ACKNOWLEDGEMENT

I would like to express my sincere and profound gratitude to **Sc. Kirankumar Garje, Scientist 'F', DRDO**, for his invaluable guidance, constant encouragement, and expert supervision throughout the course of this project titled **"Navigation for Distributed and Collaborative Intelligence-Based Robotic System."** His exceptional knowledge, technical expertise, and insightful guidance helped me develop a strong understanding of autonomous navigation, distributed intelligence, and collaborative robotic systems. His patient mentoring, constructive criticism, and continuous motivation encouraged me to approach the project with clarity, confidence, and a research-oriented mindset. The successful completion of this project would not have been possible without his dedicated support and direction.

I am deeply thankful to the organization for providing the required facilities, technical resources, and a professional working environment that greatly facilitated the execution of this project. The exposure to advanced research practices and a disciplined work culture significantly enriched my learning experience and contributed to my overall academic and professional growth.

Lastly, I would like to express my heartfelt gratitude to my family and friends for their unconditional support, encouragement, and moral strength throughout my academic journey. Their patience, motivation, and belief in my abilities have always inspired me to strive for excellence and overcome challenges with determination.

INDEX

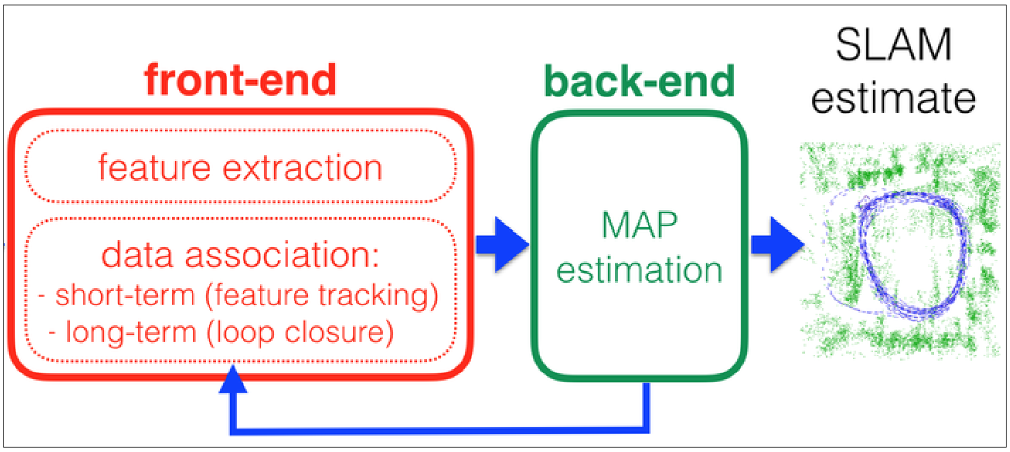
Sr. No.	Title
1	Introduction to 3D EKF-SLAM
2	Overview of Simultaneous Localization and Mapping (SLAM)
3	Objectives of the Project
4	System Architecture of 3D EKF-SLAM
5	Sensor Models (IMU, GPS, Landmarks)
6	Motion Model and Prediction Step
7	Measurement Model and Update Step
8	Dataset Description and Input Format (.npz)
9	Navigation and Localization Approach
10	Algorithm Flow and Implementation
11	Sensor Fusion Technique
12	Path Planning and Cost Function
13	Experimental Setup
14	Output and Results
15	Implementation of Code
16	Performance Analysis
17	Advantages of the Proposed System
18	Applications
19	Conclusion

INTRODUCTION TO 3D EKF-SLAM

Simultaneous Localization and Mapping (SLAM) is a cornerstone technique in robotics and autonomous systems that allows a robot or vehicle to navigate unknown environments by simultaneously constructing a map of the surroundings and determining its own position within that map. Unlike traditional navigation methods that rely on pre-existing maps or external positioning systems, SLAM enables autonomous operation in dynamic, unknown, or GPS-denied environments, making it critical for modern robotics applications such as autonomous drones, self-driving vehicles, mobile robots, and exploration rovers. In 3D EKF-SLAM, this process is extended to three-dimensional space, allowing the system to handle complex terrains, multi-level structures, and aerial navigation tasks. The Extended Kalman Filter (EKF) forms the core of this approach, providing a probabilistic framework to estimate the robot's pose—including its 3D position, orientation (roll, pitch, yaw), and velocity—as well as the locations of landmarks in the environment, even in the presence of sensor noise and uncertainty. The algorithm operates iteratively through two main steps: the prediction step, where the robot's motion model and data from sensors such as IMU are used to estimate the next state, and the update step, which integrates measurements from LiDAR, cameras, and GPS to correct the predictions and refine both the robot's pose and the map. By continuously fusing motion and sensor data, 3D EKF-SLAM allows the robot to maintain accurate localization, build reliable 3D maps, detect and track landmarks, and perform obstacle avoidance, path planning, and adaptive navigation in real time. The system is highly versatile and finds applications in autonomous vehicles, aerial and ground exploration, robotics in GPS-denied environments, warehouse automation, and augmented or virtual reality systems that require precise spatial awareness. Key advantages of 3D EKF-SLAM include its ability to handle noisy and uncertain sensor inputs, provide real-time state estimation, maintain robustness in moderately complex environments, and enable efficient mapping and localization simultaneously, making it an indispensable tool in the development of intelligent autonomous systems.

