iMorpheus.ai

Weekly Journal Club

KITTI Odometry Benchmark

iMorpheus.ai

Friday 9th Feb 2018, 12:00PM

Journal Club

Journal club介绍与自动驾驶中定位方案相关的论文， 主要关注的方向有：
SLAM算法、点云数据的处理和压缩、 特征地图、传感器数据处理和融合、 GNSS信号处理等。我们一直关注领域前沿技术， 选取得到广泛认可的、或者是在我们的实际使用中结果比较好的论文， 与大家分享，共同学习成长。

每周五 北京时间12点
http://imorpheus.ai/journalclub

iMorpheus.ai

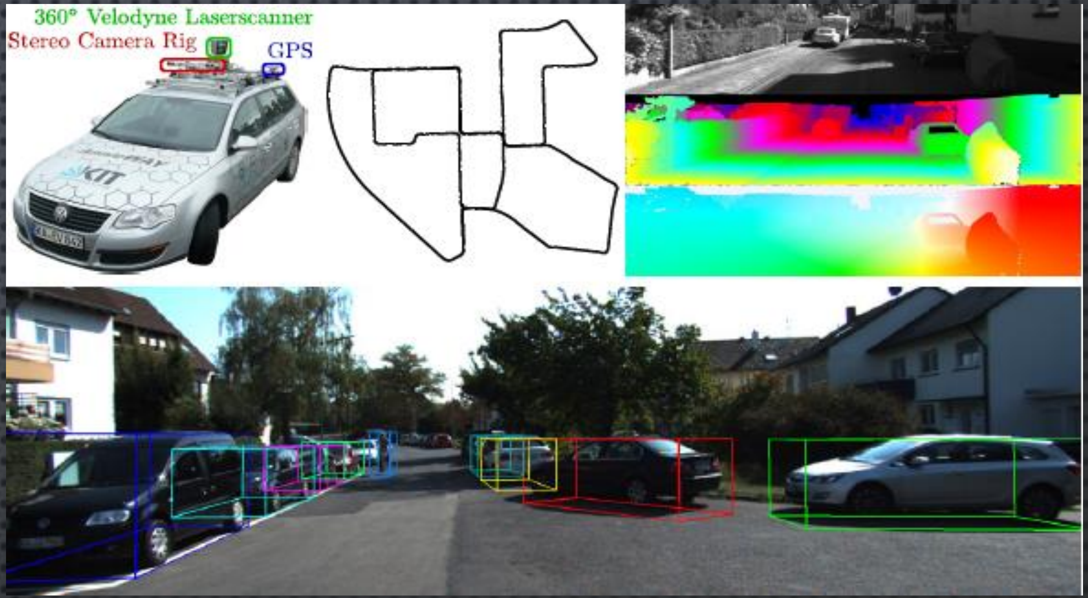
扫码加入无人驾驶技术群

Introduction to KITTI

iMorpheus.ai

- Autonomous driving dataset and benchmark project funded by Karlsruhe Institute of Technology and Toyota Technological Institute at Chicago
- Benchmarks include stereo, flow, scene flow, depth, odometry, object, tracking, road and semantics (planned)
- Volkswagen Passat equipped with LiDAR, monocular/stereo grey-scale/color cameras, GPS/IMU, workstation computer
- Data collected in and near Germany city of Karlsruhe; urban, rural and highway scenarios; all data outdoor and real world; collected in 2011
- Carefully designed error metrics that measure and rank accuracy of user submitted algorithms
- Generally considered one of the best benchmarks for autonomous driving

