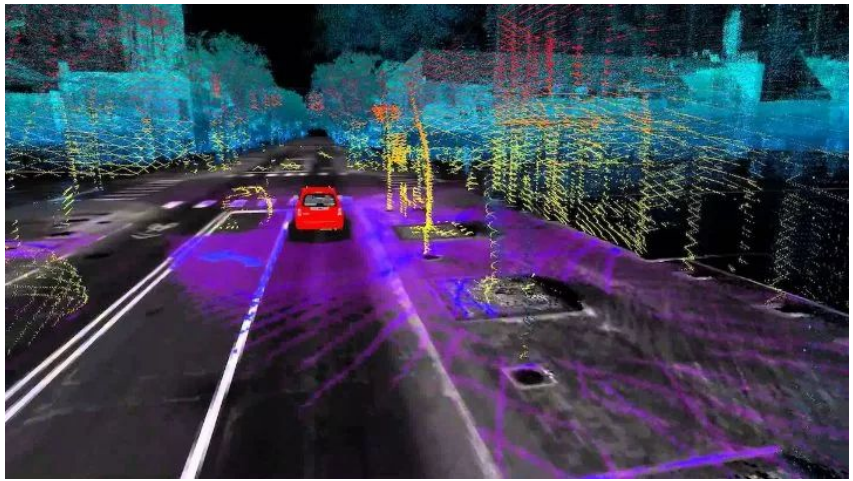
iMAP: Implicit Mapping and Positioning in Real-Time

Presenter: Geethika Hemkumar

September 7, 2023

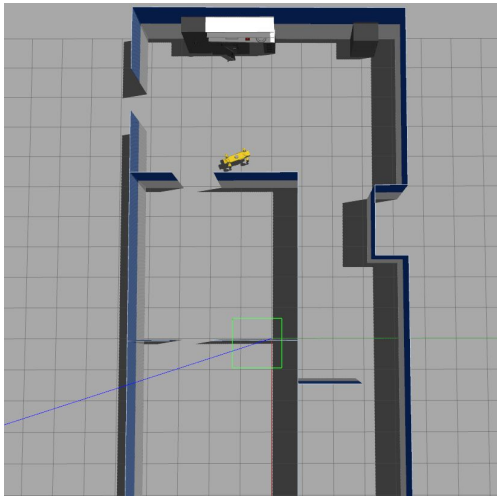
Simultaneous Localization and Mapping (SLAM)

- Process where robot builds a map of its surroundings and keeps track of its location at the same time

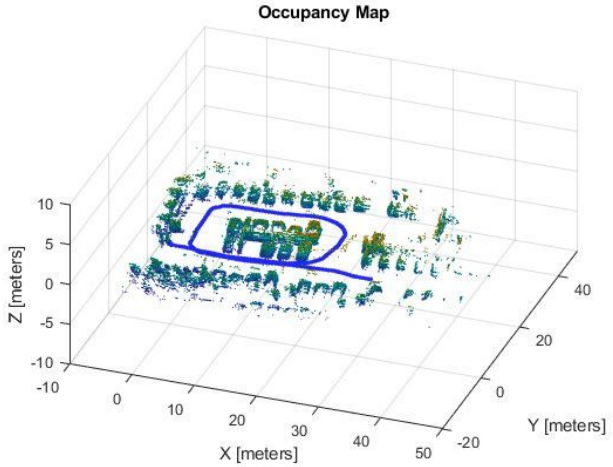


Why is SLAM Important?

- Robots must understand both where they are and what their surroundings are when navigating in environments



Gazebo simulation of apartment in AHG



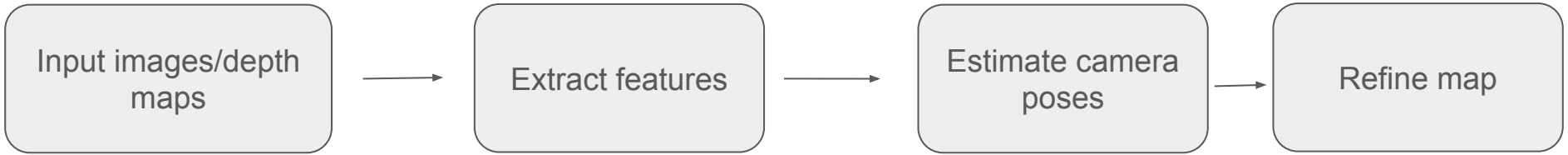
3D LiDAR SLAM for Autonomous Driving

Challenges of SLAM

- **Accuracy** of scene representation
- **Efficiency** of scene representation in terms of **memory footprint**
- Ability to **predict** shape of regions not directly observed
- **Flexibility** to new scenarios
- **Efficiency of joint optimization** of camera pose and scene representation

