MCC-EKF for Autonomous Car Security

Mushary Ali Alghamdi

Introduction

- Autonomous cars require SLAM algorithms
- SLAM (Simultaneous Localization and Mapping) algorithms allow autonomous cars to map new environments.
- They enable autonomous cars to know and navigate their environment
- Example of SLAMs include Oriented fast and Rotated Briefs-SLAM, and Convoluted Neural Network SLAM, LIDAR and Visual SLAMS.
- It is an essential component that enables to build a map and localize the car in the map.

Problem being solved

- The research attempts to solve to related problems?
- 1) Can we use the odometry information from two different Simultaneous Localization And Mapping (SLAM) algorithms to get a better estimate of the odometry?
- 2) How can the security of the SLAM be improved to minimize shot noise and attack from hackers.

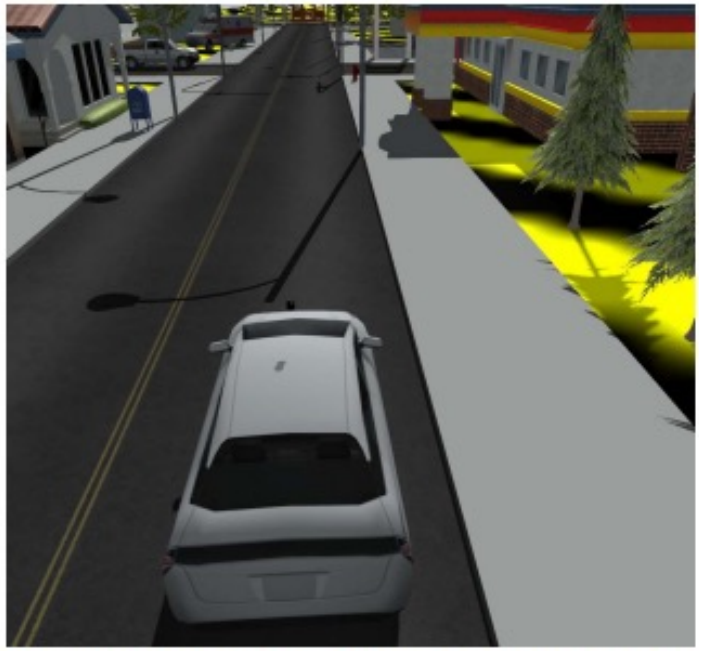
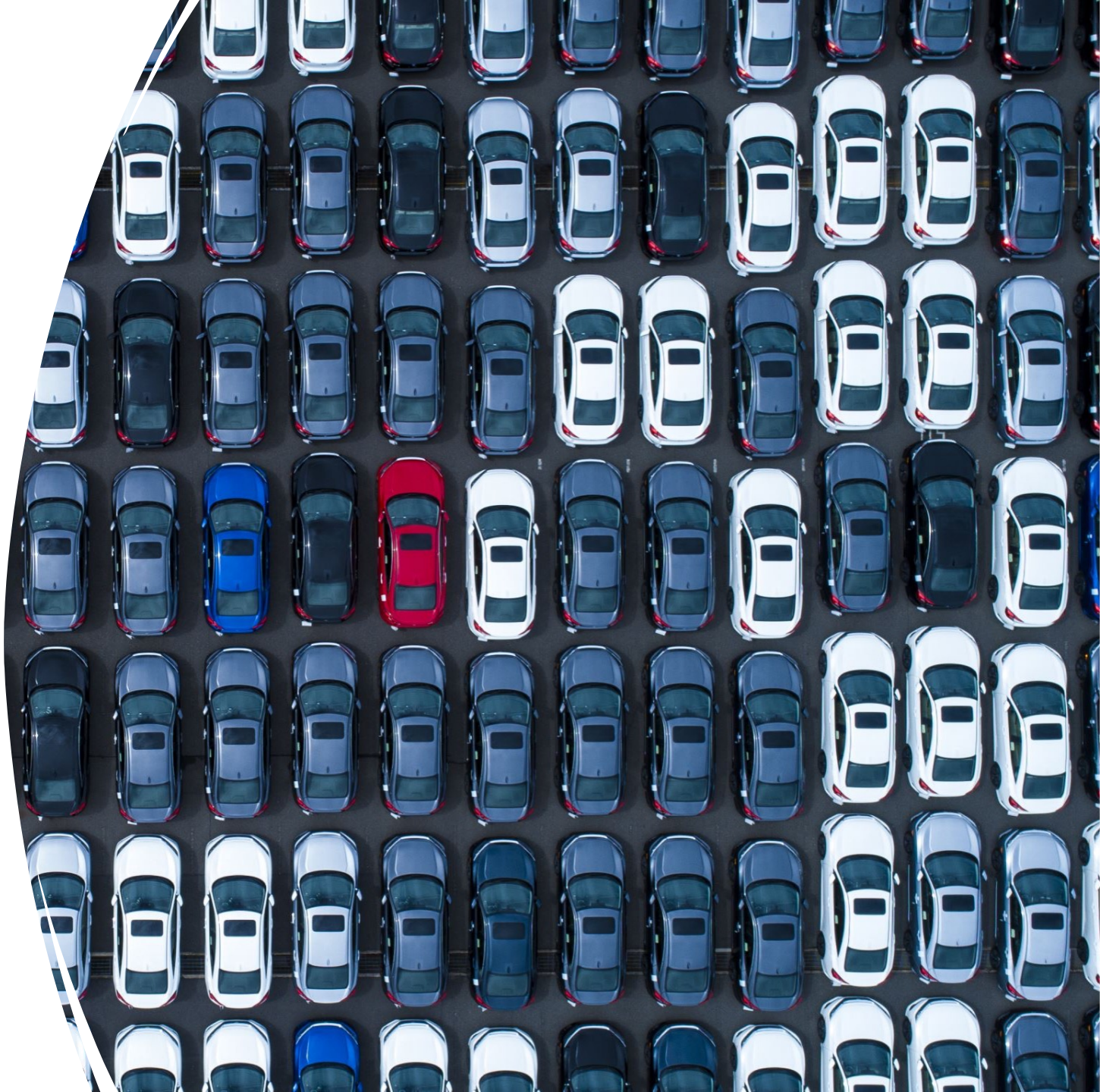


