Chapter 1

Result From Flight Data

This chapter is divided into three main sections. Firstly, the algorithm's convergence and consistency was analyzed. Secondly, the accuracy of the algorithm is examined by comparing to ground truth data. The third section summarized test results for tuning the algorithm for better accuracy and efficiency. The forth section present the advantage of using IMU data. The fifth section outline the inadequacies of the CC_EKF_SLAM algorithm identified.

To test the performace of CC_EKF_SLAM algorithm, 4 segments of video were selected from the test flight video, and 400 frames were processed in each piece. The filter initialized 40 features at the first frame, and maintain the tracked features amount at this number by initializing new features when existing features moved out of FOV.

Since all parameters are tracked in camera frame, their value is different when viewed from a fixed point in world frame. Therefore, all parameters are converted back to world frame before plotting.

1.1 Convergence and Consistency

1.1.1 Convergence and Tracking

Among the feature parameters, the feature initialization point were initialized to the zero which is the origin of the camera centric coordinate. ϕ and θ were calculated directly from the feature position on image plane, have high accuracy and don't require convergence. The only parameter that goes through a converging process is the features' inverse depth ρ, which were initialized to 0.1 for all features. Figure 1 shows the $1/\rho$ plot for video segment1 over 200 frames. The depth estimators went through rapid changes for several frames after their initialization. Within approximate 20 frames, most estimtors settles to a stable value. The estimated features distance ranged from 400 meters to about 1500 meters, confirming the algorithm's capability for estimating features at great distance. On the other hands, some features take a long time to settle, such as feature 9, 27, ??, while some other never settled, such as 12 and 20.

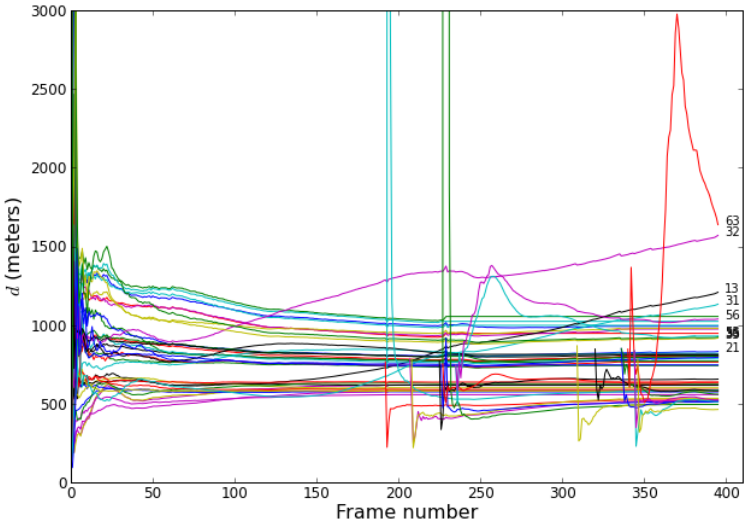

Figure 1: Inverse Depth Convergence

Although the features initialization point $[x_i, y_i, z_i]$, and the deviation-elevation

angle pair $[\phi, \theta]$ did not go through a converging stage, they do get updated and converted into the new camera coordinate using the estimated camera motion at every iteration. As a result, the accuracy of these parameters varies. Ideally, the coordinates should converge to a fixed value. However, the plot indicates a variation as the vehicle travels along. Figure 2 shows the tracking of these parameters over 200 frames. These parameters were stable for about 200 frames. Some features start to drift after 200 frames. This coninside with the introduction of new feature points into the system.

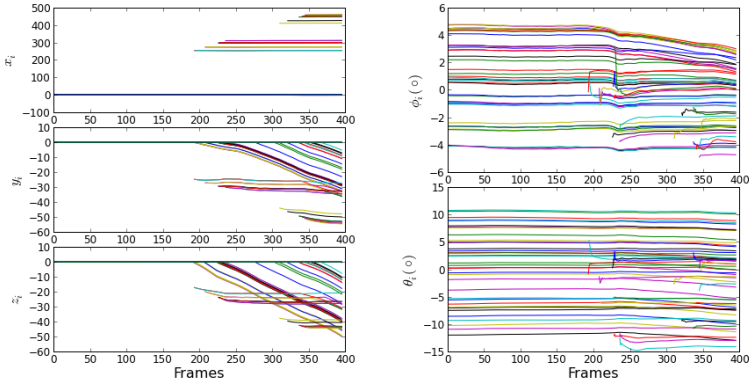

Figure 2: Inverse Depth Convergence

Comparing the pattern of the drift, it is highly correlated to the aircraft pitch and yaw rotation (figure?). The significant of the drift is measured by the maximum drift seen throughout the 200 processed frame, and are listed below (table?). The maximum drift is defined by:

Chapter 2

Error Analysis via Simulation

There are many factors impacts the accuracy in UAS localization and feature position estimation, and they can be sorted into three main categories:

1. Noise in system intrinsic parameters. The system intrinsic parameters includes
   - Camera intrinsic parameters
     - Coordinate of optical center on image plane $[c_x, c_y]$
     - Scaling factor to project feature in 3D world to image plane $[f_x, f_y]$
     - Lens distortion parameters $[k_1, k_2, p_1, p_2]$
   - Image resolution.
   - Accelerometer bias
2. Error introduced by LK tracking algorithm. Reliable vision tracking is an entire field of research in itself. Pyramid implementation of Lucas-Kanade (LK) tracking is used in this work, and there are a number of factors contribute to its performance. Firstly, LK tracking algorithm tracks features by comparing the intensity of a window of the image centered at the feature coordinate in image from one frame to another. The searching process terminates when the sum of error on the windowed image intensity is lower than a value set by user, or

when iteration of search has reached a maximum number set by user. Secondly, as scene evolve from frame to frame, the initial feature appears differently from frame to frame as the viewing distance and angle is different. Thirdly, sudden intensity change in the image sequences significant noise in the tracking. In outdoor setting, intensity change can be introduced by many factors, such as changes of sky area in a image, sun glare, UAV enters or exits cloud shades, or camera auto-adjust its shuttle speed, etc.

