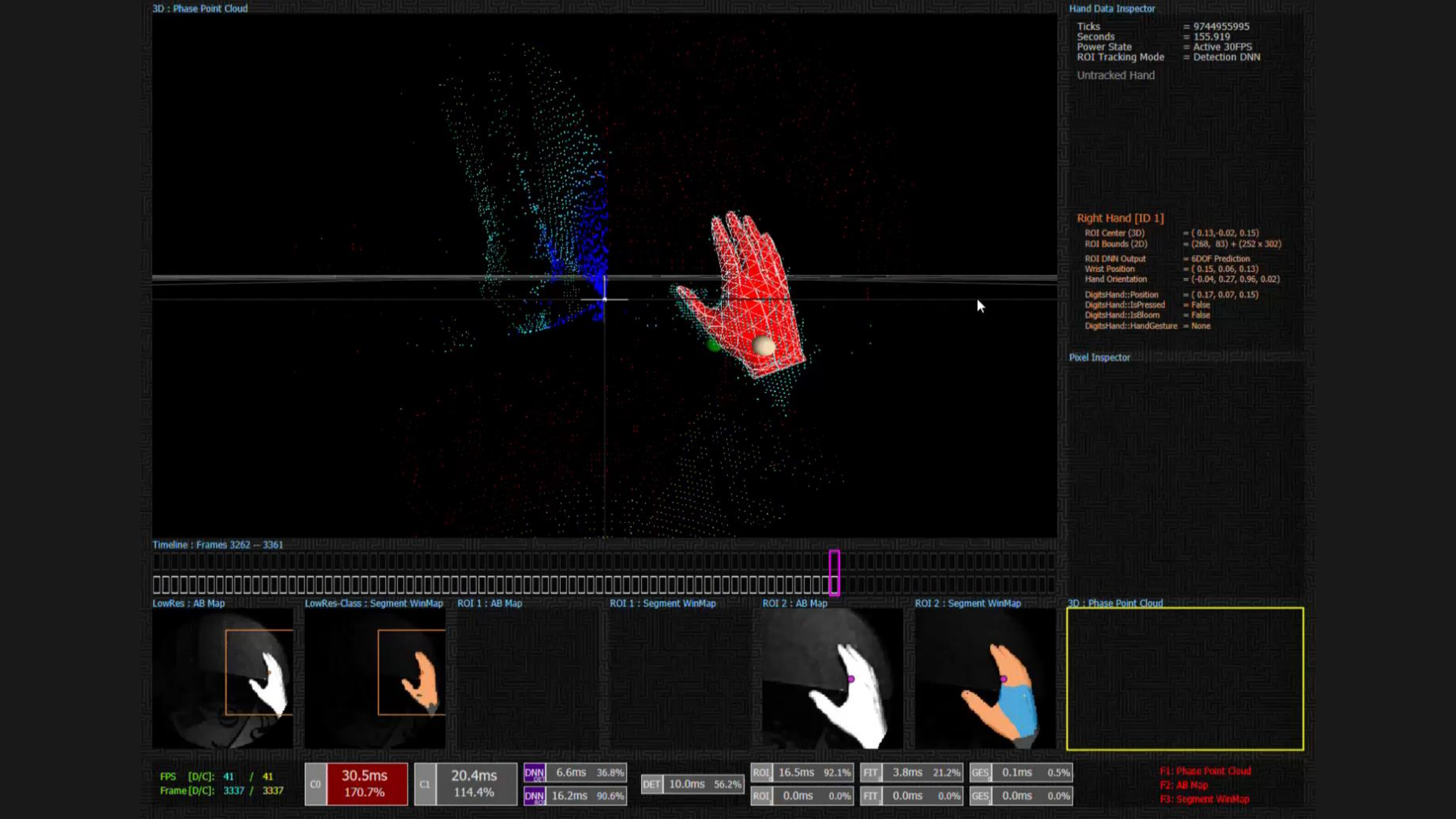
Hand-segmentation tuns on HoloLens DNN accelerator

