

RGBD Tutorial

Pat O'Keefe

[RGB]D cameras

- Primesense acquired by Apple
- Structure Sensor from Occipital
- Softkinetic
- PMD
- Mantis Vision
- Google's Project Tango

OpenNI

Occipital continues to maintain OpenNI. Submit your issues and pull requests!



STRUCTURE

[HOME](#) [DEVELOPERS](#) [PRESS](#) [PRE-ORDER](#)

“The rumors of my death have been greatly exaggerated...”

When we launched the Structure Sensor, we made a promise to support OpenNI. In order to keep doing that, we've created this OpenNI resource page (and a GitHub repo) to provide a one-stop resource for the community to get OpenNI 2 binaries and documentation. We'll be hosting this page for the indefinite future, and we'll keep it updated as OpenNI continues to evolve. This page was last updated on May 5th, 2014.

GUESS WHAT? WE'RE HOSTING

OpenNI 2 SDK Binaries & Docs

Mac OS X

- [OpenNI 2.2.0.33 Beta](#) 4.8mb, zipped
- 64-bit Intel processors only.
- No additional driver required.



Windows

- [OpenNI 2.2.0.33 Beta \(x86\)](#) 8.2mb, zipped
- [OpenNI 2.2.0.33 Beta \(x64\)](#) 8.2mb, zipped
- Includes Windows driver.



Linux

- [OpenNI 2.2.0.33 Beta \(x86\)](#) 2.9mb, zipped
- [OpenNI 2.2.0.33 Beta \(x64\)](#) 3.0mb, zipped
- [OpenNI 2.2.0.33 Beta \(ARM\)](#) 2.7mb, zipped
- No additional driver required.



Android

OpenNI Documentation

OpenCV RGBD Module

- Normals
- Planes
- DepthCleaner
- Depth image to 3D Cloud
- Odometry
 - ICP / RGBD / RGBD + ICP



Registering Depth to an External Camera



Registering Depth to an External Camera

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = z K_{ir}^{-1} \begin{bmatrix} u_{ir} \\ v_{ir} \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} u_{rgb} \\ v_{rgb} \\ 1 \end{bmatrix} = K_{rgb} [R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

(Don't forget about distortion)

