Check out our GitHub!
https://github.com/xorthat/MobileSLAM

LSD Visual Odometry on a Mobile Device

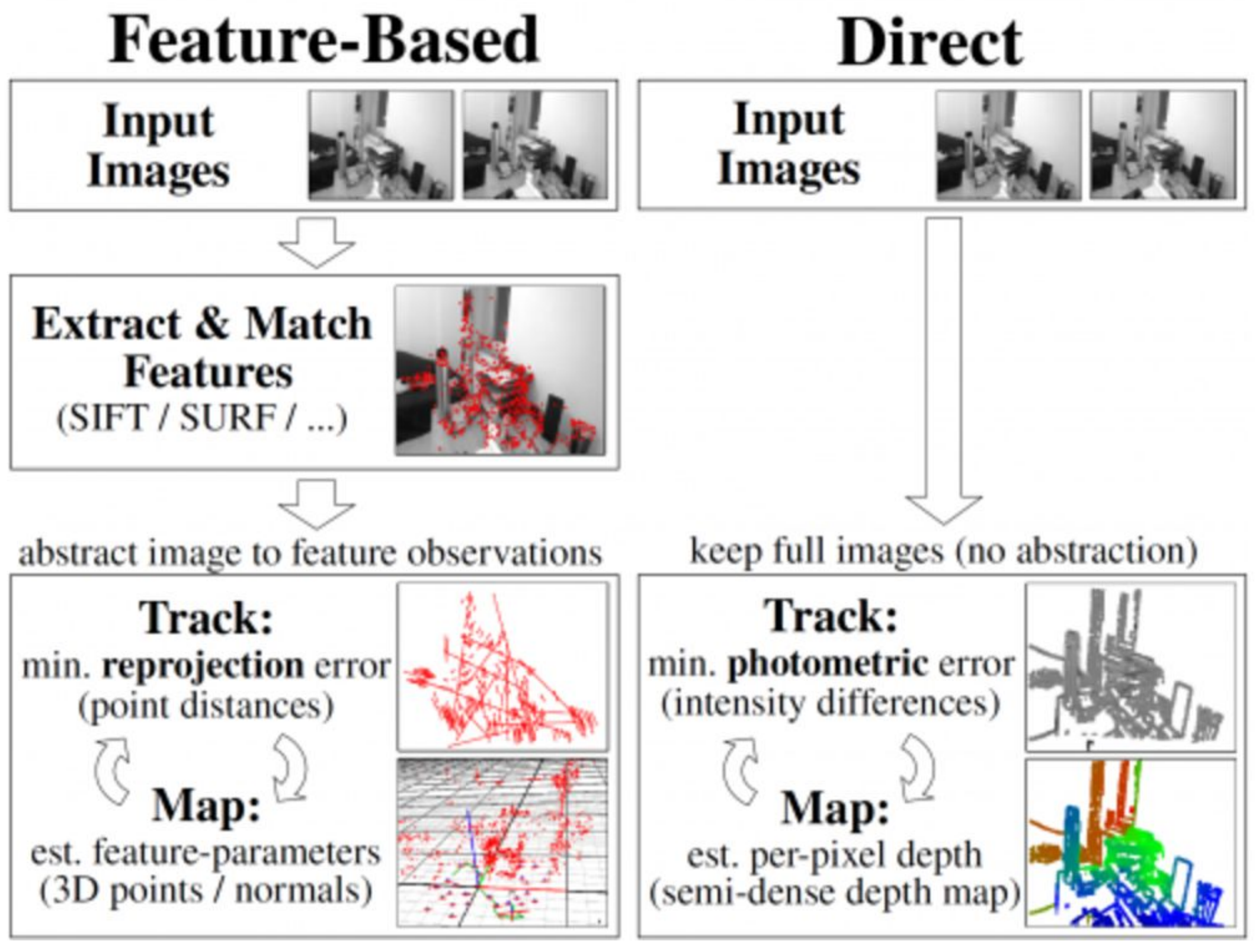
Jennifer Lake (jelake) & Guanhang Wu (guanhanw)

Check out our YouTube clip!
https://www.youtube.com/watch?v=gOveXpLiBqw

MOBILE LSD

Applications: Self-driving car, Augmented Reality, Virtual Reality

Direct vs. Feature-Based Methods



Feature-Based

Input Images

Extract & Match Features (SIFT / SURF / ...)

abstract image to feature observations

Track: min. **reprojection** error (point distances)

Map: est. feature-parameters (3D points / normals)

