3D Scanning Using STRUECTURE Sensor

July 5 2017

Jungwon Kang

Contents

3D Scanning for Modeling

- **Modeling of Indoor Space, Object-of-Interest, and Many Other Things**

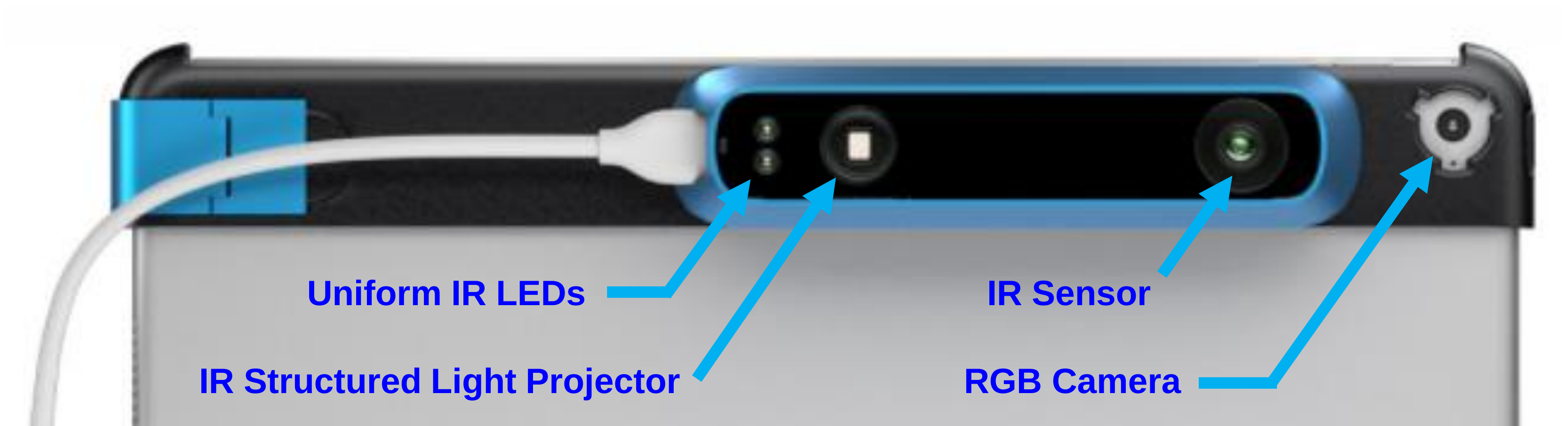
Commercial 3D Scanners

- **Many Commercial Ones, But Still Expensive (for Personal Use)**

Matterport Pro2 3D Camera (MC250):
USD 3995 [Device] + USD 100/Month [Data Processing Service]

STRUCTURE Sensor (1/3)

- **STRUCTURE Sensor: A Consumer-Grade 3D Scanner**
  - Small, lightweight IR-based 3D scanner
  - Provides RGB-D data (when combined with iPad RGB camera)
  - Developed by Occipital Inc. in 2014
  - USD 379

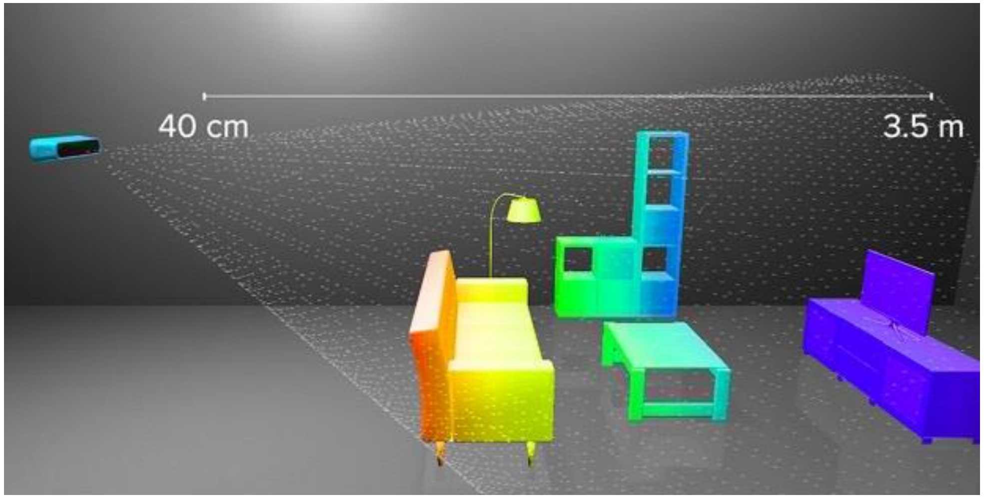


STRUCTURE Sensor (2/3)

