Real-time Instance-aware Semantic Mapping

Xuecheng Xu, Lilu Liu, Rong Xiong and Lei Jiang.

Zhejiang University

About this paper

This paper proposed a real-time framework to construct an object map within the city road scene. The framework contains lightweight algorithms such as traditional point clustering and object detection methods. Our system also applies the 2.5D representation of the environment to reduce storage consumption because of the large-scale scene in the city. Our system uses a hash table as the structure of data storage to avoid dependence upon CPU memory. With all these methods, the framework can be running at ~10fps. Besides the real-time feature, the proposed method is also born with a feature that can be used in navigation. The local traversable map, along with the global semantic map, can lead the robot to a more semantic interaction with people.

The framework uses a lidar-IMU unified method for localization and a camera for object detection. The lidar-IMU unified localization estimates the motion of the robot and enables the multiple observation of the object and improves the accuracy of the object-instance recognition. The camera provides abundance textures of the environment, which can be used to detect objects.

Outline of our system

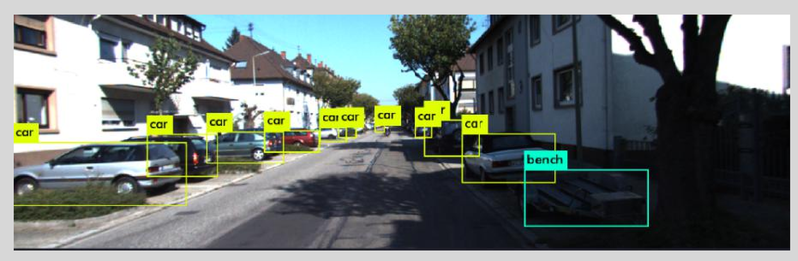
1. Object Detection: acquiring semantic objects.
2. Localization: get current pose.
3. Mapping: modelling the environment

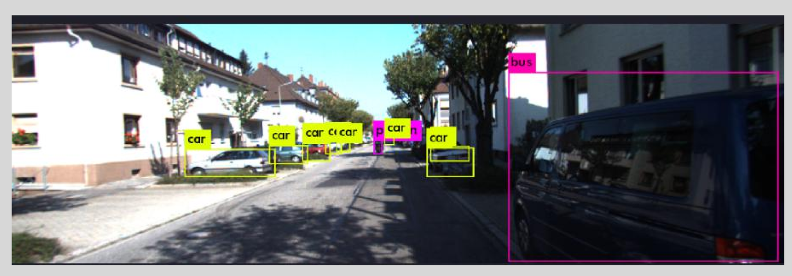
Object Detection

To get the semantic information from the streaming image use a YOLO v3 framework integrated into the ROS system, to detect objects during the movement of the car.

The YOLO v3 detector typically relies on a single view for casting and validating hypotheses, so it may produce fault semantic information in the mapping system. To overcome this, we use the detection result in a constraint way. Mainly, we only use the detection result with the probability higher than 90% and filtered those detections far away from the camera. This consideration is based on that the far-away object is too small for object detector to extract validate features for classification.