3. Error caused by the SLAM algorithm itself. The algorithm estimated features coordinate through a model that represents the relation between UAS location, feature location and UAS motion. As the model is non-linear, the linearization process introduces error into the result.

To better understand the impact of the factors listed above. A simulation is performed to examine the impact item 1 and item 3.

The simulator first generates a 3D point cloud ranging from 100 meters to 3000 meters from the camera (Figure 3). At each frame, the coordinates of the 3D points are first transformed to the new camera frame using the measured UAS motion. Next, the 3D points are projected to the image plane using a camera model defined by $[c_x, c_y, f_x, f_y, k_1, k_2, p_1, p_2]$, and digitized to a resolution of choice.

2.1 An Ideal Case

First of all, understanding the algorithm's performance under no noise, or nearly no noise condition provide a solid ground for the analysis later on. This simulation shows how much error the model itself generates under the most basic flying condition, which is moving forward at constant speed. The low noise environment is configure as such,

- UAS is moving forward (X axis) with constant speed at 60 knots.

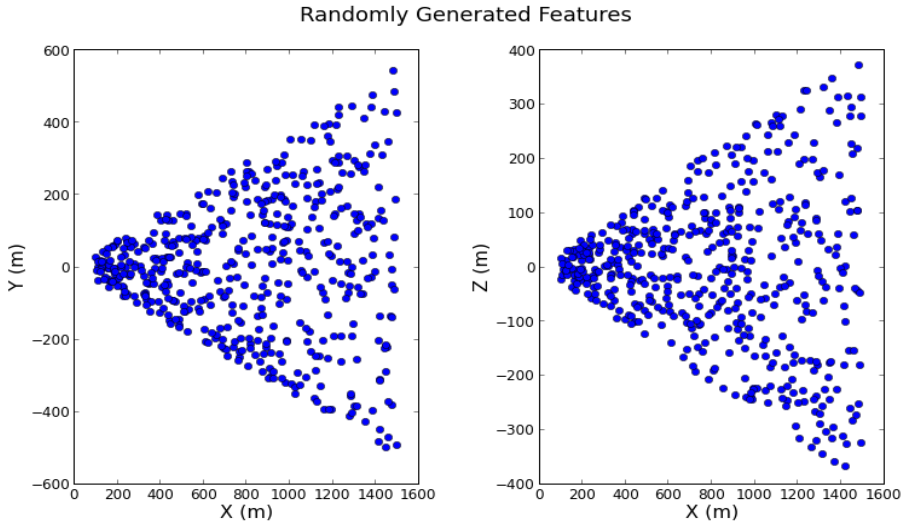

Figure 3: Randomly Generated 3D Feature Points

- Y axis and Z axis translational movement are limited to white noise with standard deviation of 0.08 meters and a mean of 0.
- UAS rotation are modelled by white noise with standard deviation of 0.01 degree and a mean of 0
- No error was introduced due to image digitization. (i.e. the projected feature position on image plane was not digitized to any sensor resolution)
- No error was introduced from camera model mismatch. (i.e. camera model used by simulator is exactly the same as the one used by CC_EKF_SLAM algorithm.

2.1.1 UAS Localization

The estimation of UAS coordinate and orientation is first analyzed, as these estimates are directly used to perform transformation between camera and world frame. Figure 4 plots the UAS translation and rotation against video frame number. The ground truth and estimated value are plotted in blue and green lines respectively. The error

defined by $Estimated - GroundTruth$ is plotted in red line. Under a simple forward only motion, the algorithm tracked the UAS status quite well, with error on translational motion less than 1 cm and error on rotational motion less than 3e-3 degree.