- **Specification**

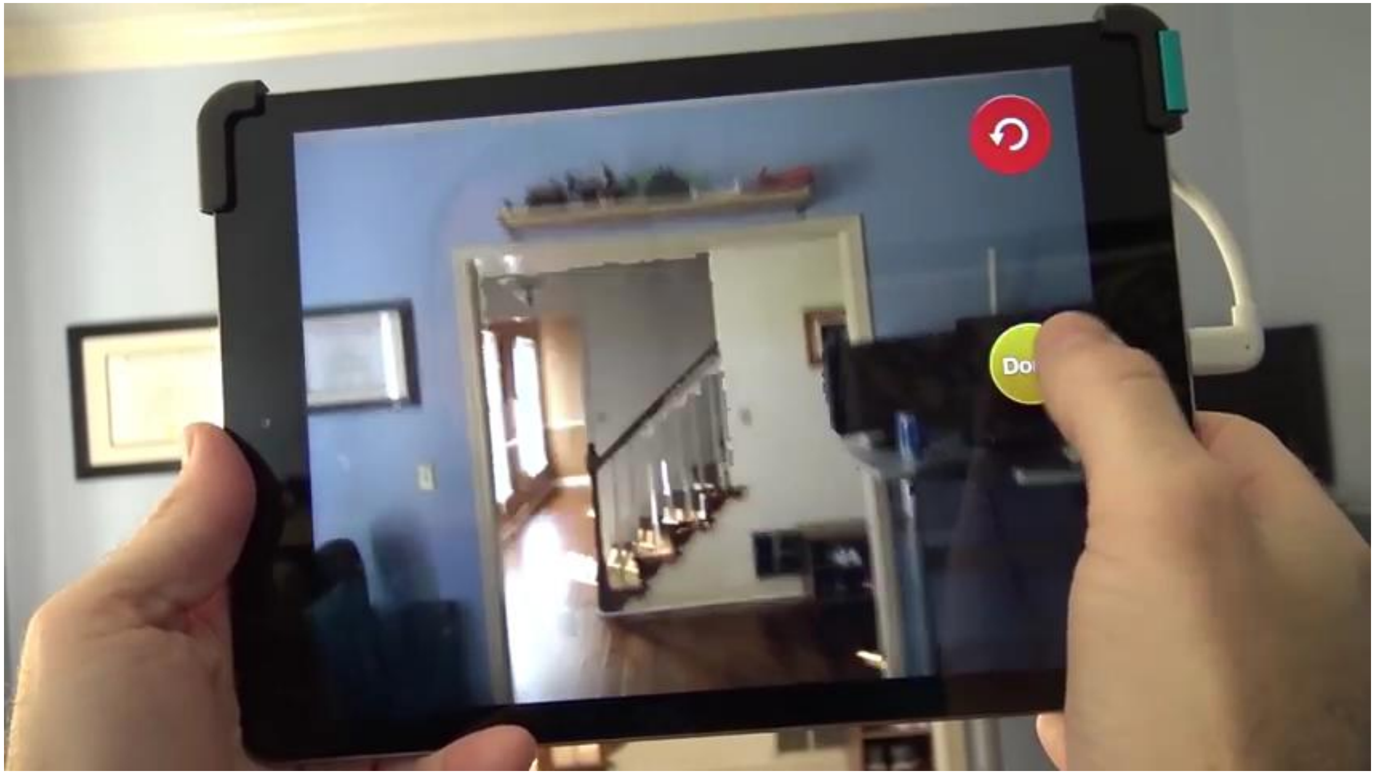
Length x Width x Height	119.2 mm x 27.9 mm x 29 mm
Weight	99.2 grams
Sensing distance	40 cm ~ 3.5 m
Precision	1% of measured distance
Resolution	VGA(640 x 480) / QVGA (320 x 240)
Framerate	30 / 60 frames per second
Battery Life	3-4 hours of active sensing
Field of View	58° (Horizontal), 45° (Vertical)