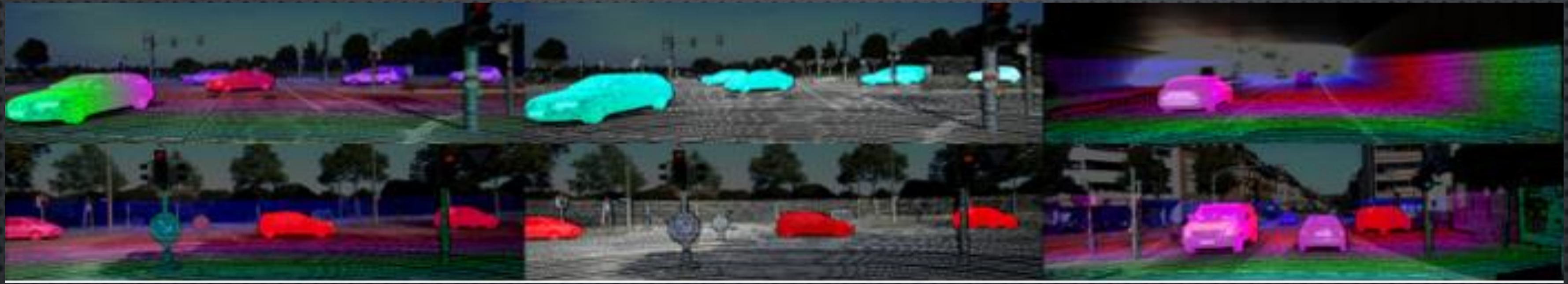
Reference and more information



- Geiger, Andreas, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.
- Geiger, Andreas, et al. "Vision meets robotics: The KITTI dataset." *The International Journal of Robotics Research*32.11 (2013): 1231-1237.
- Main site http://www.cvlibs.net/datasets/kitti/
- Video trailer at http://www.youtube.com/watch?v=KXpZ6B1YB_k

iMorpheus.ai

KITTI benchmarks



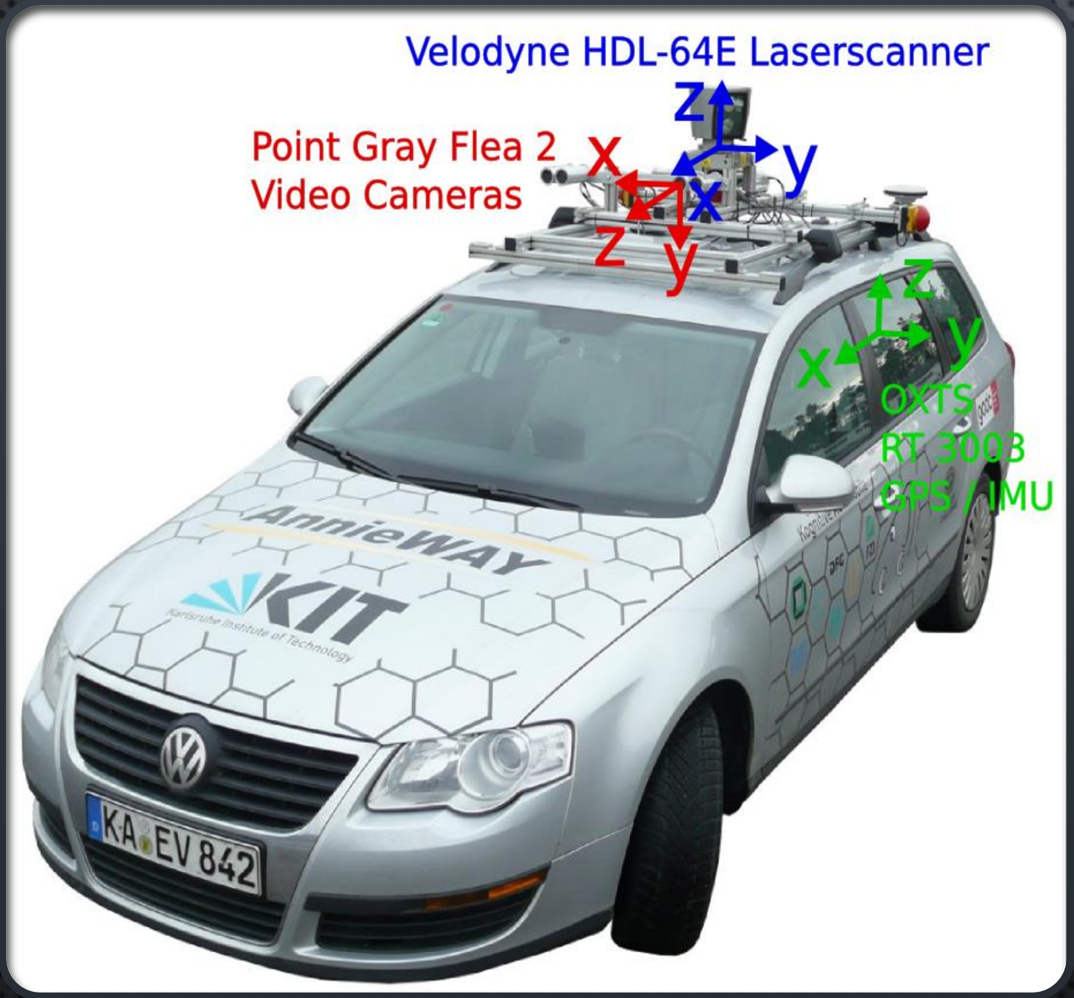
- Stereo scene flow: estimate scene motion relative to stereo camera, segment background and moving objects; ground truth semi-manually created with LiDAR point clouds
- Depth completion: based on LiDAR point clouds and RGB image; complete depth map from a sparse one
- Depth prediction: based on RGB image; predict a depth map

