Table of Contents

[Introducing SLAMBench, a performance and accuracy benchmarking methodology for SLAM 2](#_Toc17622327)

# Introducing SLAMBench, a performance and accuracy benchmarking methodology for SLAM

* The purpose of SLAMBench: a framework for quantifying quality-of-result with instrumentation of execution time and energy consump- tion
* Relate works
  + KITTI benchmark suite - the stereo, optical flow, visual odometry and 3D object recognition
  + ICL-NUIM dataset and TUM RGB-D benchmark aim to benchmark the accuracy of visual odometry and SLAM algorithms
  + No energy-constraint consideration
  + MEVBench only focuses on recognition algorithms and does not include a SLAM pipeline
  + A typical output of SLAMBench consists of the performance achieved, and the accuracy of the result along with the energy consumption (on platforms where such measurement is possible). These parameters capture the potential trade-offs for real-time vision platforms
* KinectFusion Algorithm
  + Input - depth frame
  + Preprocess - applies a bilateral filter
  + Track - estimates the new 3D pose of the moving camera by registering this point cloud with the current global map using a variant of *iterative closest point* (ICP)
  + Integrate - utilises a voxel grid as the data structure to represent the map, employing a *truncated signed distance function*
  + Raycast - The 3D surfaces are present at the zero crossings of the TSDF and can be recovered by a raycasting step
  + Rendering
* SLAMBench
  + ICL-NUIM is a high-quality synthetic dataset providing RGB-D sequences for 4 different camera trajectories through a living room model. (1)free of noise, (2)add noise to simulate realistic situation
  + Parallel patterns: map, reduce, stencil, gather, search
  + Performance evaluation methodology
    - GUI vs terminal interface
    - Pre-recorded scenes
    - Frame rate
    - Accuracy evaluation

# SLAMBench2: Multi-Objective Head-to-Head Benchmarking for Visual SLAM

* The purpose of SLAMBench 2: Dataset-agnostic, plug and play algorithms, modular user interface
* Four main components: I/O System, Integration API, Loader and User Interfaces.
* Supported artefacts: datasets, SLAM Algorithms, Performance Metrics,
* I/O System: convert dataset to uniform format
* API: interface with algorithm and other parts of SLAMBench
* Loader and UI