STRUCTURE sensor on a iPad



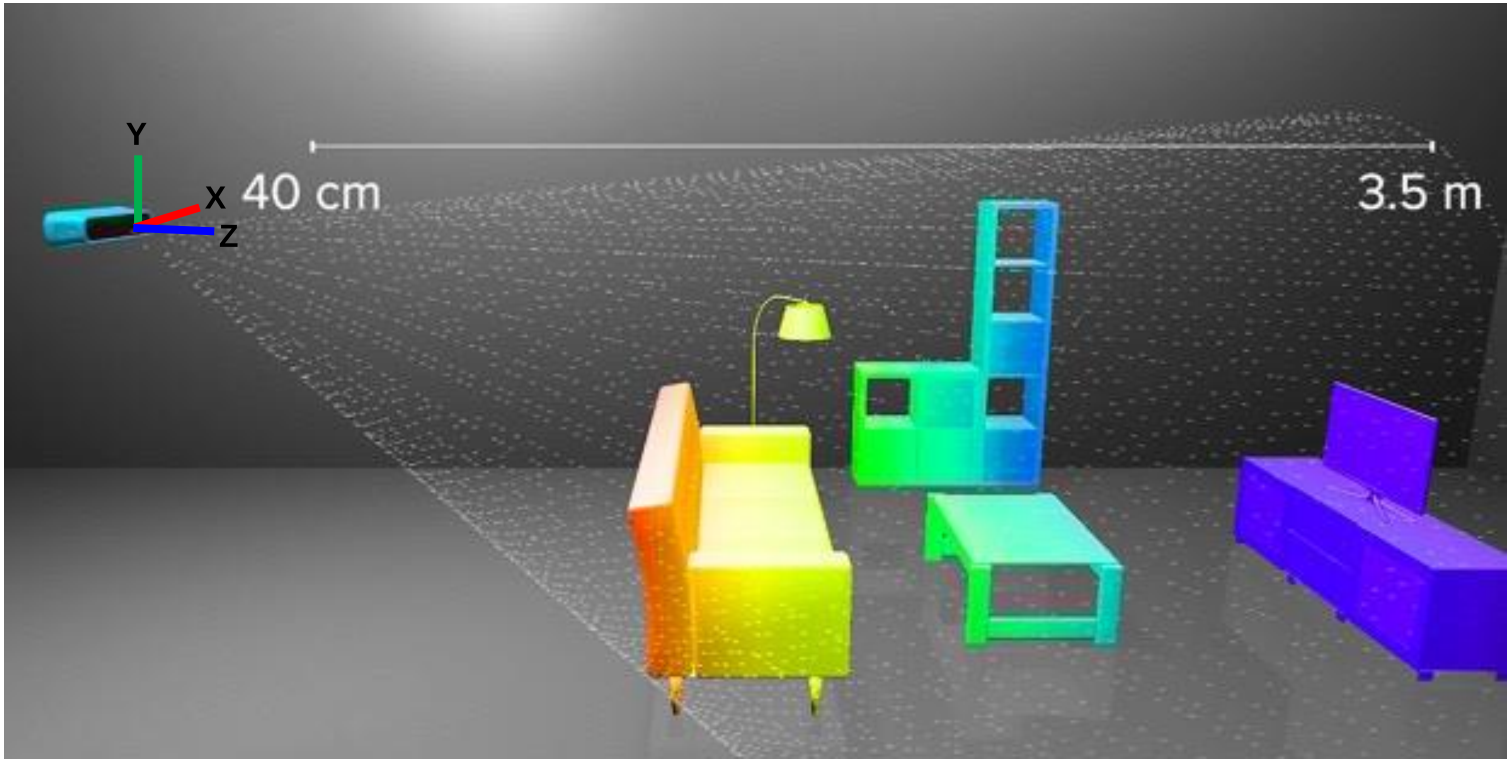
STRUCTURE Sensor (3/3)