Fig. 3: Zoom-in at one location in Gazebo Simulation Environment.

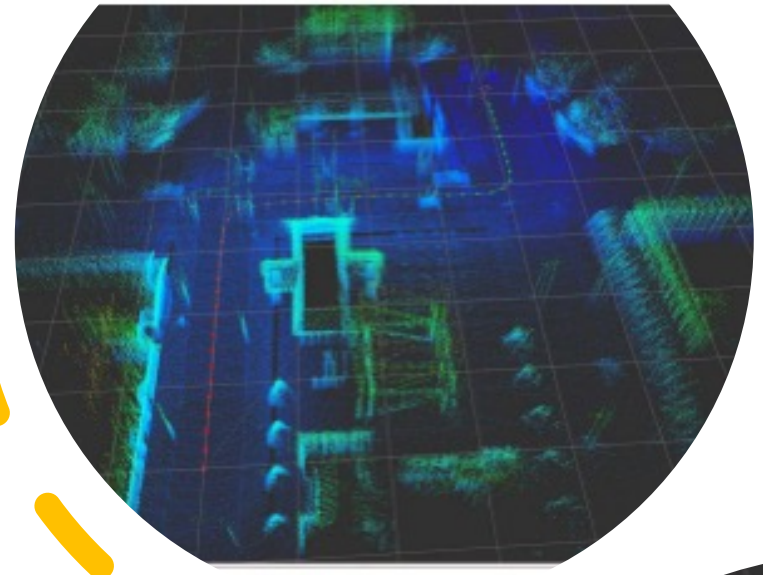
Is the problem significant

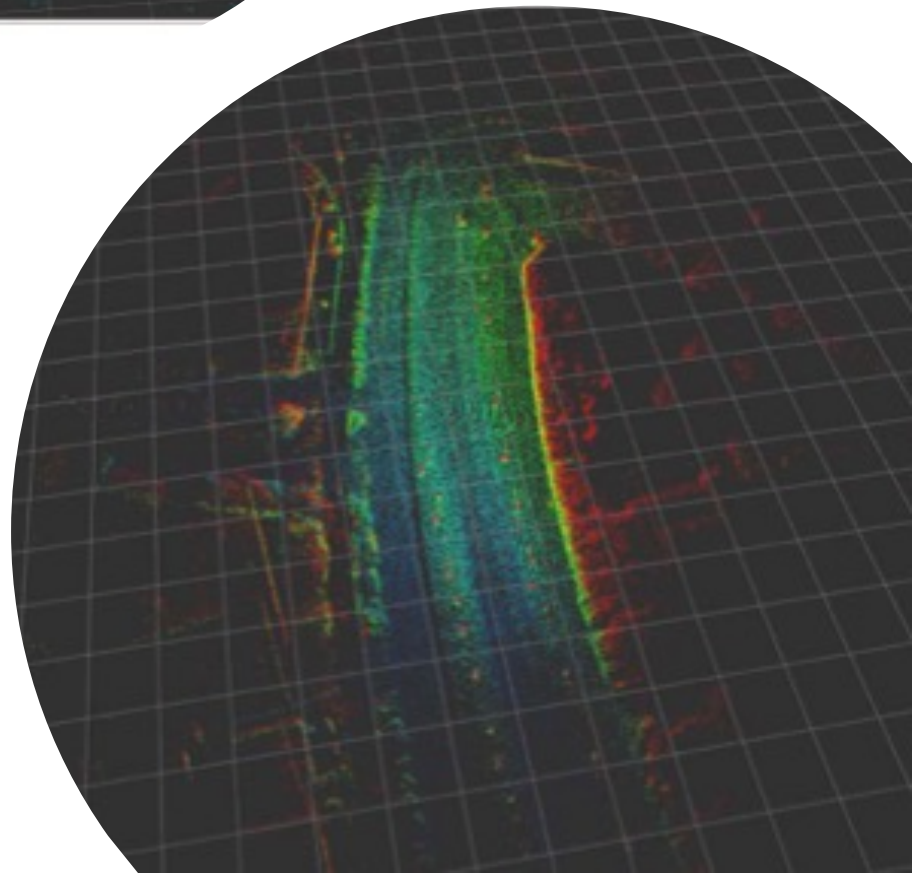
- Yes, the problem is significant in the modern technological era especially in the autonomous car industry.
- By analyzing whether it is possible to use two different SLAM algorithms, autonomous cars manufacturers can learn how to improve navigation and mapping of the cars improving the overall trajectory.
- The research will also help understand how to improve the security of the SLAMs. An unsecure SLAM can be hacked which can change its natural trajectory.
- The research will help know whether a SLAM can be self-secure.



Author’s assumptions

- Two SLAM algorithms can be used to improve the trajectory of autonomous cars. This will help to improve the localization and mapping of autonomous cars and map effectively in new environments.
- SLAM algorithms can be self-secure. The authors assume that SLAM algorithms can be self-secure and detect and thwart attacks to avoid change in natural trajectory.



