Orientation Tracking & Panorama Construction

Manoj Kumar Reddy Manchala
Department of Mechanical and Aerospace Engineering
University of California San Diego
La Jolla, U.S.A.
mmanchala@ucsd.edu

Nikolay Atanasov
Department of Electrical and Computer Engineering
University of California San Diego
La Jolla, U.S.A.
mmanchala@ucsd.edu

***Abstract*—This paper focuses on using constrained gradient descent to estimate the three dimensional orientation of a rotating body over time using the angular velocity, linear acceleration measurements from an IMU and to construct a panoramic image by stitching the RGB camera images captured by the body over time based on the estimated orientation.**

***Index Terms*—Orientation Tracking, Constrained Gradient Descent, Panorama, styling, insert**

I. Introduction

The problem this paper aims to solve is "3D orientation tracking of a rotating body using IMU data and Panorama construction based on RGB image data". This is an important step to proceed further with Localization and Mapping. The process plays its part in vast applications in autonomous robots like Simultaneous Localization & Mapping(SLAM), Environment Mapping and Monitoring, Maneuvering through unknown environments, GIS applications etc. The approach taken is to estimate the orientation trajectory of the body through optimization and then create a panorama based on the estimated orientations.

II. Problem Formulation

Consider a body undergoing pure rotation in space whose body frame orientation at time t is represented by a unit quaternion $q_t \in H_*$. Let ω_t, a_t be the measured angular velocity and linear acceleration obtained from the IMU respectively.

A. Motion Model

We can predict the quaternion corresponding to the orientation at the next step q_{t+1} using the angular velocity measurements ω_t and the differences between consecutive time stamps(obtained from IMU data) τ_t based on the Discrete-time quaternion kinematics as follows:

$$q_{t+1} = f(q_t, \tau_t\omega_t) := q_t \circ \exp\left([0, \tau_t\omega_t/2]\right) \tag{1}$$

where exp() represents quaternion exponential function.

B. Observation Model

Since the body is undergoing pure rotation, the acceleration of the body should be approximately [0, 0, -g] in the world frame of reference, where g is the gravity acceleration. Hence, the measured acceleration a_t in the IMU frame should agree with gravity acceleration after it is transformed to the IMU frame using the orientation q_t, as follows:

$$a_t = h(q_t) := q_t^{-1} \circ [0, 0, 0, -g] \circ q_t \tag{2}$$

where q_t^{-1} represents quaternion inverse.

C. Optimization Problem

In order to to estimate the orientation trajectory $q_{1:T} := q_1, q_2, ..., q_T$ based on the motion model in (1) and observation model in (2) we formulate an optimization problem whose cost function $c(q_{1:T})$ is defined as follows:

$$\frac{1}{2}\sum_{t=0}^{T-1} \left\|2\log\left(q_{t+1}^{-1} \circ f(q_t, \tau_t\omega_t)\right)\right\|_2^2 + \frac{1}{2}\sum_{t=1}^{T} \left\|a_t - h(q_t)\right\|_2^2 \tag{3}$$

where $||.||_2$ represents the two-norm, the first term measures the error between the estimated orientation and the motion model prediction, while the second term measures the error between the acceleration measurements and the observation model prediction.

The motion model error is based on the relative rotation $q_{t+1}^{-1} \circ f(q_t, \tau_t\omega_t)$ between the predicted orientation $f(q_t, \tau_t\omega_t)$ and the estimated orientation q_{t+1}. The error obtains the axis-angle parametrization of the relative rotation error using the quaternion $\log(\cdot)$ function and measures the angle of rotation as the norm of the axis-angle vector. We need to enforce the constraint that the quaternions q_t remain unit norm i.e., $q_t \in H_*$, during the optimization.

Hence, we have a constrained optimization problem:

$$\min_{q_{1:T}} \; c(q_{1:T})$$
$$\text{s.t. } \|q_t\|_2 = 1, \quad \forall t \in \{1, 2, \ldots, T\} \tag{4}$$

III. Technical Approach

A. Orientation Tracking

1) Calibration: The measurements from the IMU need to be calibrated and converted into the physical units of angular velocity and linear acceleration. The equation for conversion from raw A/D values to physical units is as follows:

$$value = (raw - bias) * scalefactor$$

$$scalefactor = V_{ref}/1023/sensitivity$$

We rely on the data from the on-ground cameras to estimate the bias. Analysing the dataset reveals that the body is static initially for a few time stamps as the values of ω_t do not change. Hence the first few values of $\omega_x, \omega_y, \omega_z$ should be zero. Since the body is undergoing pure rotation, a_x, a_y, a_z should be [0,0,1] in g units.

Starting with q_0 = [1, 0, 0, 0], a simple integration of the angular velocity ω_t is performed, computing $q_{t+1} = f(q_t, \tau_t \omega_t)$ to get the motion model trajectory. The roll, pitch, yaw angles are extracted from the estimated quaternions and rotation matrices of VICON ground truth data to verify the correctness of calibration.

