# NORTHEASTERN UNIVERSITY
**Boston, Massachusetts**

# ROBOT SENSING AND NAVIGATION
**EECE 5554**

# PROJECT REPORT
# ON
# VISUAL SLAM UTILIZING ORB SLAM 3


**GROUP1:**
Vignesh Ravikumar
Skanda Akkihebbal Prasanna
Pavan Rathnakar Shetty
Shanmugam Sankarakumar

**Instructor:**
Hanumat Singh
Professor, ECE
Northeastern University

## Introduction:

ORB-SLAM3 is abbreviated as Oriented Fast and Rotated Brief. Further 'fast' and 'brief' are abbreviated as Features from Accelerated and Segments Test and Binary Robust Independent Elementary Feature respectively. ORB-SLAM 3 is the first real-time SLAM library able to perform Visual, Visual-Inertial, and Multimap SLAM with monocular, stereo, and RGB-D cameras, using pinhole and fisheye lens models.

In all sensor configurations, ORB-SLAM 3 is as robust as the best systems available in the literature and significantly more accurate. ORB-SLAM 3 is able to survive long periods of poor visual information: when it gets lost, it starts a new map that will be seamlessly merged with previous maps when revisiting mapped areas. Compared with visual odometry systems that only use information from the last few seconds, ORB-SLAM 3 is the first system able to reuse in all the algorithm stages all previous information. This allows it to include in bundle adjustment co-visible keyframes that provide high parallax observations boosting accuracy, even if they are widely separated in time or if they come from a previous mapping session.

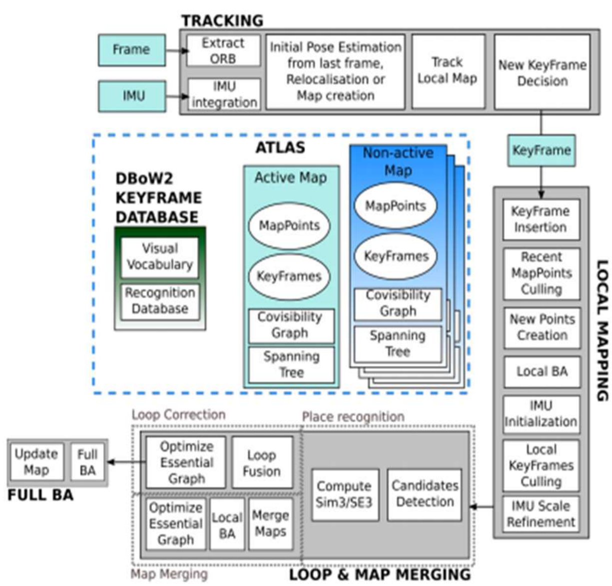



**Figure1: System Architecture of ORB-SLAM3**

The above figure represents the system architecture of ORB-SLAM3 and it has the following structures:

- **Tracking Thread**: It extracts ORB features and compute pose in the frame and locates each frame from Atlas maps
- **Atlas and DBoW2 Keyframe Database**: It stores unique keyframes and contains a localized active map
- **Local Mapping Thread**: It Adds keyframes to the active map, removes duplicate keyframes, performs bundle adjustment, and refines IMU parameters with map estimation
- **Loop & Map Merging Thread**: It Identifies common regions in the active map, stores keyframes, merges different maps, and performs loop correction
- **Full Bundle Adjustment**: It Update map for reprojection errors

## Methodology:

In this project we utilized the official github repository of ORB-SLAM3 from the authors: Carlos Campos, Richard Elvira, Juan J. Gómez Rodríguez, José M. M. Montiel, Juan D. Tardos.

- We performed Monocular ORB-SLAM3 on the morning_stereo_rgb_ir_lidar_gps.bag dataset that was collected from Nuance Car.
- We subscribed to the topic /flir_boson/camera_info to obtain IR data from the rosbag file
- Two files ORBvoc.txt and morning.yaml files were modified and used to perform ORB-SLAM3 on monocular IR camera data
- ORBvoc.txt is essentially a database of previously seen feature descriptors stored in a formatted file for fast place recognition to trigger loop closure instead of matching features one by one against all images. This is a generic file that is already trained and could be replaced with a similar file from custom training.
- Morning.yaml includes parameters like camera calibration and distortion parameters, camera width, camera height, camera frames per second, ORB extraction feature, etc.
- We chose 3000 feature points to perform ORB- SLAM3 on Monocular IR data. This was decided after tweaking the value and performing SLAM to see the best result possible.