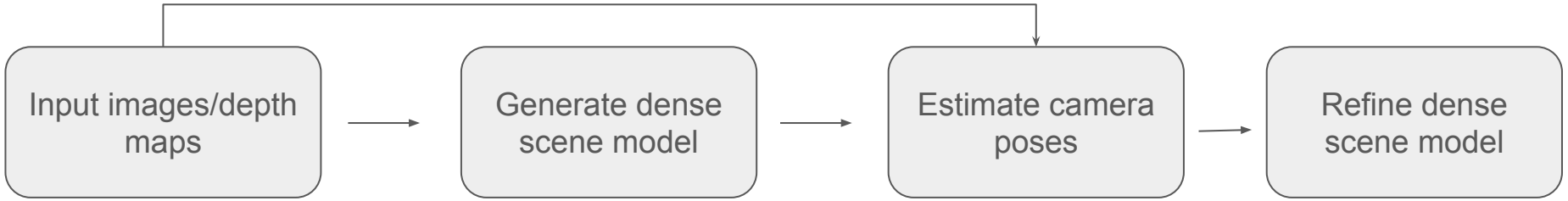
Related Work - Visual SLAM

- One common visual SLAM technique: use extracted features from input images/depth maps for localization
- Extracted features are a **sparse** representation of the environment
- Example: ORB-SLAM2



Related Work - Visual SLAM

- ‘Dense’ SLAM: *unified* dense scene representation also used for camera pose tracking

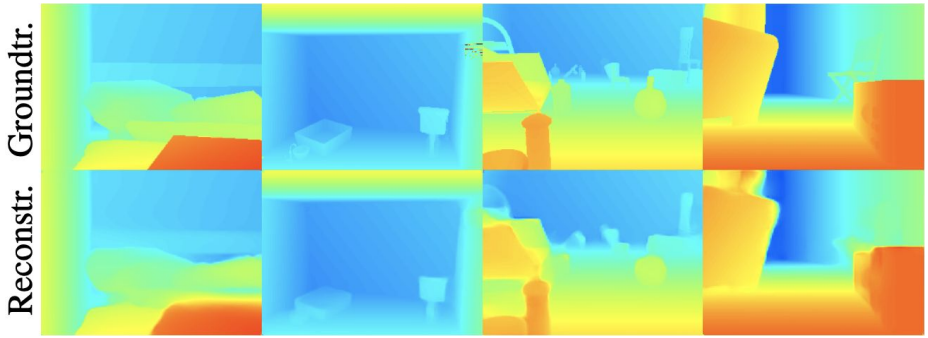


Related Work - Explicit Scene Representations

- **Explicitly representing volume** more versatile than representing surfaces but memory inefficient when fixed resolution is used
  - Occupancy mapping
  - Signed distance functions
- **Hierarchical approaches** more efficient but hard to implement and provide small range of level of detail
- Both representations are **rigid** and **cannot support joint optimization** with camera poses because they use a large number of parameters

Related Work - Explicit Scene Representations

- Machine learning can discover representations that can be jointly optimized with camera poses
  - **Low-dimensional** embeddings with **dense** structure
- Example: CodeSLAM