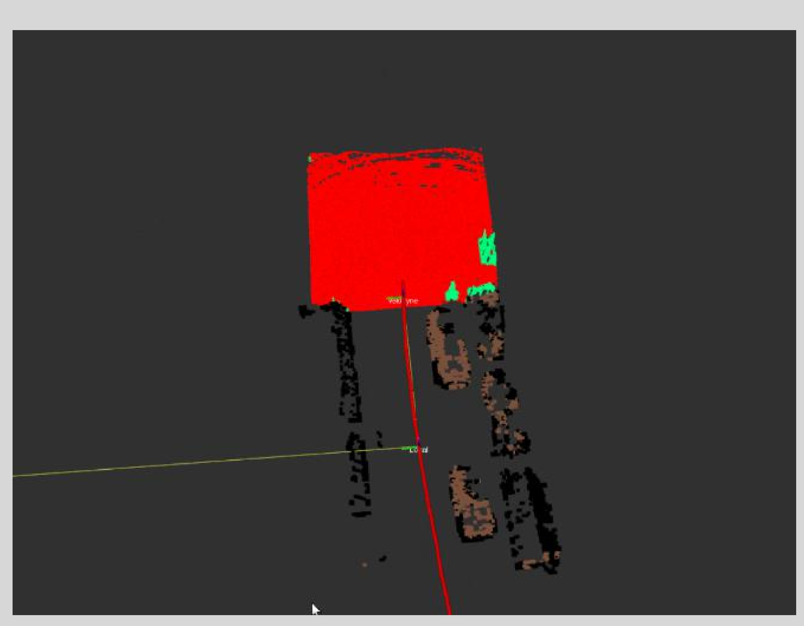
Object Detection





Localization

We adopt the online ICP based lidar-IMU front-end to localize the robot position, the poses are updated under the Extended Kalman Filter paradigm. By apply the pre-integrate algorithm of IMU data, the output localization can be running at a high rate. We focus on semantic and map representation including local mapping and global mapping. The localization module is off-shelf methods, which we will not consider in our system performance. Local Map is shown in below.

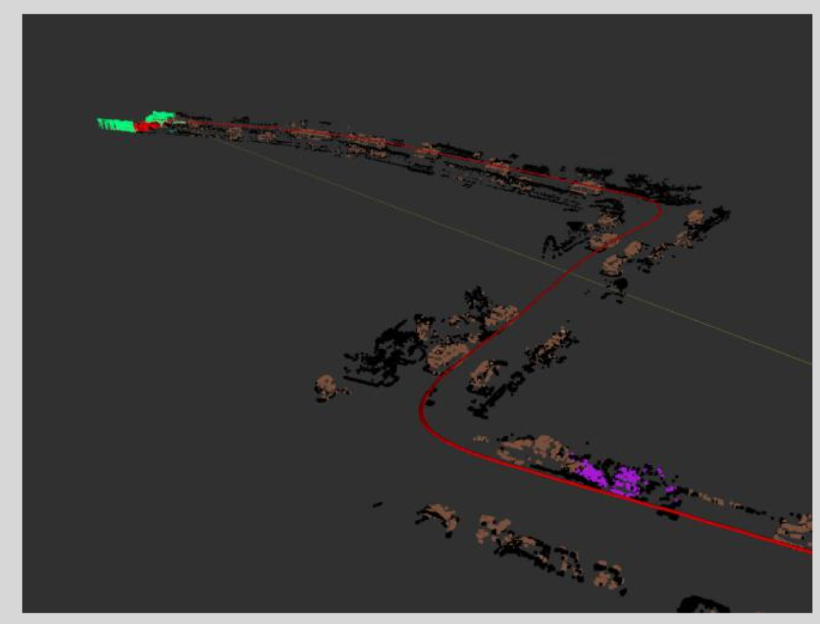


Mapping

Our mapping module process a 3D point cloud into a 2.5D grid map, it will cost about 50ms. The local mapping is shown in figure 2 with red and green grids. The red region is represented as an area that has a high traversable score, which can be passed for robots and cars, while the green area is obstacles. The local mapping module is implemented on the GPU, which can be running in real-time.

The global mapping method simply adds the outdate local map to a point cloud map with semantic information. The real-time visualization with trajectory is shown below. The global mapping module consumes little computation resources and time.

