NUS
National University of Singapore

Faculty of Engineering

EG3602/EG3612: Vacation Internship Programme
Internship Report

for reporting period 13 May 2019 to 02 Aug 2019

Name	Tay Kong Chun, Victor		
Matriculation No.	A0149540E	Year	**4**
Department	Electrical Engineering		
Internship Period	13 May 2019	to	02 Aug 2019
Mentor	Dr Lin Feng		
Company	Temasek Laboratories		

Table of Contents

Introduction and Objective

Temasek Laboratories was established mainly to maintain the research capability in selected area of defence science and technology in year 2000. They assist local defence science industry and DSO Laboratories to enhance the security and defence of Singapore.

The main objective of this internship is to assist the Control and Guidance Group in developing and improving on an existing visual odometry system for Unmanned Aerial Vehicle. In order to achieve this, full technical knowledge of the workings of the current visual odometry system used and other odometry systems available are required and subsequently to produce a design implementation of the improvement on the current visual odometry system. The resulting improved odometry system should be more robust and accurate and this can be achieved by adding a loop-closure algorithm. A proper documentation should be maintained throughout the whole design and implementation process to facilitate easier handing over of the project.

What has been achieved

During this internship, I had fully documented the technical workings of the current visual odometry used by the Control and Guidance Group and another well-tested odometry system that will be used in detail as a reference to implement the loop-closure algorithm onto the main visual odometry system. I had also designed the implementation of the loop-closure algorithm into the main visual odometry system. The designed has mostly been implemented with the exception of some code errors and bugs due to the limitation of time of the internship.

What was done during internship

This internship is very technically intensive where I have to learn most of the technical knowledge from scratch and to produce a design and implementation of an improvement of the current visual odometry system used by the Control and Guidance Group. My work will be further explained in the following few segments.

What is VO and VIO

Visual Odometry (VO) is the estimation of the camera motion in real time using sequential images. This is usually done using either mono or stereo camera setup. VO is used in GPS-denied environment and without good computing capability of the main system. The computing requirement for VO is usually much lower than other methods of odometry as it does not store the full history of previously calculated location and only uses the recent few camera positions to estimate the new camera position, which results in less accurate and robust system for error correction. Inertia-Measurement Unit (IMU) can be added to the system to further enhance the system's robustness and accuracy as momentum model can easily be determined. The system with the inclusion of IMU is called Visual Inertia Odometry (VIO).

To estimate the camera position with reference to the world frame, VO/VIO will find feature points as reference from the camera image. Using these feature points, the system will then estimate the distance and position of these feature points with respect to the camera frame as seen in Figure 1.