iMorpheus.ai

KITTI benchmarks

- Odometry: visual or point cloud odometry; ground truth provided by GPS/IMU module; average relative error metrics
- Object detection: detect, classify and assign orientation to objects such as pedestrian, car, truck; 2D, 3D and bird’s eye view
- Object tracking: track two classes of objects, pedestrian and car; 3D bounding boxes; manually marked ground truth
- Road/lane detection: manually marked ground truth
- Semantic segmentation: future plan

iMorpheus.ai

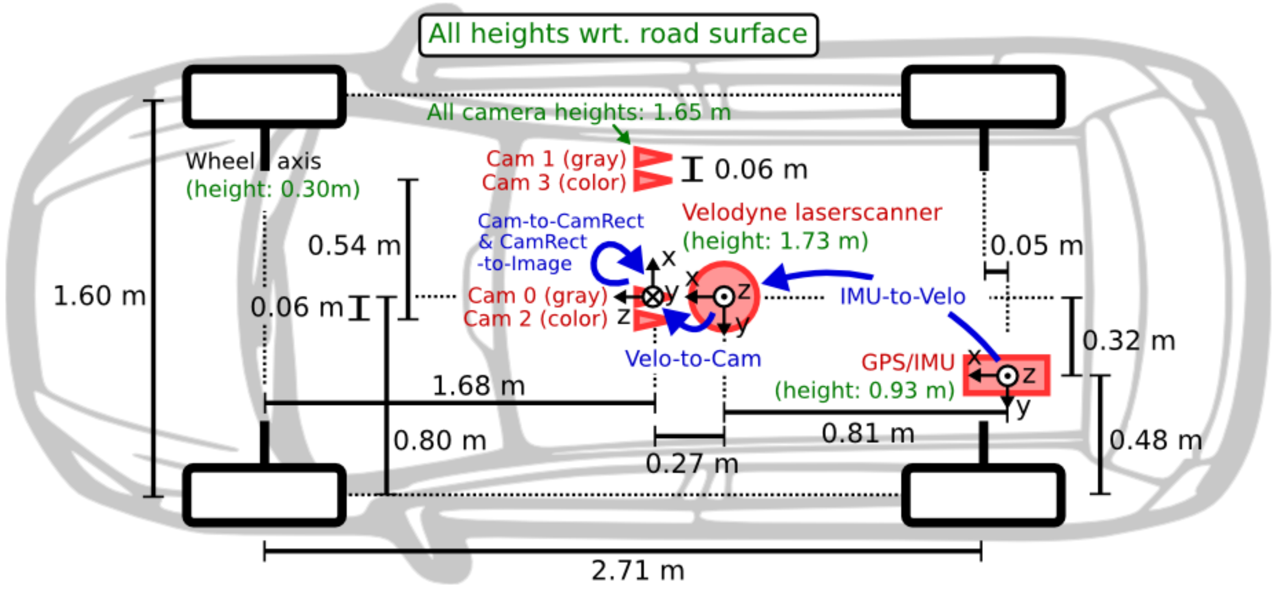


Hardware and specs

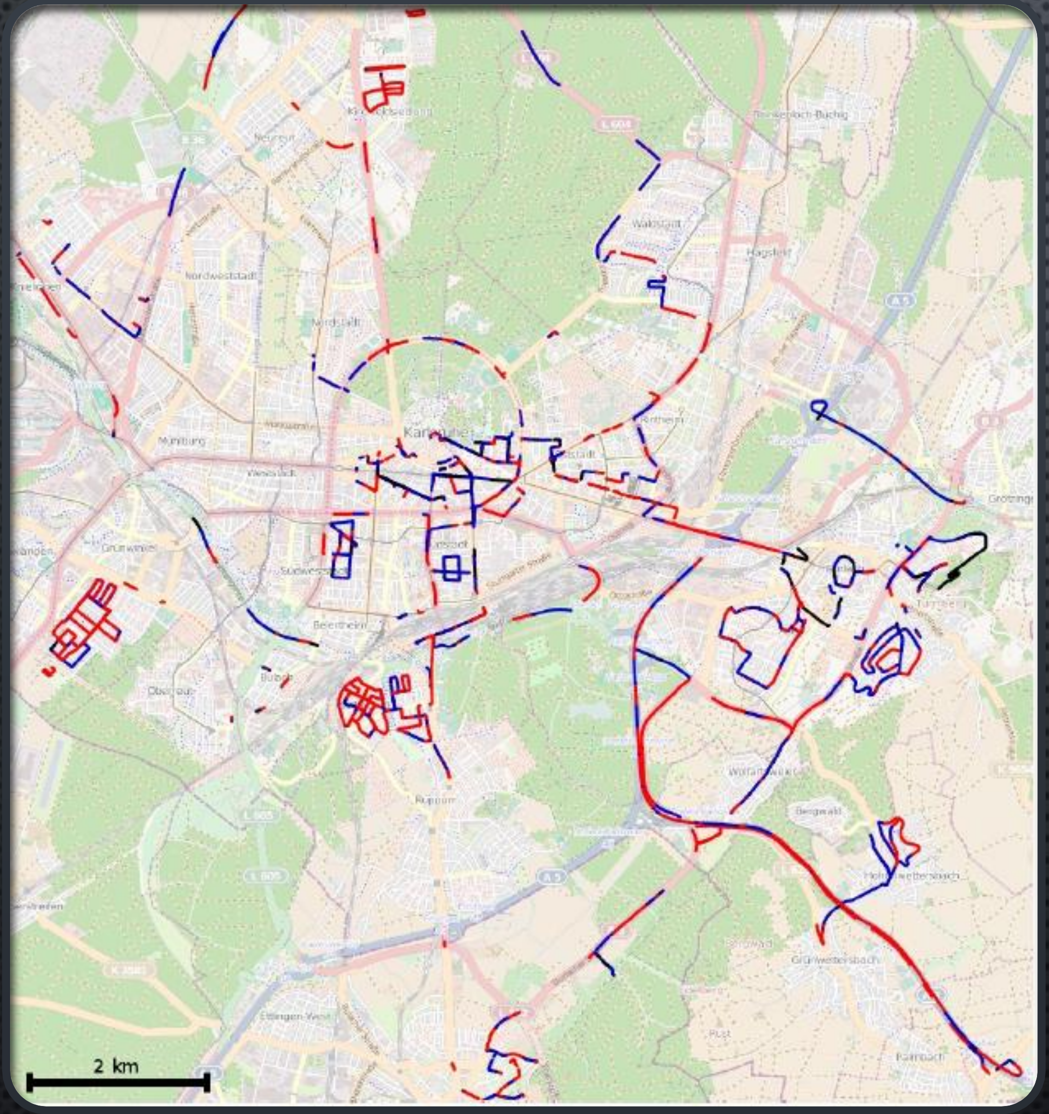
- 2x PointGray Flea2 grayscale cameras (FL2-14S3M-C), 1.4 Megapixels, 1/2" Sony ICX267 CCD, global shutter
- 2x PointGray Flea2 color cameras (FL2-14S3C-C), 1.4 Megapixels, 1/2" Sony ICX267 CCD, global shutter
- 4x Edmund Optics lenses, 4mm, opening angle ~90°, vertical opening angle of region of interest ~35°
- 1x Velodyne HDL-64E rotating 3D laser scanner, 10 Hz, 64 beams, 0.09 degree angular resolution, 2 cm distance accuracy, collecting ~1.3 million points/second, field of view: 360° horizontal, 26.8° vertical, range: 120 m

Hardware and specs

iMorpheus.ai



- 1x OXTS RT3003 inertial and GPS navigation system, 6 axis, 100 Hz, L1/L2 RTK, resolution: 0.02m / 0.1°
- 1x workstation PC: two six-core Intel XEON X5650 processors, shock-absorbed RAID 5 hard disk storage with a capacity of 4 Terabytes, Ubuntu Linux (64 bit), a real-time database to store incoming data streams