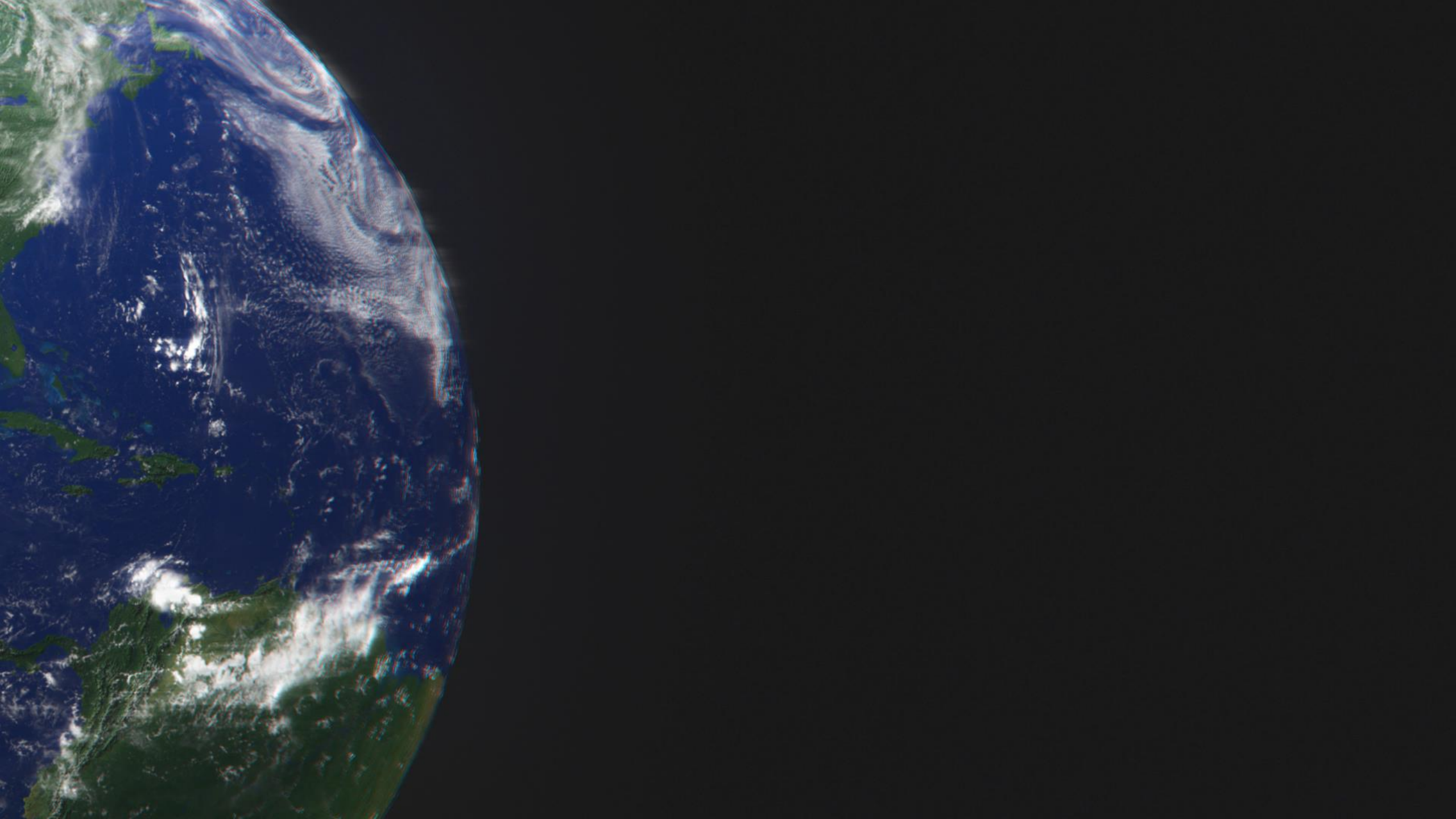
Trained on purely synthetic data



Efficient geometric fitting




3D : Phase Point Cloud
Hand Data Inspector
Ticks = 9744955995
Seconds = 155.919
Power State = Active 30FPS
ROI Tracking Mode = Detection DNN
Untracked Hand
Right Hand [ID 1]
ROI Center (3D) = (0.13,-0.02, 0.15)
ROI Bounds (2D) = (268, 83) + (252 x 302)
ROI DNN Output = 6DOF Prediction
Wrist Position = (0.15, 0.06, 0.13)
Hand Orientation = (-0.04, 0.27, 0.96, 0.02)
DigitsHand::Position = (0.17, 0.07, 0.15)
DigitsHand::IsPressed = False
DigitsHand::IsBloom = False
DigitsHand::HandGesture = None
Pixel Inspector
Timeline : Frames 3262 -- 3361
LowRes : AB Map
LowRes-Class : Segment WinMap
ROI 1 : AB Map
ROI 1 : Segment WinMap
ROI 2 : AB Map
ROI 2 : Segment WinMap
3D : Phase Point Cloud
FPS [D/C]: 41 / 41
Frame [D/C]: 3337 / 3337
C0 30.5ms 170.7%
C1 20.4ms 114.4%
DNN DET 6.6ms 36.8%
DNN ROI 16.2ms 90.6%
DET 10.0ms 56.2%
ROI 16.5ms 92.1%
ROI 0.0ms 0.0%
FIT 3.8ms 21.2%
FIT 0.0ms 0.0%
GES 0.1ms 0.5%
GES 0.0ms 0.0%
F1: Phase Point Cloud
F2: AB Map
F3: Segment WinMap