Encodings of depth maps from CodeSLAM used for scene representation

Related Work - Implicit MLP Scene Representations

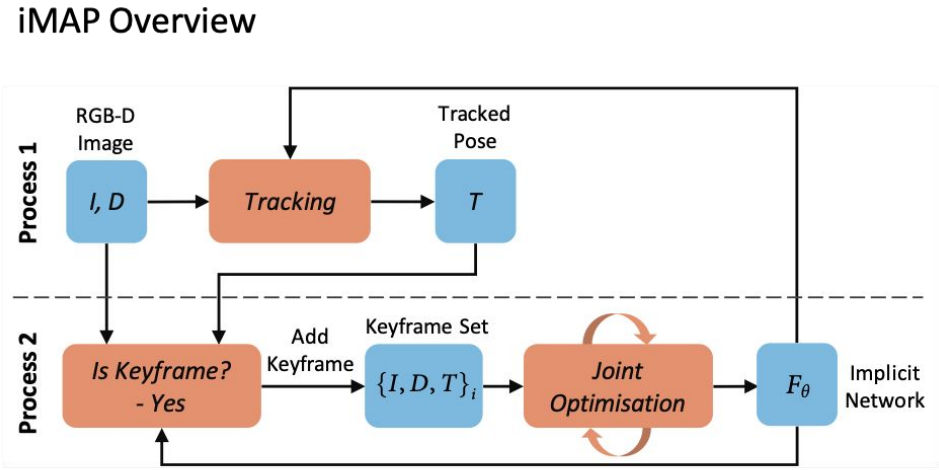
- Neural network learns function mapping that implicitly represents a scene or object
- Previously used for:
  - Object compression
  - Novel view synthesis
  - Scene completion
  - Camera pose optimization
- Previous use cases have been computationally expensive and not usable in real-time

Related Work - Continual Learning

- Learning new knowledge while retaining past knowledge
- Effective continual learning system should show **plasticity** (acquiring new knowledge) and **stability** (retaining previous knowledge)
- Catastrophic forgetting - network forgets previous knowledge as a result of new input data
- Previous ways to mitigate catastrophic forgetting:
  - Relative weighting - similar to approach used in EKF
  - Freezing or consolidating sub-networks after training on each individual task
  - Replay-based approach: store previous knowledge either in a buffer or compressed in a generative model

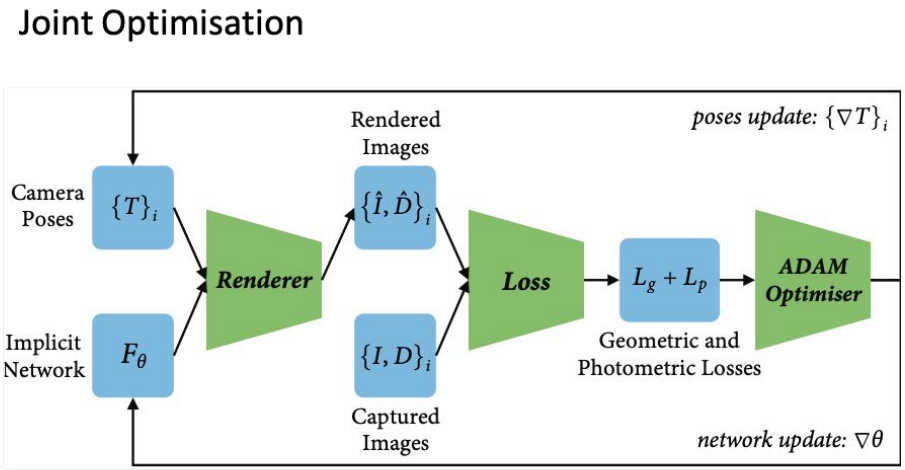
iMAP: Overview

- 3D volumetric map represented as fully connected network that maps 3D coordinate to color and volume density
- Two concurrent processes
- **Tracking**: optimizes pose from current frame using locked network
- **Mapping**: jointly optimizes network and poses of selected keyframes that are most important to 3D representation



Scene and Camera Pose Optimization

- Two optimization objectives: implicit scene network parameters and camera poses for working set of keyframes
- **Photometric loss**: captures difference in color values between rendered and captured images
- **Geometric loss**: captures difference in depth values between rendered and captured images