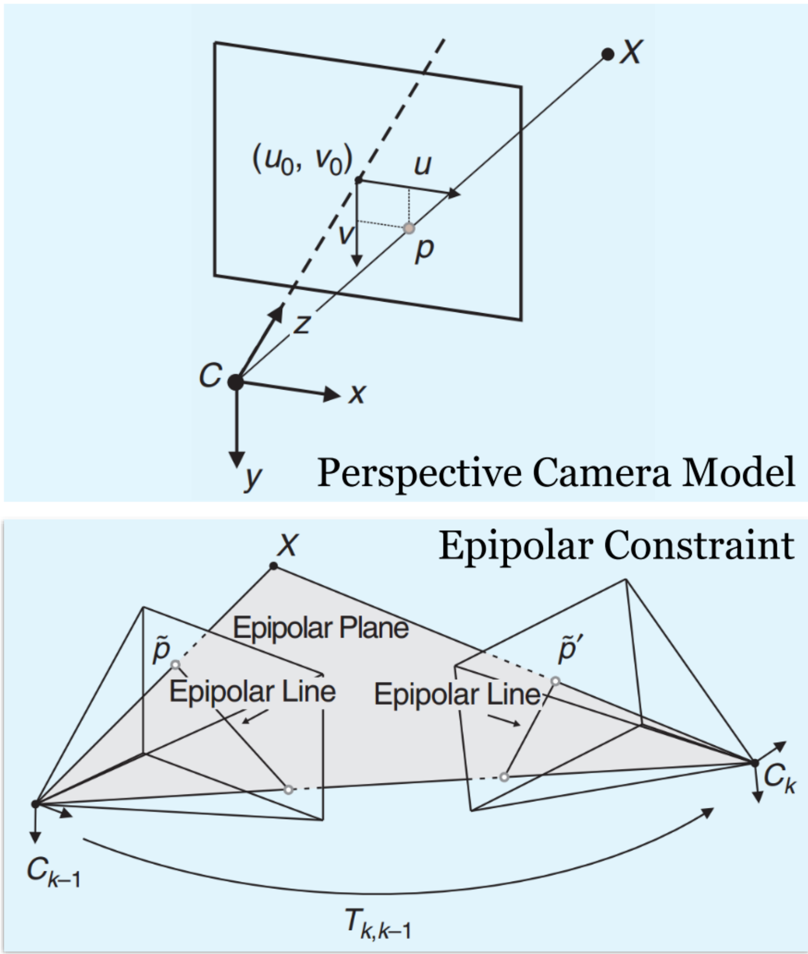

Figure 1: Feature Point position with respect to camera frame

With all the different feature points in the camera image being estimated of their position with respect to the camera frame, the system will then estimate the motion of the camera fame using a sequence of images that share similar feature points as shown in Figure 2.