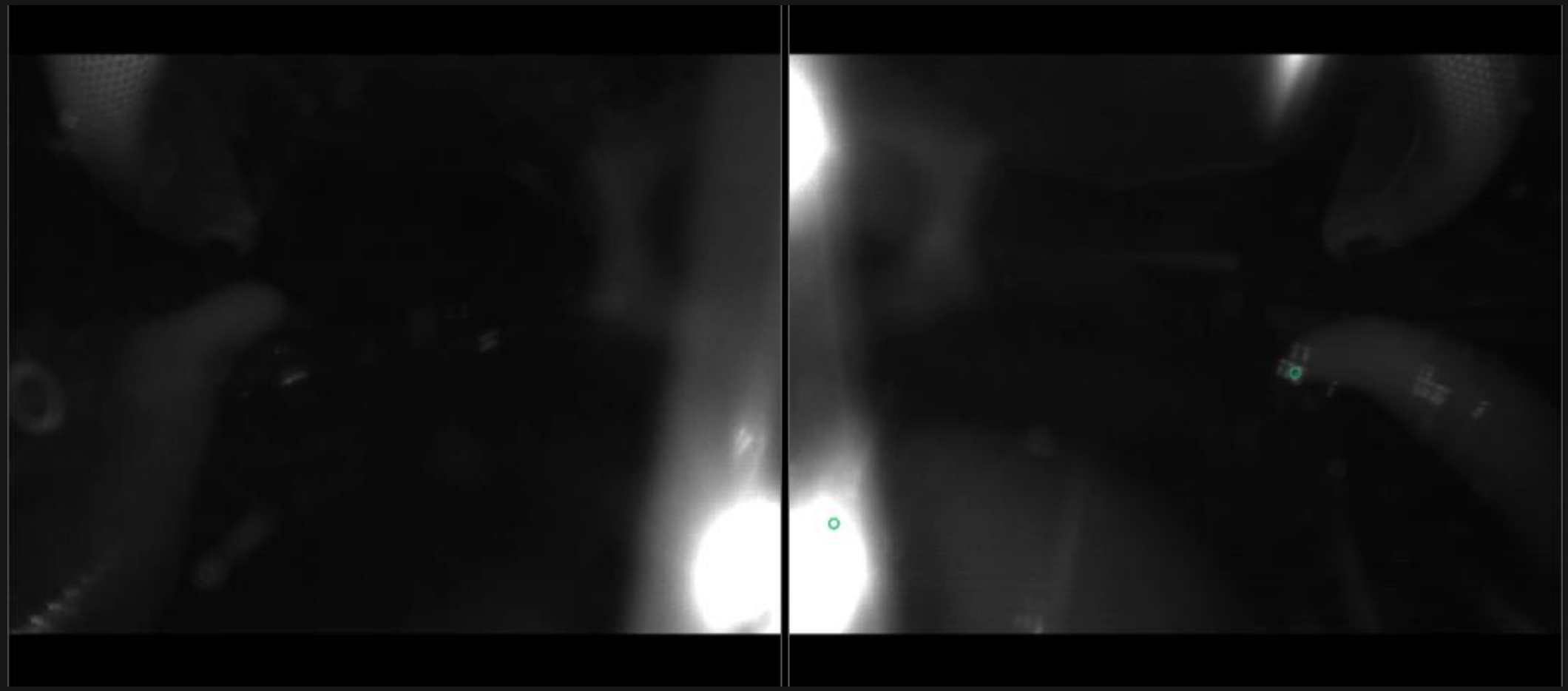



Eye Tracking


Windows Hello

Eye Tracking

2 IR cameras + IR LEDs



Environment data generally available to apps

IMU

HeT cameras

Raw Depth

Custom HPU

Head pose, map

Mesh

Hands, gestures

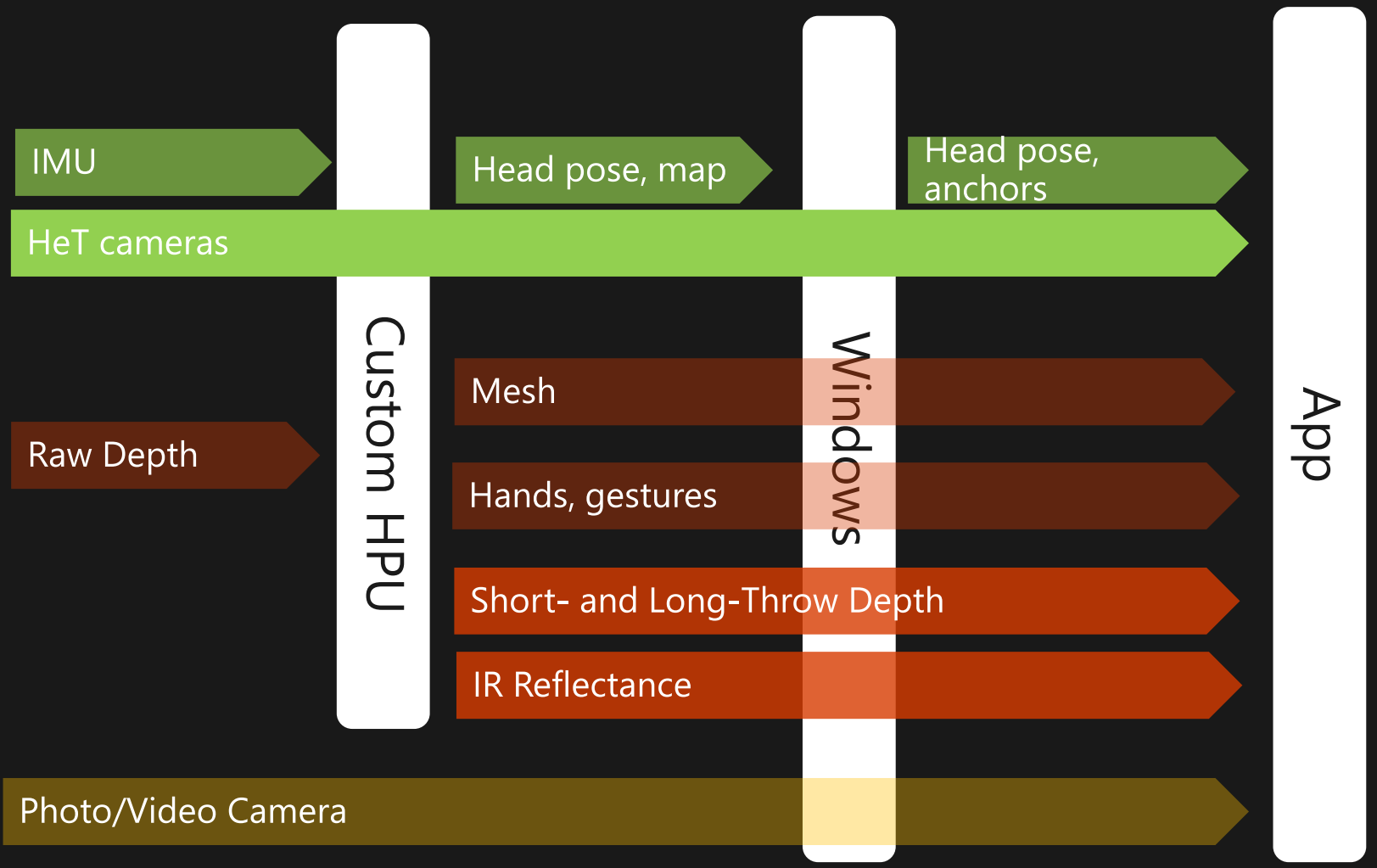
Windows

