Monocular Visual Inertial Simultaneous Localization and Mapping for Detecting Obstacles through Point Clouds

Special Study by Raunak Mukhia

Examination Committee:
Dr. Matthew N. Dailey (Chairperson)
Dr. Mongkol Ekpanyapong
Dr. Attaphongse Taparugssanagorn

May 2019

Asian Institute of Technology

Introduction

Asian Institute of Technology

Introduction

- An autonomous vehicle should be aware of its environment to take the next decision.

- SLAM (Simultaneous Localization and Mapping) systems can localize and map the environment at the same time.

Introduction

- The objective of this special study is to generate real time point clouds and robot poses by executing a monocular visual inertial SLAM algorithm.

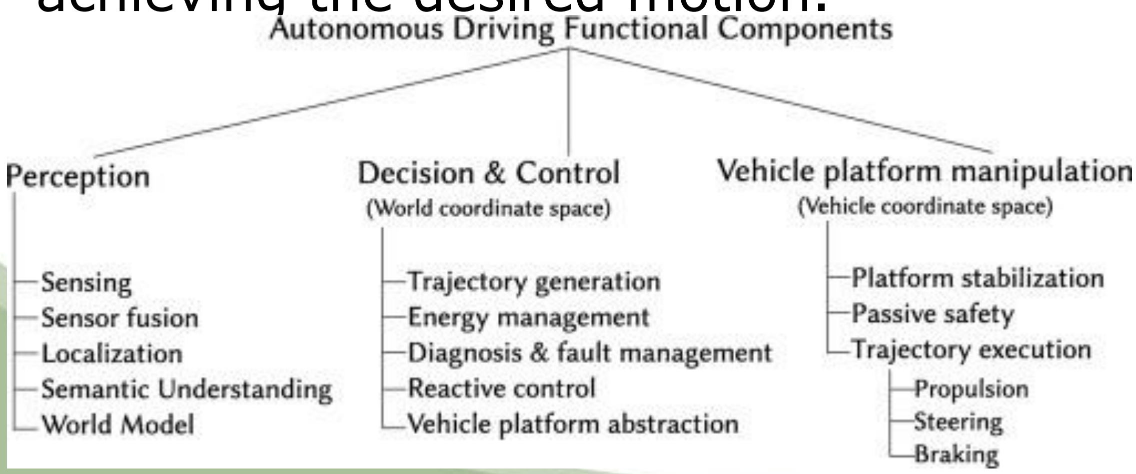
- The scope of the study is to build a visual inertial SLAM system that can provide point clouds and robot pose to other components in the perception/action pipeline of an autonomous robot.

Literature Review

Asian Institute of Technology

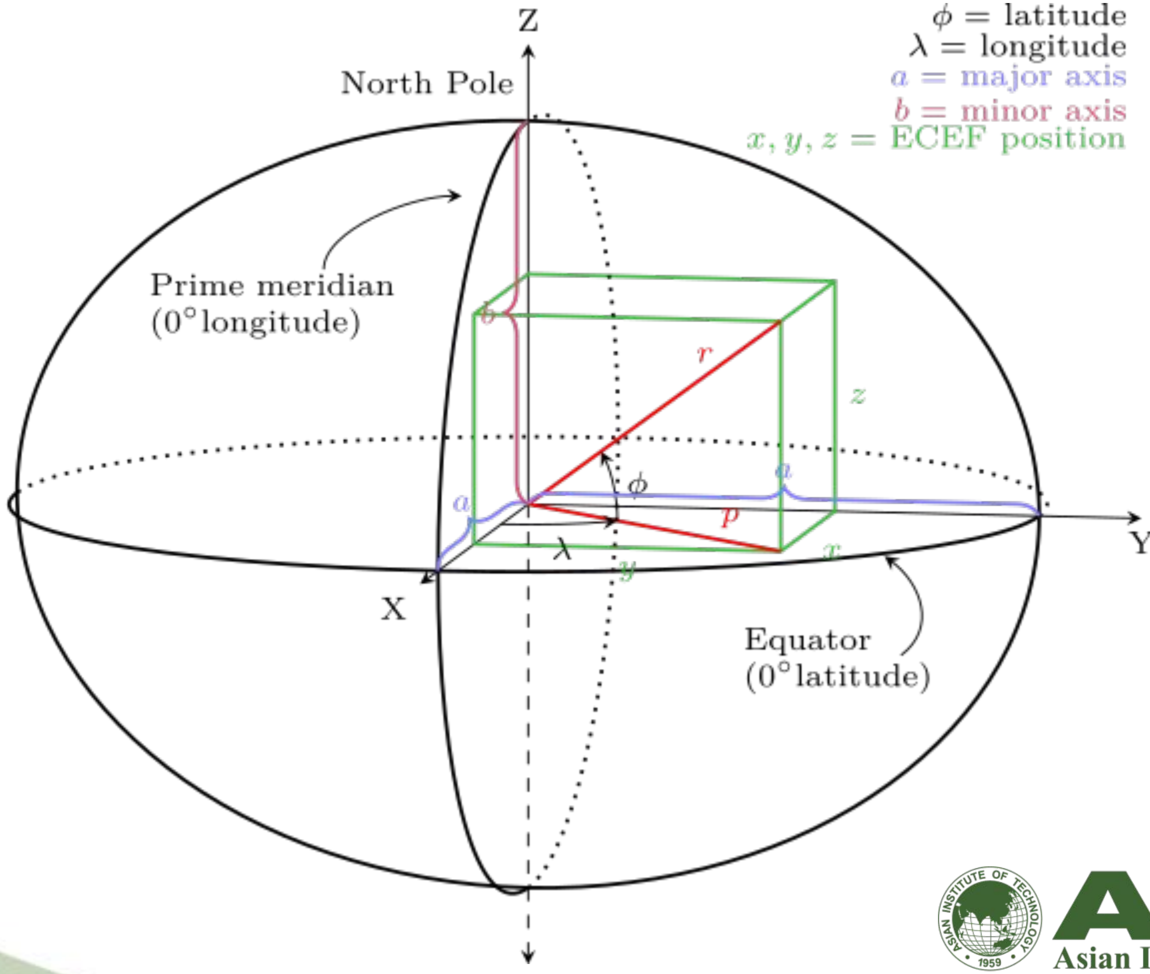
Software Architecture of Autonomous Driving Vehicles

- Perception of the environment in which the vehicle operates.
- Decision making and control of vehicle motion with respect to the external environment that is perceived.
- Vehicle platform manipulation, which deals with achieving the desired motion.