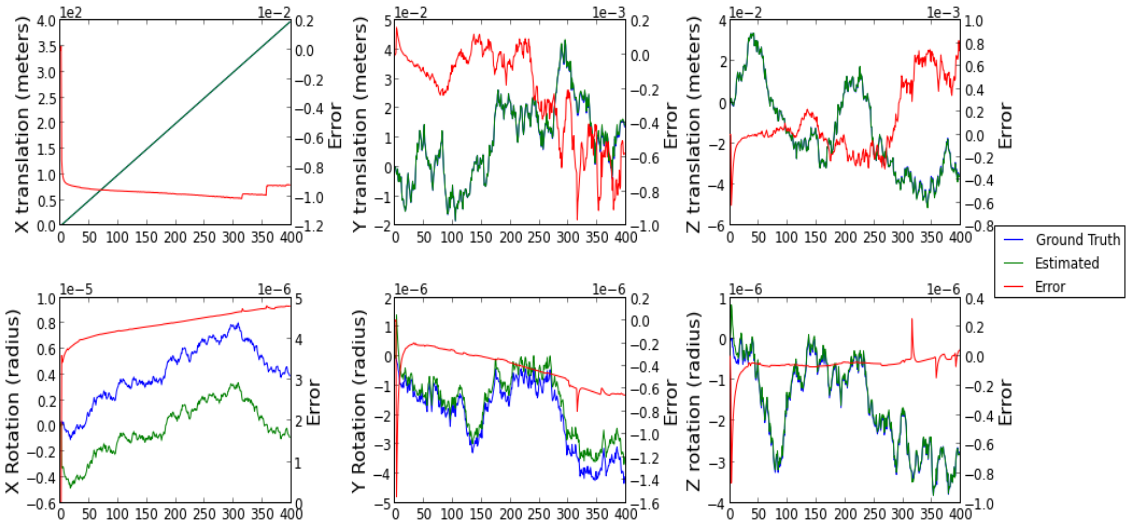

Figure 4: UAS localization error under no noise condition

2.1.2 Features Mapping - Convergence and Accuracy

Figure 5 left shows feature parameters $[d, \varphi, \theta]$ (where $d = 1/\rho$) plotted against frame number for the first 50 frames. The feature depth d for all features converged within 3 frames; elevation-azimuth angles $[\varphi, \theta]$ stay almost constant after initialization. A more detail graph can be seen from the error convergence plot for these parameters (Figure 5 right.) which shows the tracking error of these parameters for 400 frames. The error of feature distance$d$ continues to approach zero as the tracking continues. $[\varphi, \theta]$ show small drift within $+/-0.0002°$ respectively. However, as tracking continue into later frames, error of $[\varphi, \theta]$ gradually grew bigger. The resulting error for feature coordinate in world frame represented in standard Euclidean XYZ parameterization is plotted in ??. The features positions errors in world frame converge to zero as

tracking continues. During the process, certain features moved out of the FOV, and therefore its position estimation remained unchanged since then. At the end of the 400 frames, the x axis position error of the feature reduced to +/-0.2 meters; y and z axis error reduce to +/-0.02 meters

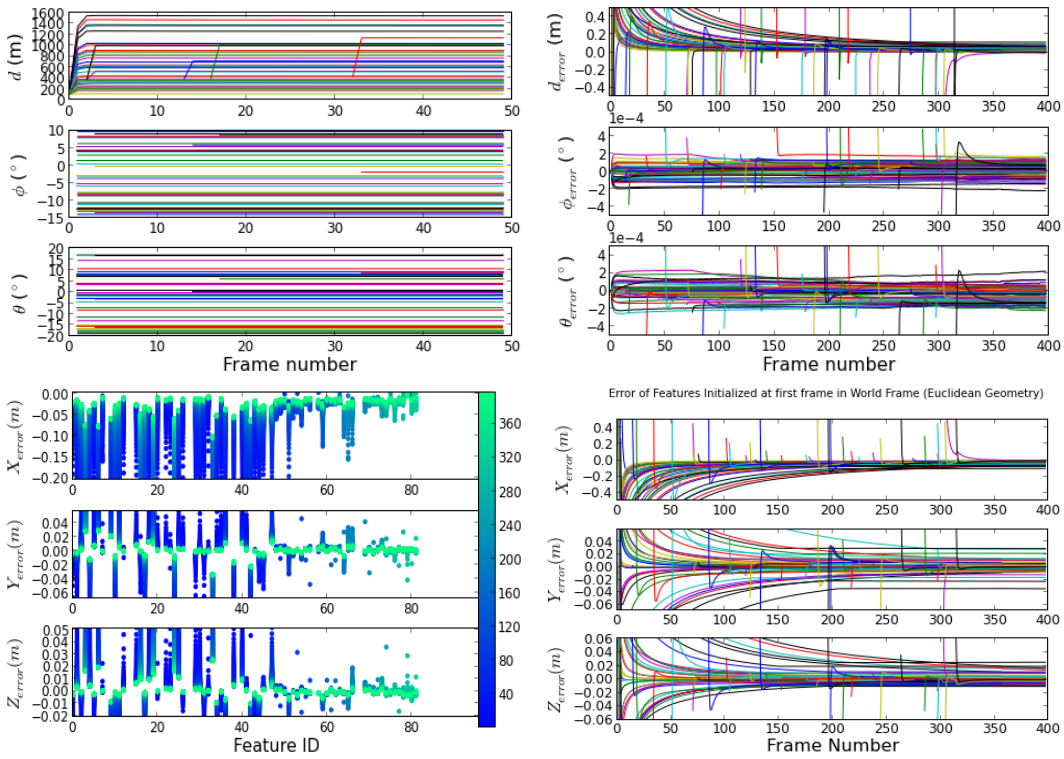

Figure 5: Features parameters and error convergence under no noise condition

2.2 Effect of UAS Motion

The simulation result from UAS forward travel shows that the CC_EKF_SLAM algorithm does feature tracking and self localization quite well under simple UAS motion. Next, the algorithm is tested with a more complex and realistic scenario. A series of motion is added to the simulation in addition to the forward motion. The remaining 5 types of maneuver are added one at a time. These maneuvers are: translation on Y, translation on Z, rotation on X, rotation on Y and rotation on Z. Each motion is

modelled by a sine wave with frequency at 1Hz, and variable amplitude. For translation on Y and Z axis, the sine amplitude varies from 1 meter to 19 meters with 2 meters increments. For X, Y, and Z axis rotation, the amplitude varies from 0.001 radius to 0.018 radius with 0.001 radius increment.