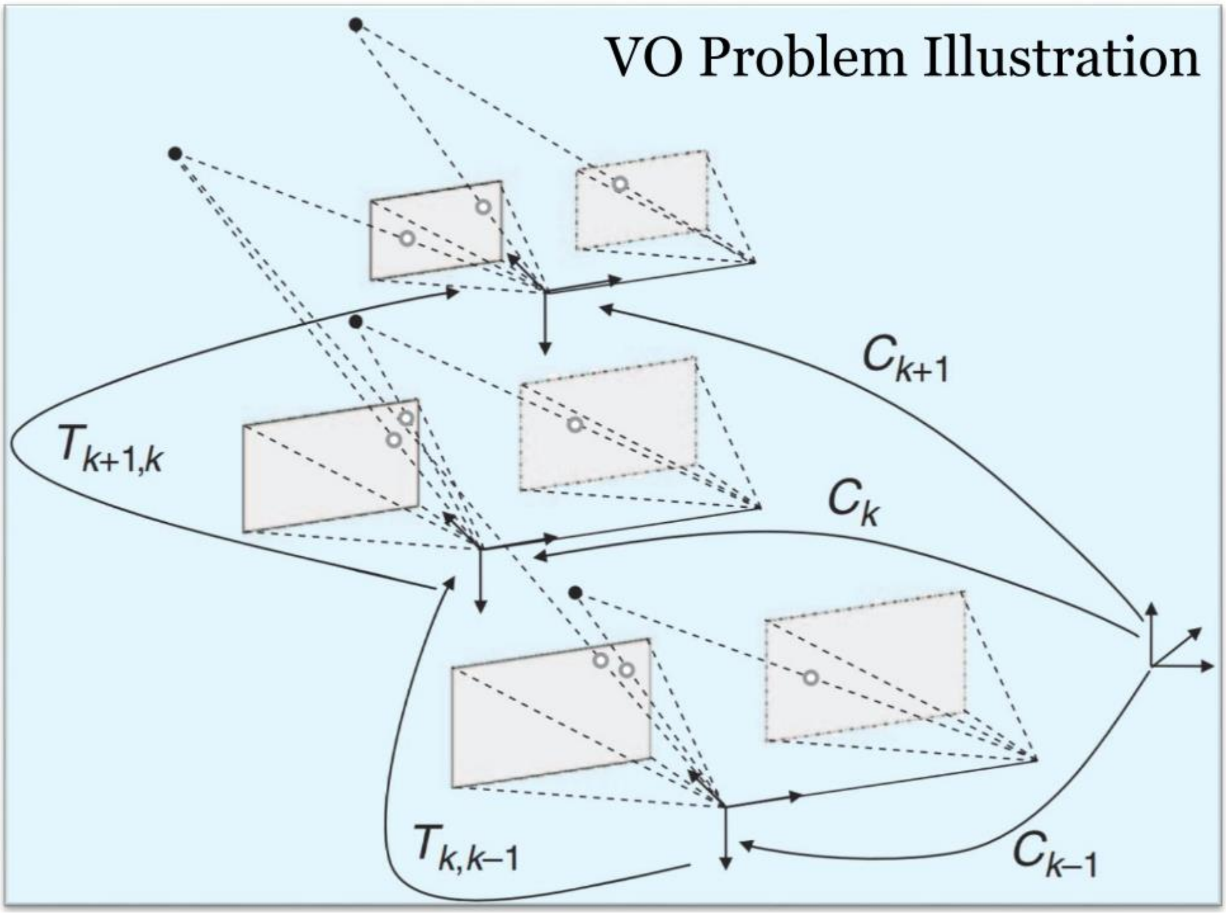

Figure 2: Estimating camera frame motion using a sequence of camera images

What is SLAM

Simultaneous Localization and Mapping (SLAM) is another method of estimating of the camera motion in real time using sequential images and a map. It is usually done using either mono, stereo or RGB-D camera setup. SLAM also used in GPS-denied environment where precision is usually of greater concern. SLAM maps the environment with respect to the camera in real time using a sequence of images and also estimating the position of the camera in the ever-growing map concurrently. This is achieved by storing most of the history of the camera position and the ever-growing map. SLAM usually have an additional error correct algorithm, loop-closure, where it will detect and correct the position error of the current camera frame when it sees a similar history camera frame. However, these increases the computation requirement on the system. Inertia-Measurement Unit (IMU) can be added to the system to further enhance the system’s robustness and accuracy as momentum model can easily be determined.

Figure 3 shows what a typical trajectory with of the camera frames (blue/green colour icons) and the feature points (black/red dots) being mapped in a map while Figure 4 shows a 3D reprojection of the surroundings (coloured feature points) using a RGB-D camera setup.

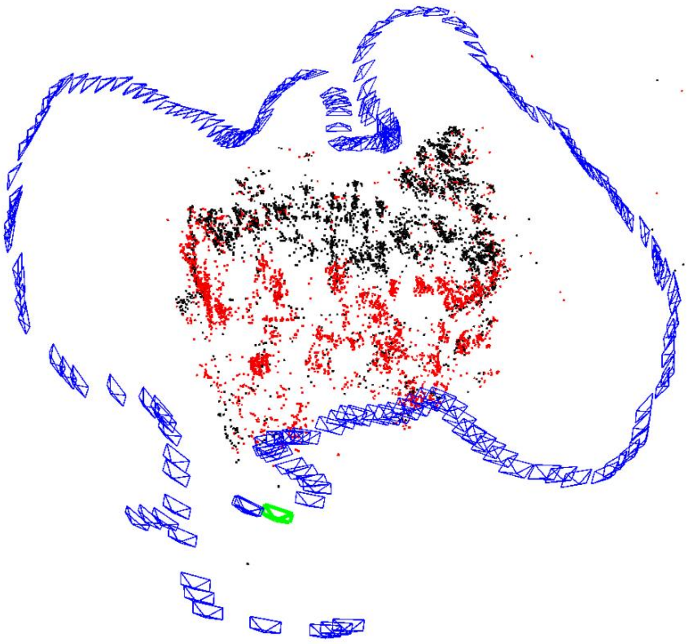

Figure 3: Map of camera frames and feature points

Figure 4: 3D reprojection of surrounding using RGB-D camera

Current System: S-MSCKF VIO

The current VIO system used by the Control and Guidance Group is the S-MSCKF VIO. It uses a stereo RGB camera setup with the left camera as the reference camera frame. The main method used in this system is a semi-direct method where it manipulates the feature points detected in the image directly, e.g. optical flow as shown in Figure 5 where the different feature points are able to be detected as the foreground or the feature points of the background that can be used as a reference to estimate the position of the camera frame. It also implements a sliding-window position and motion estimation as shown in Figure 6, where it only remembers recent few camera frames to estimate the new camera frames using the common feature points shared between the different camera frames. This is to ensure that the estimated position and motion will be accurate, and the drift error is brought to the minimum. S-MSCKF is more accurate and efficient than most state-of-art monocular solutions such as VINS-Mono and OKVIS.