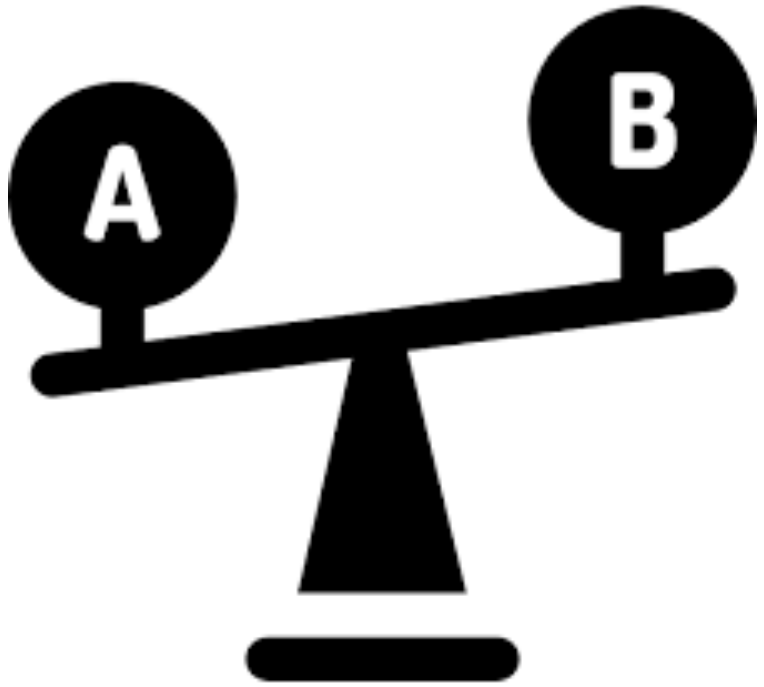


Are the assumptions realistic?

- Yes
- There are several SLAMs such as Lidar-based SLAM, Visual-based SLAM, CNN-SLAM, ORB2-SLAM
- SLAM combines different fields such as deep learning, signal processing, and computer vision. Therefore, two SLAMs can be used to get a better odometry estimate.
- Only few SLAM algorithms are self-secure. It is realistic to test whether a SLAM algorithm can be self-secure to detect and thwart attacks.

Comparison to others

- It is a unique practical study
- It addresses gaps that will help improve the efficiency and security of autonomous cars.
- With a self-secure SLAM, it has the capacity to revolutionize autonomous car industry by reducing hacking threat ensuring the systems perform as expected.



Proposed solution

- Uses two different SLAM algorithms to get the odometry.
- The set-up had simulated Lidar and stereo camera.
- The experiment is carried out in a simulated gazebo environment from an open-source repository. The experiment is also simulated in the KITTI data set
- The experiment assesses how odometry information from two SLAM methods can improve overall odometry.