2.2.1 UAS Localization under Motion

Figure 6 shows the UAS localization error statistic under translation motion on Y and Z axis and rotation motion on X, Y, and Z axis. The blue dots mark the mean value μ of the error throughout the tracking, and the error bars mark the standard deviation σ.

The translation motion clearly increases the error of UAS localization. However, the amount of increase is insignificant. With the Sine amplitude increased to 19m, UAS position error increased by less than 0.02 meter.

On the other hand, rotation motions have a big impact on the accuracy of localization. Rotation on X axis has small effect on the accuracy of UAS position and orientation estimate. No obvious increase on mean and standard deviation of the error can be observed. Rotations on Y and Z axis yield significant error increase on the position of the UAS.

- Rotation on Y axis increases the UAS X position mean error, as well as the standard deviation. For Z positioning of the UAS, the mean error stays zero, but standard deviation increase dramatically.

- Same thing happens to the X and Y positioning for rotation on Z axis.

To understand how rotation motion affect the UAS localization estimation, the UAS position on X, Y, Z in world frame are plotted below (figure 7) with rotation amplitude on X, Y and Z set at 0.01 radius. With rotation on X axis, the position

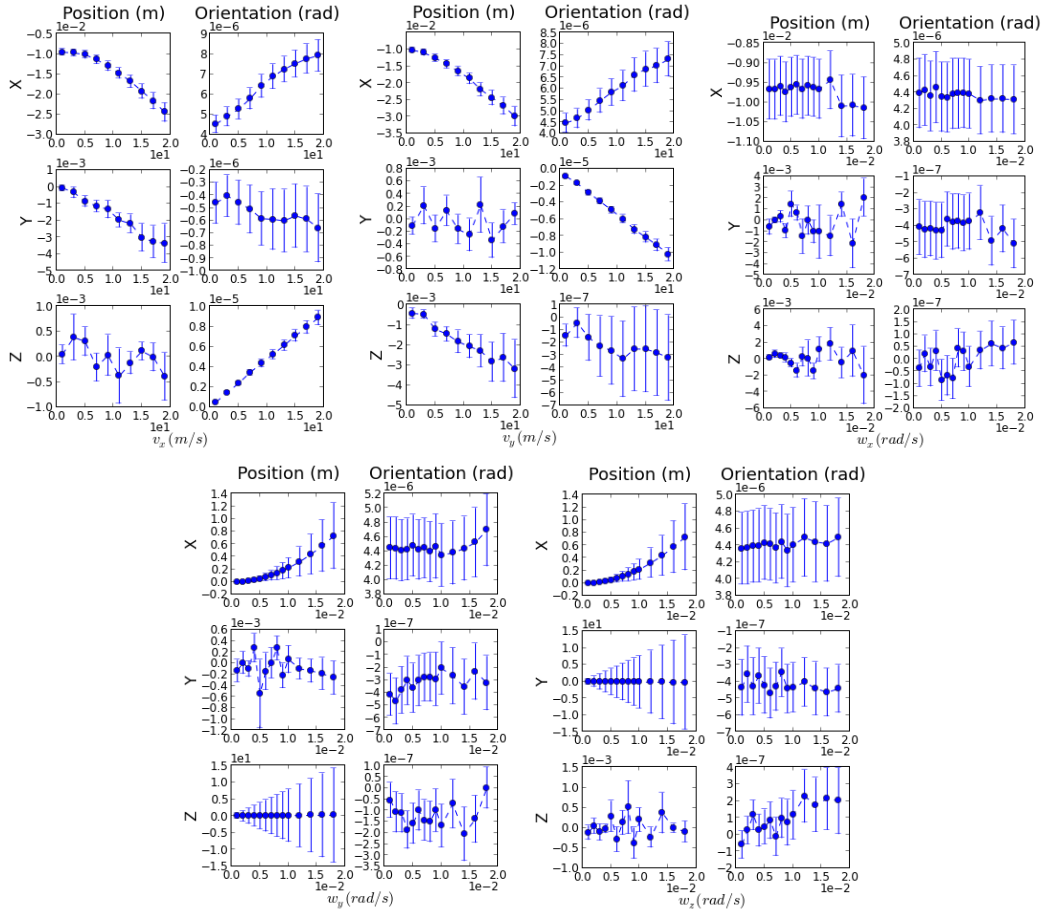

Figure 6: UAS localization error under translational motion

error of UAS shows some oscillation. The oscillation magnitude remains small (in the scale of millimeters) and around zero. For rotation on Y and Z, the situation is very different. Both rotation motions caused the UAS position error to oscillate with the oscillation amplitude increasing (diverging) as tracking goes on. The X position error of the UAS increases in positive value with both rotation on Y and Z, but the most significant impact happens on the Z position (for rotation on Y) and Y position (for rotation on Z), with error reaching 20 meters at the end of the video sequence. With rotation rate increases (amplitude of the sine wave), the rate of error diverging from zero also increases, hence, resulting in an increasing error standard deviation in error statistic plots. This simulation result suggests that CC_EKF_SLAM algorithm

is very sensitive to rotation on Y and Z axis.