Head pose, anchors

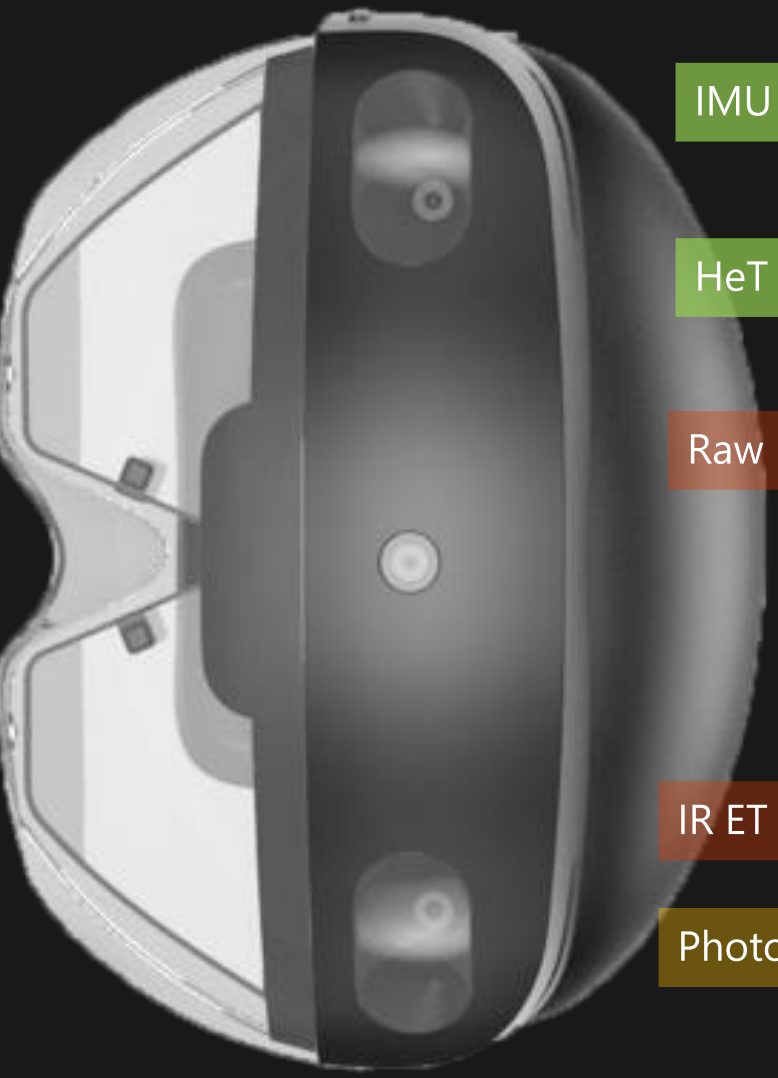
App

Photo/Video Camera

HoloLens Research Mode



HoloLens 2 Research Mode



IMU

HeT cameras

Raw Depth

IR ET cameras

Photo/Video Camera

Custom HPU

Head pose, map

Mesh

Articulated hands

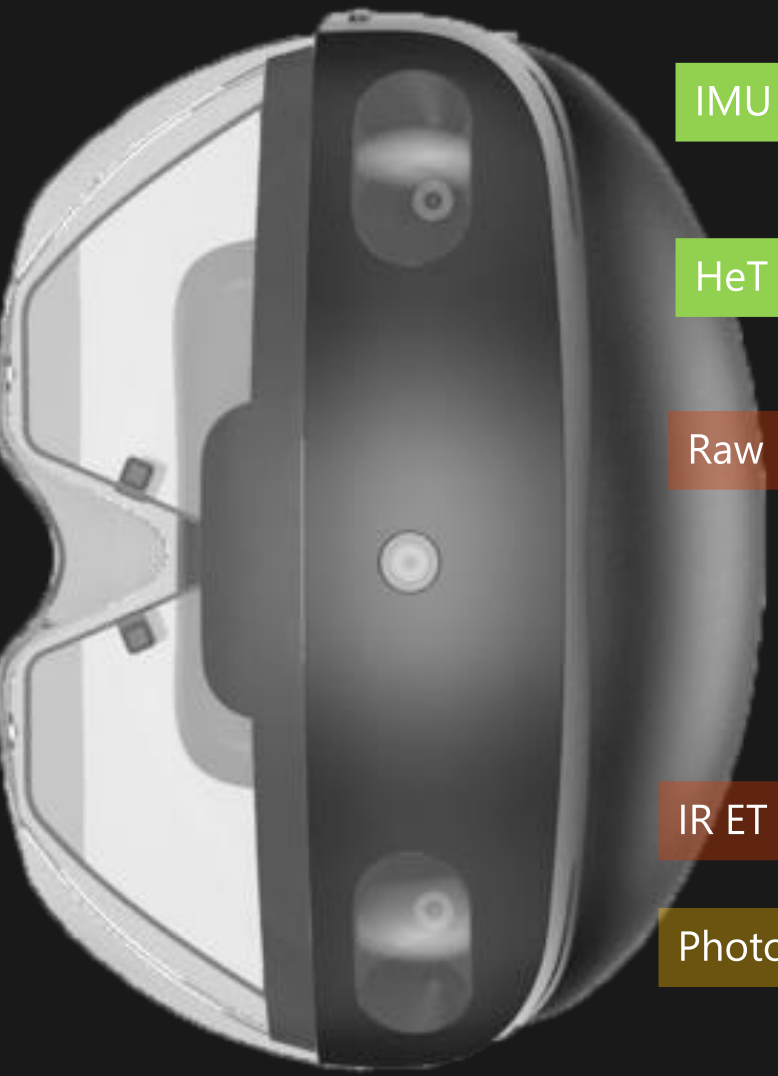