Asian Institute of Technology

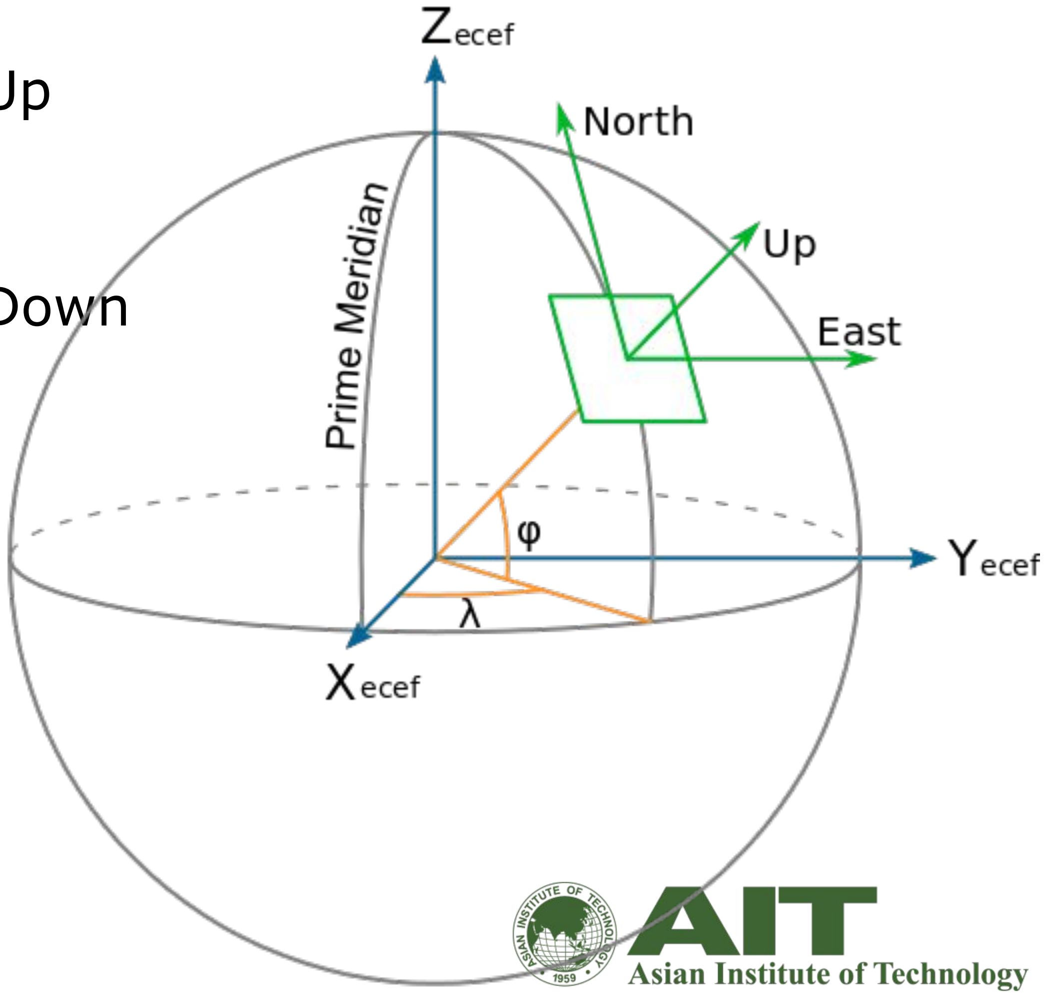
Coordinate Systems(Earth Centric, Earth Fixed)



Asian Institute of Technology

Coordinate Systems(Local Tangent Plane)

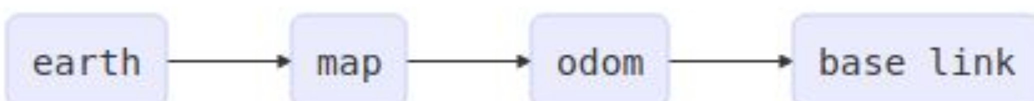
- X=East, Y=North, Z=Up (ENU).
- X=North, Y=East, Z=Down (NED)



Asian Institute of Technology

Coordinate Systems(PixHawk and ROS)

- PixHawk follows the NED convention.
- ROS follows the ENU convention.
- ROS has other reference frames
  - base_link is rigidly attached to the mobile robot base.
  - odom is a world fixed frame.
  - map is a world fixed frame. with Z axis pointing upwards.
  - earth is the origin on the ECEF frame



Simultaneous Localization and Mapping

- A method by which a mobile robot can build a map of an environment and at the same time use this map to infer its location.

- The trajectory of the platform and the location of all landmarks are estimated online without the need for any previous information of the location.

Asian Institute of Technology

Visual SLAM

- Images from one or more cameras are used for observation of the landmarks.
- Visual SLAM can be classified into feature based and direct methods.
- Feature based methods extract features from the input image and use those features as observations.
- Direct methods for vSLAM takes the whole image for tracking and mapping and produce dense point clouds.

AIT Asian Institute of Technology

Visual SLAM

- Some of the important vSLAM algorithms are
  - Parallel Tracking and Mapping (PTAM)
  - ORB SLAM
  - ORB SLAM2

- Some of the direct vSLAM algorithms are
  - Dense Tracking and Mapping (DTAM)
  - Large-Scale Direct SLAM (LSD SLAM)

Asian Institute of Technology

Visual SLAM Modules

- Initialization
- Tracking
- Mapping
- Re-localization
- Global map optimization

AIT Asian Institute of Technology

ORB SLAM

- Feature based vSLAM algorithm.

- Uses ORB feature detectors and the same features are used in mapping as well as tracking.

- It has three threads running in parallel for tracking, local mapping and loop closing.

Asian Institute of Technology

Visual Inertial ORB SLAM

- Extends upon ORB SLAM by integrating stream of IMU measurements which solves the scale ambiguity problem of monocular SLAM.

- In tracking, the IMU pose is estimated and the camera pose is predicted.

- In local mapping, the local window is retrieved by the transitory sequence of keyframes, while in ORB SLAM covisibility graph is used.

- In loop closing, the pose graph optimization is performed on 6 degrees of freedom instead of 7 degrees of freedom as scale is observed.