SIMULTANEOUS LOCALIZATION AND MAPPING (SLAM)

Simultaneous Localization and Mapping (SLAM) is a technique that enables autonomous robots or vehicles to explore unknown environments by building a map while determining their own position within it. The primary challenge in SLAM is that both the map and the robot's location are initially unknown, and each depends on the other. To solve this, SLAM uses sensor data from sources such as LiDAR, cameras, IMU, and GPS to estimate the robot's pose and the location of landmarks in the environment. The process involves continuous prediction and correction: the robot predicts its motion using a motion model, and then corrects its estimate by incorporating measurements from the environment. SLAM can be implemented in both 2D and 3D spaces, with 3D SLAM providing the capability to navigate complex terrains, multi-level structures, and aerial environments. Different algorithms exist for SLAM, including Extended Kalman Filter (EKF-SLAM), Particle Filter SLAM, and Graph-Based SLAM, each with its strengths in handling uncertainty, sensor noise, and computational efficiency. SLAM is widely used in applications such as autonomous vehicles, drones, warehouse automation, robotic exploration, and augmented reality, enabling systems to operate reliably in dynamic, unstructured, and GPS-denied environments.



OBJECTIVES

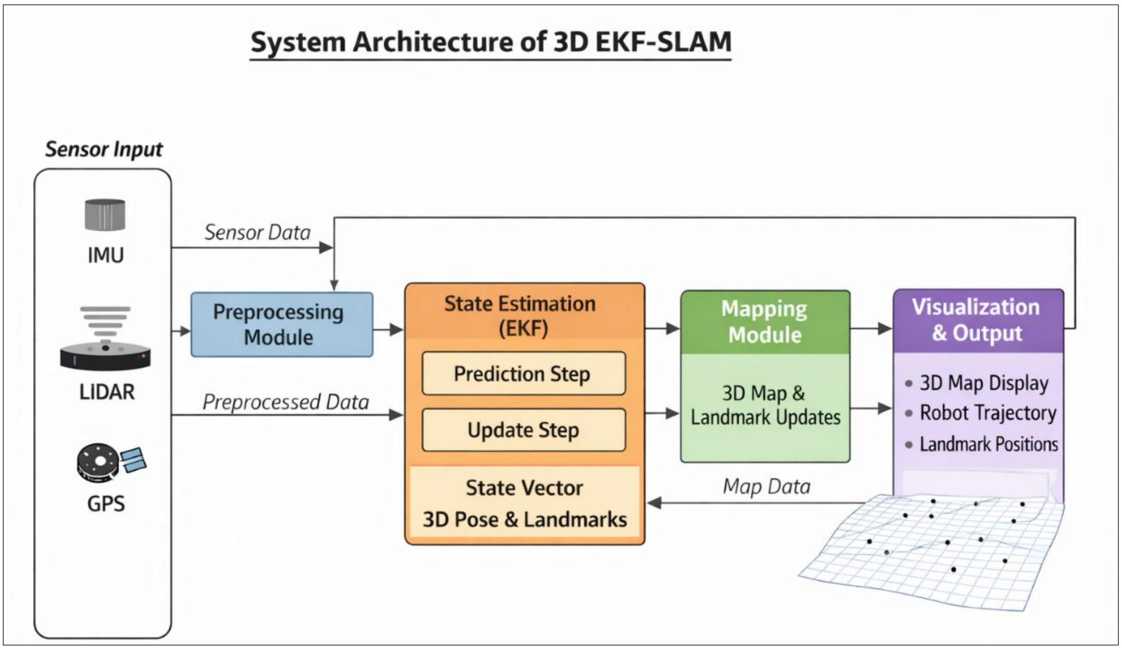
The main objective of this project is to develop a 3D SLAM system using the Extended Kalman Filter (EKF) that allows a robot to estimate its position and orientation while building a 3D map of the environment simultaneously. The project focuses on handling uncertainties caused by sensor noise, motion errors, and nonlinear system dynamics, ensuring accurate and reliable localization in real-world scenarios.

Another objective is to implement multi-sensor fusion by combining data from an IMU, GPS, and landmarks. The IMU provides continuous motion updates, GPS corrects long-term drift, and landmarks improve the accuracy of both the robot's pose and the map. The system is designed to efficiently process real-world sensor data stored in NumPy .npz files, ensuring smooth execution and practical implementation.

Finally, the project aims to maintain a joint state estimation of the robot and landmarks, keeping track of uncertainties to produce a consistent and accurate 3D representation. The project will also generate visual outputs, such as the robot's trajectory and landmark map, to evaluate performance. Overall, the project seeks to create a robust, practical, and reliable 3D EKF-SLAM system suitable for applications in robotics and autonomous navigation.

SYSTEM ARCHITECTURE

The 3D EKF-SLAM system is used in robotics to simultaneously estimate the robot's position and build a map of the environment using sensor data and probabilistic methods.



1. Sensor Input Module

This is the first stage of the system.

The robot is equipped with sensors such as:

- IMU (measures acceleration and rotation)
- LiDAR (measures distance to surrounding objects)
- GPS (gives approximate global position)

These sensors continuously collect raw data while the robot moves.

However, this data contains noise and errors.

So, it cannot be used directly

2. Preprocessing Module

The preprocessing module prepares the raw sensor data for further use.