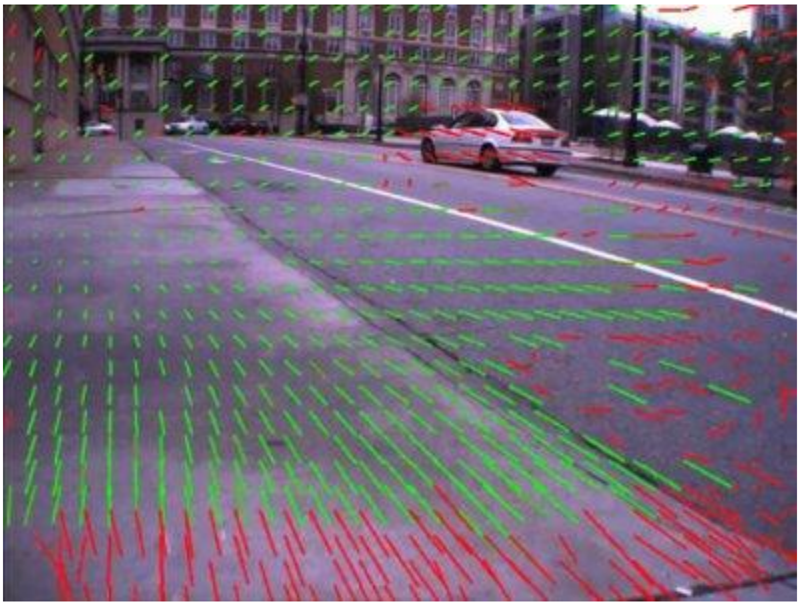

Figure 5: Optical flow vectors using feature points detected in the camera image

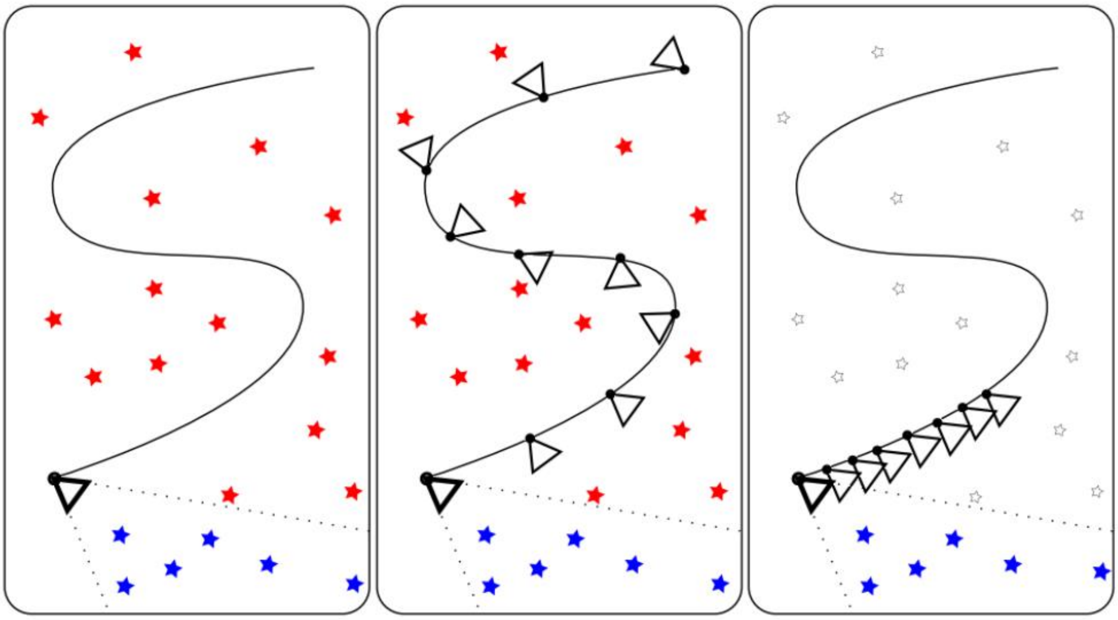

Figure 6: Difference between EKF SLAM, keyframe-based SLAM and MSCKF

Loop-closure algorithm used: ORB_SLAM2

The loop-closure algorithm that was used was inspired and referenced from ORB_SLAM2. It is a state-of-art SLAM algorithm that is well-tested and recognised. ORB_SLAM2 is capable of using different camera setup such as mono, stereo, or RGB-D. The main method of processing and estimating the position and motion of the camera frame is full feature-based method, where descriptors were used to describe the different features detected from the camera image. The system will then make of the these descriptors to determine these feature points and the camera frame's position in the map generated by the SLAM algorithm. This method enables the system to store and process the history camera frames more efficiently and quickly. With the availability of most of the history camera frames, ORB_SLAM2 is capable of detecting if the current camera frame is similar to one of the history frame to correct the drift error due to the prolong position estimation of the camera frames over time. This is a very common issue that is prevalent in all of position estimation algorithm as shown in Figure 7. As seen in Figure 8, the system is able to detect similar feature points observed between the current camera frame and the history camera frame.