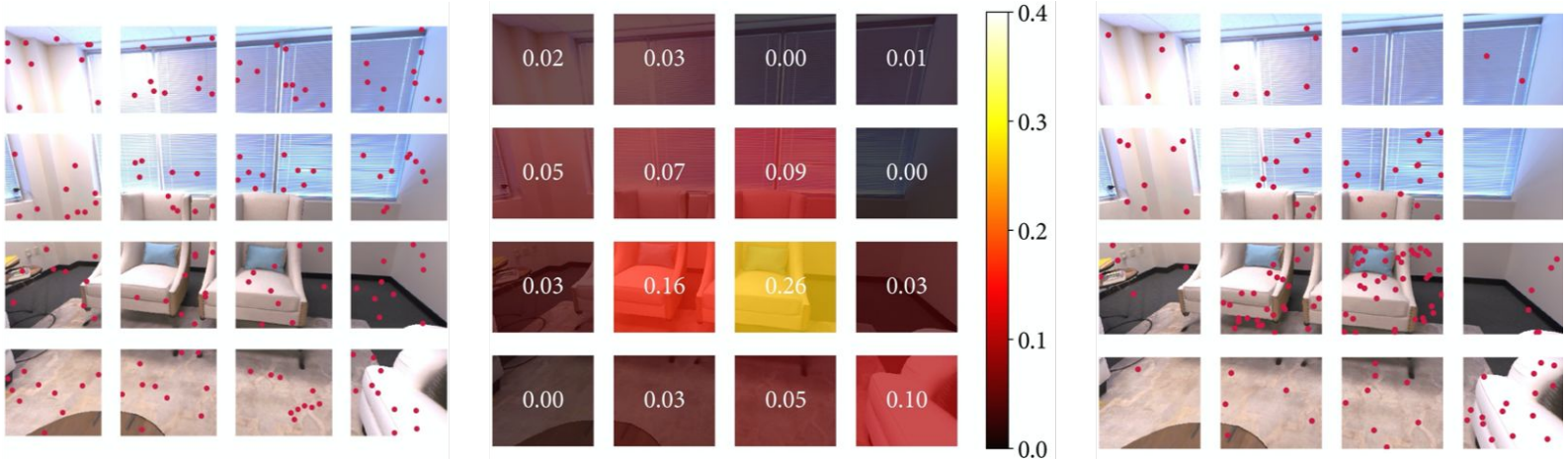
$$\min_{\theta, \{T_i\}} (L_g + \lambda_p L_p)$$

Selecting Keyframes for Optimization

- Computationally infeasible to optimize camera poses and network parameters using all captured images
- Solution: select keyframes incrementally based on the amount of new information the keyframe adds to the existing set