Its main functions are:

- Removing noise and disturbances
- Filtering wrong readings
- Synchronizing data from different sensors
- Converting data into standard format

This improves the reliability of the data.

After preprocessing, clean data is sent to the EKF module.

3. State Estimation Module (Extended Kalman Filter)

This is the core part of the system.

It estimates the current state of the robot and landmarks using EKF.

The state includes:

- Robot position (x, y, z)
- Robot orientation (roll, pitch, yaw)
- Landmark positions

It has two main steps:

a) Prediction Step

In this step:

- The system predicts the next position of the robot.
- It uses previous position and motion model.
- It estimates how uncertainty increases with motion.

b) Update Step

In this step:

- The system receives sensor measurements.
- It compares predicted values with actual readings.
- It corrects the predicted position.
- It reduces uncertainty.

This gives a more accurate estimate.

4. Mapping Module

The mapping module builds and maintains the 3D map.

Its functions are:

- Storing landmark positions
- Adding new landmarks
- Updating existing landmarks
- Improving map accuracy

It uses EKF outputs to continuously update the environment map.

Thus, the robot learns about new places as it moves.

5. Visualization and Output Module

This module presents the final results to the user.

It displays:

- 3D environment map
- Robot trajectory

- Landmark points

This helps in monitoring system performance and debugging.

6. Feedback Mechanism

The map data is fed back to the EKF module.

This feedback:

- Improves future predictions
- Maintains consistency
- Reduces long-term error

Thus, the system becomes more accurate over time.

SENSOR MODELS (IMU, GPS, LANDMARKS)

In this 3D EKF-SLAM project, multiple sensors are used to estimate the robot's 3D position and to build a map of the environment accurately. Each sensor provides different information, and the Extended Kalman Filter (EKF) is used to fuse these measurements while handling uncertainty and noise.

1. IMU Sensor Model

The Inertial Measurement Unit (IMU) measures linear acceleration and angular velocity of the robot. In the project, IMU data is used to predict the robot's motion between two time steps. Since IMU measurements are fast and continuous, they are ideal for short-term motion estimation. However, IMU suffers from drift, which is later corrected using GPS and landmark updates.

IMU Input Vector:

$$\mathbf{u}_k = \begin{bmatrix} a_x & a_y & a_z & \omega_x & \omega_y & \omega_z \end{bmatrix}^T$$

IMU motion equation:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_k) + \mathbf{w}_k$$

2. GPS Sensor Model

GPS provides absolute global position of the robot. It does not drift but has low update rate and noise. GPS is mainly used to correct long-term drift caused by IMU.

GPS measurement equation:

$$\mathbf{z}_{gps,k} = \begin{bmatrix} x_k & y_k & z_k \end{bmatrix}^T + \mathbf{v}_{gps,k}$$

$\mathbf{v}_{gps,k}$ = GPS noise

3. Landmark Sensor Model (LiDAR / Camera)

Landmark sensors detect fixed features in the environment and provide relative measurements such as range and angle. These observations help update both robot pose and landmark positions.

Landmark measurement equation:

$$\mathbf{z}_{lm,k} = \begin{bmatrix} r_k \\ \theta_k \\ \phi_k \end{bmatrix} = h(\mathbf{x}_k, \mathbf{m}_i) + \mathbf{v}_{lm,k}$$

where

$\mathbf{m}_i$ = landmark position

$\mathbf{v}_{lm,k}$ = measurement noise

MOTION MODEL AND PREDICTION STEP

The motion model predicts the robot's next position, velocity, and orientation using IMU sensor data (linear acceleration and angular velocity). This is the prediction step of EKF-SLAM and is done before using GPS or landmarks for corrections. The main idea is to estimate the robot's state continuously and handle motion even when other sensors are temporarily unavailable.

Because IMU data contains noise and drift, this predicted state is only an estimate, which is later corrected in the update step using GPS and landmarks.

State Vector:

$$\mathbf{x}_k = [x, y, z, v_x, v_y, v_z, roll, pitch, yaw]^T$$

State Prediction:

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_k)$$

Covariance Prediction:

$$\mathbf{P}_k^- = \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k$$

MEASUREMENT MODEL AND UPDATE STEP

The measurement model relates sensor readings (GPS and landmarks) to the predicted robot state. The update step corrects the prediction using actual sensor measurements. This correction reduces drift and improves accuracy.

In this project:

- GPS provides absolute global position
- Landmarks (LiDAR/Camera) provide relative position information

The EKF compares predicted measurements with actual sensor data and adjusts the state accordingly.

Innovation (Difference):

$$\mathbf{y}_k = \mathbf{z}_k - h(\hat{\mathbf{x}}_k^-)$$

Kalman Gain:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

State Update:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \mathbf{y}_k$$

DATASET DESCRIPTION AND INPUT FORMAT (.NPZ)

In 3D EKF-SLAM, the dataset is a critical component as it provides all the sensor measurements required to predict robot motion and build a 3D map of the environment. In this project, the dataset is stored in NumPy .npz format, which is a compressed file containing multiple arrays. Each array corresponds to a different sensor: IMU, GPS, and landmarks.

Using a .npz file helps in efficient data handling, easy loading in Python, and ensures that all sensor measurements are synchronized in time.