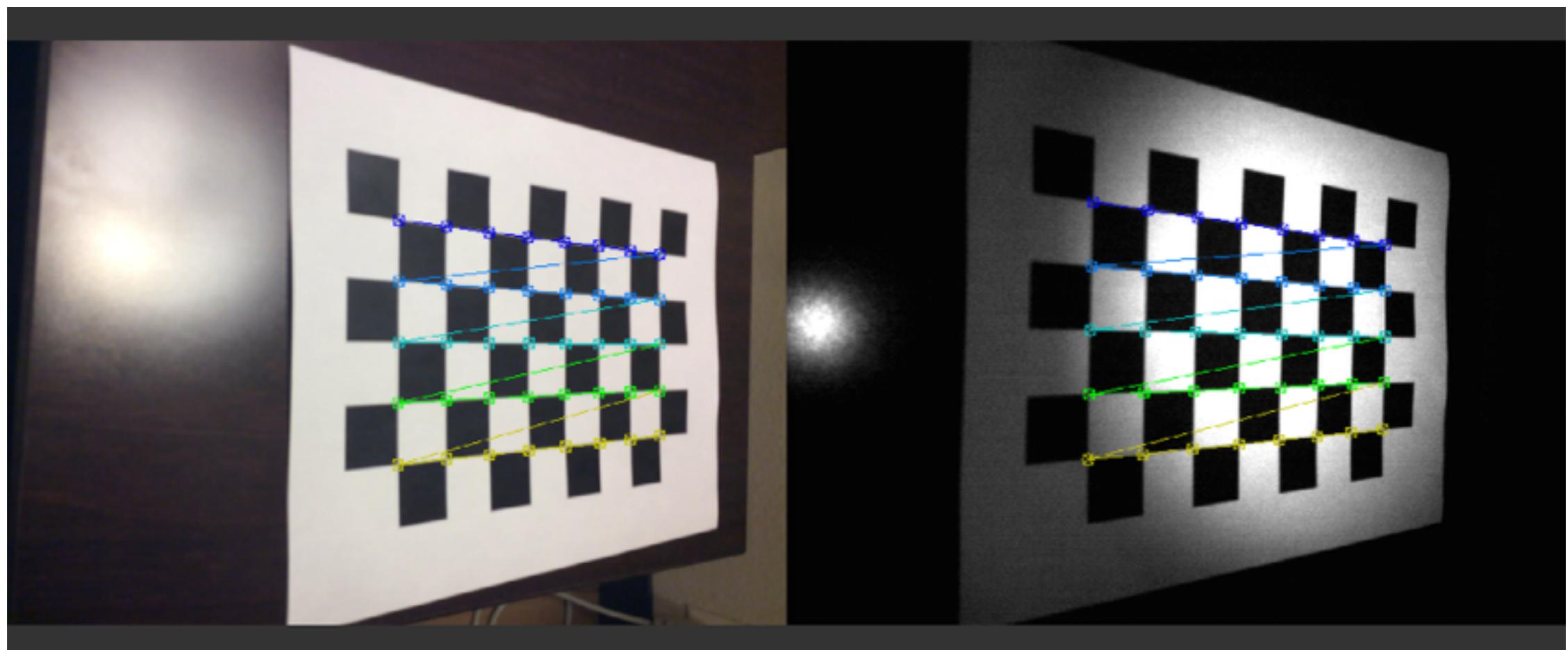
Registering Depth to an External Camera

- Unknowns are: K_{ir} , K_{rgb} , Rt
- Can be found using traditional camera/stereo calibration (and knowing some magic offsets)
- Ambient infrared illumination can be a problem

Registering Depth to an External Camera

- `cv::findChessboardCorners(...)` or
`cv::findCirclesGrid(...)`
- `cv::stereoCalibrate(...)`
`//CALIB_FIX_INTRINSIC`