Data collection

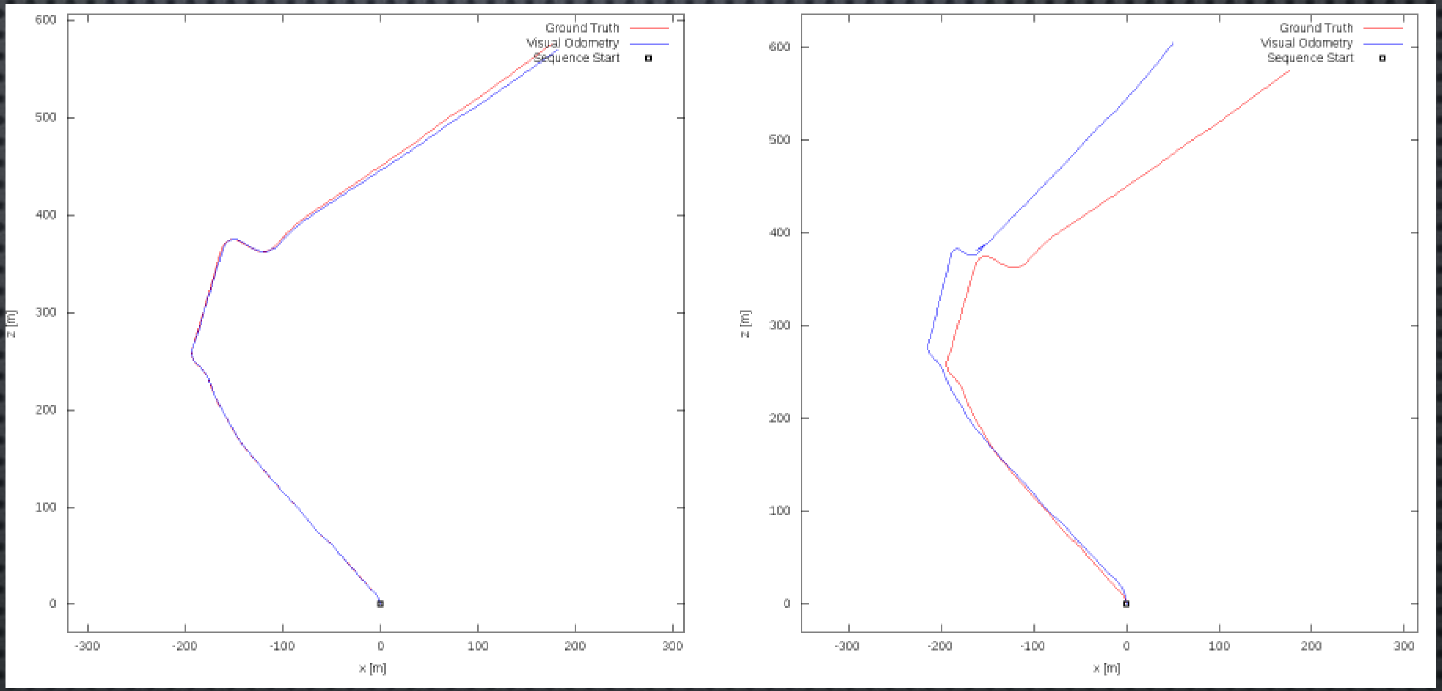
- Red = RTK fix, blue = no correction signal, black = no GPS signal (excluded from dataset)
- For short period (~ 1 sec) of GPS/IMU outage, interpolation is used
- Timestamps for visual, point cloud, GPS and IMU data; all timestamps synchronized to LiDAR
- Covers a variety of scenes such as city, residential, highway, rural and campus in and near Karlsruhe
- Total raw data size = 3 TB

iMorpheus.ai

Data collection for odometry

- GPS/IMU provide altitude, global orientation, velocities, accelerations, angular rates, accuracies and satellite information
- Grey-scale and color images in 8bit PNG format
- Point clouds are stored in LiDAR coordinates with reflectance value; about 1.9 MB (120k points) per frame
- For visual or LiDAR odometry, 22 long sequences are selected; total frames 41,000 at 10 fps; total distance 39.2 km

Odometry ground truth



- Ground truth directly from GPS/IMU, which most of the time gives RTK fixes
- Among 22 odometry benchmark sequences, ground truth are provided for 11 sequences to help training; the remaining 11 are used for evaluation
- Can choose monocular, stereo, LiDAR based algorithm, or any combination
- Algorithm should output both position and attitude of vehicle (6 DOF)

Odometry error metrics

Some desired properties of odometry error metrics

- Separate metrics: translational (vehicle position) and rotational (attitude) errors are calculated separately
- Coordinate invariant: transformation between coordinate systems introduces error, so it is better to use relative errors in original coordinates
- Average: all segments of odometry should be treated roughly equally
- Distance: SLAM errors usually accumulate over distance travelled
- Speed: also affects SLAM performance
- Fast evaluation
- Can be easily ported to other datasets