Asian Institute of Technology

Methodology

Hardware

- The Mobius Maxi action cam mounted on a rigid base.

- The PixHawk 2.4.8 flight controller with internal IMU and external GPS module is attached to the same rigid base.

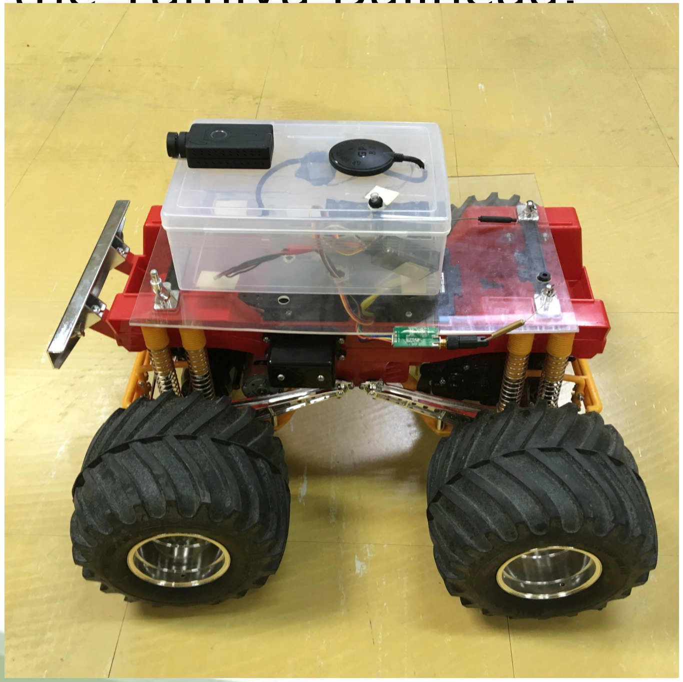
- Both devices are connected through USB to a laptop with
  - i7 CPU
  - 16 GB of RAM
  - Ubuntu 16.04

Asian Institute of Technology

Hardware

- PixHawk has PX4 as its firmware.

- The test rig is mounted on top of RC vehicle, the Tamiya Bullhead.

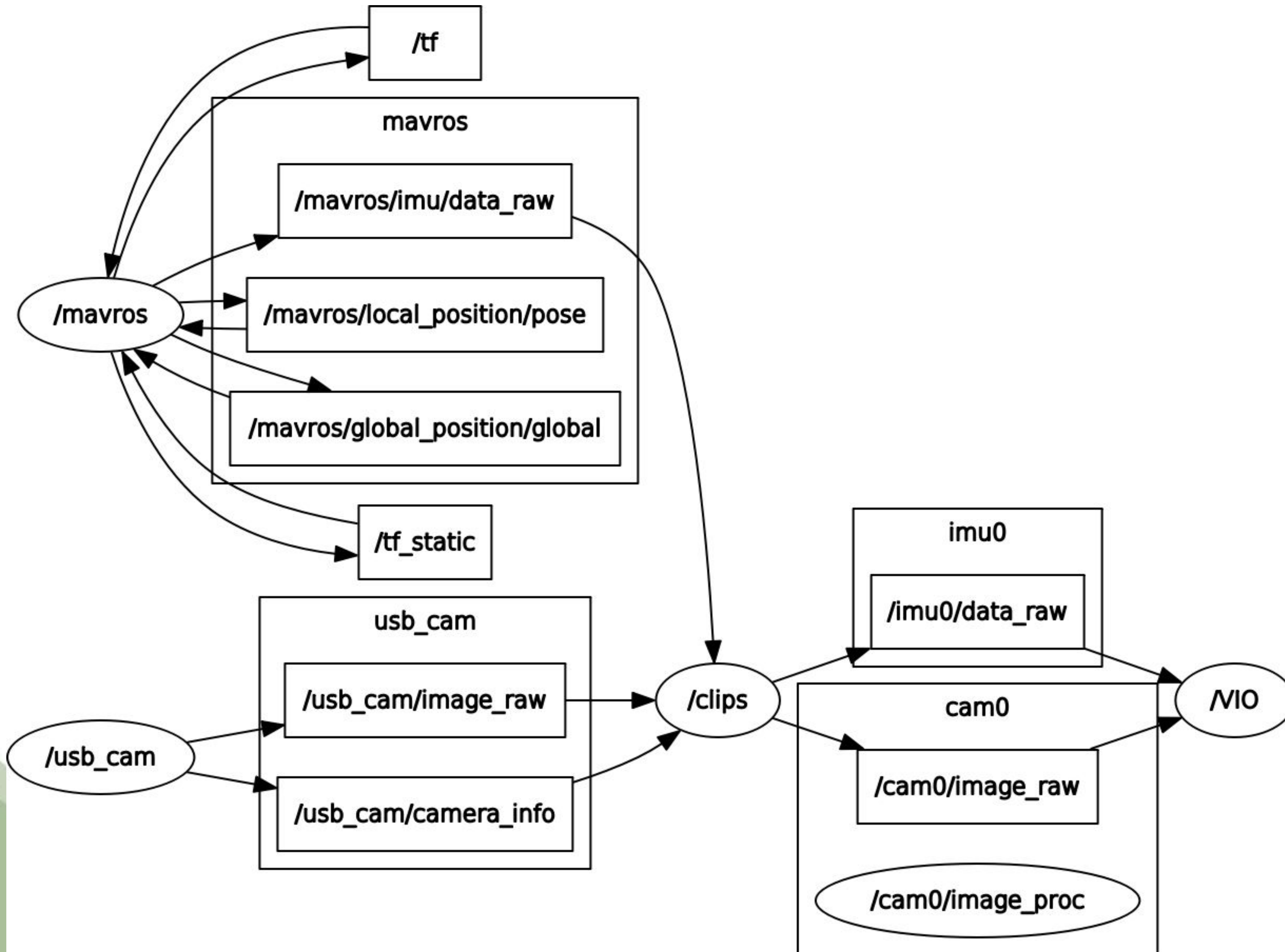


AIT
Asian Institute of Technology

Software

- Robot Operating System (ROS) version Kinetic.
- Four ROS packages running on the test laptop.
  - usb_cam: Modified to support H264 streams.
  - mavros: Communicates with PixHawk using MAVLINK protocol.
  - clips: A new middleware layer to ensure at least one IMU measurement is present between each pair of image frames.
  - LearnVIORB: Modified to publish pose, point clouds, and tf information.