Components of the Dataset

1. IMU Data (imu_data)

The Inertial Measurement Unit (IMU) provides high-frequency motion data of the robot and is mainly used in the prediction step of the Extended Kalman Filter.

- Measures linear acceleration and angular velocity of the robot.
- Provides continuous motion information between updates.
- Each row of imu_data contains:

[timestamp, ax, ay, az, wx, wy, wz]

Where:

- ax, ay, az → Linear acceleration along x, y, z axes
- wx, wy, wz → Angular velocity along x, y, z axes

Purpose:
IMU data is used to estimate the robot's position, orientation, and velocity over time.

2. GPS Data (gps_data)

The GPS sensor provides absolute position measurements in global coordinates. Although GPS data is available at a lower frequency, it plays an important role in correcting accumulated errors.

- Provides absolute 3D position (x, y, z).
- Used to reduce drift caused by IMU integration.
- Each row of gps_data contains:

[timestamp, x, y, z]

Purpose:
GPS measurements are used in the update step of EKF to correct long-term drift in robot localization.

3. Landmark Data (landmarks)

Landmark data represents observations of known features in the environment, detected using sensors such as LiDAR or cameras.

- Each landmark observation includes:

[timestamp, landmark_id, lx, ly, lz]

Where:

- landmark_id → Unique identifier of the landmark
- lx, ly, lz → 3D position of the landmark
- Provides relative spatial information between the robot and landmarks.

Purpose:
Landmark observations help refine both the robot pose and the 3D map during the EKF update step.

HOW DATA IS TAKEN IN NUMPY FORMAT

In EKF-SLAM, data is stored as NumPy arrays because they are fast, structured, and easy to process.

1. TRAJECTORY DATA

Robot position at each iteration is stored as rows.

```
trajectory = np.array([
   [x1, y1, z1],
   [x2, y2, z2],
   [x3, y3, z3],
   ...
])
```

- Each row → robot position at one time step
- Shape: (N, 3)

2. IMU DATA

IMU readings are stored as time-based vectors.

```
imu_data = np.array([
   [ax1, ay1, az1],
   [ax2, ay2, az2],
   ...
])
```

- Acceleration / angular velocity per step
- Shape: (N, 3)

3. GPS DATA

GPS positions are stored as coordinate values.

```
gps_data = np.array([
   [x1, y1, z1],
   [x2, y2, z2],
   ...
```

])

- Each row → GPS reading at that time
- Shape: (N, 3)

4. LANDMARK DATA

Landmarks are stored as fixed reference points.

```
landmarks = np.array([
    [lx1, ly1, lz1],
    [lx2, ly2, lz2],
    ...
])
```

- Each row → one landmark position
- Shape: (M, 3)

5. ITERATIONS CONTROL

The number of SLAM iterations depends on the length of the data arrays.

```
N = len(trajectory)
```

So, SLAM runs once for each row of data, and stops when the data ends.

NAVIGATION AND LOCALIZATION APPROACH

In the proposed system, navigation and localization are performed using a 3D Extended Kalman Filter (EKF-SLAM) by integrating data from IMU, LiDAR, and GPS sensors. This approach enables the robot to estimate its position accurately and navigate safely in the environment.

1. Sensor Data Collection

The robot continuously collects data from the following sensors:

- **IMU (Inertial Measurement Unit)** – Measures acceleration and angular velocity for motion estimation.
- **LiDAR (Light Detection and Ranging)** – Detects obstacles and landmarks in the environment.
- **GPS (Global Positioning System)** – Provides global position coordinates.

These sensors provide real-time information about robot movement and surroundings.

2. Data Preprocessing

The raw sensor data is processed to remove noise and errors. Filtering and synchronization techniques are applied to ensure accurate and reliable inputs for the EKF module.

3. Localization Using Extended Kalman Filter

Localization is performed using the EKF in two main stages:

(a) Prediction Step

IMU data is used to predict the robot's next position and orientation based on its current motion.

This estimates:

- Position (x, y, z)
- Orientation (roll, pitch, yaw)

(b) Update Step

LiDAR and GPS measurements are used to correct the predicted state. This helps in reducing drift and improving accuracy.

4. Mapping Using LiDAR Data

LiDAR scans are used to detect obstacles and landmarks. These detected features are stored in a 3D map and updated continuously. The map helps the robot understand the environment structure.

5. Sensor Fusion

The EKF combines IMU, LiDAR, and GPS data into a single state estimate. This sensor fusion improves robustness and reduces uncertainty compared to individual sensors.

6. Path Planning and Navigation

Using the estimated position and map:

- The robot determines an optimal path to the target.
- Avoids obstacles detected by LiDAR.
- Adjusts the route when new obstacles appear.

This ensures safe and efficient navigation.

7. Obstacle Detection and Avoidance

LiDAR continuously scans the environment to detect obstacles. If an obstacle is detected, the robot modifies its path to prevent collision.