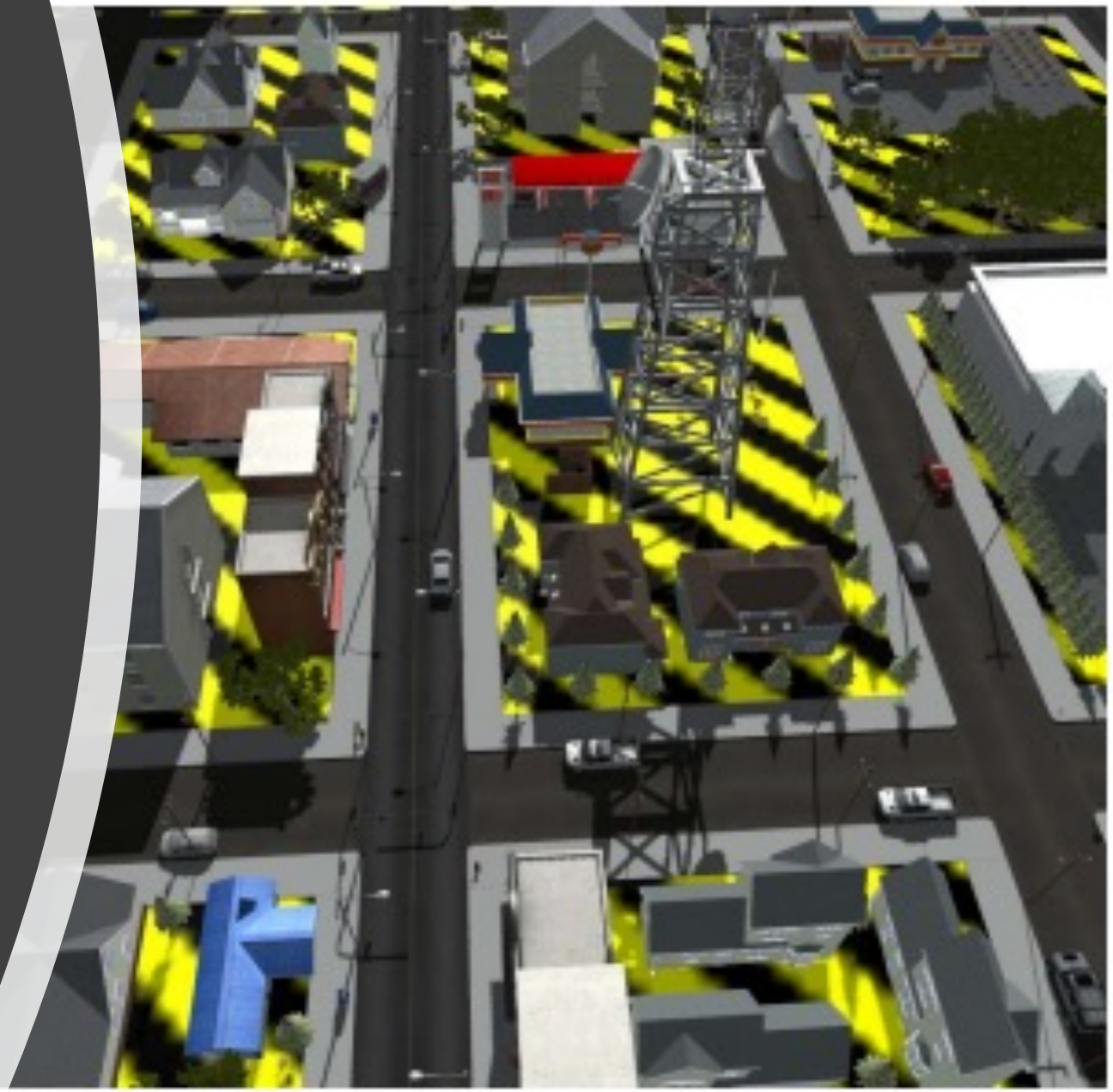


Fig. 2: Gazebo Simulation Environment.

Proposed Solution: KALMAN-FILTER CORRENTROPY

- It is important in removing large outliers
- Operates as a weighted least squares approach
- It was used to solve the problem of getting a better odometry estimate using two different SLAM algorithms.
- Lidar-based SLAM and Visual-based SLAM were fused using the Extended Kalman Filter algorithm.

Algorithm 1: MCC-EKF algorithm for autonomous system security

Result: Computed odometry from MCC-EKF SLAM
Lidar Odom (ICP) $\rightarrow L_o(x, y, z, r, p, y)$
Stereo Odom (F2M) $\rightarrow S_o(x, y, z, r, p, y)$
Initialization;
Lidar odom $\rightarrow$ (L_o)
Stereo odom $\rightarrow$ (S_o)
Compute x_0 from Equ. (13)
Compute P_0 from Equ. (14)
Prior Estimation;
Compute $\hat{x}_k^-$ from Equ. (15)
Compute $P_{k|k-1}$ from Equ. (16)
while *get L_o and S_o;*
do
  Compute L_k from Equ. (24)
  Compute Gain K_k from Equ. (18)
  Update state x_k from Equ. (19)
  Update $P_{k|k}$ from Equ. (20)
end

The remaining state estimation can be calculated as:

$$L_k = \frac{G_\sigma(||y_k - H * \hat{x}_k||_{R_k^{-1}})}{G_\sigma(||\hat{x}_k - F * \hat{x}_{k-1}||_{P_{k|k-1}^{-1}})} \quad (17)$$

$$K_k = (P_{k|k-1}^{-1} + L_k * H^T * R_k^{-1} * H)^{-1} * L_K * H^T * R_K^{-1} \quad (18)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - H * \hat{x}_k^-) \quad (19)$$

$$P_{k|k} = (I - K_k * H) * P_{k|k-1} * (I - K_k * H)^T + K_k * R_k * K_k^T. \quad (20)$$

Evaluation setup, and experiments

- The calculated odometry from the two SLAM methods is fed to the MCC-EKF (Maximum Correntropy Criterion- Extended Kalman Filter) framework.
- Shot noises and attack vectors are introduced into the system to see the response of the framework.
- The figure shows the set-up of the framework