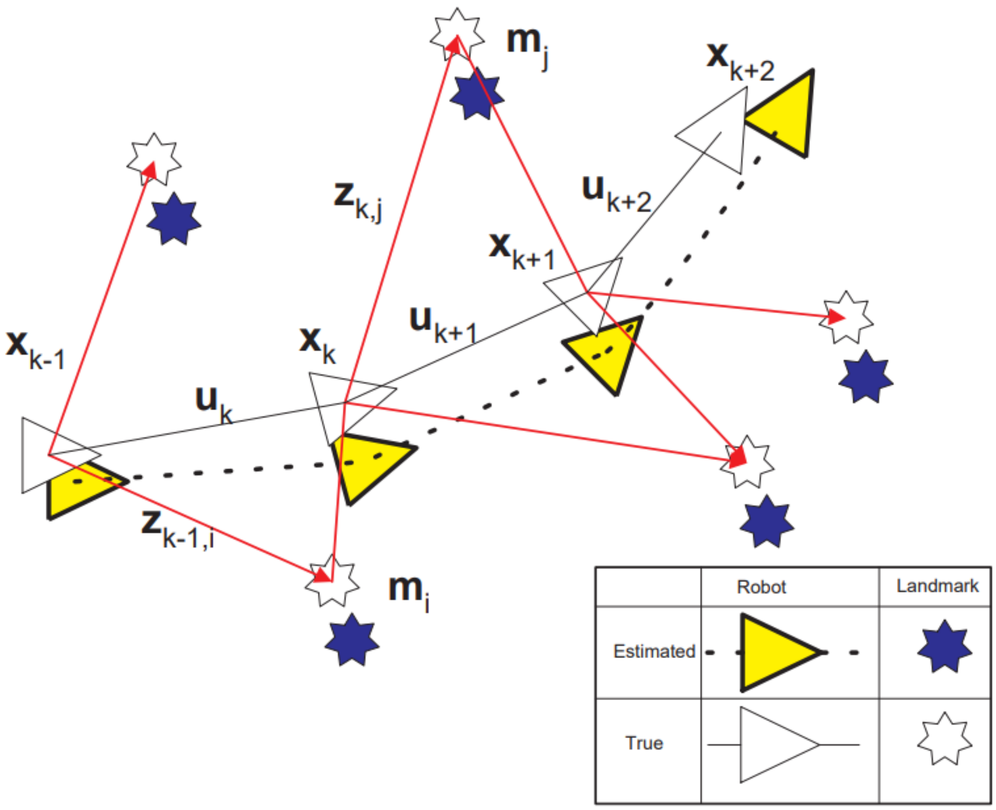

Figure 7: Position Estimation Drift Error

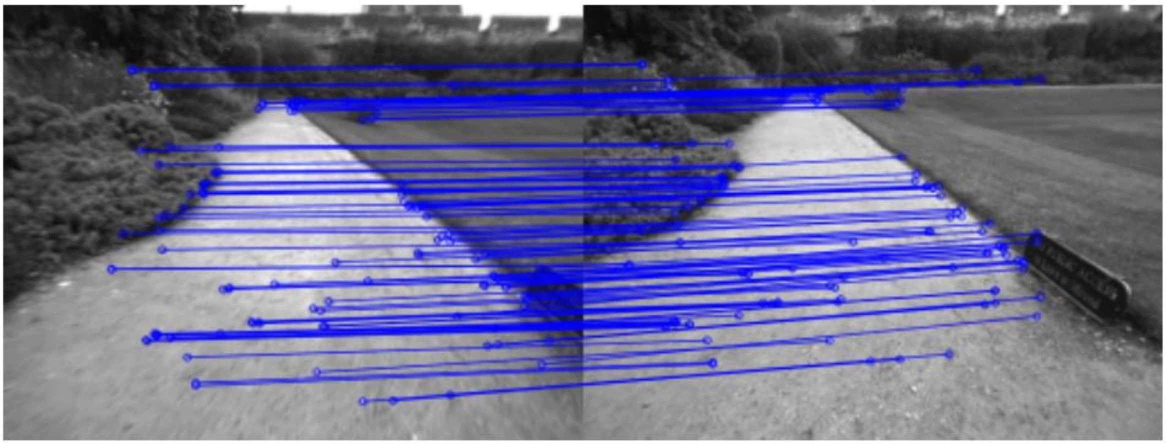

Figure 8: Loop detected due to similar feature points detected between camera frames

Once ORB_SLAM2 detected that the current camera frame is similar to the history camera frame, the system will then calculate the transformation and rotation of the current camera frame with respect to the history camera frame to remove the drift error and back project it to all the previous history frames as shown in Figure 9.

