Roger Black
MAGICC Lab 497R
Semester Report
April 22, 2020

My experience in the MAGICC Lab has been both educational and memorable. I have written this brief report to record what I have learned and transfer the work I have done on the IMU aided KLT project. I want to express my appreciation specifically to Dr. Beard and Hayden Morgan for their support and guidance.

Learning Outcomes

My time during the first half of the semester was spent reading technical papers and becoming familiar with the technologies needed to complete the assigned project. From what I studied I have identified the following learning outcomes.

- Familiarity with some of the most recent advancements in image-based target tracking
- Familiarity with the overarching concepts of Visual-MTT
- Basic knowledge of and experience implementing image transformation functions
- Basic knowledge of image feature recognition
- Working knowledge of and experience using opencv
- Experience capturing image and IMU data from a RealSense camera

Project Work

The objective of my assigned project was to implement an IMU aided KLT algorithm into the existing Visual-MTT code base. To achieve this objective the project was split into four main phases. The first was to research current methods of IMU aided KLT and become familiar with the algorithms and technologies to implement them. The second was to build out a proof of concept using python. The third was to transfer the proof of concept from python to C++. Finally, the fourth phase was to implement the C++ algorithm into the existing Visual-MTT code base.

Due to circumstances beyond my control (family emergencies and COVID-19) I have not completed all phases of the project. I have currently only completed phases 1 and 2. I started looking into phase 3, but my knowledge of C++ as this point in time is insufficient to allow me to quickly transfer what I have done in Python to C++. I trust that what I have done in Python will be a good starting point for another undergraduate student to pick up on phase 3 or for a more experienced graduate student to quickly implement phases 3 and 4.

The python proof of concept executes the following workflow

1. Opens a connection to the RealSense camera
2. Reads image and IMU data from the RealSense data streams

3. Passes the raw IMU data through a low pass filter
4. Integrates the filtered IMU data to get position (x, y, z, theta, phi, psi)
5. Warps the last retrieved image using the integrated position as input
6. Retrieves the latest image and subtracts it from the warped image for comparison

A specified number of frames can be skipped before the function will retrieve the next image frame. Meanwhile, the IMU data continues to be retrieved and integrated. The low pass filter alpha values for both acceleration and gyro data can also easily be tuned to achieve a desired performance.

Next Steps

- Add code to record and playback rosbag files. Doing so will allow for consistent testing and comparison to be performed between different frame skips and low pass filter parameters.
- Record/plot the accelerations and velocities. This would be for the purpose of establishing boundaries over which the warping provides acceptable results.
- Complete phases 3 and 4

Notes/Comments

There is a possible weakness in the IMU aided KLT algorithm if the target and vehicle are moving with similar trajectories and near the same velocity. When the last frame is warped based on the IMU data, the target will be warped in the same way. When the warped image and the next camera image are compared, the target track may be lost. This is demonstrated in Figure 1 below. I do not know enough about the applications of IMU aided KLT to know if this would be an issue or not.

A possible solution would be to warp the motionless landscape based on the vehicle IMU data but warp the area around the target using a mixture of the vehicle IMU data and the estimated target trajectory. This is demonstrated in Figure 2.