Direct

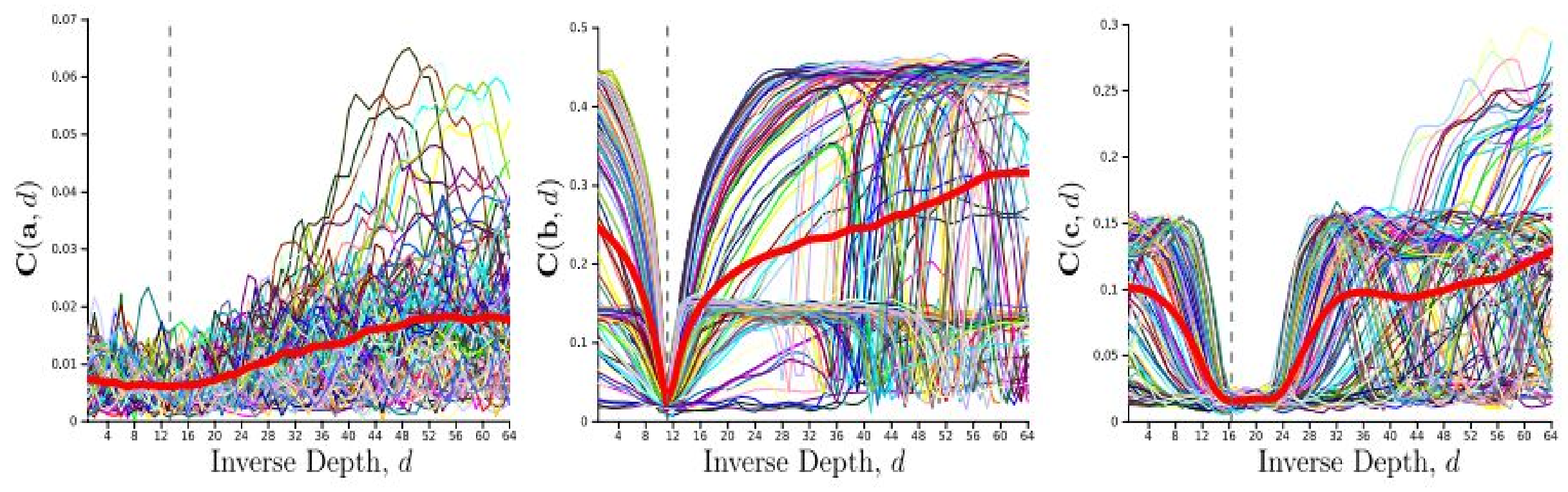
Input Images

keep full images (no abstraction)

Track: min. **photometric** error (intensity differences)

Map: est. per-pixel depth (semi-dense depth map)

Consensus Voting in Direct Methods



Example of low texture

Example of corner-like texture

Example of edge-like texture

Photometric Error Minimization

$$E(\xi) = \sum_i (I_{ref}(p_i) - I(\omega(p_i, D_{ref}(p_i), \xi)))^2$$

$$\delta\boldsymbol{\xi}^{(n)} = -(\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{r}(\boldsymbol{\xi}^{(n)}) \quad \text{with} \quad \mathbf{J} = \left.\frac{\partial \mathbf{r}(\boldsymbol{\epsilon}\circ\boldsymbol{\xi}^{(n)})}{\partial\boldsymbol{\epsilon}}\right|_{\boldsymbol{\epsilon}=0}$$