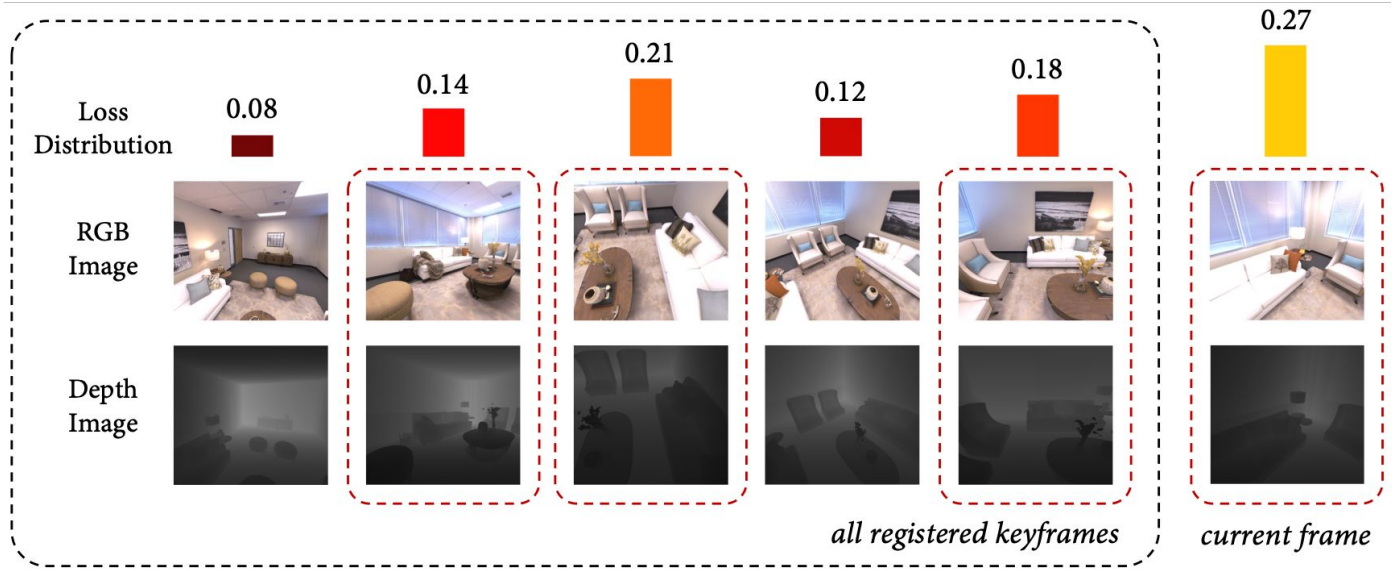
Image Active Sampling

- Inefficient to render all image pixels
- Solution: select a set of random pixels to render at each iteration
- Sample pixels in areas of higher detail or where reconstruction can be improved



Keyframe Active Sampling

- Keyframes with a higher loss value should be sampled from more heavily

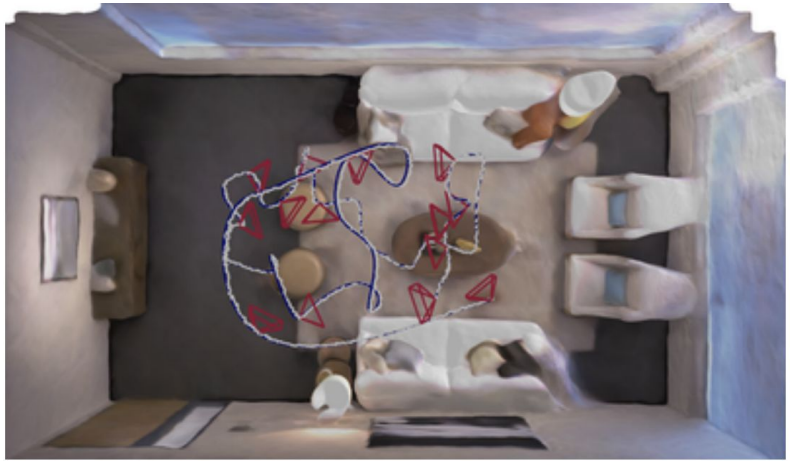


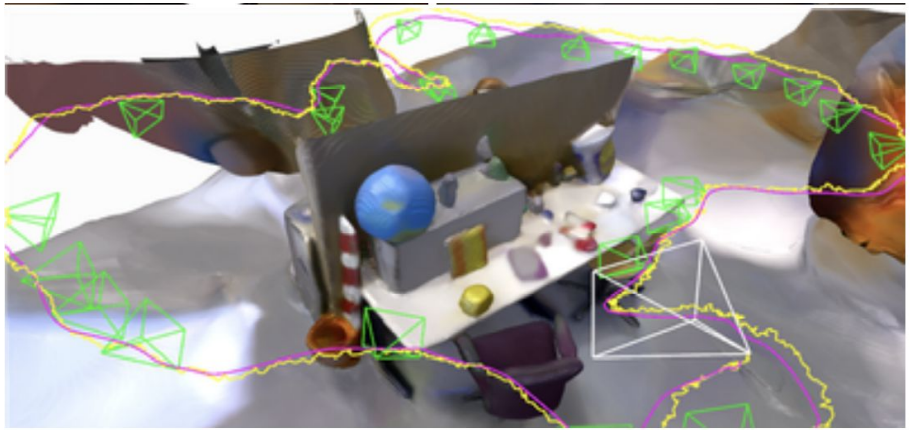
Bounded Keyframe Selection

- Bounds the number of keyframes used for optimization to ensure feasible computation

Experimental Setup: Datasets

- Replica dataset: for evaluating effectiveness of scene reconstruction
- TUM RGB-D dataset: for evaluating camera tracking
- RGB-D videos from Azure Kinect