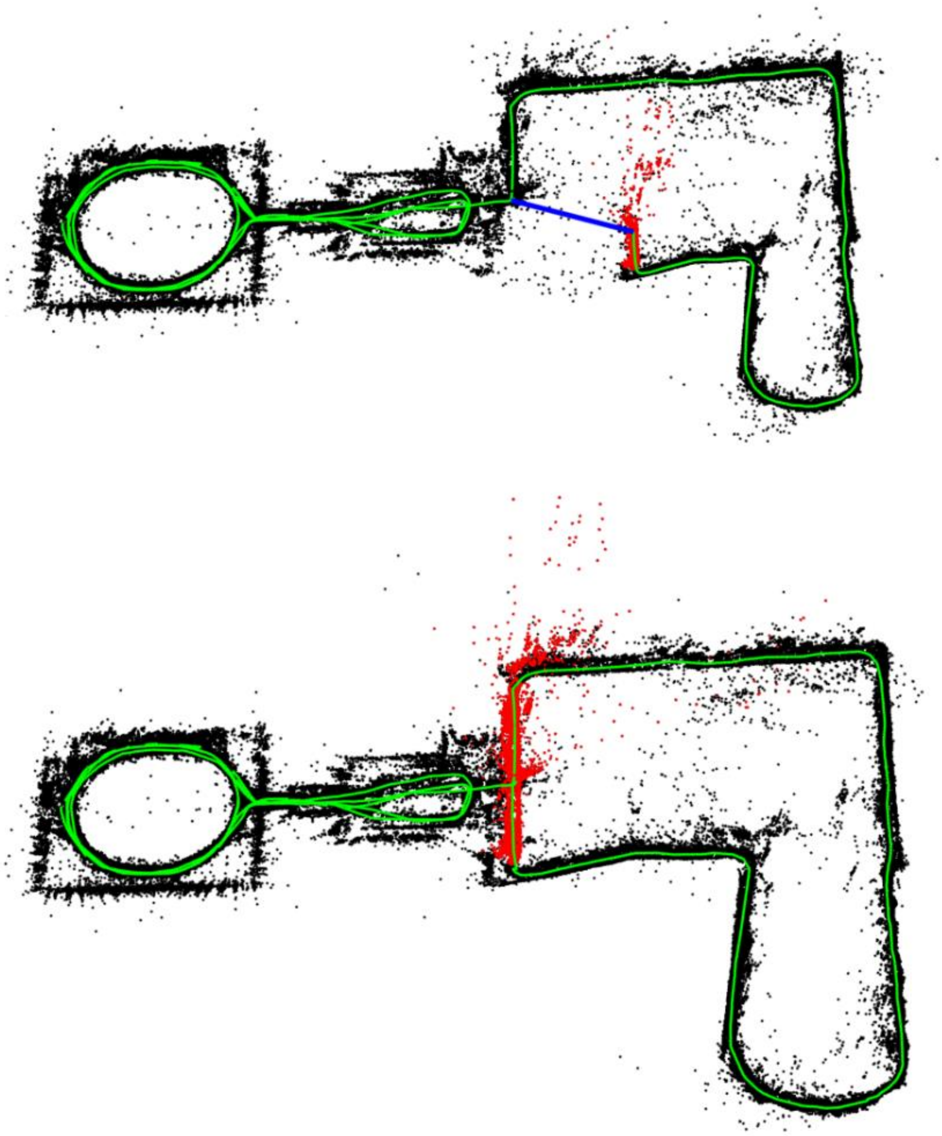

Figure 9: Loop correction for position drift error

One of the main reasons that the loop-closure algorithm of ORB_SLAM2 is chosen to be implemented on S-MSCKF is that it is able to handle and detect camera frames with severe scale changes and dynamic object in the camera image as seen in Figure 10.

Figure 10: Loop detection with severe scale change and dynamic objects in the image

Improved Implementation

To implement the loop-closure algorithm of ORB_SLAM2 into S-MSCKF, there is a huge hurdle that need to be address. S-MSCKF have a totally different working system architecture from ORB_SLAM2 where S-MSCKF uses semi-direct method, directly uses the features detected, of estimating the position and motion of the camera frame and ORB_SLAM2 uses fully feature-based method, uses descriptors to describe the feature points detected, of estimating the position and motion of the camera frame.

To resolve this issue, the backplane of ORB_SLAM2 is ported over to S-MSCKF so that it the new system is capable of using both semi-direct and feature-based methods. This will increase the amount of computational resources required but this is to test if the loop-closure algorithm is able to run on S-MSCKF.

