530.707 Robot System Programming

3D Visual SLAM and Motion Planning using AR Drone

Project Proposal

Date: Apr. 3, 2018

Students: Tianyu Song, Huixiang Li, Wenhao Gu

Project Description:

This project focuses on a fusion of monocular vision and IMU to robustly track the position of an AR drone using LSD-SLAM (Large-Scale Direct Monocular SLAM) algorithm. The system consists of a low-cost commercial drone and a remote control unit to computationally afford the SLAM algorithms using a distributed node system based on ROS (Robot Operating System). Upon finishing this project, it is expected that we are able to reconstruct the 3D environment around AR drone and localize the drone.

Background:

Simultaneous Localization and Mapping (SLAM) for Unmanned Aerial Vehicles (UAVs) in the context of rescue and/or recognition navigation tasks in indoor environments has been a hot topic for several years.

LSD-SLAM is a novel, direct monocular SLAM technique developed by TUM: Instead of using keypoints, it directly operates on image intensities both for tracking and mapping. The camera is tracked using direct image alignment, while geometry is estimated in the form of semi-dense depth maps, obtained by filtering over many pixel-wise stereo comparisons.

Software:

Systems: Ubuntu 16.04, ROS Kinetic

Existing packages:

- lsd_slam

  TUM's LSD-SLAM package for ROS Indigo under Ubuntu 14.04. It is used to generating the 3D visual SLAM point cloud based on either stereo or monocular camera feed. This package consists of two sub-packages: lsd_slam_core and lsd_slam_viewer. lsd_slam_core contains the full SLAM system whereas lsd_slam_viewer is used for 3D visualization.

- ardrone_autonomy

  The driver of the AR Drone. It can let us control the AR Drone's takeoff and landing as well as its behavior after takeoff.

  To let AR Drone takeoff, landing or emergency stop, we just need to publish `std_msgs/Empty` message to `ardrone/takeoff`, `ardrone/land` and `ardrone/reset` topics respectively. In order to fly the drone after takeoff, we can publish a message of type `geometry_msgs`::Twist to the `cmd_vel` topic.

- Gazebo simulator

  Simulate the behavior of the AR Drone. It was originally built under Fuerte and compiled with rosmake. We also need to make modification on it so that it is able to compile for Kinetic using catkin make.

New package:

- lsd_slam (new)

  Modify and upgrade the existing lsd_slam package to make it compile and work for ROS Kinetic under Ubuntu 16.04

- Joystick Controller

  Control the movement of the drone using joystick connected to our laptop.

- Rviz visualizer

  Visualize the movement of the AR Drone as well as the constructed 3D map.

- automatic path planning

  After the 3D map is constructed, we are going to let the AR Drone do path planning to avoid obstacles. This package will probably involve existing libraries in Moveit! motion planning framework.

<u>Hardware:</u>

Existing hardware: AR Drone Parrot, Joystick, Laptops with ROS Kinetic and Ubuntu 16.04, smart phones with the drone controller App(AR.FreeFlgiht)

Testing done: Use App and Laptop with joystick to control and get image from AR Drone Parrot.

<u>Management Plan:</u>

Bi-weekly meeting on Monday and Friday at Kreiger 70.

Sensors:

mono-camera, Parrot drone's IMU

Tasks:

Map and reconstruct the environment, localize the drone, automatic path planning(if possible)

Safety risks:

Risks	***Risk managements and mitigation***
Drone out of control	Only operate the AR Drone in the safety net, and make sure that the net is fully lowered before flying the drone. Add emergency stop option to the joystick so that we can stop the propellers at any time.
Drone's propellers hurt human body	Always put the drone propeller guard on before flying the drone, and always maintain visual line of sight for the drone. Before the propellers start spinning, make sure that everyone is outside the net with eye protection and the safety net is fully lowered to the floor. No one is allowed to enter the net when the propellers are spinning.
Unintentionally start up	After finishing flying, make sure that the battery is off and disconnect from the drone, and put the drone back into the box.

Schedule:

Items	Start Date	End date
Install necessary packages in ROS Kinetic	March 27	March 28
Convert the LSD SLAM from rosbuild+Indigo to catkin+Kinetic	March 28	April 4
Be able to implement LSD-SLAM under ROS Kinetic on our own laptop independently with sample data as input	April 4	April 11

Communicate with AR Drone and get the image data	April 11	April 13
Tesing the LSD SLAM on the AR Drone	April 13	April 15
Path planning package	April 15	April 30
Test LSD SLAM & path planning on AR Drone	May 1	
Wrinting final report and making poster	May 1	

References:

"Informatik IX Chair for Computer Vision & Artificial Intelligence." *Computer Vision Group - Visual SLAM - LSD-SLAM: Large-Scale Direct Monocular SLAM*, 8 Mar. 2016, vision.in.tum.de/research/vslam/lsdslam.

"Wiki." *Migrating From Rosbuild*, wiki.ros.org/catkin/migrating_from_rosbuild.