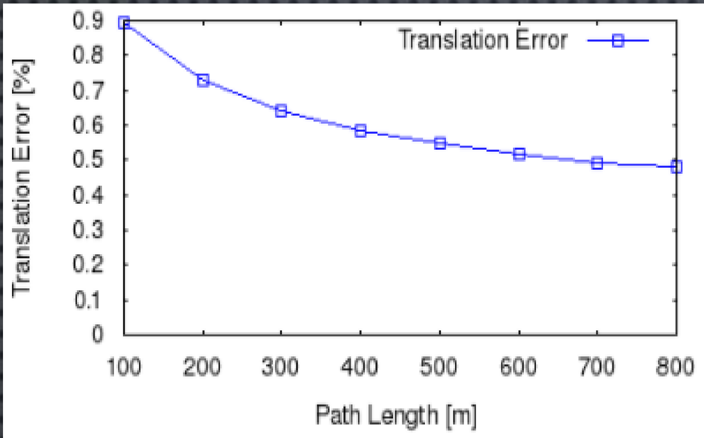
iMorpheus.ai

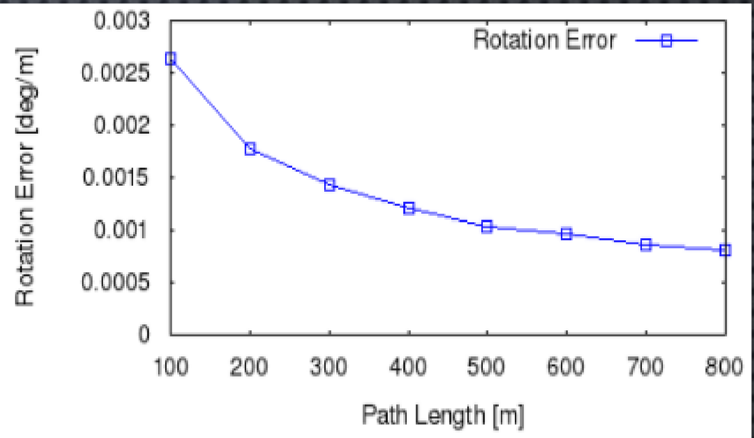
KITTI odometry error metrics

- Vehicle pose relative to a fixed (usually starting) frame is given by elements in SE(3), $P = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$, where R is in SO(3) and t is 3x1 translation vector
- Difference between poses are calculated as $P_2 \ominus P_1 = P_1^{-1} P_2$
- Relative pose error from frame i to j is given by $E_{ij} = (T_j \ominus T_i) \ominus (P_j \ominus P_i)$, where P is result and T is ground truth
- Rotation error is $\cos^{-1}(\frac{1}{2}(\mathrm{tr}(E_{ij}) - 2))$, which is the angle between result relative pose and ground truth relative pose
- Translation error is $|\langle T_j \ominus T_i \rangle - \langle P_j \ominus P_i \rangle|$, where $\langle \cdot \rangle$ denotes 2-norm of the translation vector in the SE(3) matrix form; this is comparison of distance between frame i and j, regardless of route covered

iMorpheus.ai

KITTI odometry error metrics





- Rotation and translation errors are averaged over all distance segments and starting frames (moving average with varying window size)
- Distance segments are chosen as 100, 200, …, 800 meters
- Starting frames increment step size is 10, which means 1 second interval for 10 Hz sample rate
- Average translation error is given as percentage over distance and speed
- Average rotation error is given as °/m over distance and speed
- Finally, error is averaged over all distances/speeds for ranking

iMorpheus.ai

KITTI odometry ranking

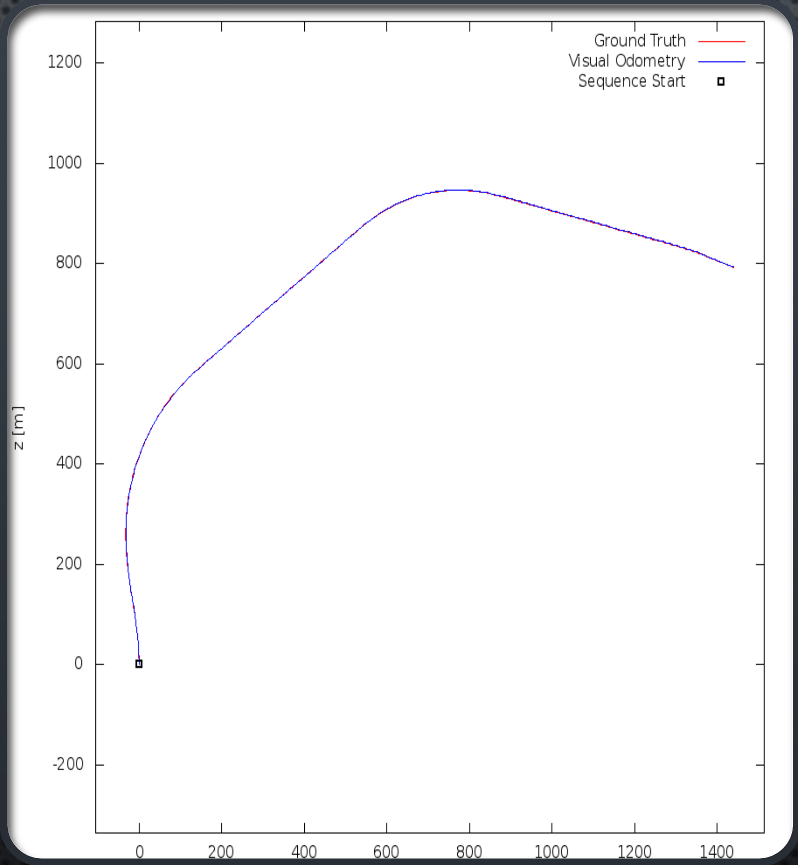
- Ji Zhang's methods ranked top two
- Stereo methods are more popular, but LiDAR methods perform better
- Most methods are real time
- Most methods do not provide open source code
- No loop closure used