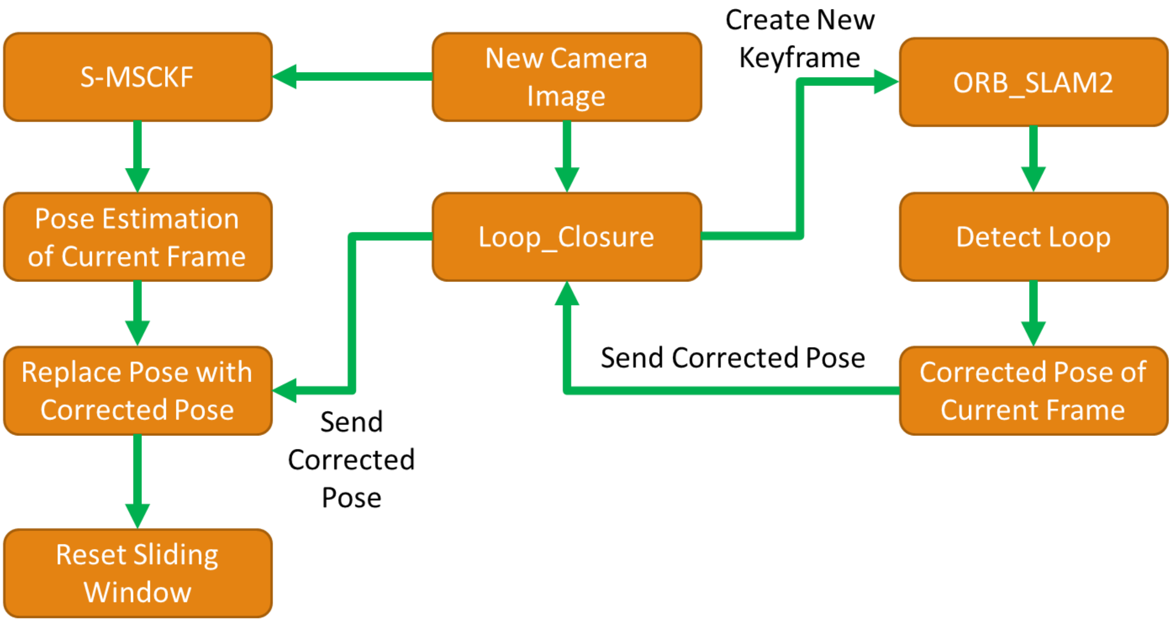

Figure 11: Simplified system overview of the implementation of loop-closure algorithm onto the current system

From Figure 11, whenever a new camera image is received by the system, the image will be passed to S-MSCKF and the Loop_Closure algorithm. S-MSCKF will proceed to process and estimate the current position and motion of the current camera frame and wait for the Loop_Closure algorithm to acknowledge if the current camera frame needs to have its position changed due to drift error. Loop_Closure on the other hand, will create a new camera keyframe in the ORB_SLAM2 backplane to process if the current camera keyframe is similar to any of the history camera keyframes. If there is a history camera keyframe that is similar to the current camera keyframe, it will then estimate the new position of the current keyframe and correct the position of the history camera keyframes in the database. The new position of the current camera keyframe will then be passed back to S-MSCKF to correct the position of the current camera frame and reset the sliding window reference as the positions of the previous camera frames are inaccurate and only the current camera frame have an accurate position.

Implementation Level

On the code level implementation, most of the algorithm for the improved system has been implemented. However, there are some major issues that have yet to be rectified due to the time constrain of the short internship period. Some of the major issues includes, multi-thread randomly kills the algorithm node, system is very computationally heavy which results in the system being very sluggish and the code is not optimised.

Future Works

Some of the future works that could be added onto the current system includes fixing the issue of multi-thread randomly kills the algorithm node, making the system less computationally heavy, optimizing the code, adding feature point detection under severe lighting conditions and removal of dynamic objects from the system in the camera image to increase the robustness of the system.

Reflection and Learning Points

During the course of the whole internship, I have learnt that continuous self-learning is important as the knowledge needed to address the problem is never sufficient. There will be no one guiding the learning process of the required knowledge like the structured lessons in NUS and everything are learnt mostly on need to know basis. Self-motivation on learning new knowledge is the main drive to keep myself updated to the knowledge that I need to finish my work, and this is observable with all the colleagues whom I had interacted throughout the whole internship.

Documentation of everything is very important as there are many times where I had to re-learn some information due to the lack of documentation in the work or things that I had read, and this wasted lots of my productive time. Due to the vast knowledge that I always need to keep updated with, there are times where I will just forget certain things and would need to refer again to refresh my memory. With the help of proper documentation, these knowledges could be easily referred to and will greatly reduce the time needed to re-search of the relevant information again.