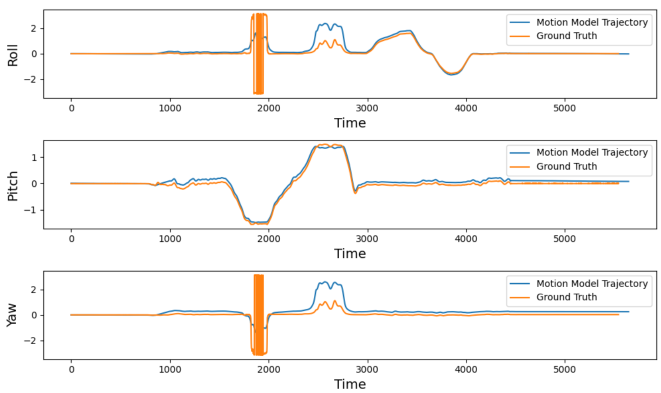

Fig. 1. Orientations of Motion Model Vs Ground Truth - Dataset 1

2) Motion Model Error: There exist two quaternions q, $-q$ corresponding to every rotation. So the condition that all the quaternions follow the same convention has to be enforced. The distance between two quaternions $q_1, q_2 \in H_*$ can be measured by the rotation angle $||\theta_{12}||_2$ of the axis-angle representation θ_{12} of the relative rotation $q_{12} = q_1^{-1} q_2$:

$$d(q_1, q_2) = ||\theta_{12}||_2 = ||2log(q_1^{-1} q_2)||_2$$

3) Acceleration Error: Acceleration error ensures that the optimization converges to the ground truth. The ground truth acceleration is known to be $[0, 0, -g]$. The measured acceleration is in the IMU frame. Hence the ground truth acceleration is converted into the body frame acceleration using the observation model.

$$error = ||a_t - q_t^{-1} \circ [0, 0, 0, -g] \circ q_t||_2$$

4) Gradient Descent Optimization: The optimization can be initialized with all quaternions $q_{1:T}$ to be [1,0,0,0] or from the orientation trajectory estimated from the motion model. Both approaches should converge to the true orientation with the second case converging faster as the start is from a closer orientation trajectory estimate.

Initializing from the motion model trajectory, Gradient Descent Algorithm is implemented as follows:

$$q_{1:T}^{0} = q_{motion}$$

$$q_{1:T}^{(k+1)} = q_{1:T}^{(k)} - \alpha^{(k)} \nabla c(q_{1:T}^{(k)})$$

$$q_{1:T}^{(k+1)} = q_{1:T}^{(k+1)} / ||q_{1:T}^{(k+1)}||_2$$

where $\nabla c(q_{1:T})$ is the gradient of the cost function in (3) w.r.t the quaternions $q_{1:T}$, $\alpha^{(k)}$ is the learning rate.

After every iteration, the quaternions are normalized to enforce the unit quatenrion condition. Both the first term of the cost function and the second part of the cost function are given equal weightage as it is equally important to not go in the negative direction as it is to be close to the ground truth.

B. Panorama

The resolution of the images captured is 240x320. Camera field of view is known to be 60^o along the horizontal and 45^o along the vertical. RGB intensity data of all the images is available. The world coordinates of each pixel of image need to be computed from image coordinates(u,v). Assuming the image lies on a sphere, the longitude(λ) and latitude(ϕ) of each pixel are calculated as follows:

$$\lambda = \frac{60u}{320} - 30 : \phi = \frac{45v}{240} - 22.5 \tag{5}$$

Assuming a depth of 1, these spherical coordinates $(\lambda, \phi, 1)$ are now converted to cartesian coordinates according to: The equations are:

$$x = \cos\phi \cos\lambda : y = \cos\phi \sin\lambda : z = \sin\phi$$

The above conversion helps us to perform the rotation to the world frame:

$$RotatedCoordinates = q_t \circ [0, x, y, z] \circ q_t^{-1}$$

where the image x axis is considered to be along the IMU z-axis and y & z axis in the opposite direction. These rotated coordinates are converted back to spherical coordinates. This sphere of images is now inscribed into a cylinder so that a point $(\lambda, \phi, 1)$ on the sphere has height ϕ on the cylinder and longitude λ along the cylinder circumference. These coordinates are normalized and then the image is unwrapped on to a rectangle with width 2π radians and height π radians.

IV. Results

A. Training

The algorithm has been trained based on 9 datasets, of which 4 datasets contain IMU data, Camera data, VICON ground truth data and the rest only have IMU & VICON data.

1) Calibration Results: The following plots show the Roll, Pitch & Yaw angles of the orientation based on Motion Model and Ground Truth data, plotted to ensure that the calibration has been done properly.

For almost all the datasets, the calibration has been done properly, except some of the datasets where there seems to be more noise. For the dataset 9, roll and pitch seem to be differing.

2) Analysing - Calibration Results: From the ground truth data in the plots it can be seen that - dataset 1 & 2, has mostly pitch motion, some roll and no yaw motion, i.e the panorama will have images on top of each other at the same location but rotated about the centre. Datasets 3,4,5,6,7,9 have all three motions. Dataset 8 seems to only have yaw motion.

