- ## Sensors Selection:

A variety of sensors are employed to capture data about the environment and the robot's movement:

1. **Lidar (Light Detection and Ranging):** Lidar sensors emit laser beams and measure the time taken for the reflections to return, constructing a detailed 3D point cloud of the surroundings. This information aids in mapping and obstacle avoidance.
2. **Camera:** Cameras capture images or video frames of the environment. Advanced computer vision algorithms analyze these images to identify visual features, which are then used for mapping and localization.
3. **IMU (Inertial Measurement Unit):** IMUs house accelerometers and gyroscopes that monitor the robot's acceleration and angular velocity. This data provides insights into the robot's orientation and movement.
4. **Wheel Encoders:** These sensors measure wheel rotations, allowing the robot to estimate its distance traveled. Wheel encoders are particularly useful for tracking movement in indoor environments.

- ## Algorithm Implementation:

1. **Extended Kalman Filter (EKF) SLAM:** This algorithm combines sensor data with the Extended Kalman Filter to simultaneously estimate the robot's position and create a map. It is commonly used with wheel encoders, IMU, and sometimes Lidar or camera data.
2. **Graph-Based SLAM:** Graph-based approaches model the environment as a graph, where nodes represent robot poses and landmarks, and edges represent measurements. This method optimizes the graph to find the most likely robot trajectory and map configuration.
3. **Particle Filter (Monte Carlo Localization):** Particle filters employ a set of particles to represent possible robot poses. As the robot moves and observes the environment, particles are updated based on sensor data, eventually converging to the robot's true position.
4. **Visual SLAM:** Focusing on camera data, visual SLAM tracks visual features in images to determine the robot's movement and position. It excels in environments with varying lighting conditions.
5. **Lidar-Based SLAM:** Algorithms like Iterative Closest Point (ICP) and Iterative Closest Feature (ICF) leverage Lidar data to match point clouds, estimating the robot's pose and map.