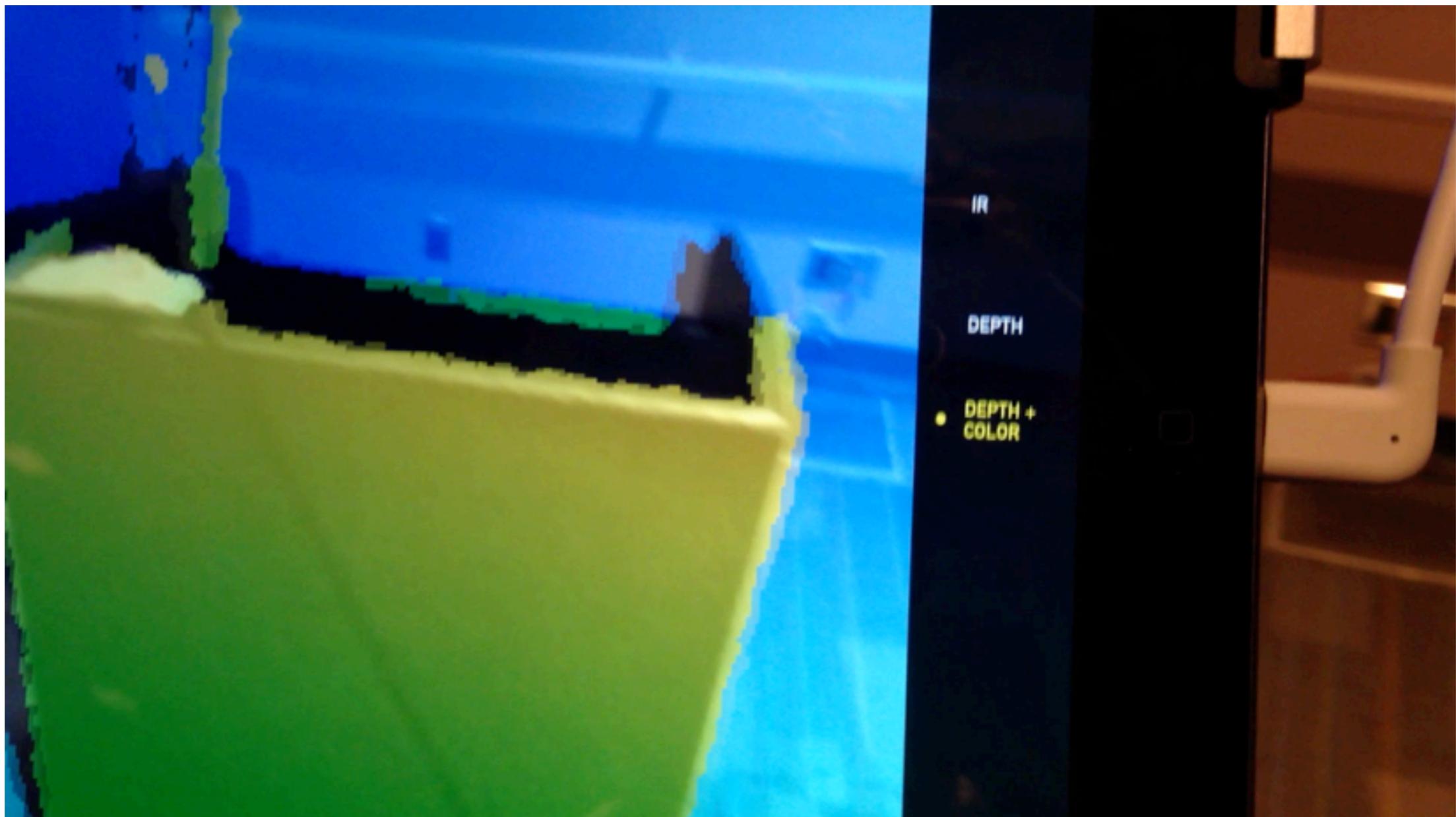
Registering Depth to an External Camera



Registering Depth to an External Camera

Putting it all together (code!)

Registering Depth to an External Camera



RGBD Odometry

- Depth makes odometry "easy". OpenCV makes it even easier.
- We've got scale!
- Needed: Great registration + frame synchronization

RGBD Odometry

```
// Creating an instance of RGBD Odometry
RgbdOdometry odometryAlgorithm;

// Ensure we're using the right camera matrix
odometryAlgorithm.set("cameraMatrix", registeredCameraMatrix);

// The algorithm needs two registered depth+color frames as input
prevOdometryFrame = Ptr<OdometryFrame>(new OdometryFrame());
currOdometryFrame = Ptr<OdometryFrame>(new OdometryFrame());

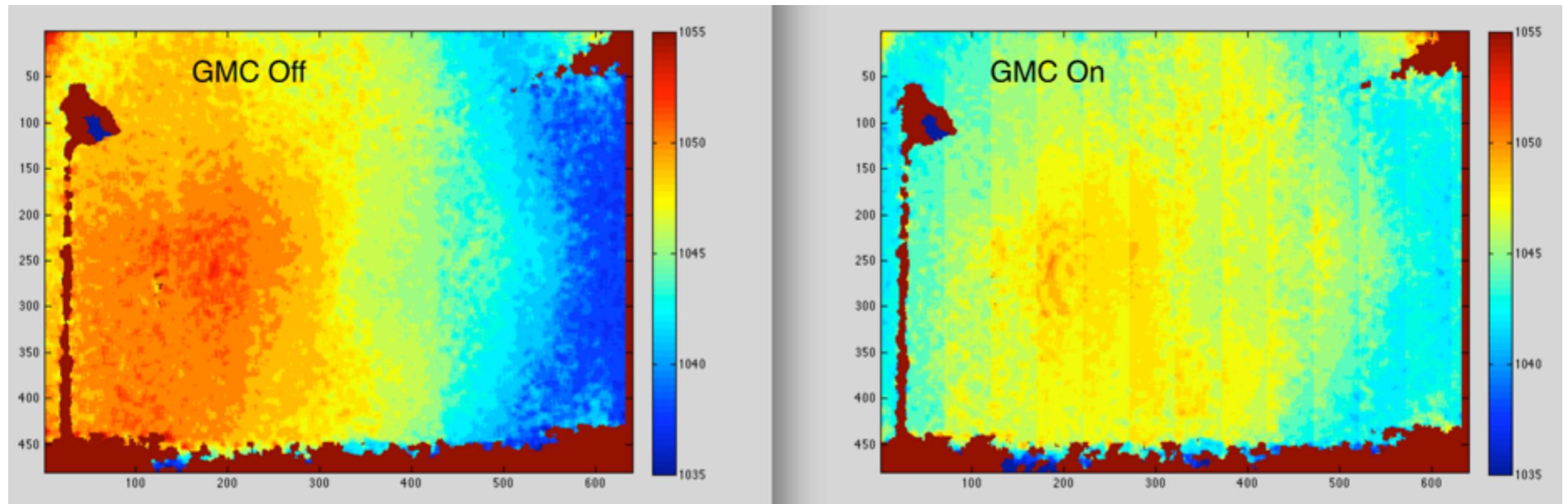
// Fill frames with data...

// Single call to compute the RBT between frames
bool res = odometryAlgorithm.compute(currOdometryFrame, prevOdometryFrame, deltaRt);
```