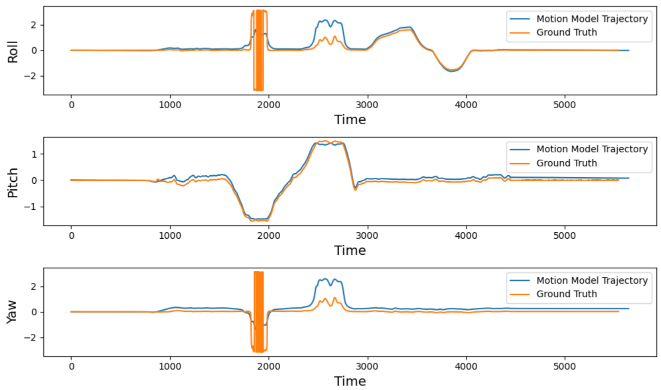

Fig. 2. Orientations of Motion Model Vs Ground Truth - Dataset 1

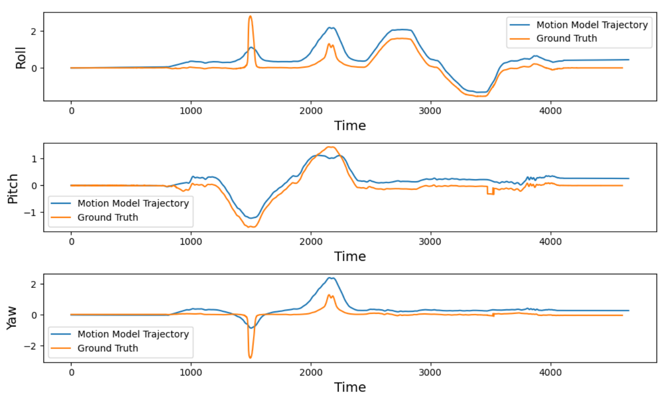

Fig. 3. Orientations of Motion Model Vs Ground Truth - Dataset 2

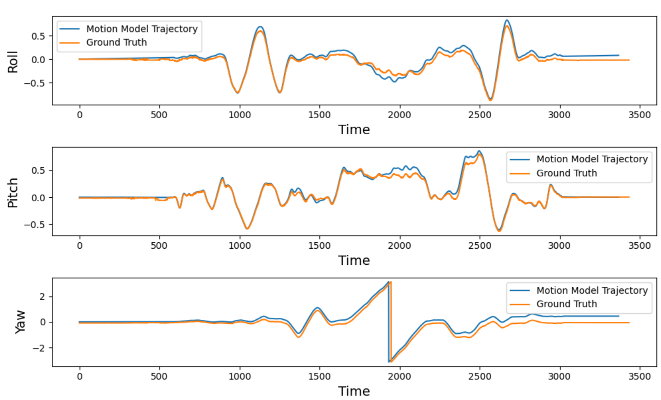

Fig. 4. Orientations of Motion Model Vs Ground Truth - Dataset 3

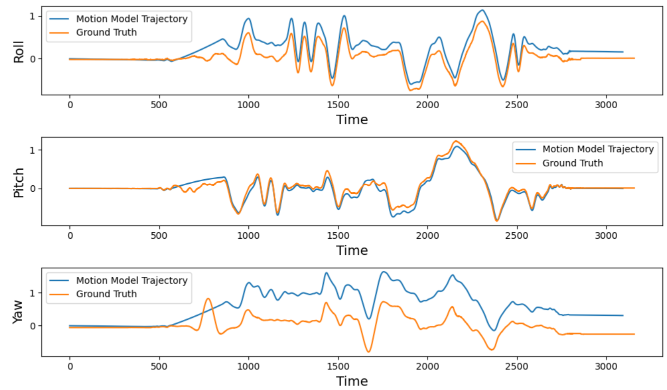

Fig. 5. Orientations of Motion Model Vs Ground Truth - Dataset 4

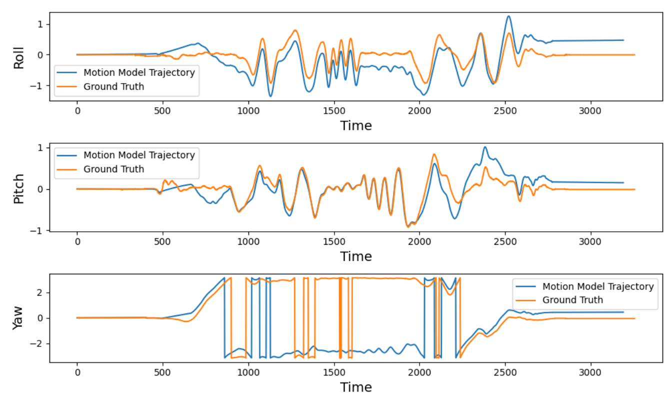

Fig. 6. Orientations of Motion Model Vs Ground Truth - Dataset 5

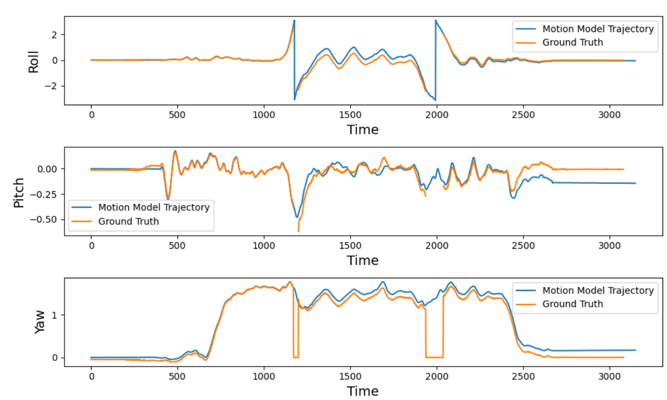

Fig. 7. Orientations of Motion Model Vs Ground Truth - Dataset 6