- **What You Can Do Using STRUCTURE Sensor**
  - Scanning indoor space (using *Room Capture* app), object-of-interest (using *Scanner* app)
  - Augmented reality (using *Fetch* app)



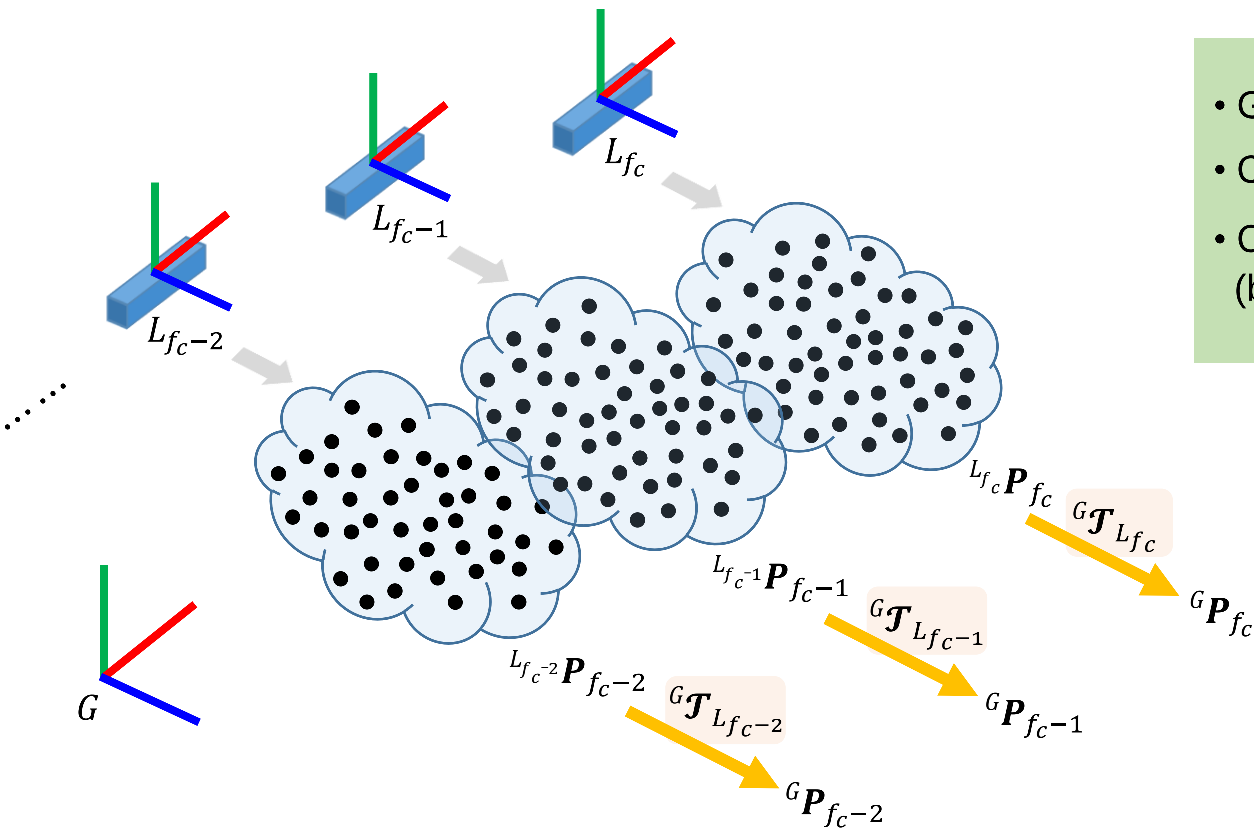
3D Modeling from Scanning (1/6)

- **Point Cloud Generation (At a Certain Frame)**
  - Generation of a set of points wrt a local sensor coordinate at a certain frame



3D Modeling from Scanning (2/6)