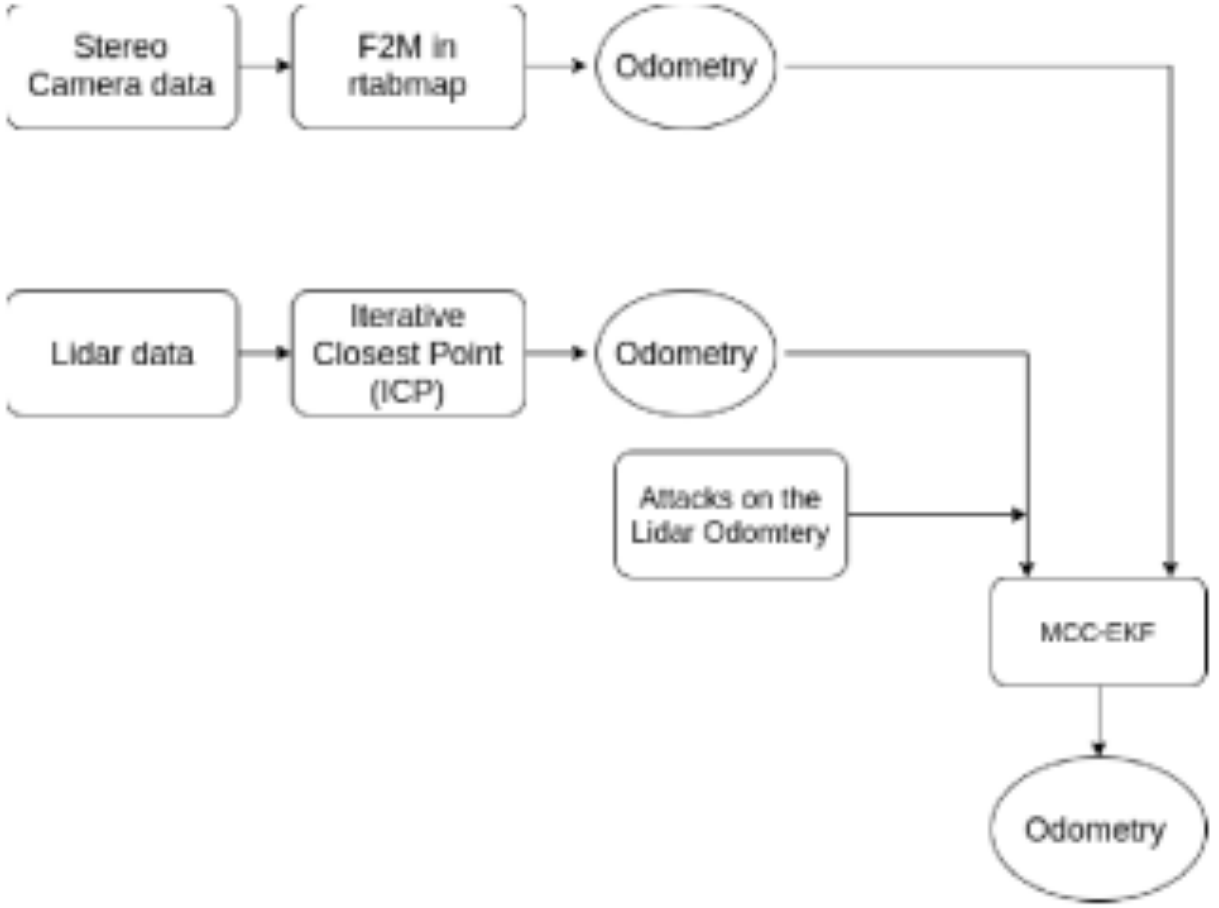