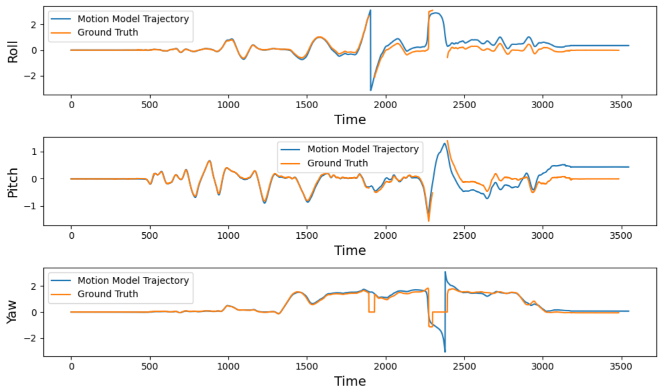

Fig. 8. Orientations of Motion Model Vs Ground Truth - Dataset 7

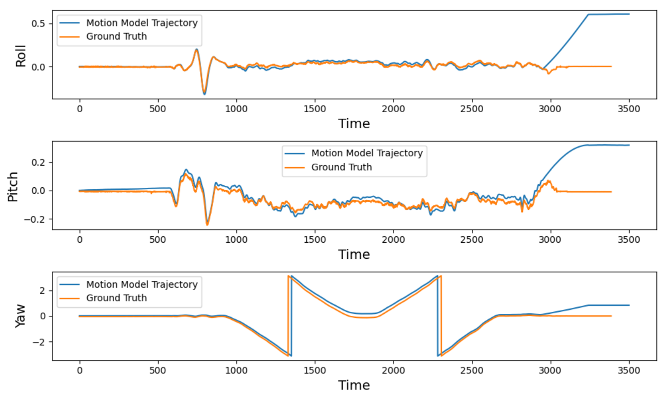

Fig. 9. Orientations of Motion Model Vs Ground Truth - Dataset 8

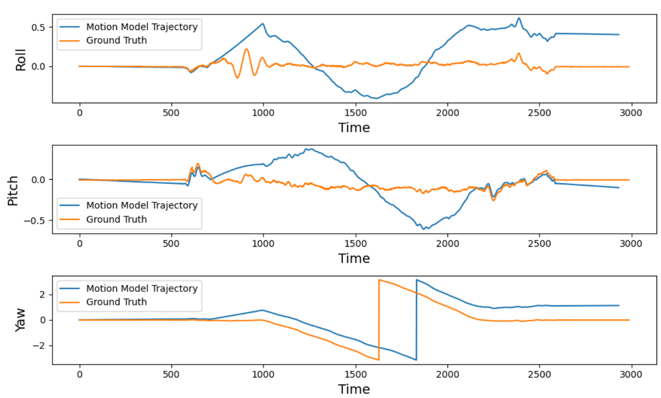

Fig. 10. Orientations of Motion Model Vs Ground Truth - Dataset 9

B. Optimized Trajectory

In the following plots, the roll, Pitch & Yaw angles from the optimized trajectory are plotted along with Motion model and Ground truth. For the Gradient descent, the learning rate is chosen to be 0.0003 after experiment with multiple values and the algorithm seems to be converging at about 100 iterations for all the datasets.