Software


/tf
mavros
/mavros/imu/data_raw
/mavros
/mavros/local_position/pose
/mavros/global_position/global
/tf_static
imu0
/imu0/data_raw
usb_cam
/usb_cam/image_raw
/clips
/VIO
cam0
/usb_cam
/usb_cam/camera_info
/cam0/image_raw
/cam0/image_proc

Software (usb_cam)

- Mobius Maxi supports H264 hardware encoding.
- Video stream captured through H264 stream is smoother and supports higher framerates than raw stream.
- The library libavutils used by usb_cam has support for H264 streams.
- Added support for h264 pixel format.

Software (mavros)

- The PixHawk connection is shown as serial port /dev/ttyACM0 on the test laptop.
- The baud rate is 921600.
- PX4 publishes coordinates using the NED convention.
- mavros handles the conversion of NED coordinates to ENU coordinates to use with ROS.
- sys_time mavros plugin handles time synchronization between the host and the remote flight controller.

Asian Institute of Technology

Software(sys_time)

- A timesync message M is sent at a consistent frequency from the host system to FCU with with timestamp t_s.
- The remote FCU adds its timestamp t_c and replies.
- It is assumed that the round trip time for the message is equal in both ways,

$$t_{offset} = t_s + t_{now} - 2t_c$$

Asian Institute of Technology

Software(sys_time)

- t_{offset} is used for interpolating the time offset and time skew with a window of size n. k = {0 … n-1} is the index of the timesync message used for interpolation before convergence.

$$I = k/n$$

$$P = 1 - e^{\frac{1}{2}\left(1 - \frac{1}{(1-I)}\right)}$$

$$\alpha = P \times \alpha_f + (1 - P) * \alpha_i$$

$$\beta = P \times \beta_f + (1 - P) * \beta_i$$

Asian Institute of Technology

Software(sys_time)

- Online exponential smoothing filter is used to calculate interpolated time offset t and time skew t_{skew}.

$$t_k = \alpha \times t_{offset} + (1-\alpha) \times (t_{k-1} + t_{skew_{k-1}})$$

$$t_{skew_k} = \beta \times (t_k - t_{k-1}) + (1-\beta) \times t_{skew_{k-1}}$$

- If a maximum number of consecutive observations are received that is higher than a threshold deviation from the present estimate, the interpolation is reset.

- If the interpolation has not converged, the intermediate t and t_{skew} are used.

Asian Institute of Technology

Software (clips)

- The IMU and camera are not tightly bound.
- At least one IMU measurement between any two consecutive image frames is necessary for VIORB to predict the next keyframe location.
- This custom package which I created ensures that there is IMU measurement between two image frame.
- Drops the image frame if IMU measurement is not received.

Asian Institute of Technology

Software (LearnVIORB)

- LearnVIORB is a community-developed implementation of Visual Inertial ORB SLAM.
- I added support for publishing pose, point cloud, and tf information as ROS topics.
- I referred to the open source project ORB_SLAM2_CUDA (Nguyen)

Software(PX4 firmware)

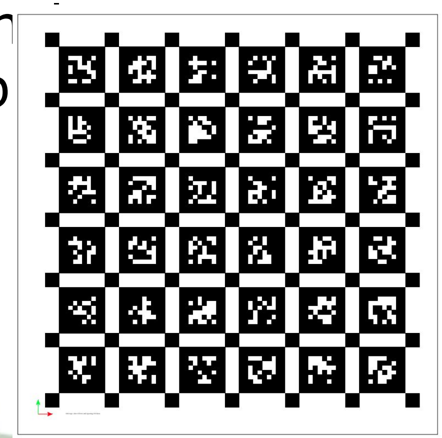
- PX4 by default does not support more than 50 Hz IMU rate.
- In the source code ROMFS/px4fmu_common/init.d/rcS had to be edited and re-compiled to increase IMU measurement rate to 200 Hz.

Calibration

- Camera intrinsic parameters were calibrated with the ROS camera_calibration package, using a checkerboard pattern.

- Noise density, and random walk for the accelerometer and gyroscope was calculated using kalibr_allan through Allan variance method.

- Camera-to-IMU tran as calculated using kalibr tool, using ap MU noise analysis results.



Asian Institute of Technology

Integration

- A configuration yaml file configured with the results of calibration.

- IMU noise density and random walk constants in LearnVIORB src/IMU/imudata.cpp was updated.

- A launch file in clips starts usb_cam, mavros, and clips and publishes /imu0/data_raw and /cam0/image_raw.

- The topics can be stored in a bag file or directly subscribed by LearnVIORB.

Experimental Results

Asian Institute of Technology

Experiment Environment

- The test rover was manually controlled in indoor and outdoor environment.
- The test laptop was connected to the camera and the flight controller.
- The data collected was stored in a ROS bag file.
- The bag file was replayed and the collected data was used to run VIORB and ORB SLAM.