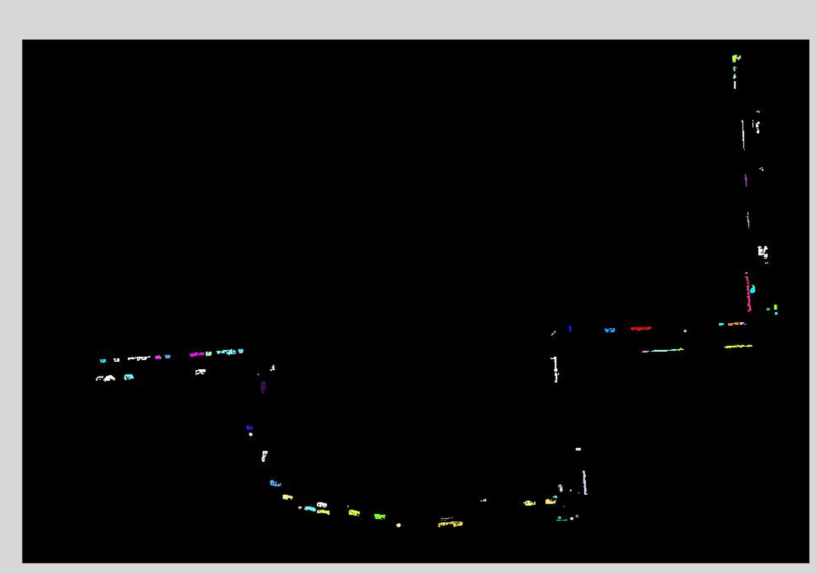
Mapping



The YOLO v3 detector typically relies on a single view for casting and validating hypotheses, so it may produce fault semantic information in the mapping system. To overcome this, we use the detection result in a constraint way. Mainly, we only use the detection result with the probability higher than 90% and filtered those detections far away from the camera. This consideration is based on that the far-away object is too small for object detector to extract validate features for classification.

Our experiment is conducted on PC with Nvidia GPU RTX2060 and AMD 3700X. Also, we test the framework on TX2, a common platform for mobile robots. Our test dataset is KITTI car datasets containing lidar, camera and IMU.

First, we evaluate our object detection module, we adopt YOLO v3 under the ROS framework and test it. The performance of YOLO is shown below and it can be running on real-time ~30fps on our experiment platform.



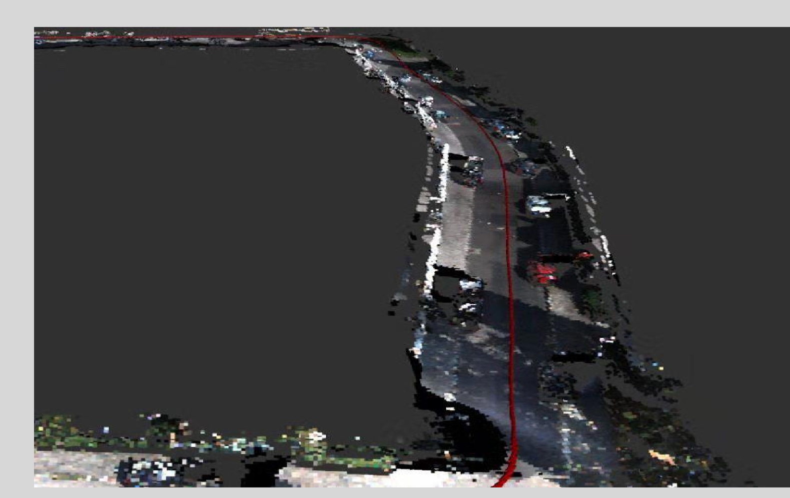
The point cloud segmentation is efficient in removing outliers in semantic point cloud acquired by the global mapping step. The different clusters are shown in a different color.

	Time [ms]
Object Detection	30
Point cloud clustering	23
Local mapping	78
Global mapping + Fusion	34

More experiments

The point cloud segmentation is efficient in removing outliers in semantic point cloud acquired by the global mapping step. The different clusters are shown in a different color.

The final semantic map combining the raw object detection result and point cloud segment can be very clear in the object, with this clear point cloud, we can also perform object pose estimation on it. Along with this object map, we can also get a global colored map.



Conclusion

In this article, we propose a real-time semantic object map method with additional information about drivable navigation information. The method is based on multi-modal data such as lidar, IMU and camera. With the real-time object detection and clustering pipeline, we fuse the semantic information into our global object-centric map. The output of our system can generate cost maps for classic planning algorithms, such as A* and RRT. Also, this map representation can be used in a more complex motion planning situation of quadruped robots.

Contact information

Email: xuechengxu@zju.edu.cn

Website: https://maverickpeter.github.io/