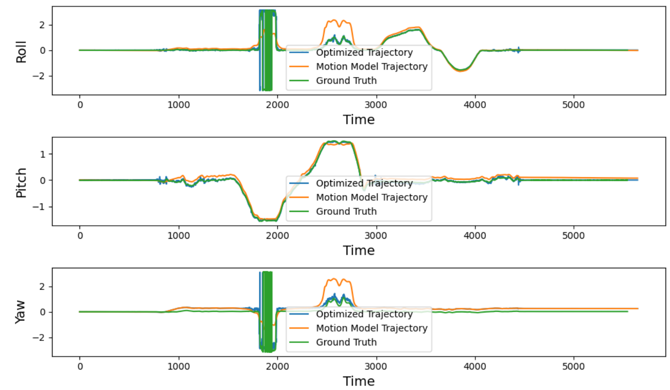

Fig. 11. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 1

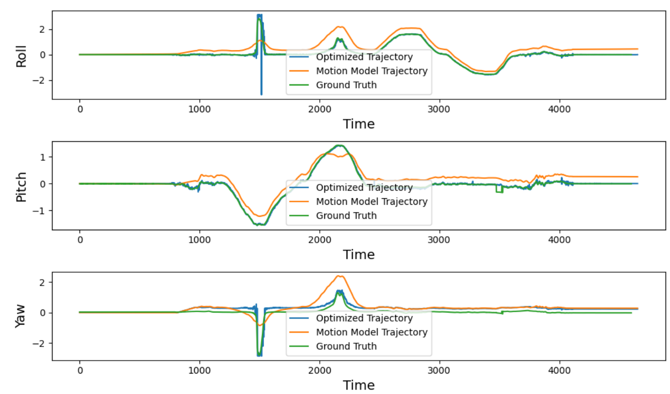

Fig. 12. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 2

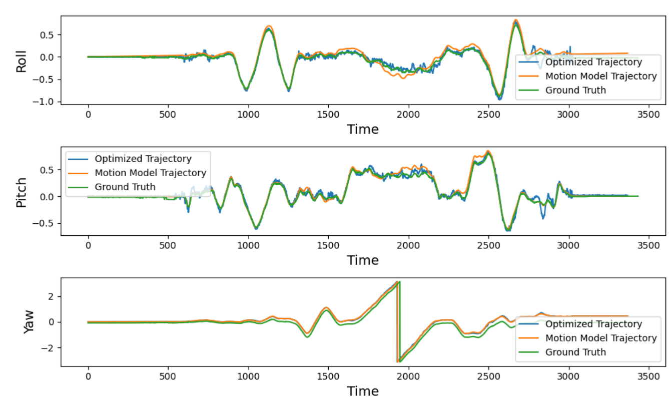

Fig. 13. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 3

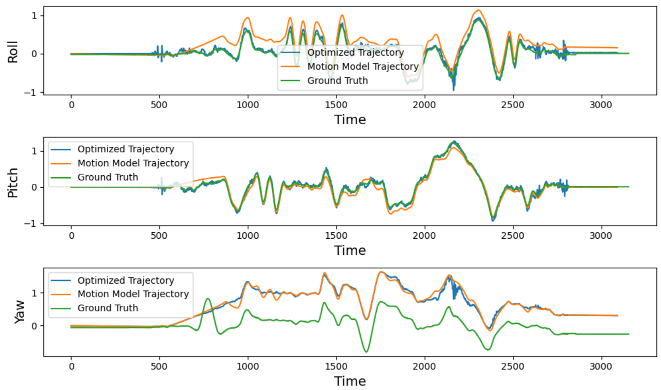

Fig. 14. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 4

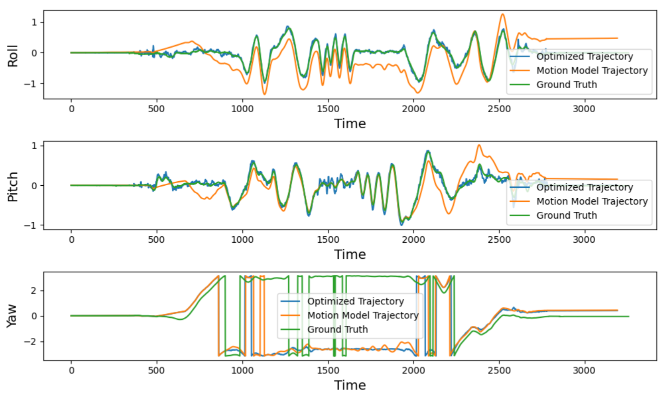

Fig. 15. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 5

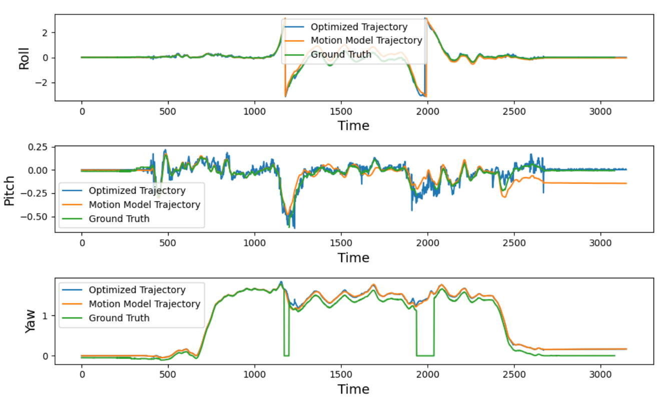

Fig. 16. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 6

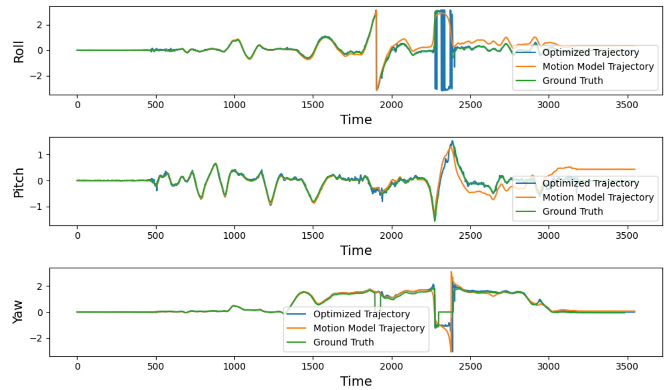

Fig. 17. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 7

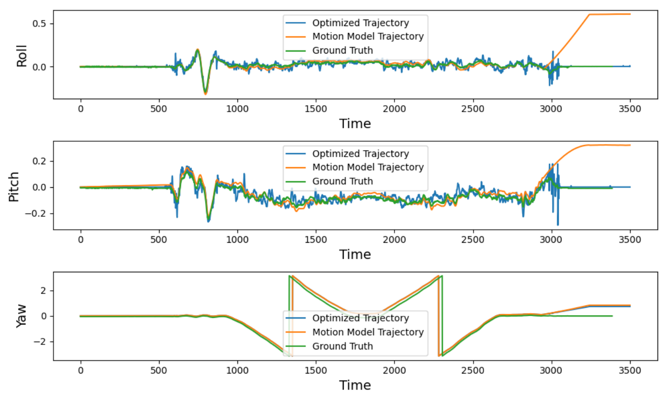

Fig. 18. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 8

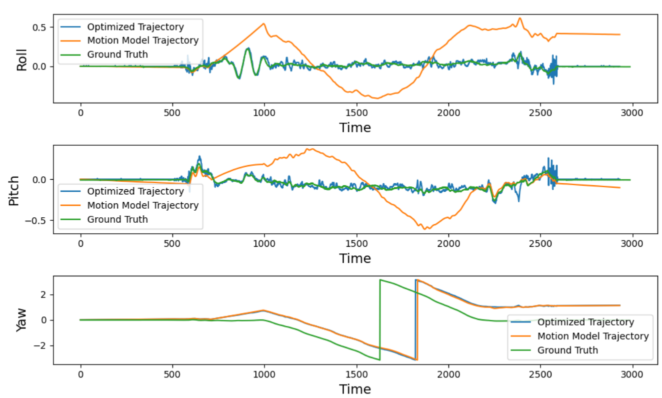

Fig. 19. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 9

C. Analysing - Optimized Trajectory