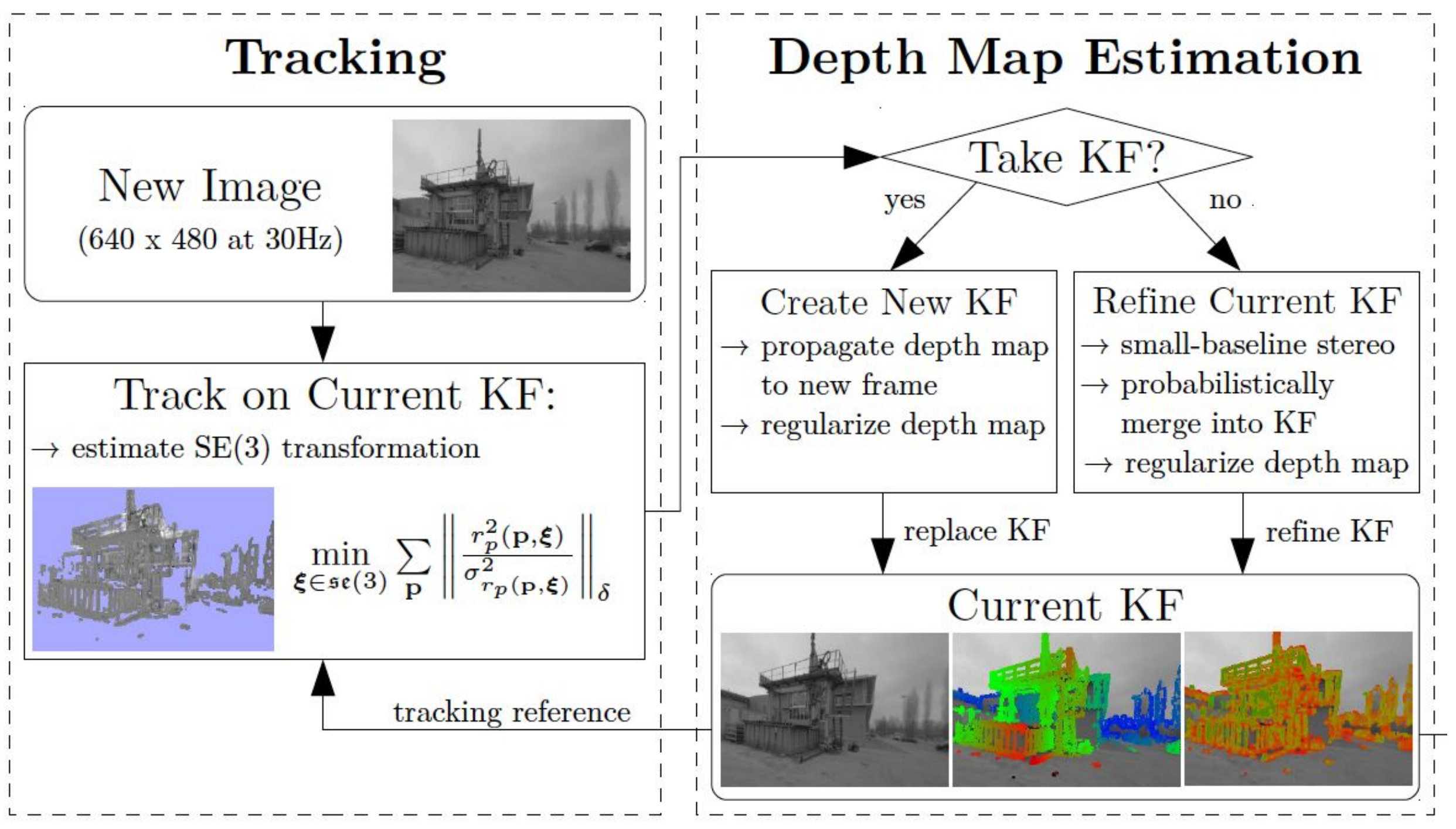
$$\delta\boldsymbol{\xi}^{(n)} = -(\mathbf{J}^T\mathbf{W}\mathbf{J})^{-1}\mathbf{J}^T\mathbf{W}r(\boldsymbol{\xi}^{(n)}).$$

Inverse Depth Estimation

$$\lambda(p) = argminC(p, \lambda*)$$

$$C(p, \lambda*) = ||I_{ref}(p) - I(\omega(p_i, \lambda*(p_i), \xi))||_1$$

System Overview



Tracking

New Image (640 x 480 at 30Hz)

Track on Current KF:
→ estimate SE(3) transformation

$$\min_{\boldsymbol{\xi}\in\mathfrak{se}(3)} \sum_{\mathbf{p}} \left\| \frac{r_p^2(\mathbf{p},\boldsymbol{\xi})}{\sigma_{r_p(\mathbf{p},\boldsymbol{\xi})}^2} \right\|_\delta$$

tracking reference

Depth Map Estimation

Take KF?

yes

no

Create New KF
→ propagate depth map to new frame
→ regularize depth map

Refine Current KF
→ small-baseline stereo
→ probabilistically merge into KF
→ regularize depth map