## Results:

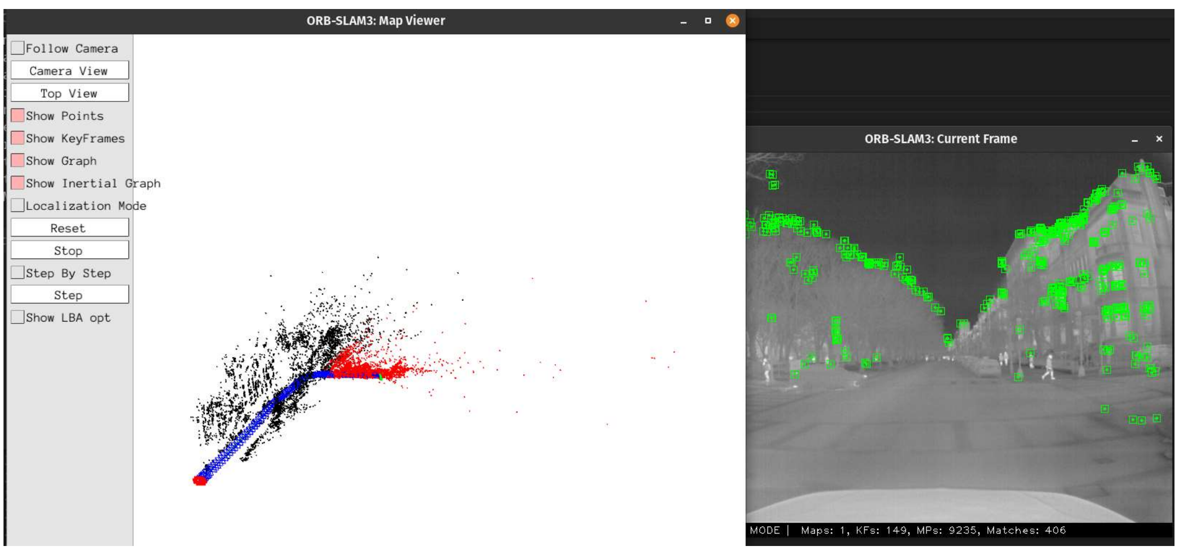
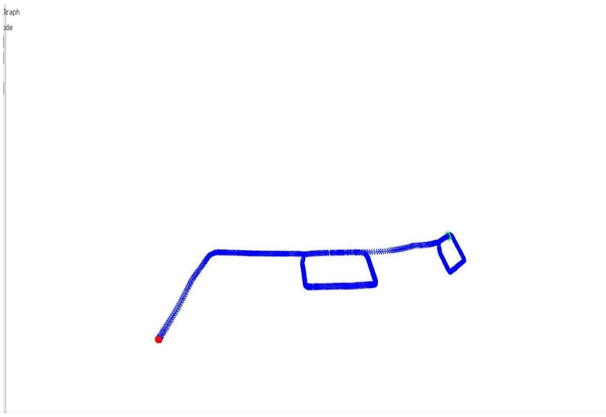