From the plots, it can be seen that the optimization led to very close convergence of the trajectory to the ground truth. Datasets 1,2,3,7 show almost complete convergence including some of the extremities. Datasets 4,5,6,9 have both the roll & pitch motion predicted but the yaw motion has an offset. Datasets 6,8,9 even though resulted in good convergence the results are noisy seemingly caused by gradient descent trying to estimate high frequency noisy motion.

D. Panorama - All Images

The figures 20,21,22,23 have been constructed based on the entire image dataset. As predicted Datasets 1 & 2 have images overlapped at the centre and images at the top and bottom because there is very little motion in yaw and significant Roll, Pitch motion

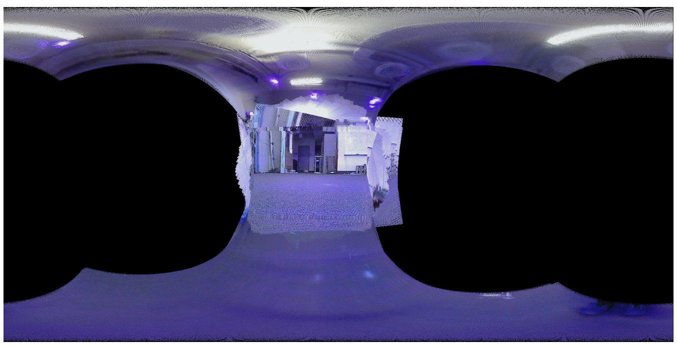

Fig. 20. Panorama - Dataset 1

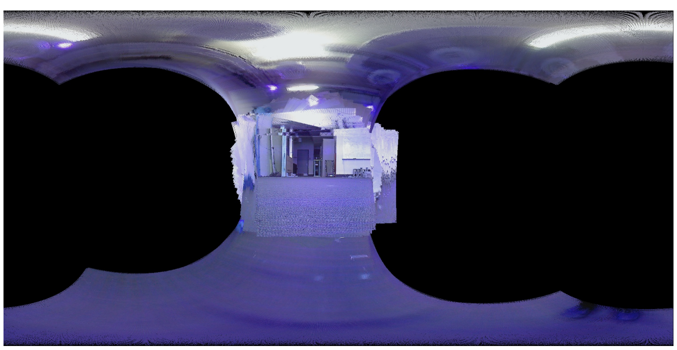

Fig. 21. Panorama - Dataset 2

E. Panorama - Clipped

To eliminate the overlap(Fig 24,25) so that the whole of yaw motion can be captured, the image dataset has been clipped at the start and the end. We can clearly see the yaw motion being captured and the entire panorama being constructed. We can also see that these panoramas are noisy, which should be the case from the orientation predictions earlier.