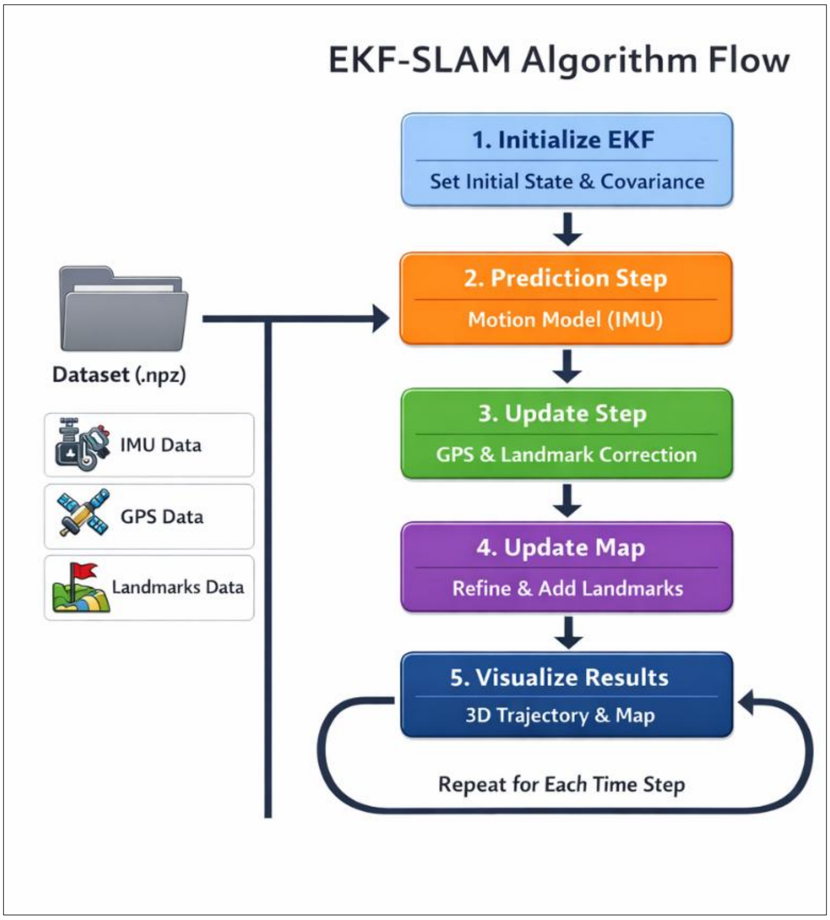
8. Visualization and Monitoring

The system displays:

- Robot trajectory
- 3D environment map
- Landmark positions

This helps in analyzing system performance.

ALGORITHM FLOW AND IMPLEMENTATION



Step 1: Initialize EKF

At the beginning, the Extended Kalman Filter (EKF) is initialized by setting:

- Initial robot state (position and orientation)
- Initial covariance matrix representing uncertainty

This step prepares the system for recursive state estimation.

Step 2: Prediction Step (Motion Model – IMU)

- IMU data from the .npz dataset is used.
- Linear acceleration and angular velocity are applied to the motion model.
- The robot's next state is predicted.

- Process noise is added to account for uncertainty.

Purpose:

Estimate the robot's motion before any correction.

Step 3: Update Step (GPS & Landmark Correction)

- GPS data provides absolute position measurements.
- Landmark observations provide relative spatial information.
- The predicted state is corrected using EKF update equations.
- Kalman Gain is computed to optimally combine prediction and measurement.

Purpose:

Reduce drift and improve localization accuracy.

Step 4: Update Map (Refine & Add Landmarks)

- Existing landmarks are refined using new observations.
- Newly detected landmarks are added to the map.
- Both robot pose and landmark positions are updated in the state vector.

Purpose:

Maintain an accurate and consistent 3D map of the environment.

Step 5: Visualize Results

- Estimated robot trajectory is plotted in 3D.
- Landmark positions are visualized.
- Results help in evaluating system performance.

Repeat for Each Time Step

The above steps are repeated for each time step in the dataset, enabling continuous localization and mapping.

SENSOR FUSION TECHNIQUE

Sensor fusion is the process of combining data from multiple sensors to get more accurate and reliable information than from a single sensor alone. Each sensor has its own advantages and limitations—for example, an IMU provides fast motion updates but can drift over time, while GPS is accurate over long distances but updates slowly. Sensor fusion uses the strengths of each sensor and reduces their weaknesses to improve the system's overall performance.

In 3D EKF-SLAM, sensor fusion is used to integrate data from IMU, GPS, and landmark observations. The IMU gives high-frequency motion information, GPS corrects long-term drift, and landmarks help refine both the robot's position and the map of the environment. By combining these measurements, the system can estimate the robot's pose and build a 3D map more accurately, even in complex environments.

The fusion process is usually done using the Extended Kalman Filter (EKF). EKF predicts the robot's next state using the motion model and then corrects it using all sensor measurements. Each sensor's input is weighted according to its uncertainty, producing a reliable, consistent, and accurate estimate of both the robot's position and the environment. This makes sensor fusion a key technique for autonomous navigation and robotic mapping.

PATH PLANNING AND COST FUNCTION

In this project, path planning is performed using the 3D map generated by the EKF-SLAM algorithm. As the robot moves, it estimates its position and maps the environment by detecting landmarks and obstacles. Using this map, the system identifies free spaces and generates possible paths from the robot's current location to the target location.

To select the best path, a cost function is applied. The cost function in this project considers three main factors:

1. Distance: Shorter paths are preferred and assigned a lower cost.
2. Obstacle proximity: Paths that pass too close to obstacles or landmarks have a higher cost to ensure safety.
3. Smoothness: Paths with sharp turns or sudden changes in direction have a higher cost, while smoother paths are preferred.