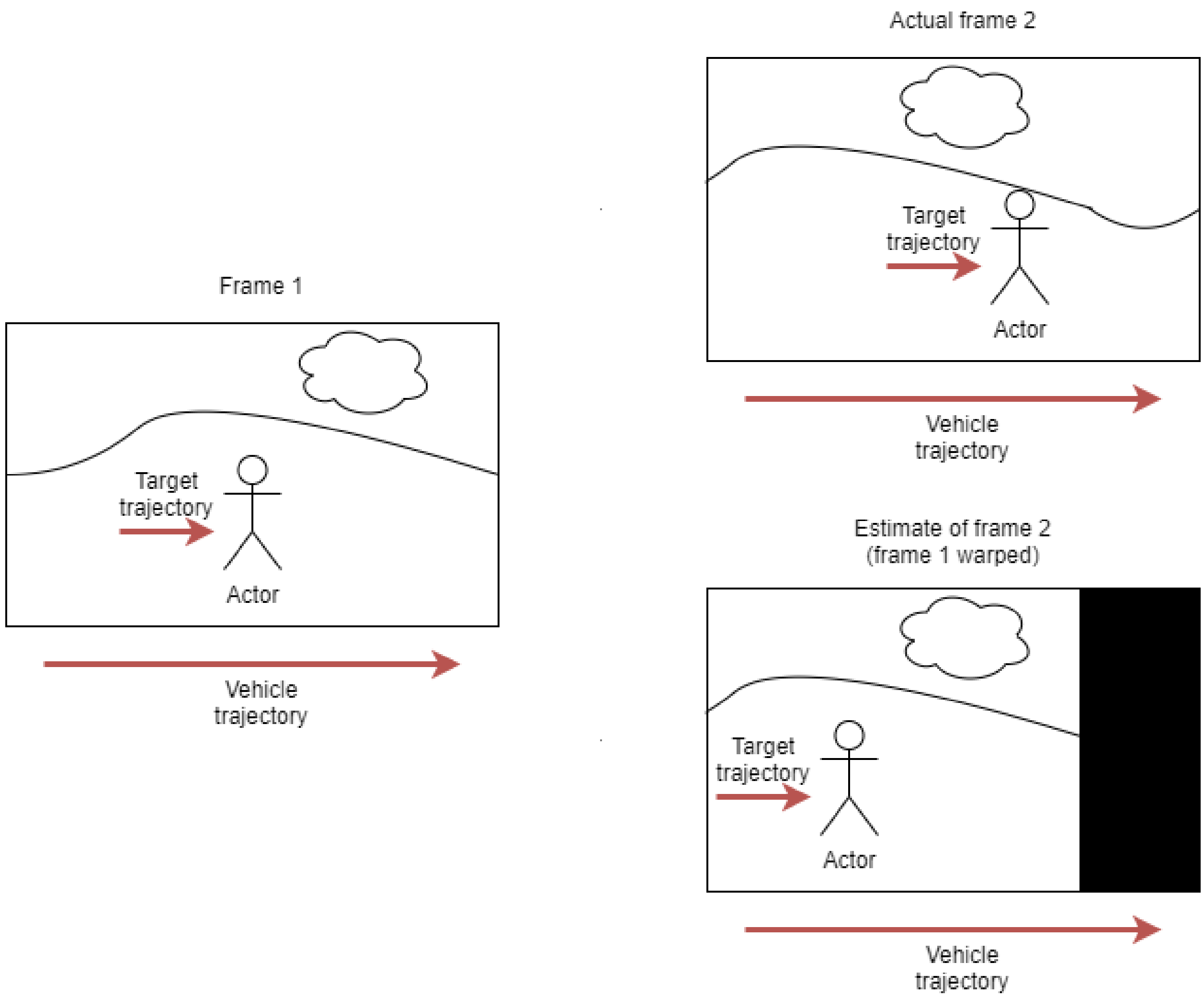

Figure 1. Possible weakness in the IMU aided KLT algorithm. The target track is lost in the warping.

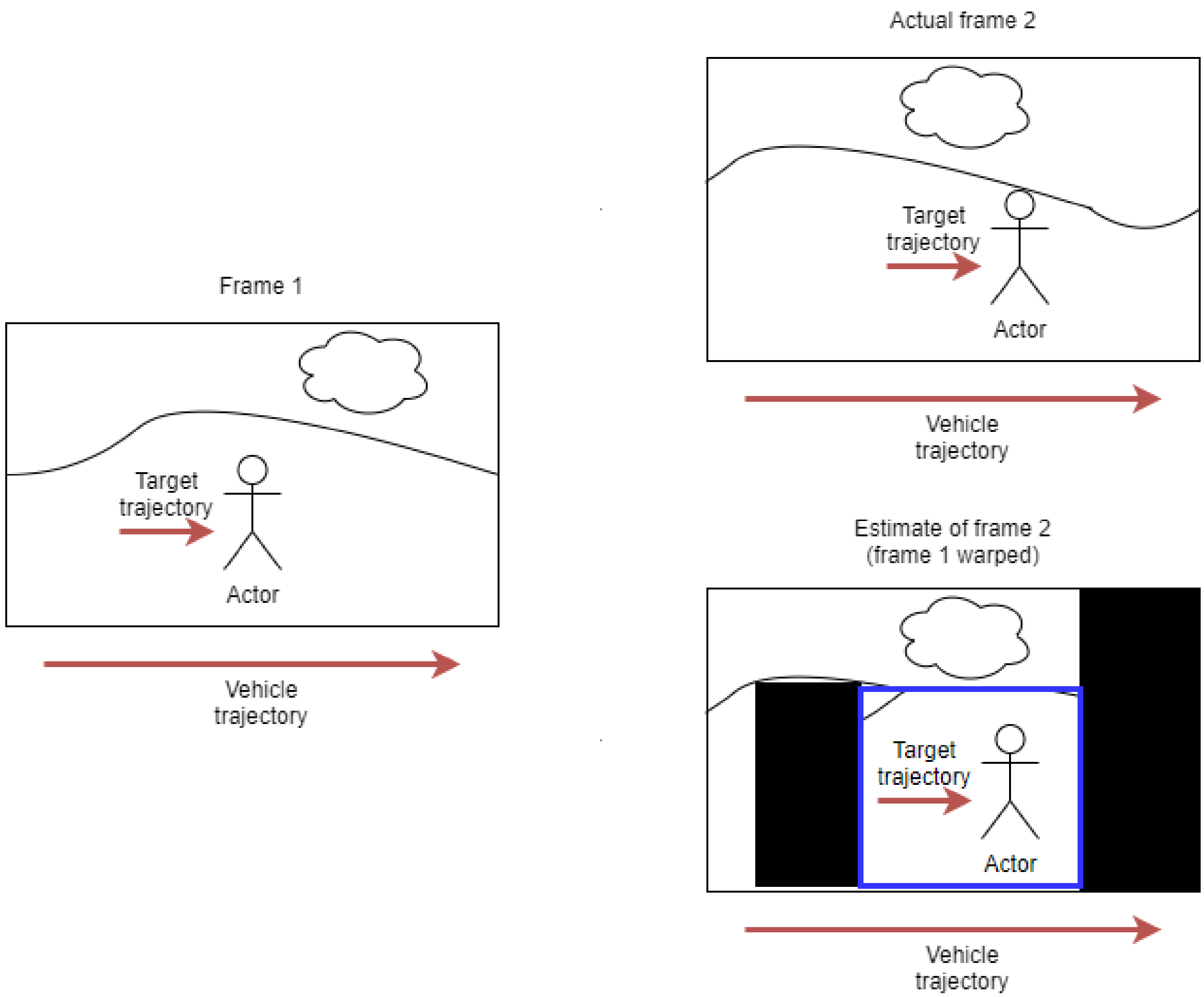

Figure 2. A possible solution for when the target and vehicle have similar trajectories and velocities. The motionless landscaped is warped based on the vehicle IMU data, but the area around the target is warped using a mixture of the vehicle IMU data and the target trajectory.