replace KF

refine KF

Current KF

Supported Devices: iPad, iPhone

Frameworks and Libraries Used

boost

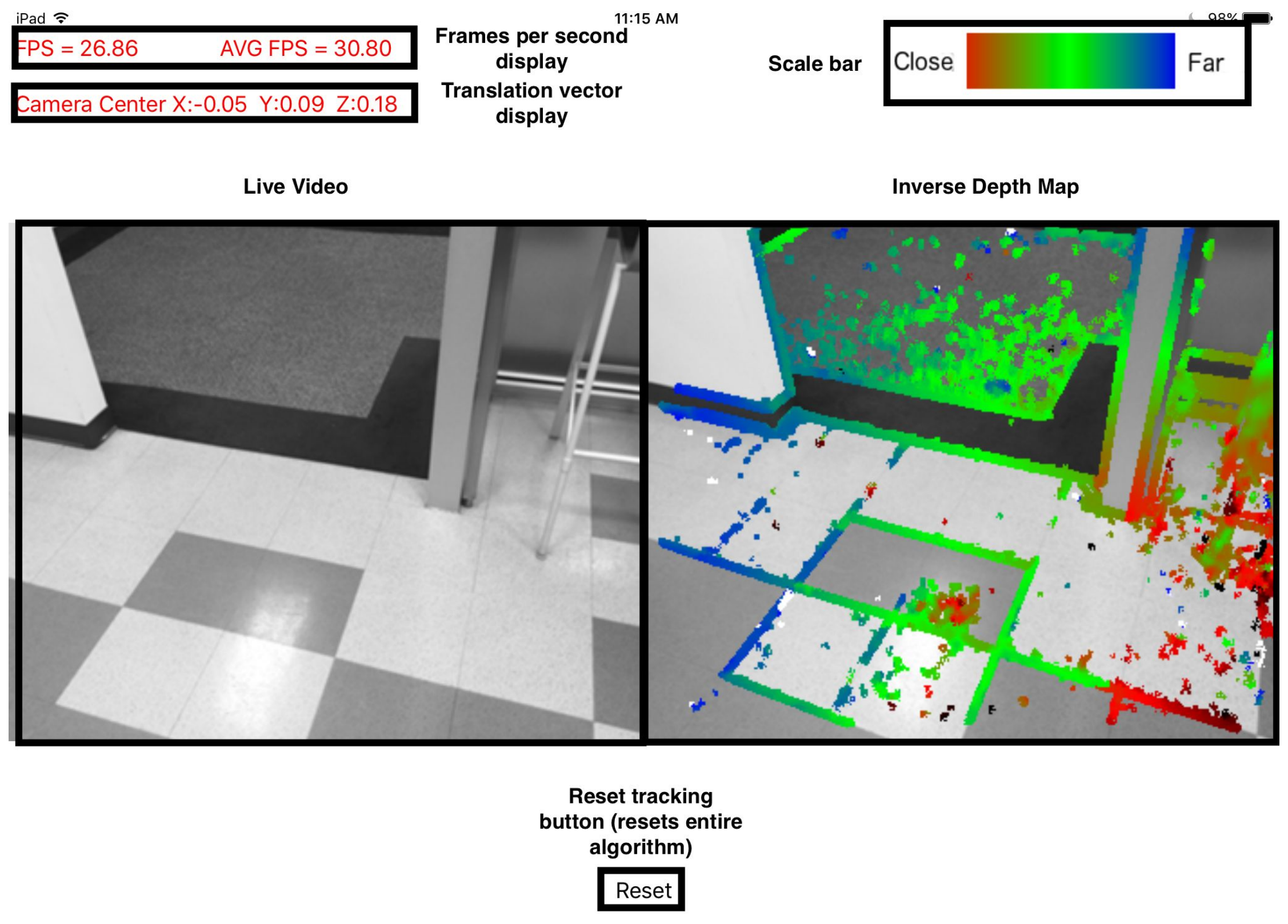
Sophus

Objective-C

OpenCV

Eigen

User Interface



FPS = 26.86 AVG FPS = 30.80

Frames per second display

Camera Center X:-0.05 Y:0.09 Z:0.18

Translation vector display

Scale bar

Close Far

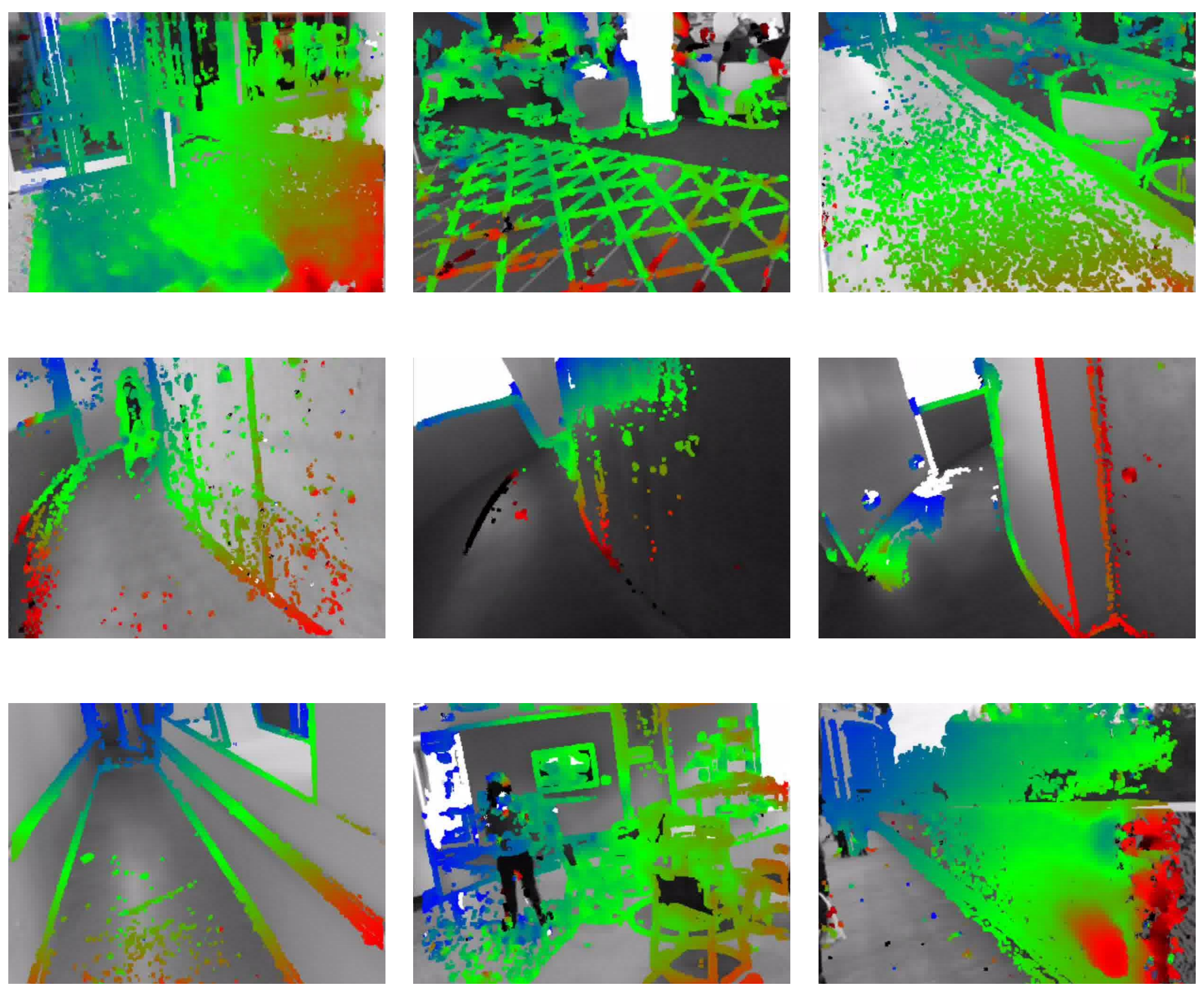
Live Video

Inverse Depth Map

Reset tracking button (resets entire algorithm)

Reset

Results: Gates Inverse Depth Map



Performance Results

Optimization Level	Flag	FPS
Fastest, aggressive optimizations	-Ofast	31.20
Fastest, smallest	-Os	27.32
Fastest	-O3	30.62
Faster	-O2	30.61
Fast	-O,-O1	2.23
None	-O0	0.63

References

D. A. Beman Dawes et al. Boost c++ libraries. `http://www.boost.org`, 2001.

G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

J. Engel, T. Schöps, and D. Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision (ECCV)*, September 2014.

J. Engel, J. Sturm, and D. Cremers. Semi-dense visual odometry for a monocular camera. In *IEEE International Conference on Computer Vision (ICCV)*, Sydney, Australia, December 2013.

O. S. R. Foundation. Robot Operating System. `www.ros.org`, 2016. [Online; accessed 08-Dec-2016].

G. Guennebaud, B. Jacob, et al. Eigen v3. `http://eigen.tuxfamily.org`, 2010.

T. Horikawa. Camera calibration ios. `https://github.com/thorikawa/camera-calibration-ios`, 2016.

A. Inc. Tuning for performance and responsiveness. `https://developer.apple.com/library/content/documentation/General/Conceptual/MOSXAppProgrammingGuide/Performance/Performance.html`, 2015.

R. A. Newcombe, S. J. Lovegrove, and A. J. Davison. Dtam: Dense tracking and mapping in real-time. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 2320–2327, Washington, DC, USA, 2011. IEEE Computer Society.

D. Rosser. ofxiosboost. `https://github.com/danoli3/ofxiOSBoost`, 2016.

T. Schöps, J. Engel, and D. Cremers. Semi-dense visual odometry for AR on a smartphone. In *International Symposium on Mixed and Augmented Reality*, September 2014.

H. Strasdat. Sophus. `https://github.com/strasdat/Sophus`, 2016.