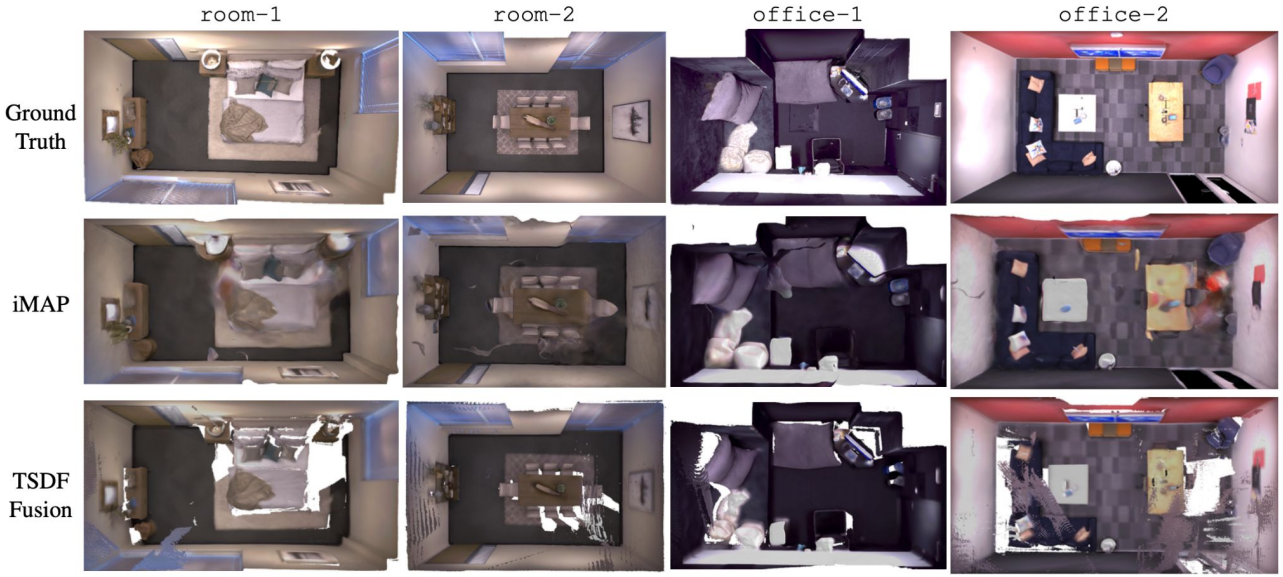




Experiments: Metrics

- **Accuracy**: measures, on average, how well the reconstruction reflects the ground truth
- **Completion:** measures how “full” the reconstruction is (i.e., the amount of gaps present)
- **Completion ratio:** percentage of points in reconstructed mesh with completion under 5 cm

Experimental Results: Reconstruction



- iMAP can fill unobserved regions, which are shown as holes in TSDF Fusion reconstruction

Experimental Results: Tracking

- iMAP does not outperform existing SLAM methods in terms of tracking

	fr1/desk (cm)	fr2/xyz (cm)	fr3/office (cm)
iMAP	4.9	2.0	5.8
BAD-SLAM	1.7	1.1	1.73
Kintinuous	3.7	2.9	3.0
ORB-SLAM2	1.6	0.4	1.0

Table 3: ATE RMSE in cm on TUM RGB-D dataset.

Experimental Results: Ablative Analysis

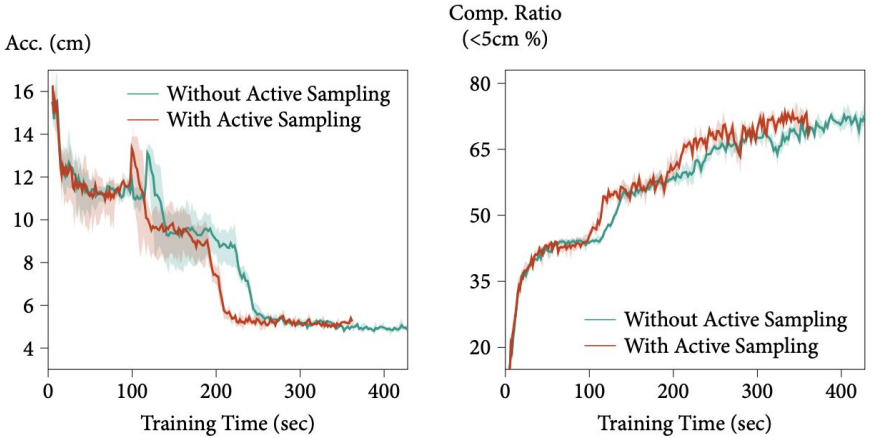
	Default	Width		Window		Pixels	
		128	512	3	10	100	400
Tracking Time [ms]	101	80	173	84	144	74	160
Joint Optim. Time [ms]	448	357	777	373	647	340	716
Comp. Ratio [<5cm %]	77.22	75.79	76.91	75.82	77.35	77.33	77.49