Eye gaze

Windows

Head pose, anchors

App

HoloLens 2 Research Mode



IMU

Head pose, map

Head pose, anchors

HeT cameras

Raw Depth

Mesh

Articulated hands

"Short"- and Long-Throw Depth

IR Reflectance

IR ET cameras

Eye gaze

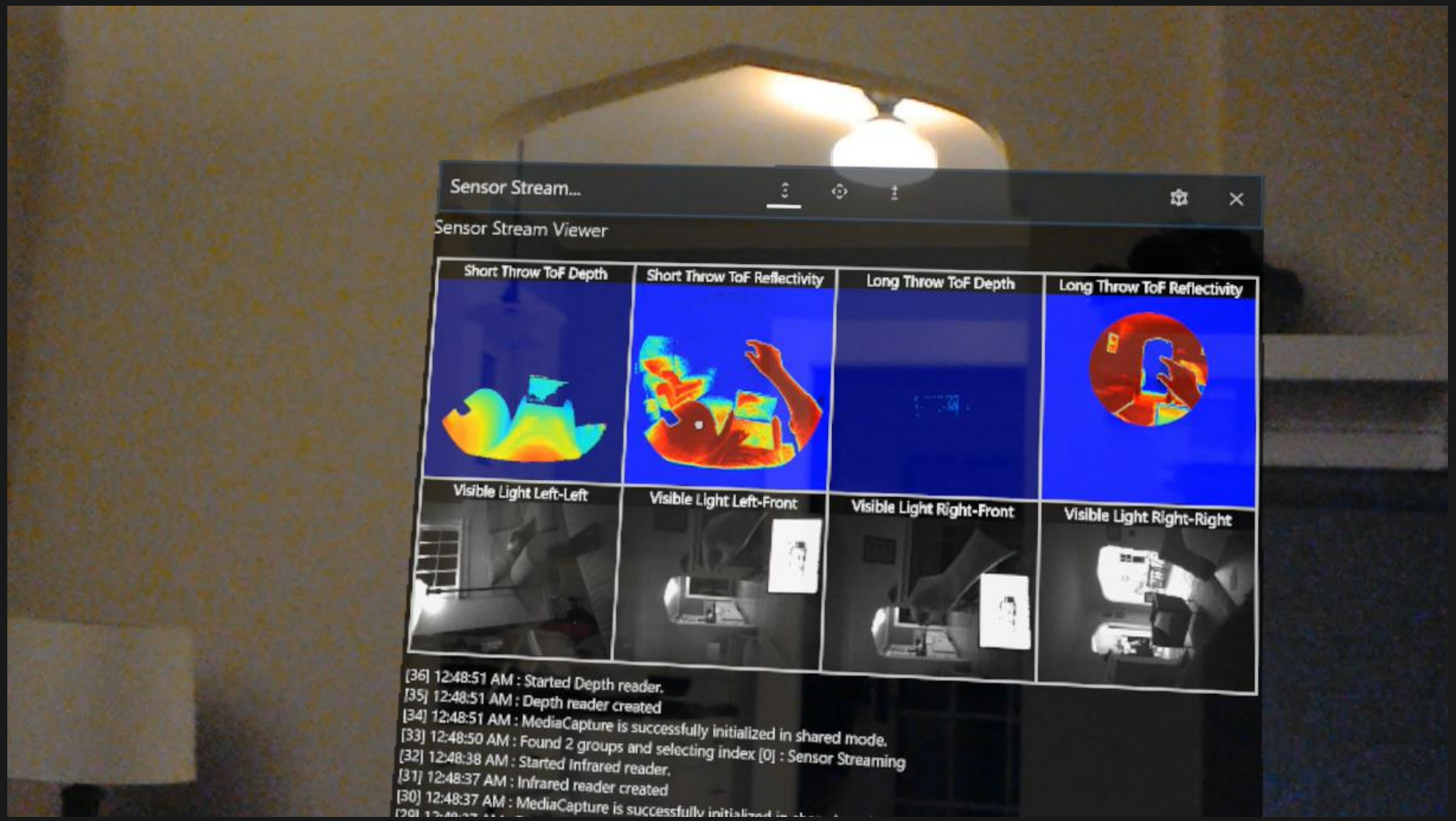