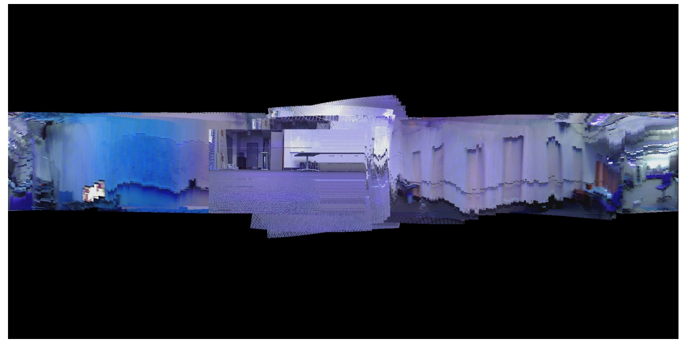

Fig. 22. Panorama - Dataset 8

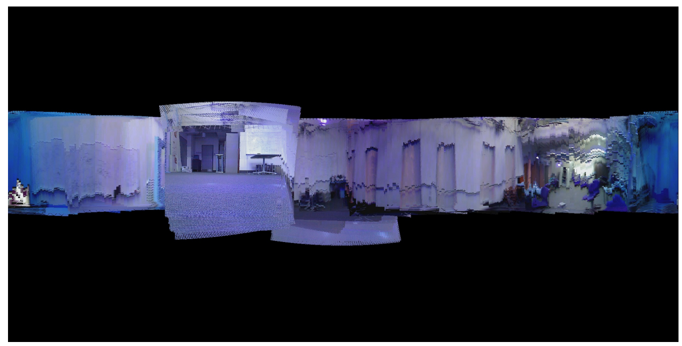

Fig. 23. Panorama - Dataset 9

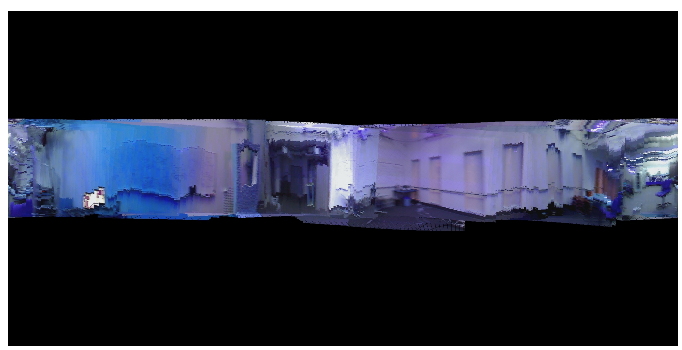

Fig. 24. Panorama - Dataset 8

Fig. 25. Panorama - Dataset 9

F. Test Data Results

1) Optimized Trajectory: In the following plots, the roll, Pitch & Yaw angles from the optimized trajectory are plotted along with Motion model. For the Gradient descent, the learning rate is chosen to be 0.0003 after experiment with multiple values and the algorithm seems to be converging at about 100 iterations for both the datasets.

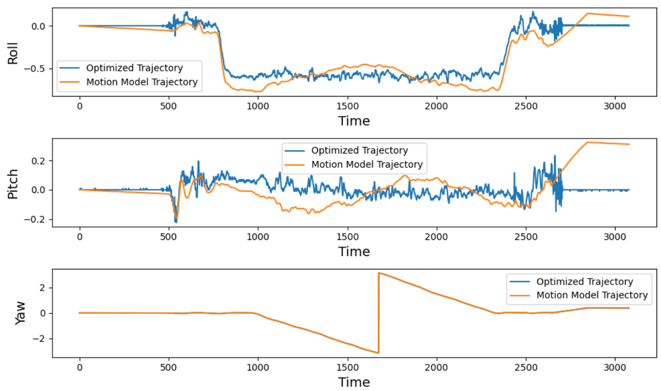

Fig. 26. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 10

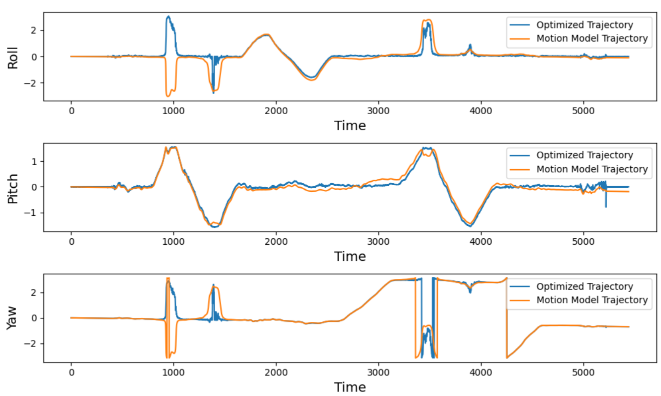

Fig. 27. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 11

2) Optimized Trajectory - Analysis: From the plot of Test dataset 10, it can be observed that there is both Roll(rotated, held constant & rotated back) & Yaw motion. Also the estimated trajectory is quite noisy. The dataset 11 seems to have all kinds of motion but at different time stamps. Hence dataset 10 should produce a tilted panorama(due to the roll), the panorama of dataset 11 should have some part tilted, some part of it at the top & bottom due to the pitch and adjacent images due to yaw motion.