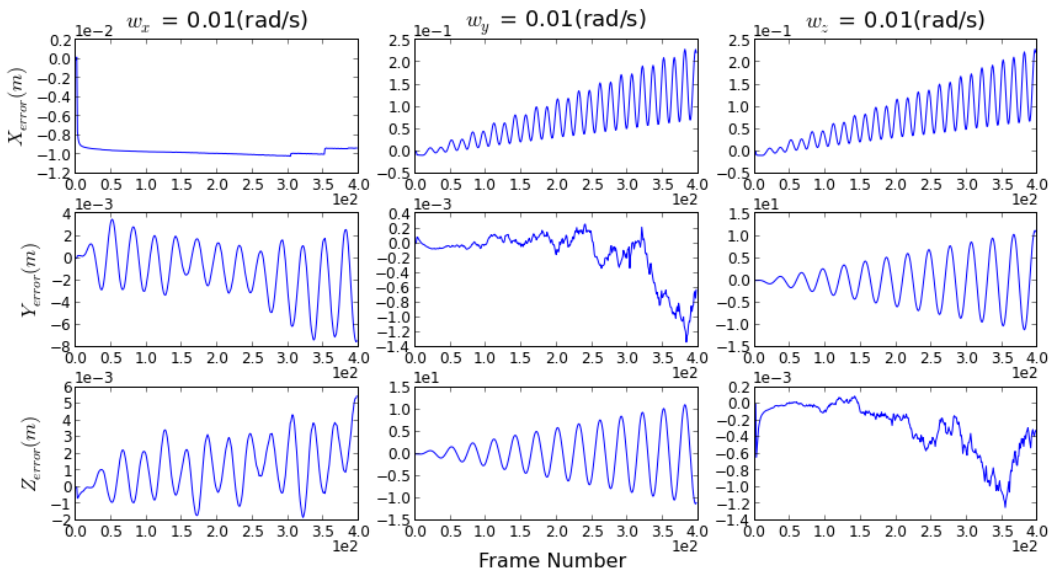

Figure 7: UAS estimated position in world frame

2.2.2 Feature Mapping Accuracy under Motion

Feature mapping error statistic are drawn from the feature error at last frame, since features error converge to zero as tracking goes on. Figure 8 shows the feature mapping error statistic with added motions. Translation motions increase both the error mean and standard deviation, but not by much. With the motion maximum amplitude ranging from 1 meters to 19 meters, the increases of features position error mean and standard deviation are both in the scale of centimeters.

With maximum rate of rotation ranging from 0.001 rad/frame to 0.018 rad/frame

- Rotation on all three axis yields significant error increase for feature position estimation
- X axis rotation causes increase on standard deviation of Y and Z axis feature position error, and by similar amount. The error are in scale of meters with maximum rotation setting.

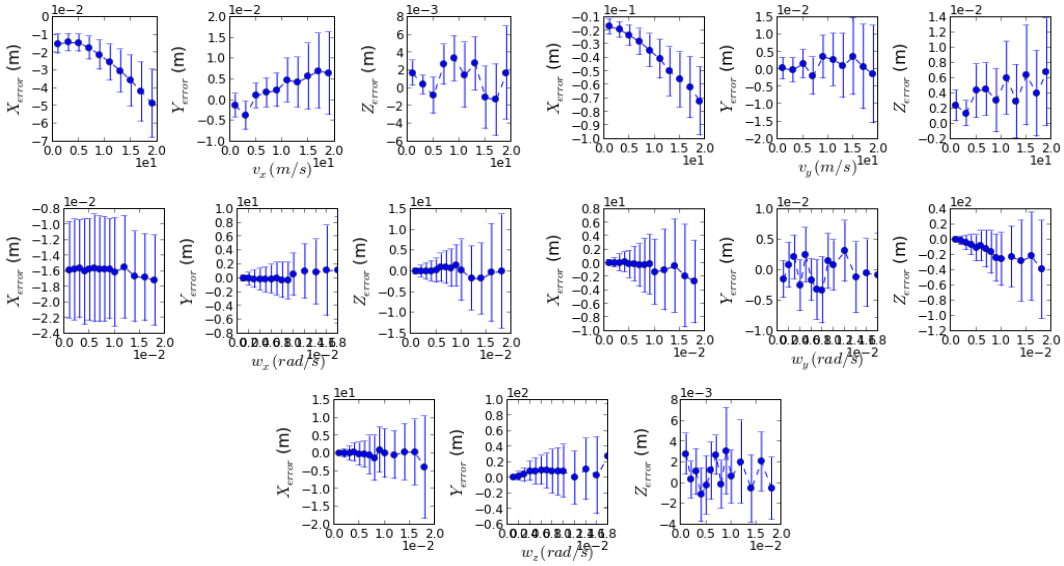

Figure 8: Feature mapping error under added motion

- Y axis rotation causes increase on mean and standard deviation of X and Z axis feature position error. Z axis feature position receives the biggest impact with error in the scale of hundreds of meters with maximum rotation setting

- Z axis rotation impacts on X and Y axis feature position in a similar way as the Y axis rotation.

Figure 9 shows the features position error at the last tracking frame for all three type of rotation motion. The errors are plotted against feature ID and it revealed some more characteristic of the features mapping errors.

- With rotation on Y and Z, the tracked features easily went out of FOV. This can be observed from total number of features went from 80 to over 110 with Y axis rotation setting varied from 0.001rad/s to 0.018rad/s. This caused frequent addition of new features.