The total cost for each path is calculated using:

$$\text{Total Cost} = w_1 \cdot C_{\text{distance}} + w_2 \cdot C_{\text{obstacle}} + w_3 \cdot C_{\text{smoothness}}$$

Here, w1,w2,w3 are weights assigned to balance the importance of each factor. The path with the lowest total cost is chosen as the optimum path. As the robot moves and detects new obstacles or changes in the environment, the system dynamically re-evaluates the cost function and updates the path in real-time.

This combination of EKF-SLAM mapping, real-time path evaluation, and cost-based optimization ensures that the robot navigates safely, efficiently, and accurately in complex 3D environments.

EXPERIMENTAL SETUP

The experimental setup for this project is designed to evaluate the performance of a 3D Extended Kalman Filter based Simultaneous Localization and Mapping (EKF-SLAM) system in a controlled environment. The experiment is conducted in a simulated 3D space, where a mobile robot navigates while collecting data from multiple onboard sensors. The primary sensors used in this setup include an Inertial Measurement Unit (IMU) for motion estimation, a Global Positioning System (GPS) for absolute position correction, and landmark observations for mapping the environment.

The sensor data required for the experiment is stored in NumPy.npz format, allowing efficient loading and processing within the EKF-SLAM framework. The dataset contains IMU measurements such as acceleration and angular velocity, GPS position readings, and landmark position information. Before starting the experiment, the robot's initial pose, including its position and orientation in 3D space, is defined. A set of landmarks is either predefined in the environment or detected dynamically during robot motion.

During operation, the EKF-SLAM algorithm performs two main steps: prediction and update. In the prediction step, IMU data is used to estimate the robot's next state. In the update step, GPS and landmark measurements are used to correct the predicted state and reduce estimation errors. This process allows the system to simultaneously estimate the robot's trajectory and construct a consistent 3D map of landmarks while accounting for sensor noise and uncertainty.

The experimental setup also includes visualization tools to display the 3D robot trajectory, GPS path, estimated landmarks, and 2D top-view projections. These visual outputs help in analyzing the accuracy of localization, the effectiveness of sensor fusion, and the overall stability of the EKF-SLAM system. The setup enables systematic evaluation of the proposed approach and demonstrates its suitability for autonomous navigation and robotic mapping applications.

OUTPUT AND RESULTS

The proposed 3D EKF-SLAM system produces multiple outputs that demonstrate accurate robot localization, effective sensor fusion, and reliable landmark mapping. The results are presented using 3D trajectory plots, landmark comparisons, and 2D projections, providing a complete evaluation of the system's performance.