- **Concatenation of Point Cloud at Each Sensor Pose**
  - Formally, called point cloud registration
  - Registering point cloud wrt a local sensor coordinate in a global coordinate

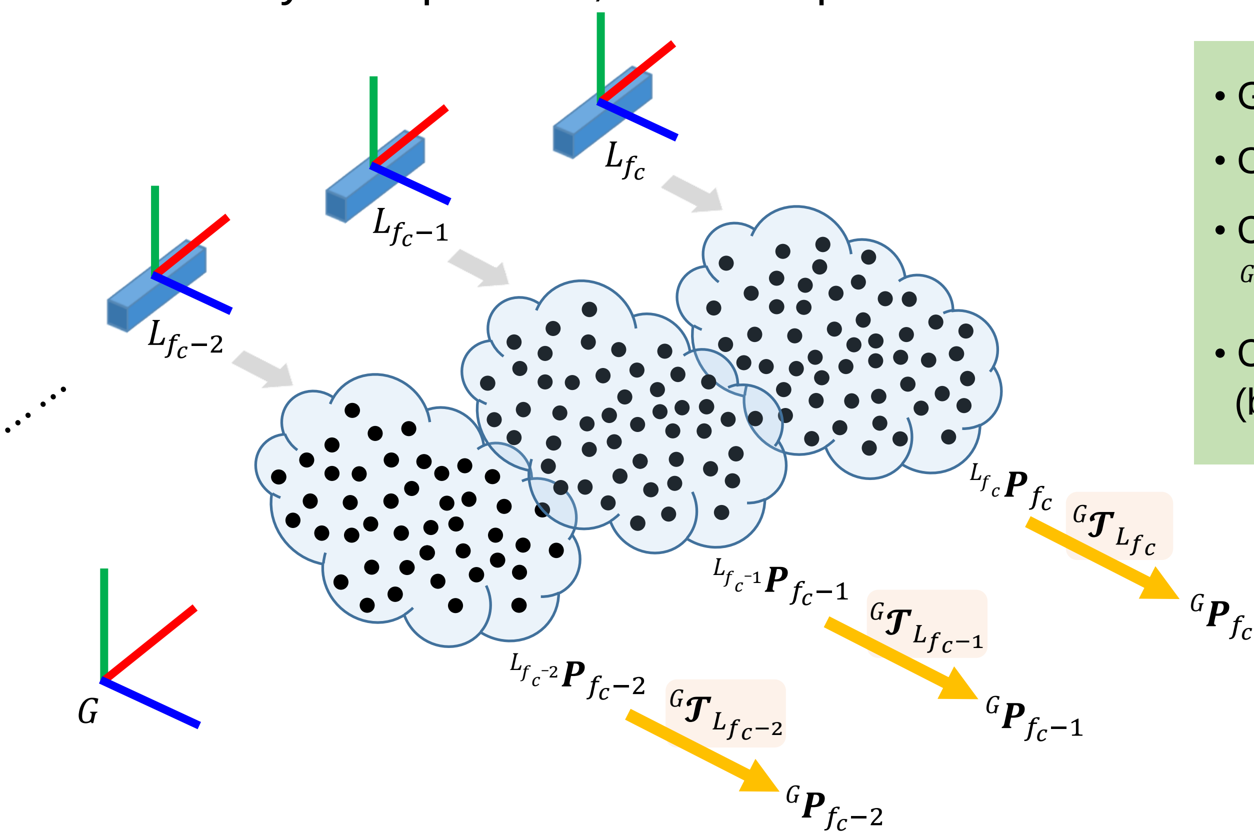


- Generate ${}^{L_f}\boldsymbol{P}_f$
- Compute ${}^{G}\boldsymbol{\mathcal{T}}_{L_f}$
- Convert ${}^{L_f}\boldsymbol{P}_f$ into ${}^{G}\boldsymbol{P}_f$ (by applying ${}^{G}\boldsymbol{\mathcal{T}}_{L_f}$ to ${}^{L_f}\boldsymbol{P}_f$)

$$ {}^{G}\boldsymbol{P} = \{{}^{G}\boldsymbol{P}_f\}_{f=1}^{f_c} $$

3D Modeling from Scanning (3/6)

- **Point Cloud Registration Based on Two Consecutive Frames**
  - ICP (Iterative Closest Point) algorithm for finding a transformation that best aligns one point cloud wrt the other point cloud.
  - Easy to implement, but error-prone due to accumulated error.