- Features added after first frame has much bigger error than features added at first frame. At 1^{st} frame, 40 features added to the filter. Error plots from Y

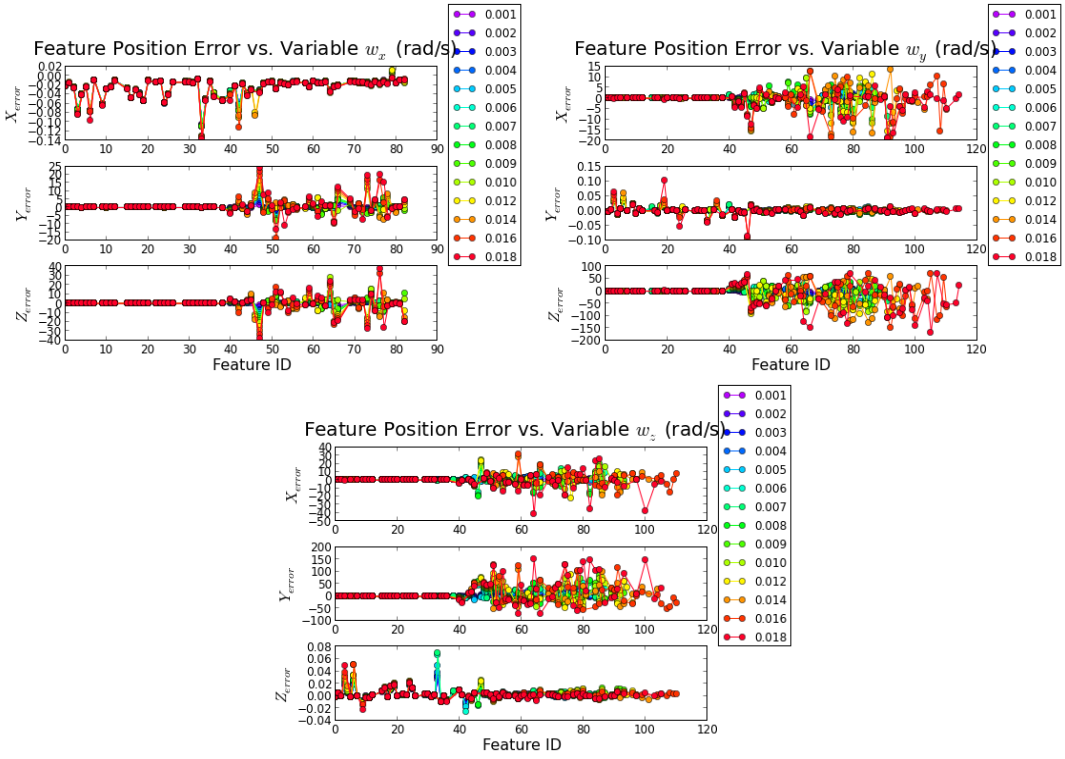

Figure 9: Feature mapping error under rotatioanl motion

and Z axis rotation both shows that major feature mapping error came from features added after the 1 st frame with ID bigger than 40.

To investigate how does rotation motion results in bigger error on features added after first frame, feature parameters error (converted to world frame) with $wy = 0.01$ are plotted in figure 10. It is found that the most significant error happen to parameter ϕ which is the feature elevation angle. This angle has the same definition as rotation angle around Y axis. The second contributor is z_i (the Z axis coordinate of the feature initialization point). The both parameters have an offset error at initialization, and were never corrected throughout the tracking.

Figure 11 shows the ϕ error at initialization in camera frame and world frame. The blue line shows the error in camera frame. The red line shows the error in world frame transformed using the estimated UAS position and orientation. It is clear that

it is the transformation process that introduced the offset error in ϕ. Offset error in z_i is due to the same reason. Recall that with Y axis rotation, UAS localization estimate has biggest error in X and Z axis coordinates estimate. Features initialized at first frame don't carry any offset error is because the transformation process is using the same parameters in both way. During tracking, these features are transformed to the new camera frame using the estimated UAS position and orientation. These are the same estimation being used to transform feature position from camera frame into world frame. Therefore, although the UAS localization estimations are different from the ground truth, feature initialized at first frame were not affected. To conclude, the major contributor for feature mapping error came from error in UAS localization estimation.