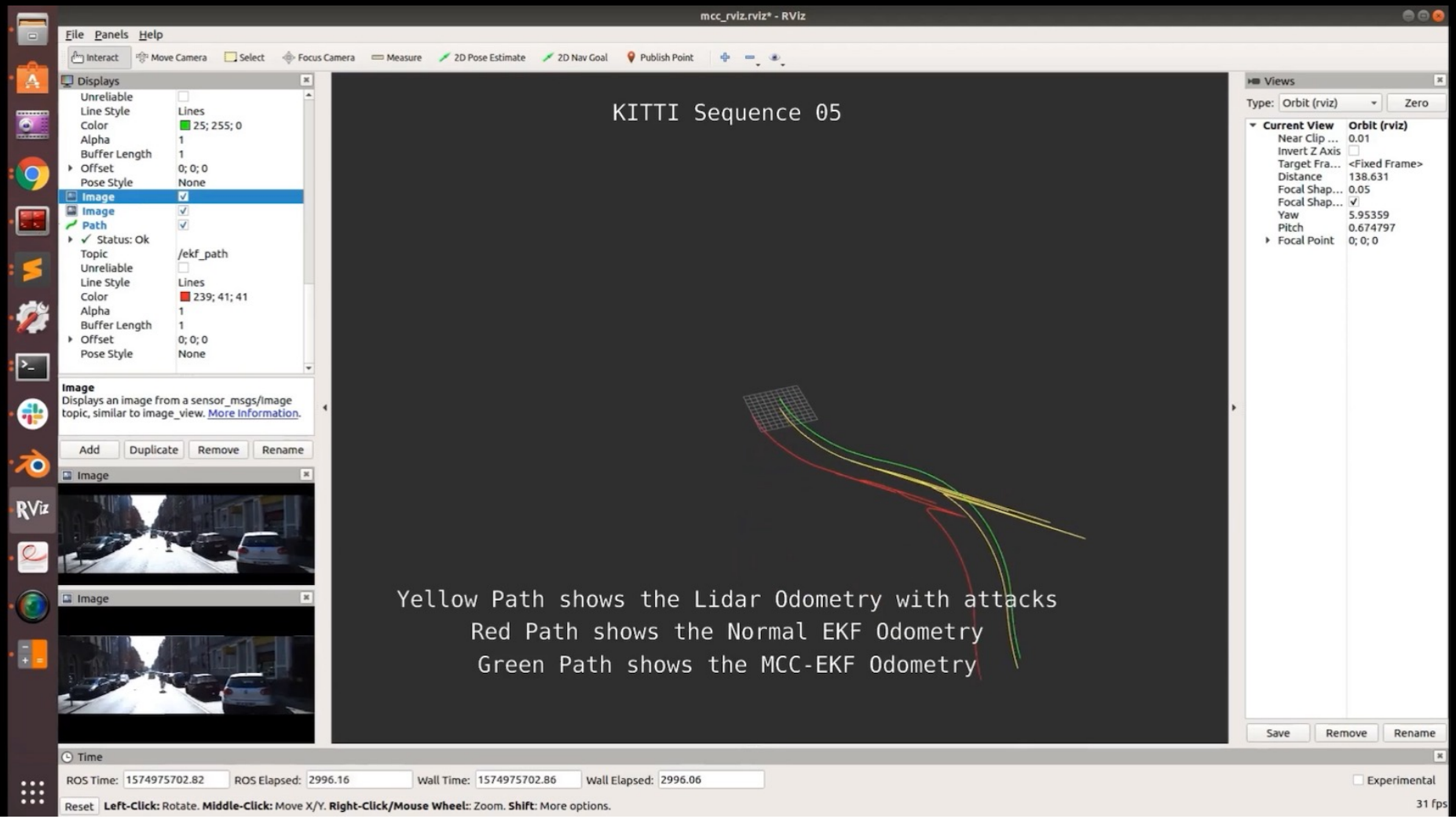