Asian Institute of Technology

Indoor Experiment

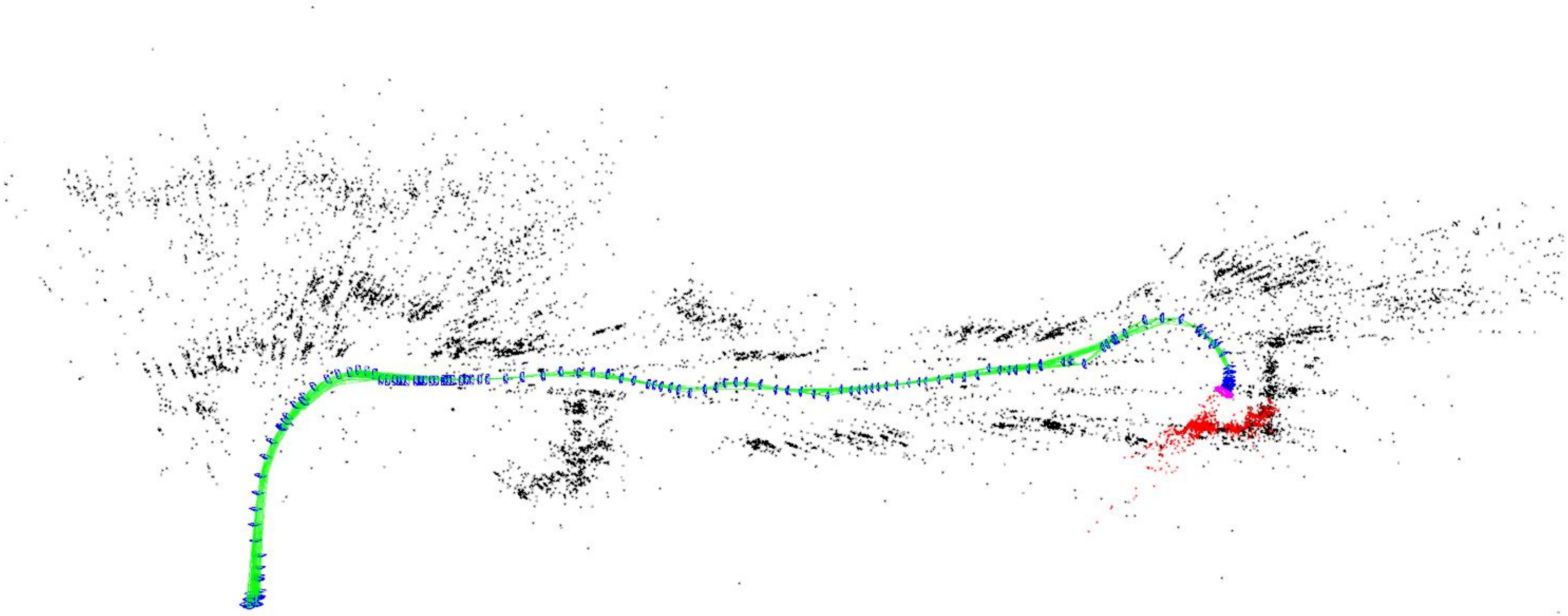
- The test rover was run in indoor environment for 170 seconds.
- The number of ORB features to extract was kept at 2000.
- The approximate length of the path was 17 meters.

Asian Institute of Technology

Indoor Experiment

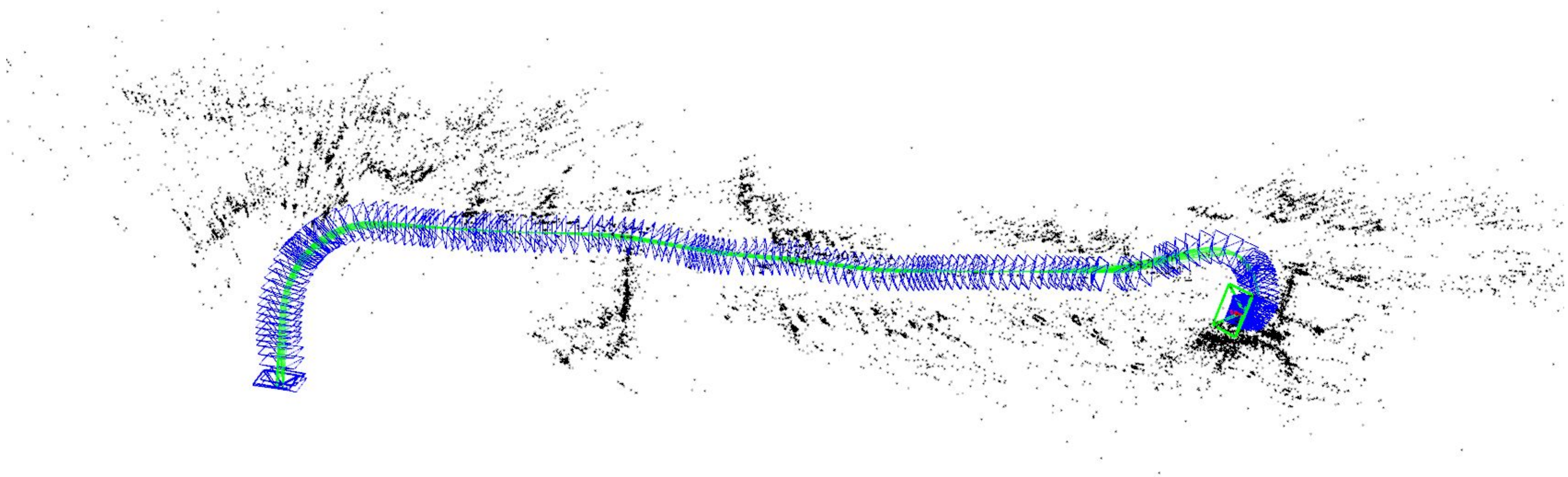
- VIORB SLAM relocalized a couple of times.
- VIORB SLAM lost tracking and could not be relocalized after 117.5 seconds.
- ORB SLAM was more robust and did not lost track.
- ORB SLAM reported path length of 3.92 meters. VIORB SLAM reported path length of 13.875 meters.

Indoor Experiment (VIORB SLAM)



Asian Institute of Technology

Indoor Experiment (ORB SLAM)