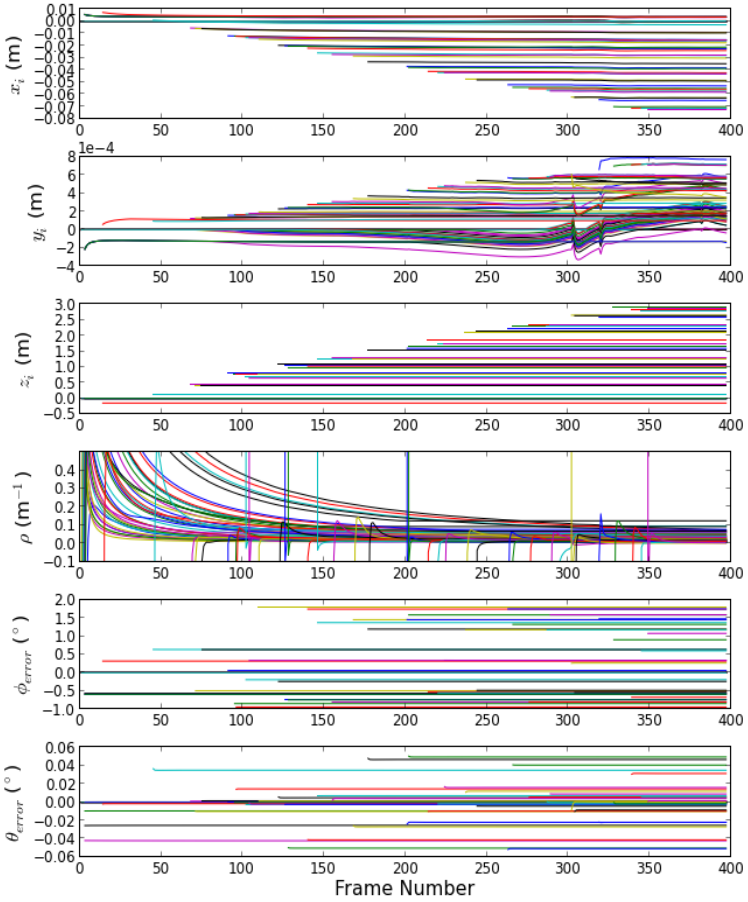

Figure 10: Feature parameters error under rotatioanl motion

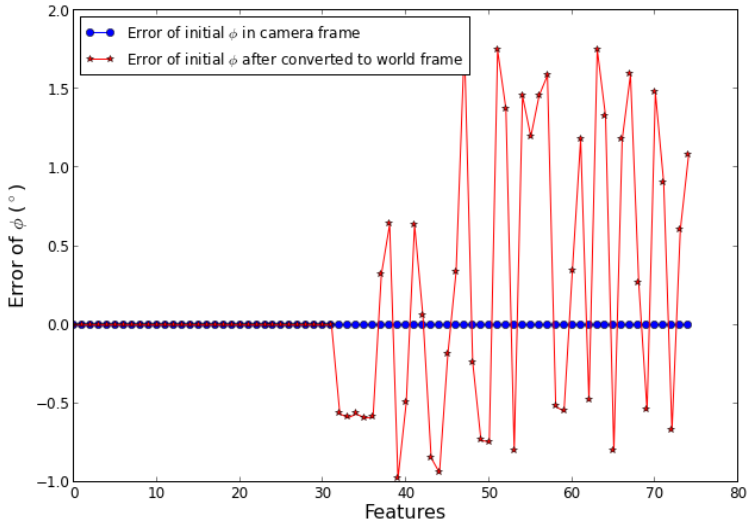

Figure 11: Error of ϕ in camera frame and world frame at initialization

2.3 Camera Intrinsic Parameters

This section summarizes the impact of inaccurate camera parameters estimations. Error on camera intrinsic parameters is simulated by using different values for camera models. One model is used in the simulator to project 3D points onto image plane, and the other is used in the measurement model of CC-LK-SLAM. c_x, c_y, f_x, and f_y are simulated individually and distortion parameters $[k1, k2, p1, p2]$ are simulated as a group. Using the calibrated camera model (see section ???) as a base model,c_x, c_y, f_x, and f_y in simulator camera model varied from -50% to 50% of the base model. Distortion parameters varied from 0% to 140% of the base model.

2.3.1 Effect from (c_x, c_y)

Figure 12 show an over view of UAS localization error statistics with incorrect estimates of (c_x, c_y).

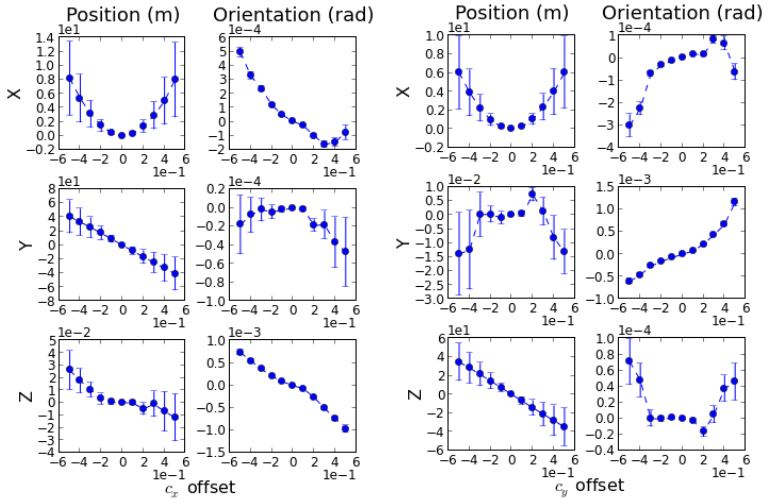

Figure 12: UAS localization error statistic with varying (c_x, c_y)

UAS position error is dependent on (c_x, c_y) and time (figure 13). The UAS position error is diverging (increases in time), and can be modeled by 1 st order polynomial function, with the rate of diverging decided by the error of (c_x, c_y). c_x affects UAS

position on X and Y axis, and c_y affects UAS position on X and Z axis.

