Capstone Proposal

Student: Jianglong LIAO
Supervisor: Professor Bruno Bodin

Title

Real-time filtering for benchmarking multi-sensory simultaneous localization and mapping (SLAM) systems

Subject Areas

Software engineering, computer vision, robotics, embedded systems, benchmarking.

Challenges

Theoretical knowledge

Understanding and integrating the filtering algorithm requires a strong basis on linear algebra and image processing methods.

The student has taken advanced courses such as machine learning and computer-assisted image processing and familiarized himself with topics such as simultaneous localization and mapping, embedded systems, and advanced algorithms via previous internships, independent reading, and self-studying.

Implementation

Implementing the real-time filtering in C++ in the large, on-going, open-source project SLAMBench requires a thorough understanding of software architecture, coordination ability with the global research team and versioning control.

The student has done numerous open source projects in Linux environment, especially in the area of performance benchmarking, and is currently taking the courses such as software engineering and C. Although this will be his first attempt to develop a module in C++, previous experiences in other object-oriented languages (Python, R, Java, Javascript) should assist with the process.

Scope

Benchmarking a SLAM system for each specific type of SLAM algorithm requires a combination of deep understanding of computer vision and special coding adjustment for various filters. Creating and integrating a generic filtering module into a SLAM benchmarking system allows SLAM researchers to test their SLAM systems with little specific knowledge of certain filtering techniques and start benchmarking with minimum configuration.

The first goal of this project is to investigate the current SLAMBench package and consider restructuring the architecture with minimum impact on the end users. Then implement three fundamental filters (random drop, blur, resize) to test their compatibility with the SLAMBench program. The second goal is to develop a more versatile filtering system using machine learning / advanced algorithms for benchmarking purposes.

Expectations associated with grade achievement

A detailed description of the restructuring process of the SLAMBench program demonstrates adequate understanding of the SLAM benchmarking, and software engineering as it is essential for integrating the filtering module.
A filtering module compatible with SLAMBench program

Semester 1 plan with time allocation

Estimated consultation time is one hour per week. Consultation consists of communication via git repository as well as one-on-one meetings.

Time	Task	Deliverable
week 1	Draft project proposal.	
week 2	Finalize proposal with supervisor.	
week 3	Learn about the architecture of SLAMBench program.	
week 4	Start restructuring the architecture of SLAMBench program.	
week 5	Read papers on the SLAM filtering techniques.	
16 Sep 5pm		Proposal
week 6	Read papers on the SLAM filtering techniques.	
week 7	Implement the filtering in SLAMBench.	
week 8	Implement the filtering in SLAMBench.	
week 9	Implement the filtering in SLAMBench.	
18 Oct 5pm		Draft Report 1
week 10	Implement the filtering in SLAMBench.	
week 11	Organize the filtering and run testing in SLAMBench.	Presentation
week 12	Write a documentation for the filtering.	Presentation
week 13	Write a documentation for the filtering.	
15 Nov 5pm		Final Report 1