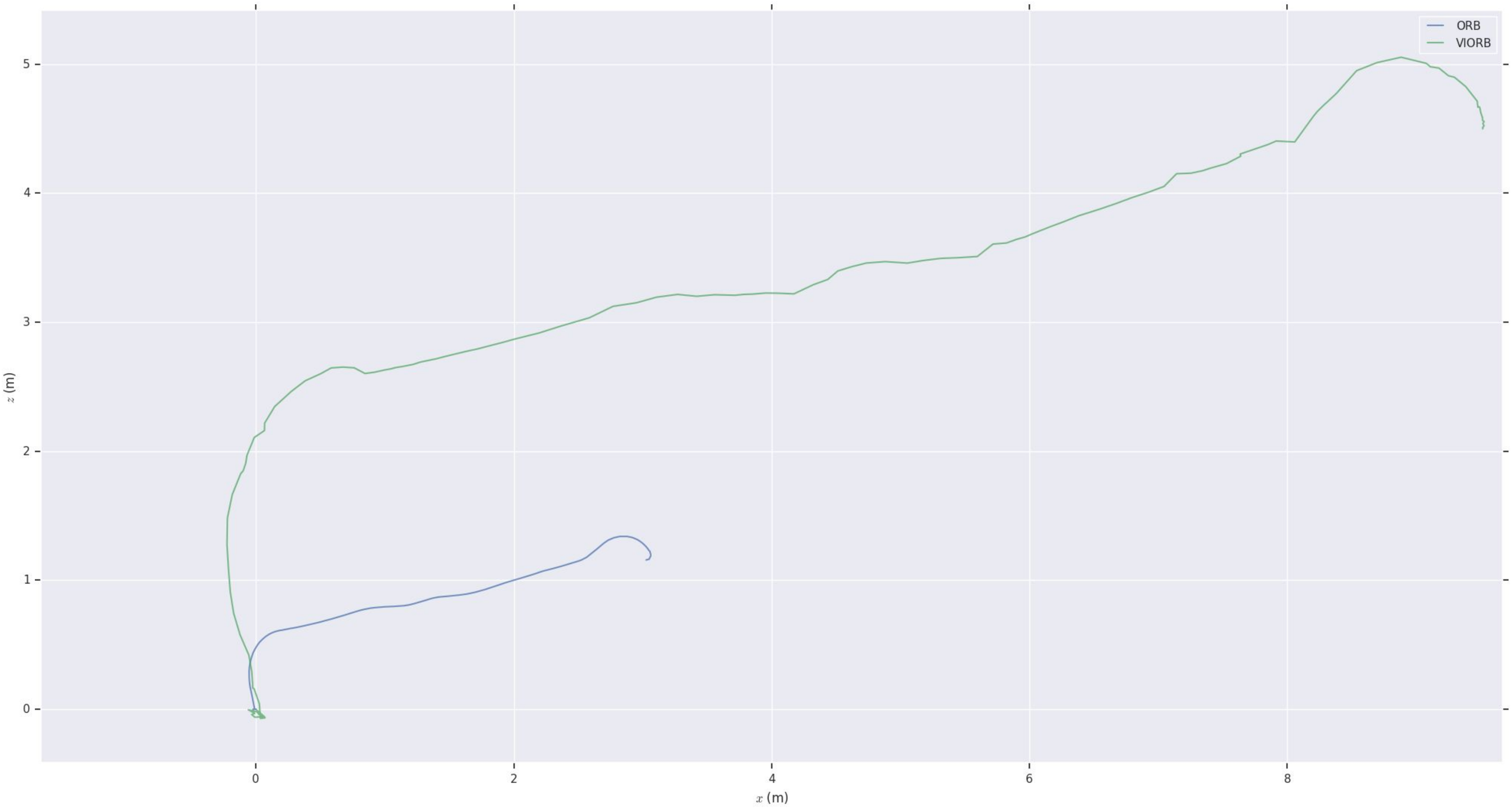
Asian Institute of Technology

Indoor Experiment

Algorithm	Total Time	IMU initialization time	Duration tracked	Poses	Path length
ORB	170s	NA	161.148s	249	3.920m
VIORB	170s	15s	117.459s	155	13.875m

Asian Institute of Technology

Indoor Experiment



Outdoor Experiment

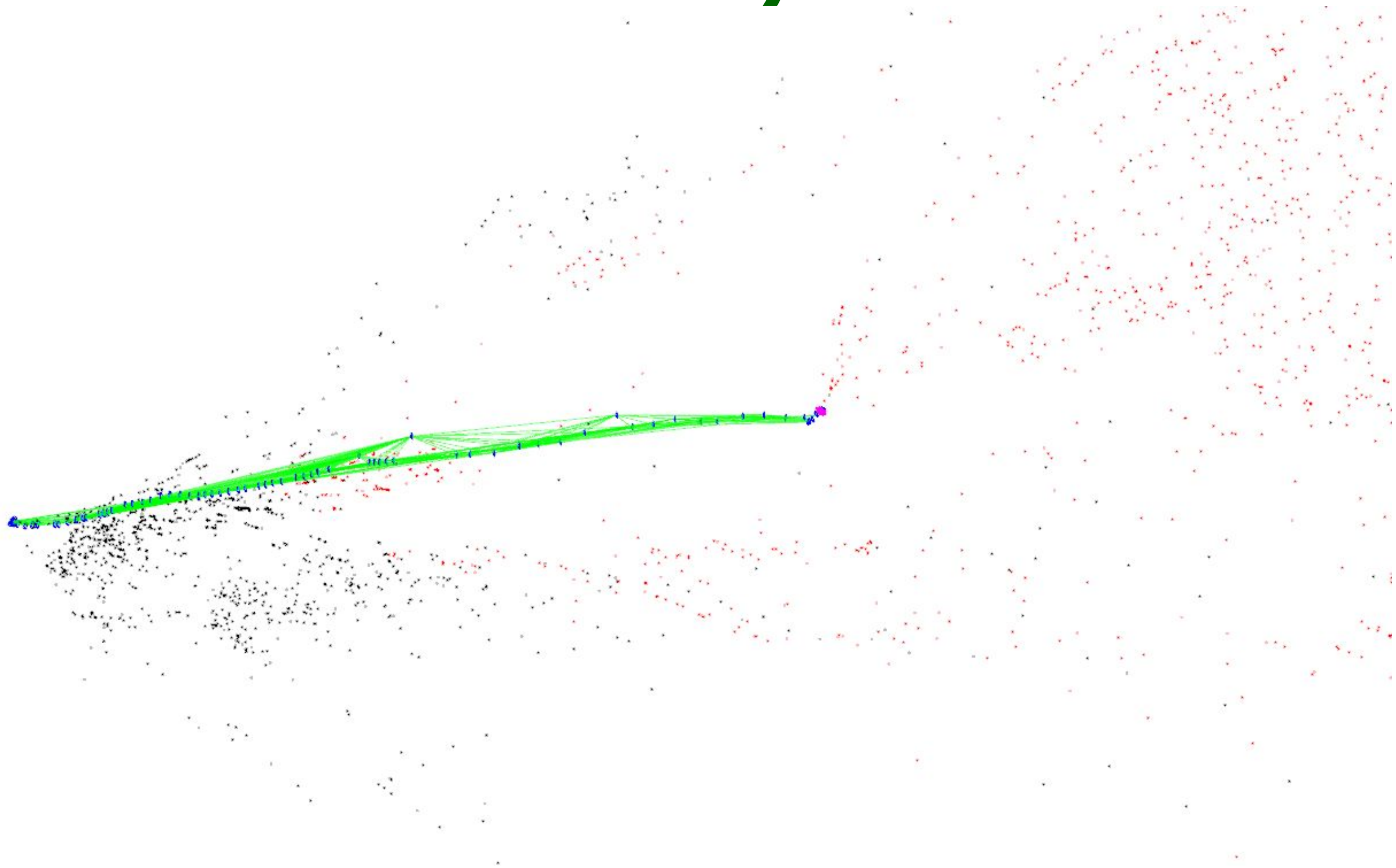
- The test rover was run in outdoor environment for 174 seconds.
- The number of ORB features to extract was kept at 2000.
- The rover faced more jerky movements than the indoor experiment.