G. Panorama - All Images

The figures 28,29 have been constructed based on the entire image dataset. As predicted Dataset 10 resulted in tilted panorama & dataset 11 has some part of the panorama tilted, some part at the top & bottom and the adjacent images.

Fig. 28. Panorama - Dataset 10

Fig. 29. Panorama - Dataset 11

H. Panorama - Clipped

The figures 30,31 have been constructed based on their clipped image datasets to eliminate pitch motions and capture the yaw motion. From the above plots we can see that dataset 11 has two pitch motions both at the start and the end. For illustration purposes only one of the pitch motions has been clipped.

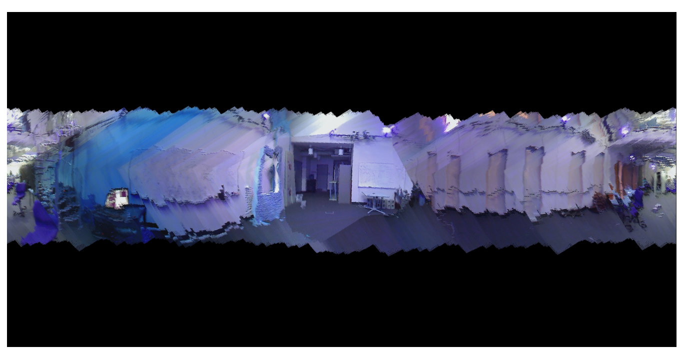

Fig. 30. Panorama - Dataset 10

V. Miscellaneous

A. Initializing with all quaternions as [1,0,0,0]

As discussed earlier, the initialization can be done with all quaternions set to [1,0,0,0] and they should converge to the ground truth with more iterations. The plot in fig.32 for dataset 1 is shown to illustrate the same.

Fig. 31. Panorama - Dataset 11

The value of the cost function at 32 iterations is 387.26 for motion model initialization Vs 444.04 for initialization with unit quaternion.

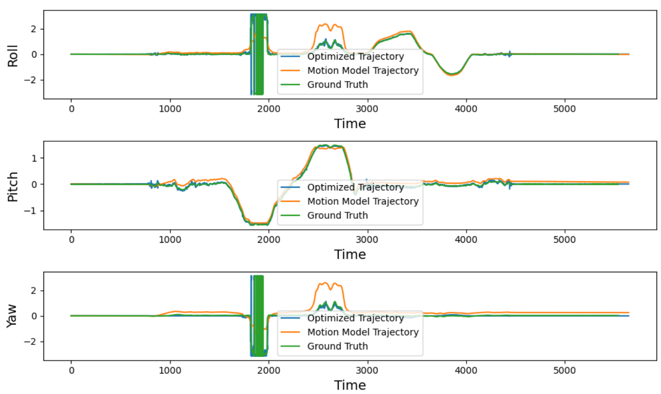

Fig. 32. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 1

B. Changing the weights of cost function terms

Earlier it has been discussed that the weights of motion model error and acceleration are chosen to be equal, to ensure both convergence and to avoid different convention. It can be seen that making the second term zero does not result in any optimization.

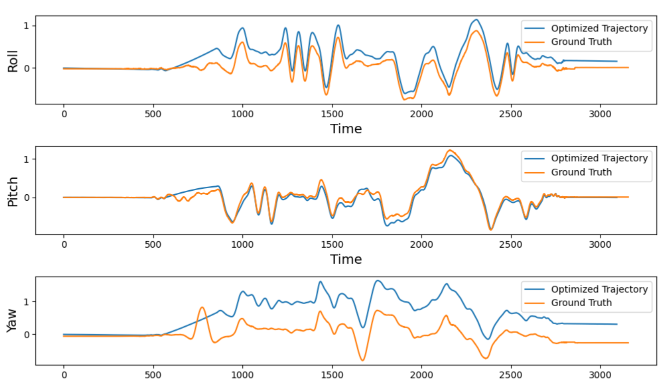

Fig. 33. Optimized Trajectory Vs Motion Model, Ground Truth - Dataset 4

References

[1] https://math.stackexchange.com/questions/54855/gradient-descent-with-constraints
[2] https://jax.readthedocs.io/
[3] Lecture 5 - Factor Graph SLAM - page 8 to 11
[4] https://matplotlib.org/stable/gallery/mplot3d/scatter3d.html

Acknowledgment

I helped my friends Khagesh Bhardwaj and Raghul Sreeram with their doubts & issues in part 1 of the project. The code has not been shared.