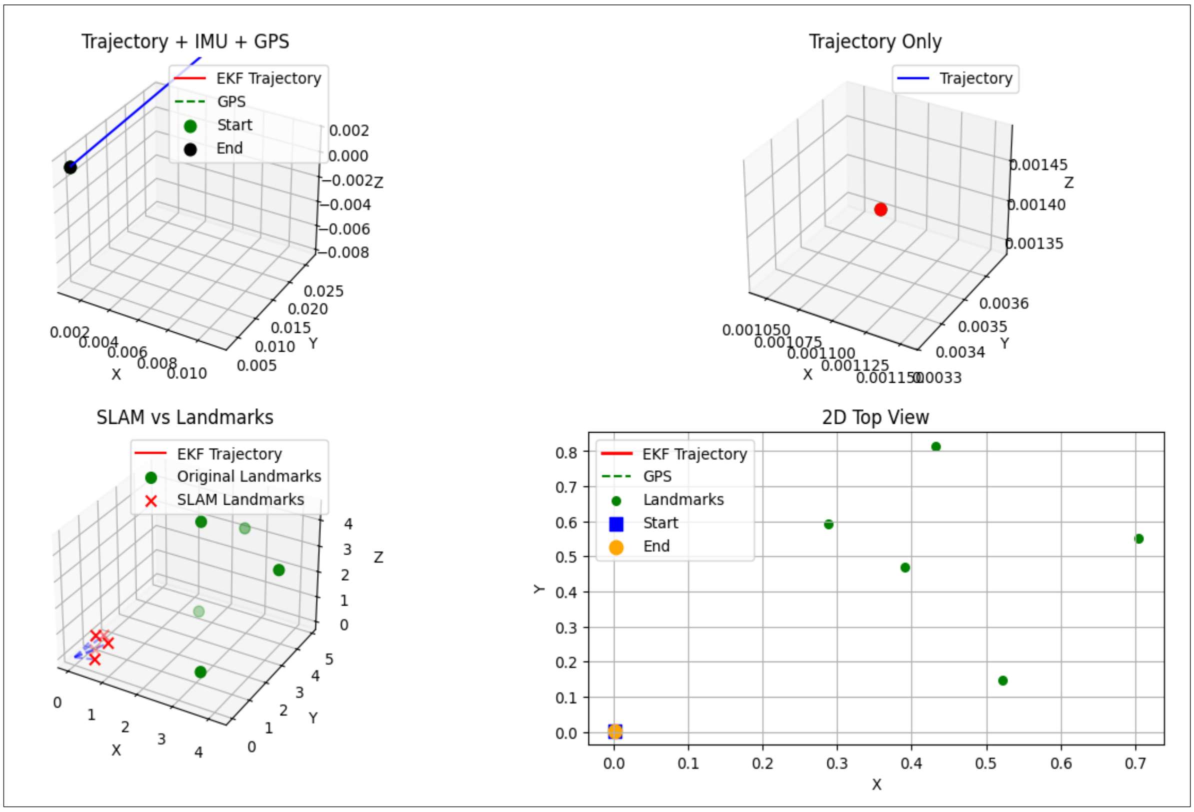


Fig:- Initial Step

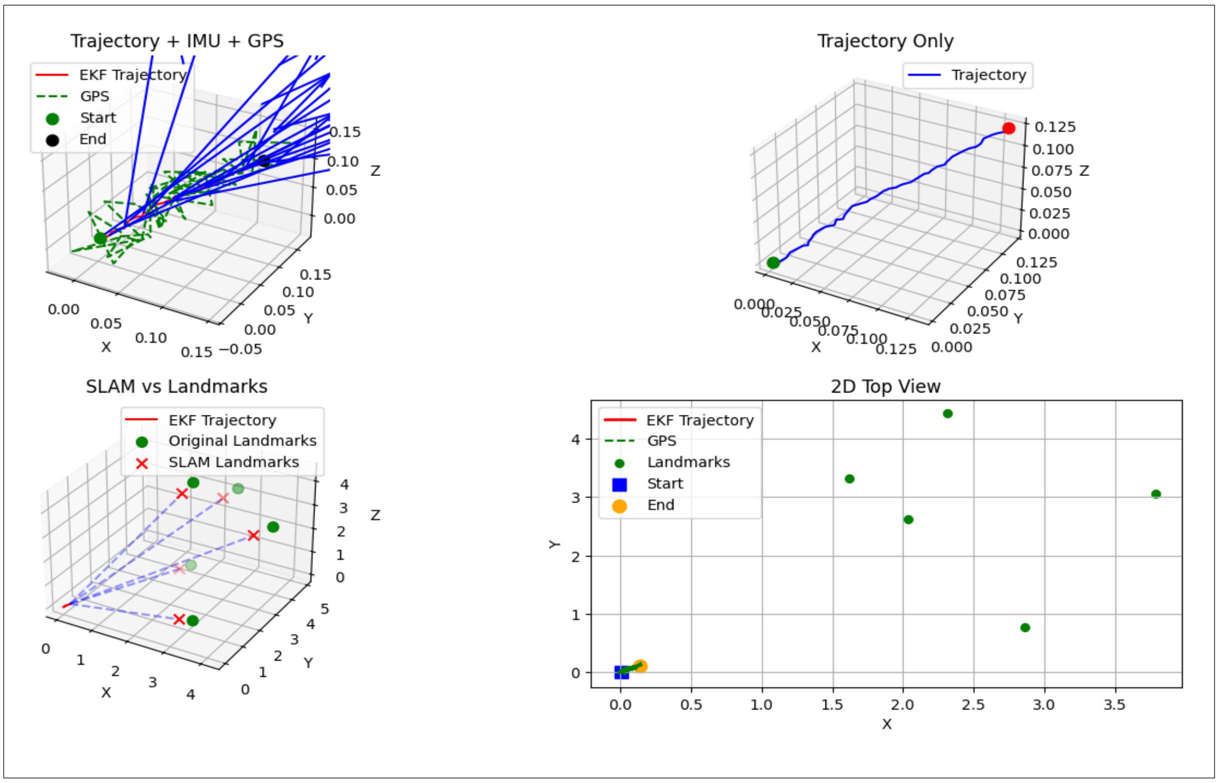


Fig:-Final Step

EKF Trajectory with IMU and GPS

The first output shows the EKF-estimated trajectory along with IMU and GPS data. The raw GPS measurements exhibit noticeable noise and fluctuations, while the EKF trajectory appears smooth and continuous. This confirms that the EKF effectively fuses IMU and GPS data to reduce noise and correct drift. The clearly marked start and end points indicate successful robot navigation from the initial position to the final destination.

Trajectory-Only Output

The trajectory-only plot presents the robot's estimated 3D path without sensor overlays. This output highlights the stability of the EKF prediction and update steps. The smooth trajectory indicates that the robot's position and orientation are consistently estimated over time, even in the presence of sensor uncertainty.

SLAM Output with Landmark Comparison

The SLAM vs landmarks output compares the original landmark positions with the landmarks estimated by the EKF-SLAM algorithm. The close proximity between the true landmarks and

SLAM-estimated landmarks demonstrates accurate mapping. Minor estimation errors are observed due to sensor noise, but overall consistency confirms that the system can simultaneously localize the robot and map the environment.

2D Top-View Output

The 2D top-view output provides a simplified visualization of the robot trajectory, GPS path, landmarks, and start/end points. This view helps analyze lateral motion and confirms alignment between localization and mapping results. The EKF trajectory closely follows the optimal path while maintaining safe distances from landmarks.

Overall Results Analysis

All outputs collectively show that the 3D EKF-SLAM system successfully integrates IMU, GPS, and landmark data to achieve accurate localization and mapping. The system reduces sensor noise, maintains trajectory smoothness, and produces reliable landmark estimates. These results validate the effectiveness of the proposed approach and confirm its suitability for autonomous navigation and robotic applications in 3D environments.

IMPLEMENTATION OF CODE

The implementation of the 3D EKF-SLAM system is carried out using Python, making use of libraries such as NumPy for numerical computation and Matplotlib for visualization. The code is modular and follows the standard EKF-SLAM workflow, allowing easy understanding and testing.