Fig. 1: System Architecture.

Live Example



Results 1

- The approach is assessed on two systems
  - Simulated gazebo system
  - KITTI dataset
- Gazebo environment uses Prius car model with lidar and camera system
- The result for the lidar and stereo odometry is shown in the figure to the right.

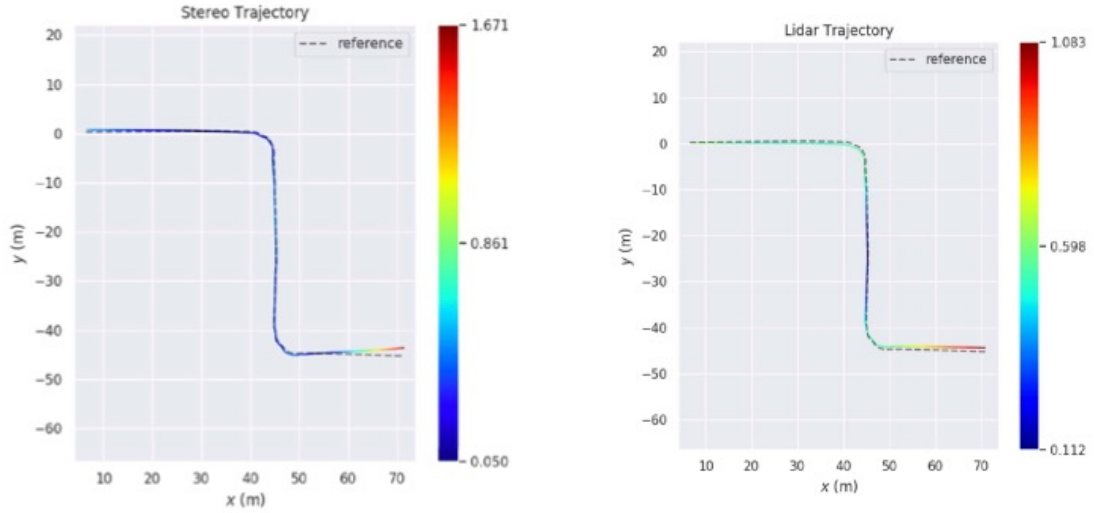


TABLE I: RMSE comparison

	Traj 1	Traj 2
Stereo (RMSE)1	0.478404	1.840472
Lidar (RMSE)	0.557076	0.505731
MCC-EKF (RMSE)	0.419823	0.49761

Results 2

- Root Mean Square Error (RMSE) values of individual trajectories with ground truth are compared.
- The MCC-EKF (a combination of both) performs better than the individual SLAM algorithms.
- MCC-EKF also handles injections of false odometry values better
- MCC-EKF is not affected when the attacks are introduced.

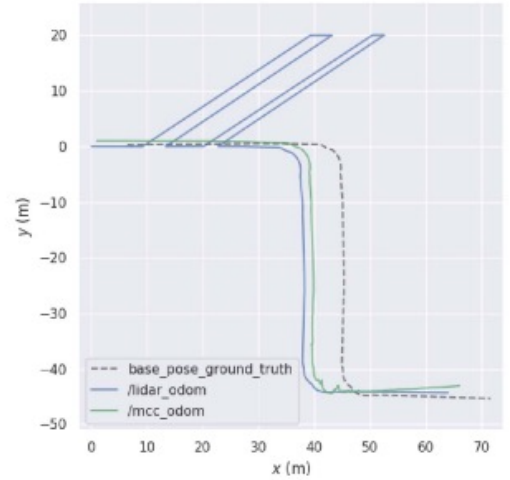


Fig. 7: MCC-EKF response to attacks on Lidar data.

TABLE II: RMSE comparison after attack vectors

	Traj 1	Traj 2
Random attacks 1		
Normal EKF (RMSE)	3.624596	5.01472
MCC-EKF (RMSE)	0.420437	0.81037
Random attacks 2		
Normal EKF (RMSE)	4.624596	7.01472
MCC-EKF (RMSE)	0.43317	0.85132

Conclusion

- Utilize the MCC-EKF approach
- Results in a better odometry estimate as it uses two different SLAM algorithms.
- It is also not vulnerable to false value injections therefore its trajectory cannot be changed.
- The experiment is novel and aims to solve a genuine autonomous car problem.
- The proposed solution is logical and applicable in the current autonomous car industry.
- The results are well-represented in figures and graphs making them easy to understand.

Discussion Points

What are the examples of SLAM algorithms?

What are the results of the study?

What is the importance of improving the security of autonomous cars?

THANK YOU

Reference

- Singandhupe, A., & La, H.M. (2020). MCC-EKF for Autonomous Car Security. *2020 Fourth IEEE International Conference on Robotic Computing (IRC)*, pages 306-313.