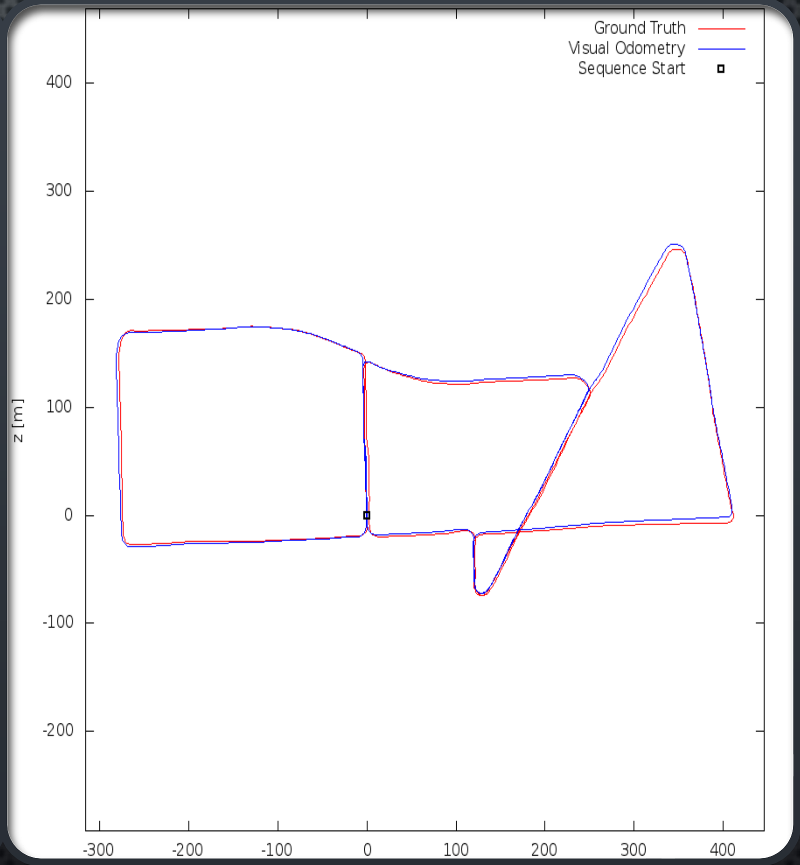
	Method	Setting	Code	Translation	Rotation	Runtime	Environment
1	V-LOAM			0.63 %	0.0014 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
J. Zhang and S. Singh: Visual-lidar Odometry and Mapping: Low drift, Robust, and Fast. IEEE International Conference on Robotics and Automation(ICRA) 2015.							
2	LOAM			0.64 %	0.0014 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
J. Zhang and S. Singh: LOAM: Lidar Odometry and Mapping in Real-time. Robotics: Science and Systems Conference (RSS) 2014.							
3	SOFT2			0.65 %	0.0014 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
I. Cvišić, J. Ćesić, I. Marković and I. Petrović: SOFT-SLAM: Computationally Efficient Stereo Visual SLAM for Autonomous UAVs. Journal of Field Robotics 2017.							
4	IMLS-SLAM			0.69 %	0.0018 [deg/m]	1.25 s	1 core @ >3.5 Ghz (C/C++)
5	sGAN-VO			0.81 %	0.0025 [deg/m]	0.1 s	1 core @ 2.5 Ghz (C/C++)
6	LG-SLAM			0.82 %	0.0020 [deg/m]	0.2 s	2 cores @ 2.5 Ghz (C/C++)
7	RotRocc+			0.83 %	0.0026 [deg/m]	0.25 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: Flow-Decoupled Normalized Reprojection Error for Visual Odometry. 19th IEEE Intelligent Transportation Systems Conference (ITSC) 2016.							
8	GDVO			0.86 %	0.0031 [deg/m]	0.09 s	1 core @ >3.5 Ghz (C/C++)
J. Zhu: Image Gradient-based Joint Direct Visual Odometry for Stereo Camera. International Joint Conference on Artificial Intelligence, IJCAI 2017.							
9	SOFT			0.88 %	0.0022 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
I. Cvišić and I. Petrović: Stereo odometry based on careful feature selection and tracking. European Conference on Mobile Robots (ECMR) 2015.							
10	RotRocc			0.88 %	0.0025 [deg/m]	0.3 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: Flow-Decoupled Normalized Reprojection Error for Visual Odometry. 19th IEEE Intelligent Transportation Systems Conference (ITSC) 2016.							
11	SSO			0.93 %	0.0021 [deg/m]	0.1 s	1 core @ 2.5 Ghz (C/C++)
12	ROCC			0.98 %	0.0028 [deg/m]	0.3 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: How to Distinguish Inliers from Outliers in Visual Odometry for High-speed Automotive Applications. IEEE Intelligent Vehicles Symposium (IV) 2016.							
13	cv4xv1-sc			1.09 %	0.0029 [deg/m]	0.145 s	GPU @ 3.5 Ghz (C/C++)
M. Persson, T. Piccini, R. Mester and M. Felsberg: Robust Stereo Visual Odometry from Monocular Techniques. IEEE Intelligent Vehicles Symposium 2015.							
14	FPVO			1.10 %	0.0023 [deg/m]	0.08 s	4 cores @ 2.3 Ghz (C/C++)
15	MonoROCC			1.11 %	0.0028 [deg/m]	1 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: Monocular Outlier Detection for Visual Odometry. IEEE Intelligent Vehicles Symposium (IV) 2017.							
16	RI MVO			1.13 %	0.0032 [deg/m]	0.07 s	1 core @ 2.5 Ghz (Python + C/C++)
17	DEMO			1.14 %	0.0049 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
J. Zhang, M. Kaess and S. Singh: Real-time Depth Enhanced Monocular Odometry. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2014.							
18	ORB-SLAM2		code	1.15 %	0.0027 [deg/m]	0.06 s	2 cores @ >3.5 Ghz (C/C++)
Mur-Artal and J. Tard\'os: ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. IEEE Transactions on Robotics 2017.							