**Figure2: ORB SLAM3 on Monocular IR Data**



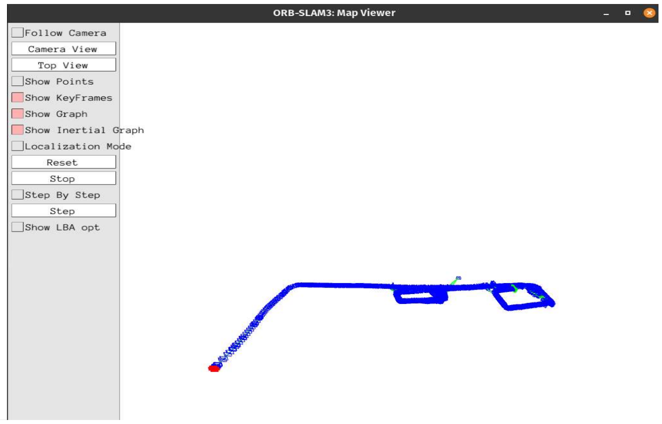



**Figure3: Single Session** **Figure4: Multi Session**

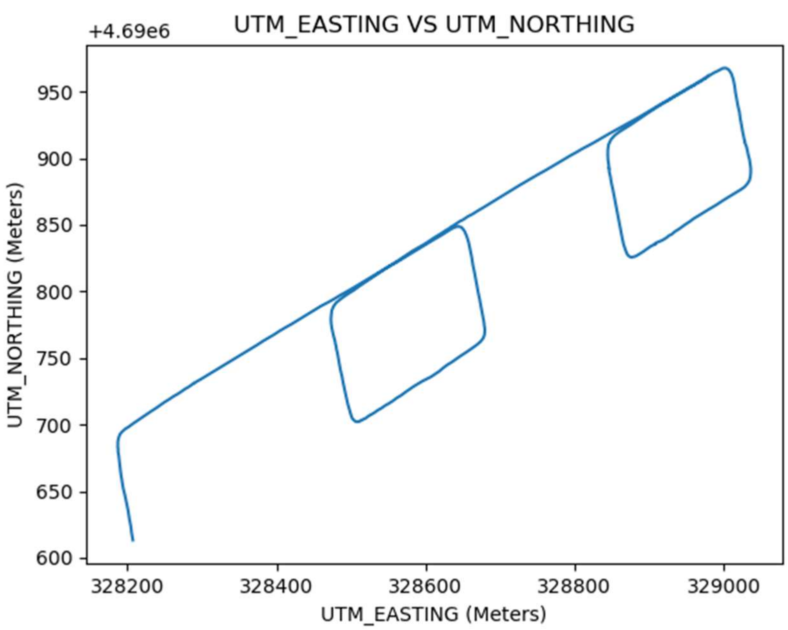



**Figure5: GPS Ground Truth**

## Observations:

Feature Detection:
Most of the features recorded were edges of the buildings and trees where there is a sharp contrast to the neighboring surrounding and very few features from the sky and roads were recorded since the contrast wasn't obvious. The algorithm ignores features on moving objects that are temporarily in the frame. This is an extended advantage of loop closure where the temporary object is ignored to be a part of keyframes.

Single Session SLAM:
Identifies the two individual loops but loses tracking at the end of the first and terminating stages of the second loop. Multimaps are created since tracking was lost at different stages of the path. A new map was created at the start of the second loop, and a third map at the end of the second loop.

Multi-Session SLAM:
We ran multiple sessions of SLAM and with each iteration, the output got progressively better. We were able to observe how well the map merges and loop closures at high frequency. Initially tracking was consistently lost at the sharp turns that were taken at high speeds, which is a limitation of VO systems. As more keyframes were stored, due to long-term data association, with subsequent iterations, tracking during turns was more consistent and reliable. There was a significant improvement in the map post the second iteration.

## Conclusions:

ORB-SLAM3 works well in low-speed and moderate-speed scenarios. In high-speed scenarios, it is difficult to identify and track features in corners. This is clear in the turns, where the feature extraction is relatively less when compared to straight lines where it is easy to keep track of repeating features which are then cached and used in the multi-session slam for improvement and tracking of the maps when lost.
Long-term data association in ORB-SLAM3 helps create accurate maps when there is an occasional loss of tracking. This is seen in the car_ir_rgb dataset where subsequent iterations of SLAM rely on past keyframes to match maps despite occasionally losing tracking.

ORB-SLAM3 is likely to display higher accuracy and robustness in indoor conditions where speed is lower, and more features are clearly visible. Moreover, the indoor dataset used has room for a lot of feature extraction at a particular position making the tracking easier without the need for Multi-Session SLAM. However, this does not appear to be the case with outdoor data at higher speeds. The advantage of IR data is that it is reliable in identifying features, this is because surface features are smoothed out and the edges of objects are in sharp definition to their backgrounds. This might not be the case with RGB data where there will be reliability issues caused by pixelation and noise.

## References:

https://github.com/UZ-SLAMLab/ORB_SLAM3
https://arxiv.org/pdf/2007.11898.pdf%5C
https://drive.google.com/drive/folders/1KOj2JToFH7r_fF0BH2X8qWngw4rd6YSc?usp=sharing