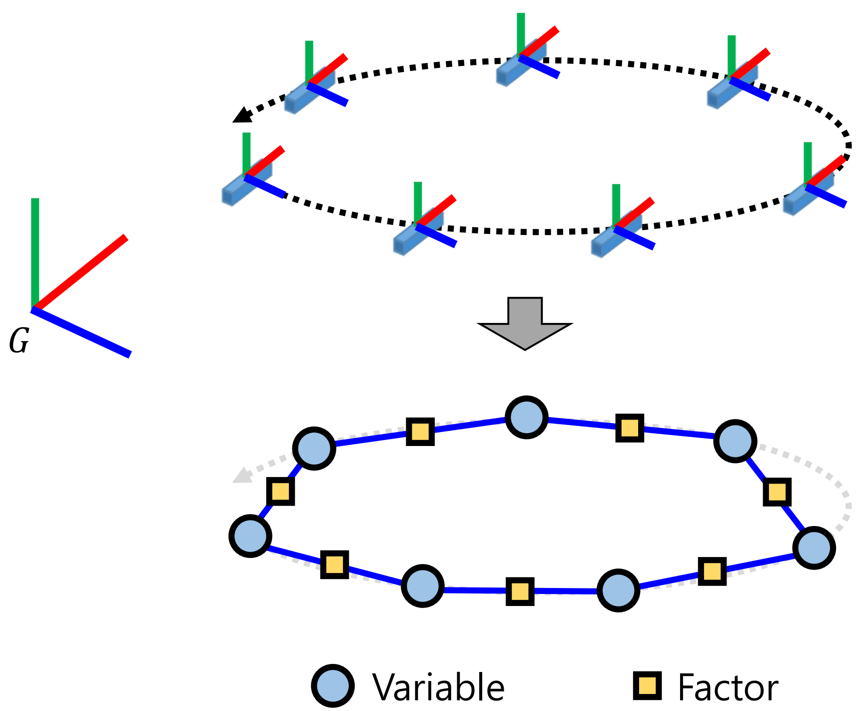


- Generate $^{L_f}\boldsymbol{P}_f$
- Compute $^{L_{f-1}}\boldsymbol{\mathcal{T}}_{L_f}$ using ICP
- Compute $^{G}\boldsymbol{\mathcal{T}}_{L_f}$ from
  $^{G}\boldsymbol{\mathcal{T}}_{L_f} = {}^{G}\boldsymbol{\mathcal{T}}_{L_0} \cdots\cdots {}^{L_{f-2}}\boldsymbol{\mathcal{T}}_{L_{f-1}} \, {}^{L_{f-1}}\boldsymbol{\mathcal{T}}_{L_f}$
- Convert $^{L_f}\boldsymbol{P}_f$ into $^{G}\boldsymbol{P}_f$ (by applying $^{G}\boldsymbol{\mathcal{T}}_{L_f}$ to $^{L_f}\boldsymbol{P}_f$)

$$^{G}\boldsymbol{P} = \{^{G}\boldsymbol{P}_f\}_{f=1}^{f_c}$$

3D Modeling from Scanning (4/6)

- **Point Cloud Registration Based on All Frame Adjustment**
  - Formally, called pose graph optimization
  - Using a factor graph where the *variables* are sensor poses to be estimated, and the *factors* are relative pose measurements between sensor poses
  - Reducing the accumulated errors when a loop closing is detected.



