

# P-AgSLAM: In-Row and Under-Canopy SLAM for Agricultural Monitoring in Cornfields

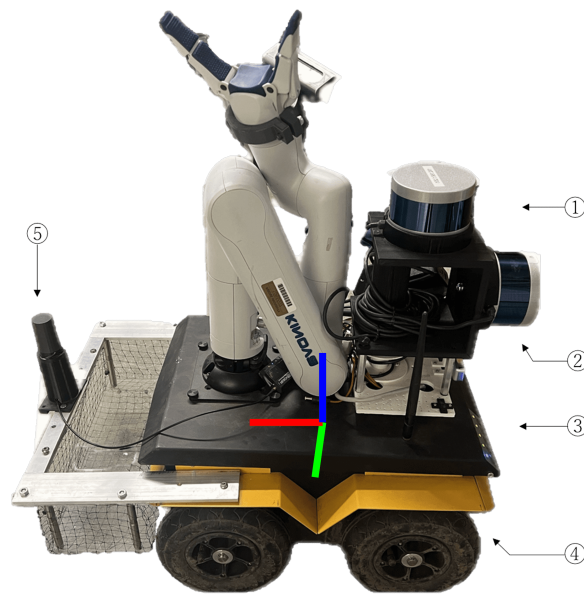
---

Purdue AgSLAM or P-AgSLAM is an in-row and under-canopy Simultaneous Localization and Mapping (SLAM) framework which is designed for robot pose estimation and agricultural monitoring in cornfields. Our SLAM approach is primarily based on a 3D light detection and ranging (LiDAR) sensor and it is designed for the extraction of unique morphological features of cornfields which have significantly different characteristics from structured indoor and outdoor urban environments. The performance of the proposed approach has been validated with experiments in simulation and in real cornfield environments. P-AgSLAM outperforms existing state-of-the-art LiDAR-based state estimators in robot pose estimations and mapping.

## System Overview

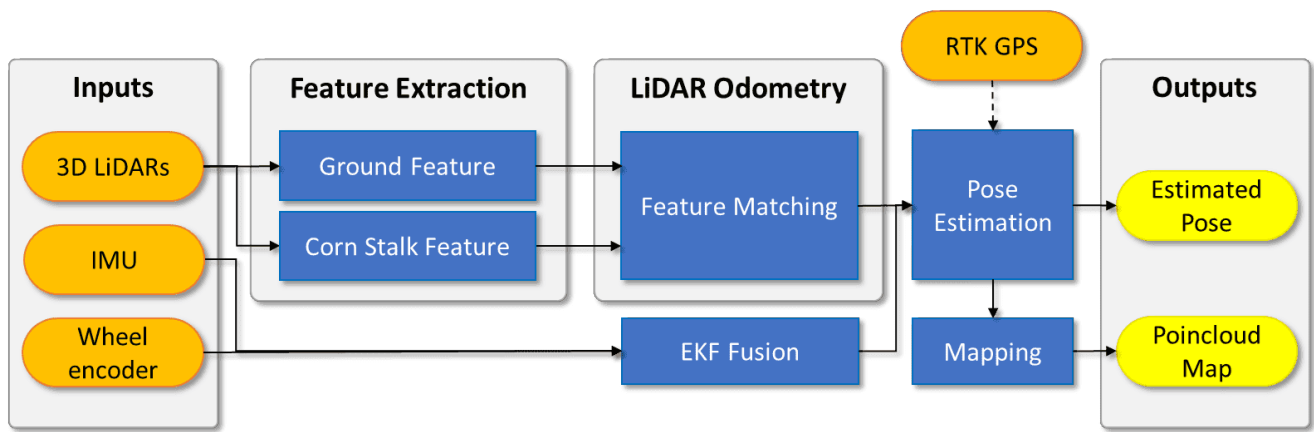
### Robot platform

P-AgBot is a customized platform with a variety of sensors and a robotic arm. Its physical size is 40 cm in width and 60 cm in length, including the front leaf storage bin. In this research, five sensors are used for P-AgSLAM: 1. Horizontal 3D LiDAR, 2. Vertical 3D LiDAR, 3. Internal IMU, 4. UGV wheel encoders, and 5. RTK GPS module. Each axis of the robot frame, represented by the colors red, green, and blue, corresponds to the X, Y, and Z translational and rotational axes, respectively.



### Framework

There are two primary modules in P-AgSLAM. (1) LiDAR-based feature extractor and (2) a robot state estimator using a factor graph with optional GPS measurements (Section IV). The two modules collaboratively minimize drift and publish accurate robot poses and maps.



## Dependency

- [ROS](#) (Tested with noetic.)
- [gtsam](#) (Georgia Tech Smoothing and Mapping library)

```
sudo add-apt-repository ppa:borglab/gtsam-release-4.0
sudo apt install libgtsam-dev libgtsam-unstable-dev
```

## Install

Use the following commands to download and compile the package.

```
cd ~/catkin_ws/src
git clone https://github.com/kimkt0408/pagslam.git
cd ..
catkin_make
```