Photo/Video Camera

ustom HPU

Windows

App

Research mode Sensor Stream test app



Check out https://github.com/Microsoft/HoloLensForCV

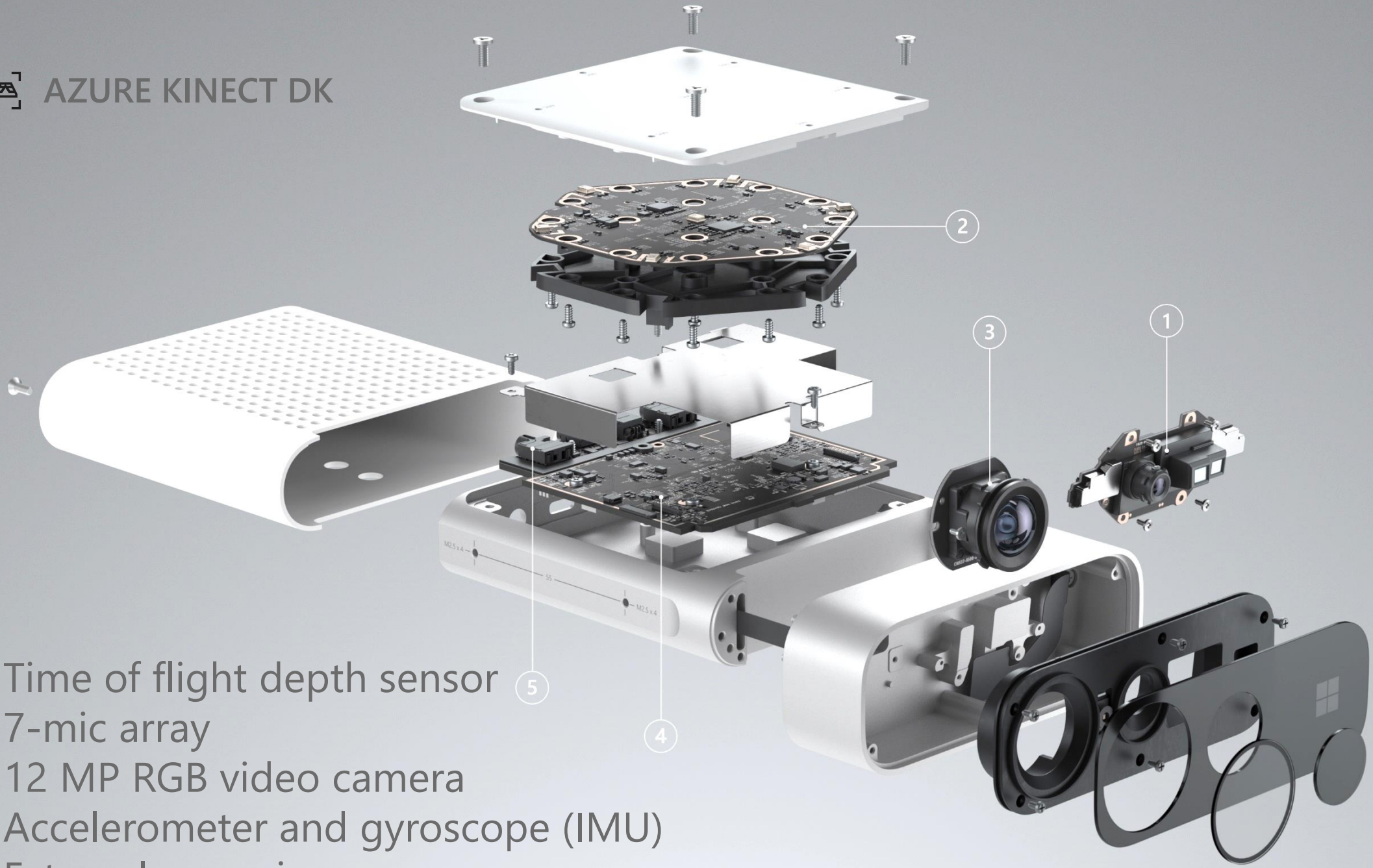
Person identification and tracking

AZURE KINECT DK

Build computer vision and speech models using a developer kit with advanced AI sensor

- Get started with a range of SDKs, including an open-source Sensor SDK.
- Experiment with multiple modes and mounting options.
- Add cognitive services and manage connected PCs with easy Azure integration.

AZURE KINECT DK



(1) Time of flight depth sensor
(2) 7-mic array
(3) 12 MP RGB video camera
(4) Accelerometer and gyroscope (IMU)
(5) External sync pins

Tutorial overview

Introduction

HoloLens2 Research Mode

- API overview
- HoloLens2forCV repository

Azure Kinect

- Open Source Sensor SDK
- Body Tracking SDK
- Demos