Clear communication of ideas and information is critical to prevent misunderstanding and double working on the mistakes due to misunderstanding between colleagues. This wasted lots of time during the whole internship period. For the department that I had worked in, most of my colleagues are not proficient in English and uses Mandarin as their primary language. I am not very proficient in speaking Mandarin and hence, there is a small language barrier that I had encountered during my work as an intern in Temasek Laboratories. There were many miscommunications between my colleagues and I and hence, mistakes were made. This greatly emphasise that clear communication of ideas and information is crucial to get work done fast and efficiently within a working environment.

And lastly, thinking outside the box for solutions can save lots of time and effort in solving most of the problems as they can be resolved more easily and more efficiently than some traditional solution. There were many times where I faced problem that were very hard to solve using the traditional method. With the constant reminder from my colleagues to think outside-the-box for potential solutions, I was able to resolve these problems using unorthodox solutions and it is more elegant and efficient than orthodox solutions.

Overall, this internship has made me realized that working is very different from the studying that we do in NUS and it is a very fresh experience for me. This allows me to have a mental preparation of what to expect when I enter the workforce in the future and to prepare for the job that I will be working in.

NUS National University of Singapore

VIP REPORT CLEARANCE FORM

Please ensure that this form is attached at the back of your VIP report before the submission to the faculty.

Student Information

Name of Student: Tay Kong Chun, Victor

Department: Electrical Engineering

Matriculation No: A0149540E

Report: End-Term

Company Information

Name of Company: Temasek Laboratories

Name of Supervisor: Gao Zhi

Designation: Research Scientist (A)

Email / Telephone: tslgz@nus.edu.sg / 97709185

Report Clearance by Company

Signature	Company Stamp	Date
Gao		18 Aug 2019

Note: If the company wishes to have a copy of the report, the arrangement is left between the company and the intern.

Faculty Internship Team
Email: enginternship@nus.edu.sg

NUS
National University of Singapore

WEEKLY JOURNAL

<u>The weekly journal should serve as your work log diary. Highlight *one* key learning point you gain each week. The point should be kept succinct. Bullet points are accepted.</u>

As a guide, you may think of how your activities performed in the week fit within the larger project/goals of your business unit. You may also think of how your perspectives about a particular matter (e.g. academic goals, life goals, motivations) have strengthened or changed following your work experiences including interactions with your managers/colleagues.

Attach this journal log to your internship report for submission to your internship mentor. While the internship report must be signed by your workplace supervisor, supervisor's endorsement of the journal is <u>not</u> required.

Note: If you are absent due to In-Camp Training or other approved reasons, please indicate so and the dates of your absence.

Period	Key learning point
Week 1	• Familiarize with the company and project team • Familiarize with the platform the project team used • Familiarize with the system that the project team used • Planned the workplan for this project
Week 2	• Understanding the required knowledge needed to familiarize with the system that the project team used
Week 3	• Understanding the required knowledge needed to familiarize with the system that the project team used • Critically analyzed part of the system • Documented the knowledge gained from the analysis
Week 4	• Understanding the required knowledge needed to familiarize with the system that the project team used • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis
Week 5	• Understanding the required knowledge needed to familiarize with the system that the project team used • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis

Week 6	• Understanding the required knowledge needed to familiarize with the loop-closure algorithm that will be implemented into the main system • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis • Came up with the initial design of the improved system
Week 7	• Understanding the required knowledge needed to familiarize with the loop-closure algorithm that will be implemented into the main system • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis • Modify and update the design of the improved system
Week 8	• Understanding the required knowledge needed to familiarize with the loop-closure algorithm that will be implemented into the main system • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis • Starts to implement the designed improved system
Week 9	• Understanding the required knowledge needed to familiarize with the loop-closure algorithm that will be implemented into the main system • Correcting the wrong analysis • Critically analyzed part of the system • Documented the knowledge gained from the analysis • Modify, update and fixes the issued with the implementation of the designed improved system
Week 10	• Continue to implement the designed system • Fixes and update the issues experienced during the implementation of the designed system
Week 11	• Continue to implement the designed system • Fixes and update the issues experienced during the implementation of the designed system • Most of the implementation is done and testing is made onto the system (system is not stable)
Week 12	• Continue to implement the designed system • Fixes and update the issues experienced during the implementation of the designed system • Most of the implementation is done and testing is made onto the system (system is not stable)

Student's Signature / Date (of last journal log written):

2nd Aug 2019