iMorpheus.ai

KITTI odometry error analysis

- Cumulative drift error
- Some ground truth might not be accurate enough
- At 300 m, LOAM error is about 1.92 m and 0.42°
- For comparison, SPAN-CPT error is 6.44 m and 0.12° (during GNSS outage and assuming 5 m/s speed)





iMorpheus.ai

Our test

- Low speed outdoor robot moving along Gubei Road, Shanghai; 280 meters covered in 324 seconds
- Modified LOAM algorithm with 16 channel LiDAR, 0.1° horizontal accuracy, 5 Hz
- Ground truth provided by Novatel OEM 628 receiver; all outputs are RTK fixes, 1 cm accuracy; no attitude
- Distance segments 50, 100, …, 250 m; frame step size 10
- Overall average translation error is 0.58%

iMorpheus.ai

Final thoughts

- Overall, KITTI odometry offers an excellent benchmark with dataset, ground truth, error metrics and ranking; different methods can compete fairly
- Real world environment (Shanghai) can be more challenging, regarding dynamic noise (vehicle, pedestrian) and degenerate scene (tunnel)
- Ground truth is provided by RTK with some interruptions; better GNSS-INS system can be used; preferably all frames have RTK fixes
- Translation is $|\langle T_j \ominus T_i \rangle - \langle P_j \ominus P_i \rangle|$, not $\langle (T_j \ominus T_i) \ominus (P_j \ominus P_i) \rangle$; this is not as accurate, but is coordinate invariant and independent of rotation error
- For short distances, ground truth error can not be neglected, even for RTK fixes; therefore, KITTI changed distance segments for evaluation from 5, 10, 50, 100, ..., 400 to 100, 200, ..., 800 meters in 2013

iMorpheus.ai

Journal Club

iMorpheus.ai Weekly Journal Club

Friday 23/02/2018 12:00PM GMT+8

Improving Poor GPS Area Localization for Intelligent Vehicle

Website : **http://imorpheus.ai**
Email Address : **live@imorpheus.ai**

iMorpheus.ai