- Generate ${}^{L_f}\boldsymbol{P}_f$
- Compute ${}^{L_{f-1}}\boldsymbol{\mathcal{T}}_{L_f}$ using ICP
- Compute ${}^{G}\boldsymbol{\mathcal{T}}_{L_f}$ by solving

$$\mathcal{X}^* = \arg\min_{\mathcal{X}} \frac{1}{2}\sum_f \left\| h_f(\chi_f) - z_f \right\|_\Sigma^2$$

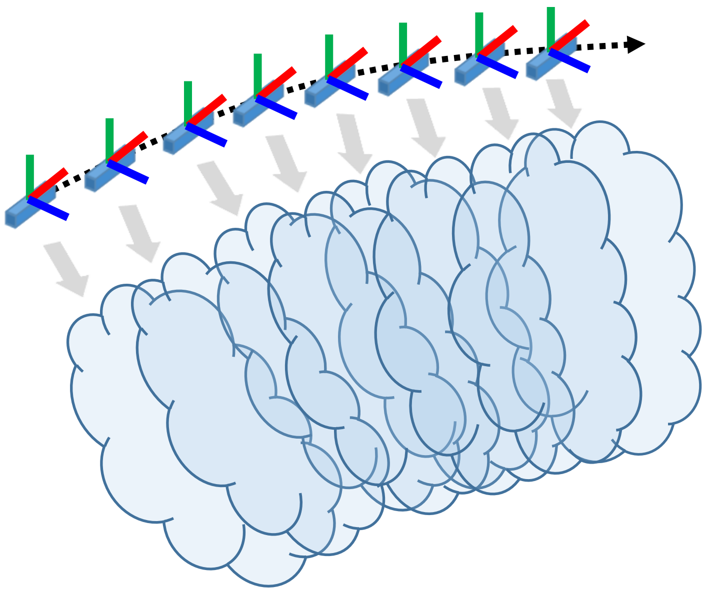
where $\mathcal{X}^* = \left\{ {}^{W}\boldsymbol{\mathcal{T}}_{L_f} \right\}_{f=1}^{f_c}$, $z_f = {}^{L_{f-1}}\boldsymbol{\mathcal{T}}_{L_f}$

- Convert ${}^{L_f}\boldsymbol{P}_f$ into ${}^{G}\boldsymbol{P}_f$ (by applying ${}^{G}\boldsymbol{\mathcal{T}}_{L_{f_c}}$ to ${}^{L_f}\boldsymbol{P}_f$)

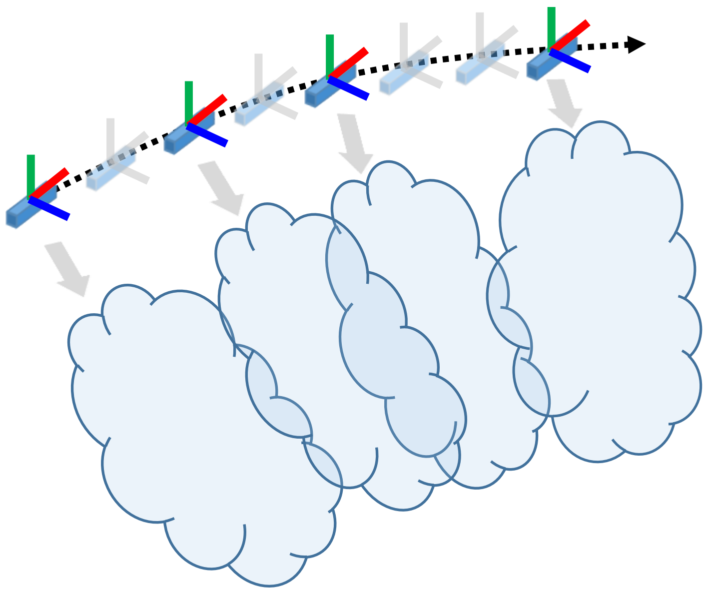
$${}^{G}\boldsymbol{P} = \left\{ {}^{G}\boldsymbol{P}_f \right\}_{f=1}^{f_c}$$

3D Modeling from Scanning (5/6)

- **Using Keyframes for Pose Graph Optimization**
  - Selecting informative frames as *keyframes* for efficient operation with low data redundancy
  - Using a factor graph for sensor poses at keyframes