Default configuration (network width 256, window size 5, and 200 samples per keyframe) provides best balance between convergence speed and accuracy

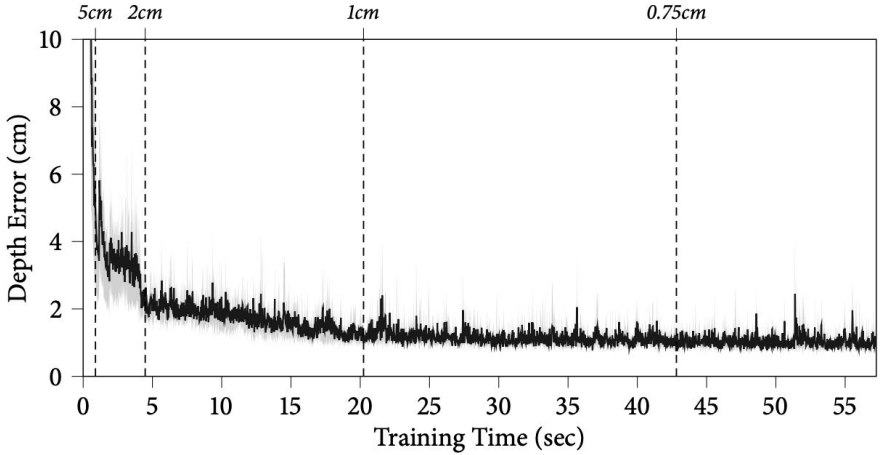
Using more than 8 keyframes does not significantly improve scene completion ratio

	$t_P = 0.55$	$t_P = 0.65$	$t_P = 0.75$	$t_P = 0.85$
# Keyframes	8	10	14	24
Comp. Ratio [<5cm %]	74.11	77.22	76.84	78.03

Experimental Results: Ablative Analysis



Better completion ratio and faster accuracy convergence achieved with active sampling compared to random sampling



Depth error falls to 5 cm quickly (~ 1 sec) and error reduces further more slowly

Limitations

- As shown from the experiments, iMAP's camera pose tracking does not outperform the SOTA SLAM techniques referenced (ORBSLAM-2, BAD-SLAM, Kintinuous), possibly due to the increased amount of data used for tracking

Unobserved regions in iMAP are blurry in reconstruction

Future Work

- Further examination of tradeoff between accuracy and efficiency
- Improving camera pose tracking accuracy
- Creating scene representations that explicitly reason about self-similarity

Further Reading

- Zhu, Zihan et al. "NICE-SLAM: Neural Implicit Scalable Encoding for SLAM." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022.
- Newcombe, Richard A. et al. "DTAM: Dense tracking and mapping in real-time." *2011 International Conference on Computer Vision*. 2011.
- Mur-Artal, Raul et al. "ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras". *IEEE Transactions on Robotics 33*. 5(2017): 1255–1262.
- M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison. CodeSLAM — learning a compact, optimisable representation for dense visual SLAM. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Summary

- iMAP is a real-time dense SLAM system that represents scenes using an implicit scene MLP
- iMAP is capable of jointly optimizing camera poses and scene representation by active sampling of keyframes and pixels within keyframes with the aim of maximizing information gain

Discussion

- Advantages/disadvantages of sparse vs dense SLAM techniques
  - Dense SLAM utilizes more (or all) of input image/depth data
  - Sparse SLAM may be more time/memory efficient (lower-dimensional representation of scene)
- Advantages/disadvantages of explicit vs implicit scene representations
  - Implicit scene representation may be more compact in terms of memory
  - Explicit representations may give more information about the scene