RGBD Odometry

- Multiple flavors available
 - ICP
 - RGBD
 - RGBD + ICP

Depth Accuracy



Mobile Depth Sensing

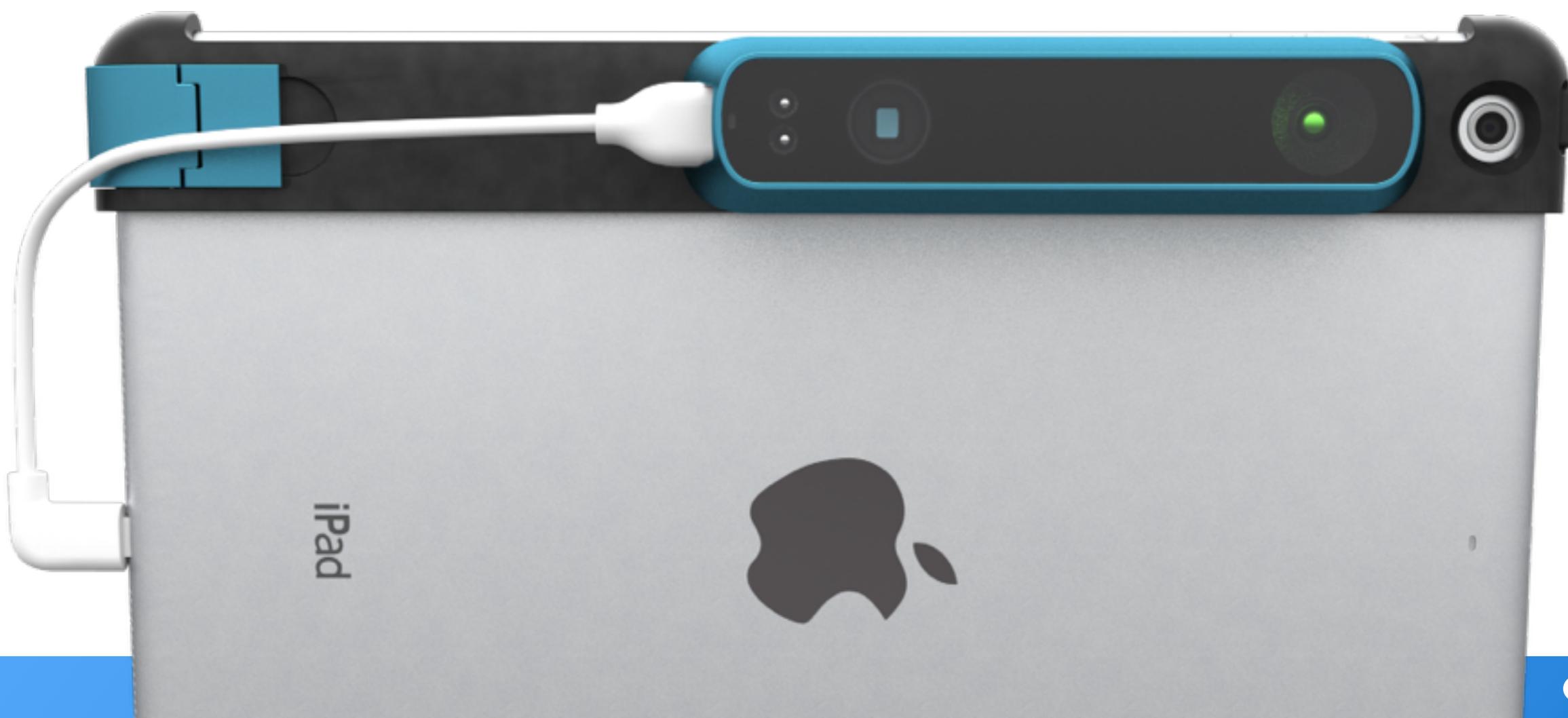
Despite a lot of cool hacks ...



There was no 3D sensor designed specifically for mobile devices.



Structure Sensor



OCCIPITAL



- Sample code for **augmented reality**.
- Sample code for **3D scanning**.
- SLAM built-in.
- Low-level access.



Create by Structure SDK user Bob Berkebile in ~6 hours

https://www.youtube.com/watch?v=_xhhXVJ89FA





STRUCTURE SDK

SOFTWARE DEVELOPMENT KIT

Let's do some coding!