Asian Institute of Technology

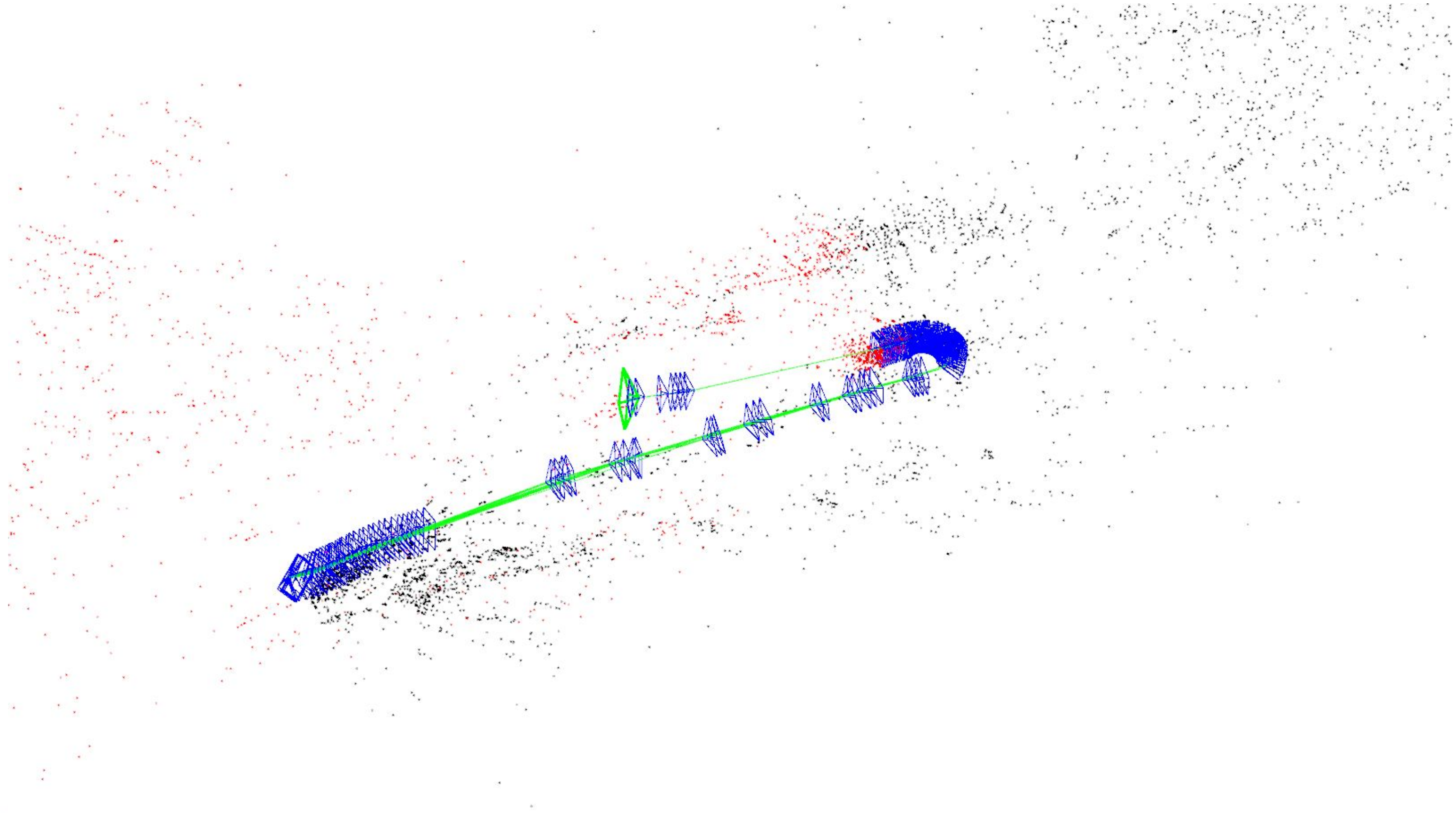
Outdoor Experiment

- VIORB SLAM relocalized a couple of times.
- VIORB SLAM lost tracking and could not be relocalized after 96.871 seconds.
- ORB SLAM was more robust and did not lost track.
- ORB SLAM reported path length of 3.15 meters. VIORB SLAM reported path length of 14.427 meters.

Outdoor Experiment (VIORB SLAM)



Outdoor Experiment (ORB SLAM)