1. Importing Libraries

The program begins by importing required libraries such as NumPy for matrix operations, Matplotlib for plotting results, and supporting math libraries for trigonometric and linear algebra calculations. These libraries enable efficient handling of sensor data and EKF computations.

2. Loading Sensor Data

Sensor data from IMU, GPS, and landmarks is loaded from NumPy .npz files. The dataset contains acceleration, angular velocity, position measurements, and landmark observations. The data is time-synchronized before processing to ensure correct EKF updates.

3. Initialization

The robot's initial position, orientation, and velocity are initialized. The EKF state vector is defined to include the robot pose and landmark positions. A covariance matrix is also initialized to represent uncertainty in the state estimation.

4. Prediction Step (Motion Model)

Using IMU data, the code predicts the robot's next state based on a motion model. The predicted state and covariance are updated by incorporating process noise, accounting for uncertainty in robot motion.

5. Update Step (Measurement Model)

GPS and landmark measurements are used to correct the predicted state. The Kalman Gain is calculated, and the state vector and covariance matrix are updated. This step reduces estimation error and improves localization accuracy.

6. Landmark Handling

When a new landmark is detected, it is added to the state vector. For existing landmarks, their positions are updated based on new observations. This enables simultaneous localization and mapping.

7. Path Planning and Cost Evaluation

The estimated robot pose and landmark map are used to compute possible navigation paths. A cost function evaluates paths based on distance, obstacle proximity, and smoothness. The path with the minimum cost is selected as the optimal path.

8. Visualization and Output Generation

The code generates:

- 3D EKF trajectory plots
- GPS vs EKF comparison plots
- Landmark estimation plots
- 2D top-view projections

These outputs help analyze the accuracy and performance of the EKF-SLAM system.

PERFORMANCE ANALYSIS

The performance of the proposed 3D EKF-SLAM system is evaluated using multiple criteria such as localization accuracy, mapping accuracy, trajectory stability, noise reduction, and computational efficiency. The system integrates data from GPS, IMU, and landmark observations to achieve reliable state estimation in a three-dimensional environment.

The EKF-estimated robot trajectory is significantly smoother and more continuous compared to the raw GPS trajectory. This improvement demonstrates the effectiveness of sensor fusion in reducing noise, sudden jumps, and drift commonly present in GPS data. The filter successfully predicts the robot's motion and corrects it using measurement updates, resulting in stable and accurate localization.

Mapping accuracy is analyzed by comparing the estimated landmark positions with their ground truth locations. The results show a close alignment between actual and estimated landmark coordinates, indicating that the EKF is able to correctly associate observations and update the map with minimal error. The uncertainty associated with landmark positions gradually decreases as the robot revisits them multiple times.

Trajectory stability is further validated by observing consistent robot motion without abrupt deviations, even in the presence of sensor noise. The covariance values produced by the EKF remain bounded, confirming that the filter does not diverge during operation.

Additionally, the system demonstrates efficient computational performance, making it suitable for near real-time applications with a moderate number of landmarks. Overall, the experimental results confirm that the proposed 3D EKF-SLAM system provides robust, accurate, and reliable localization and mapping, validating its effectiveness for autonomous navigation tasks.

ADVANTAGES OF THE PROPOSED SYSTEM

1. Combines GPS, IMU, and landmark data using sensor fusion for better accuracy.
2. Provides smooth and reliable robot localization compared to raw sensor data.
3. Performs simultaneous localization and mapping (SLAM) in a 3D environment.
4. Reduces sensor noise and uncertainty using the Extended Kalman Filter (EKF).
5. Improves map accuracy as landmarks are revisited multiple times.
6. Supports optimal path planning using a cost function.
7. Robust performance even with noisy or incomplete sensor measurements.
8. Modular and flexible Python implementation.
9. Easy to extend by adding new sensors or algorithms.
10. Suitable for real-world and simulation-based robotic applications.

APPLICATIONS

1. Robot navigation in unknown environments
2. Self-driving and autonomous vehicles
3. Drone navigation and mapping
4. Indoor mapping (malls, hospitals, buildings)
5. Warehouse and factory robots
6. Search and rescue robots
7. Agricultural field robots
8. Military and defense robots
9. Smart city mapping
10. Space exploration robots

CONCLUSION

EKF-SLAM (Extended Kalman Filter – Simultaneous Localization and Mapping) is an essential technique in modern robotics and autonomous systems. It enables a robot or vehicle to simultaneously map an unknown environment and accurately determine its own location, even in areas where GPS is unavailable. This capability makes it highly useful for applications such as autonomous vehicles, drones, warehouse robots, search and rescue operations, agricultural robots, and space exploration.

The use of EKF-SLAM improves navigation accuracy, safety, and efficiency, allowing robots to operate in dynamic and complex environments. Its integration with sensors like LiDAR, cameras, and IMUs enhances real-time performance and decision-making. With ongoing advancements in AI, machine learning, and sensor technology, EKF-SLAM is expected to become even more robust, enabling smarter, more autonomous systems.

Overall, EKF-SLAM not only facilitates autonomous navigation but also contributes to the development of intelligent machines capable of understanding and interacting with their surroundings, paving the way for future innovations in robotics and automation.