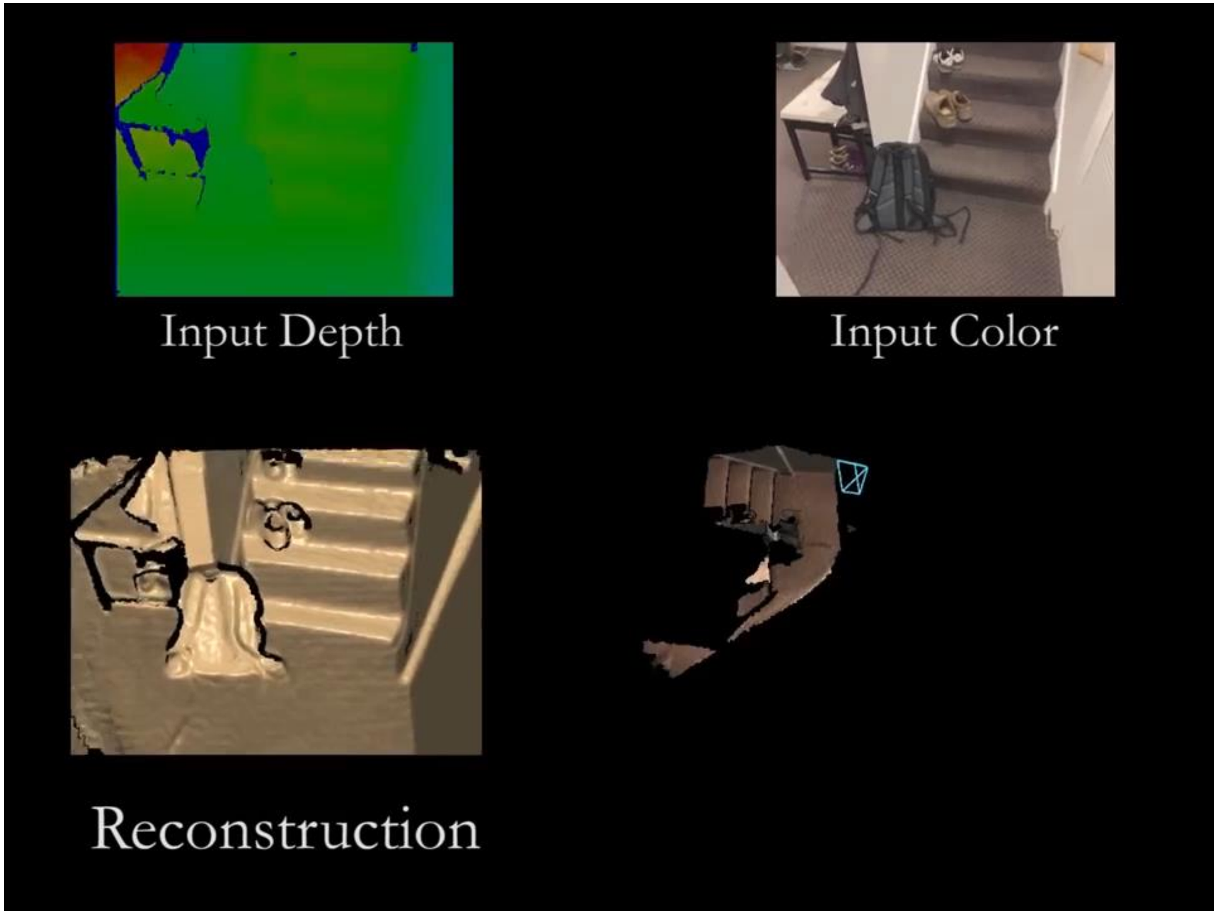
Using all frames



Using keyframes

3D Modeling from Scanning (6/6)

- **Point Cloud Registration Based on All Frame Adjustment**



[A. Dai, 2017]

Conclusion

- **STRUCTURE Sensor**
  - A consumer-grade 3D scanner (for personal use)
  - Provides several apps for modeling of indoor space, object-of-interest, and augmented reality.

- **Some Notes on Your Practice**
  - See how certain materials affect scanning.
    - ✓ Tip: Try scanning a mirror or a glass.
  - See when generated 3D model is inaccurate.
    - ✓ Tip: In order to enhance the quality of 3D model, please simplify your own motion while scanning.