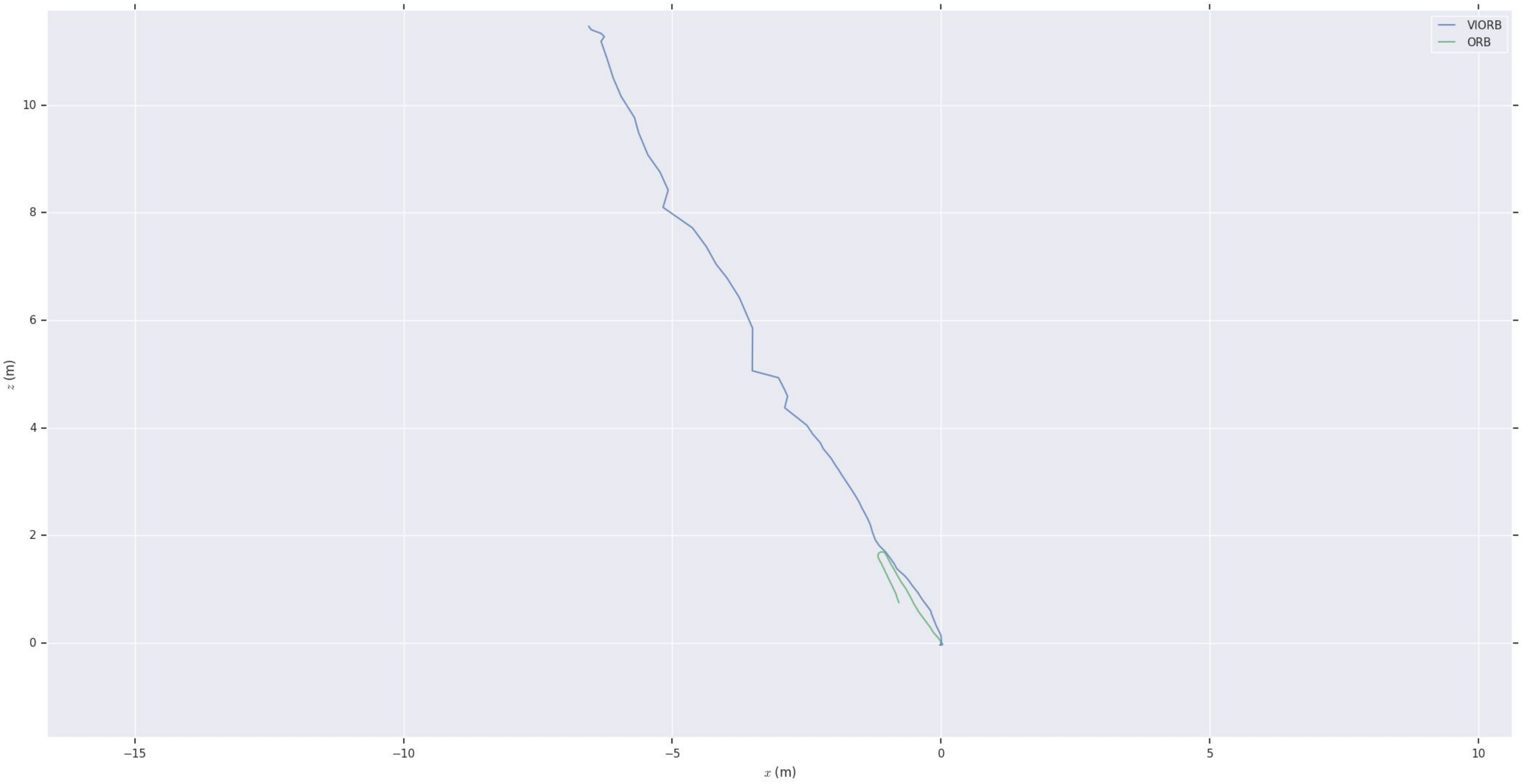
Asian Institute of Technology

Outdoor Experiment

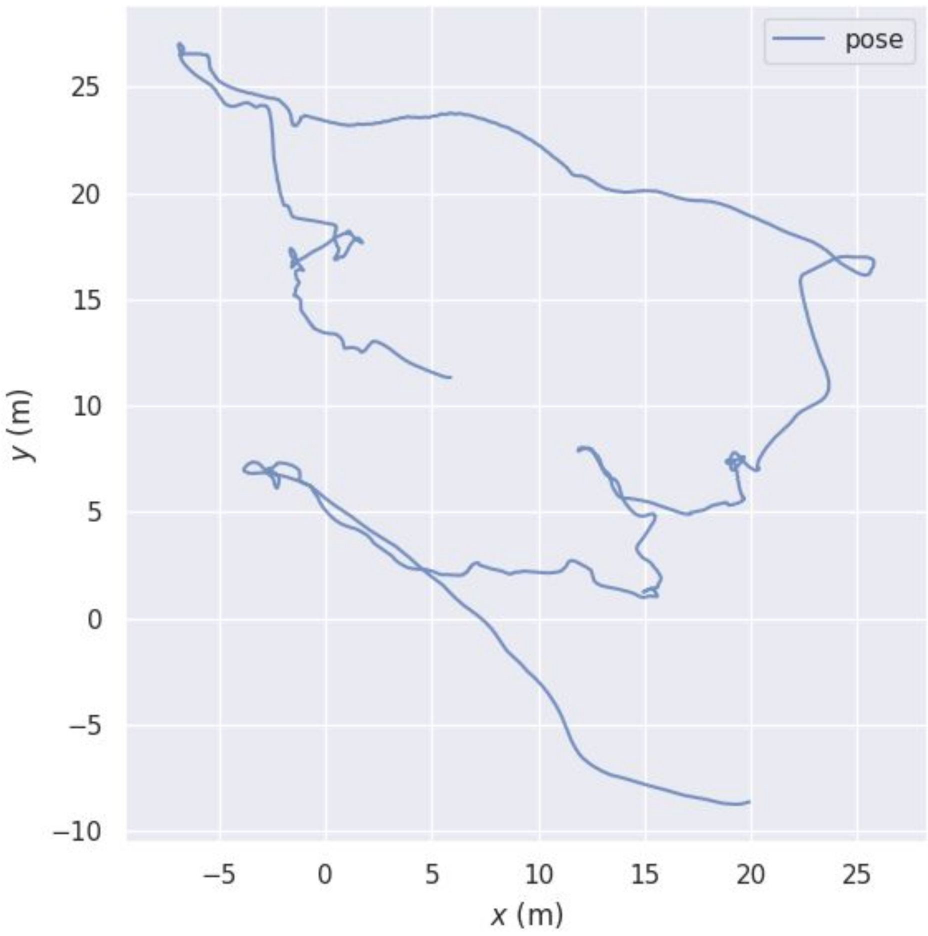
Algorithm	Total Time	IMU initialization time	Duration tracked	Poses	Path length
ORB	174s	NA	160.637s	134	3.150m
VIORB	174s	30s	96.871s	85	14.427m

Asian Institute of Technology

Outdoor Experiment



Outdoor Experiment



Asian Institute of Technology

Conclusion and Recommendations

Conclusion

- VIORB SLAM is closer to the ground truth in approximating the scale of the constructed map than ORB SLAM.
- ORB SLAM is more robust than VIORB SLAM and does not lose track as easily.
- Setting up VIORB SLAM is more complex than ORB SLAM. This increased complexity may be one reason VIORB SLAM is less robust than ORB SLAM.

Conclusion

- The interpolated time offset interpolated by sys_time plugin may not represent the actual time offset. This may affect VIORB SLAM.
- The GPS module attached to the PixHawk drifts and is not precise over short distances.
- VIORB and ORB SLAM publish pose in a zyx coordinate system, where the z axis protrudes out in the front direction of the camera or the IMU.

Recommendations

- Proper calibration may make VIORB more robust.
- Replacing the internal IMU in PixHawk with an external IMU having low noise, may make the calibration more accurate.
- A tightly coupled camera-IMU system, where the camera is triggered through the host computer or the flight controller may help in time synchronization.
- Reducing jerky movements through mechanical means while the rover is moving on the ground may make it more robust.

AIT Asian Institute of Technology

Thank You