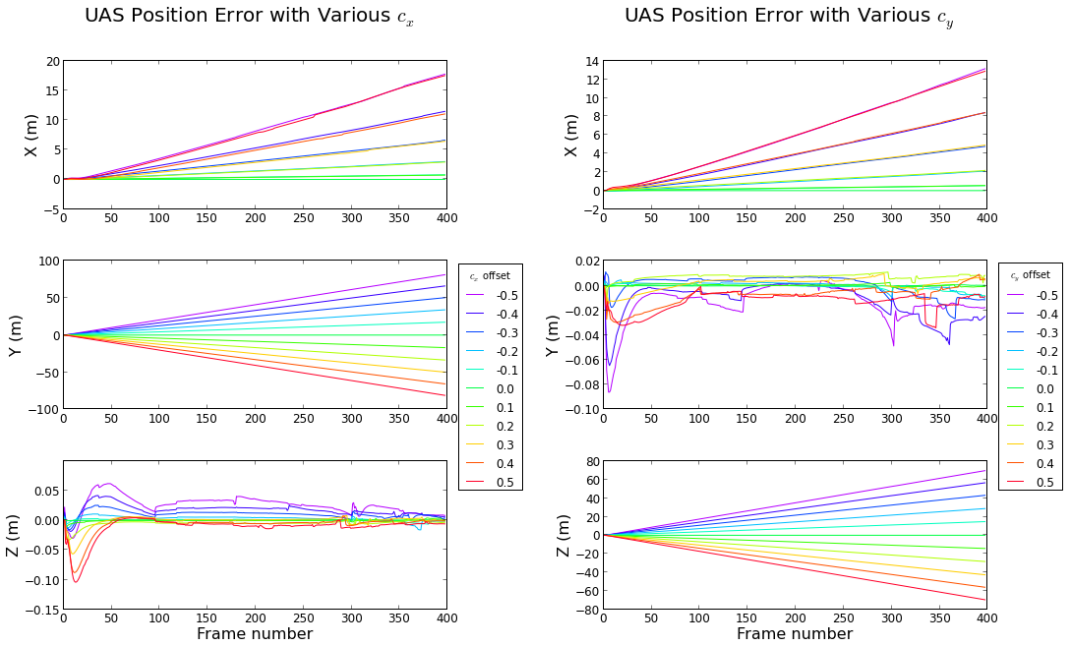

Figure 13: Diverging UAS position error

The feature mapping error statistics from incorrect estimate of (c_x, c_y) are plotted in figure 14. The following characters can be observed from the plots:

- Incorrect c_x affect feature position on all axis, among which, X and Y axis see the most significant error.
  - The further c_x deviate from the true value, the further the features appear (positive X axis error).
  - On Y axis, smaller c_x make feature appear further to the optical axis than ground truth (positive error), and bigger c_x make features appear closer (negative error).
  - Incorrect c_x also affect Z axis feature position, but by a much smaller amount.

- Incorrect c_y affect feature position on all axis similarly to c_x. Estimates on X and Z axis show most amount of error.

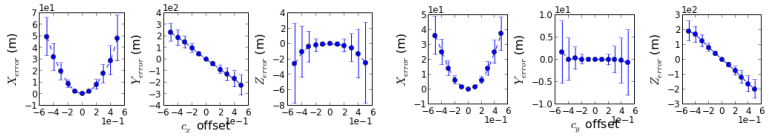

Figure 14: Feature mapping error statistic with varying (c_x, c_y)

Plotting feature position error as a function of features ground truth positions reveals more information on how incorrect c_x and c_y affect feature mapping. Feature position error is a function of its ground truth position, and c_x (or c_y).

- Error on X axis is proportional to the ground truth position on X. The further the feature is, greater the error. The degree of incorrectness in c_x decide the slope of the error plot, greater the error in c_x, steeper the slope (figure 15, plot (a), subplot $[1, 1]$).

- Feature position error on Y axis is also proportional to the ground truth position on X with the slope polarity dependent on the polarity of the error of c_x, and error plot slope dependent on the error of c_x.

- Feature position error on Z axis is proportional to the ground truth position on Z, with slope polarity dependent on the polarity of error of c_x, and slope dependent on the error of c_x.

c_y affects feature position similarly to c_x (figure 15, plot (b)), except feature position error on Z is dependent on ground truth position on X, and error on Y is dependent on the ground truth position on Y.

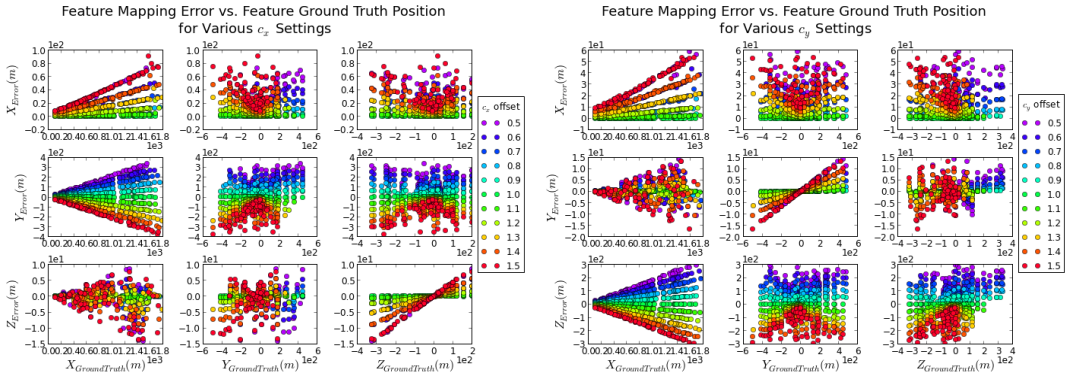

Figure 15: Feature mapping error vs. ground truth feature position

2.3.2 Effect from (f_x, f_y)

With (f_x, f_y) varying from -50% to +50% of the calibrated value, the UAS localization error is shown in 16. For all f_x and f_y settings, UAS position error remained less than +/-0.05 meters, and orientation error remained in less than 8e-6 radius. Compared to the error obtained from the ideal case simulation, Error in (f_x, f_y) estimation does not introduce any additional error into UAS localization estimate.

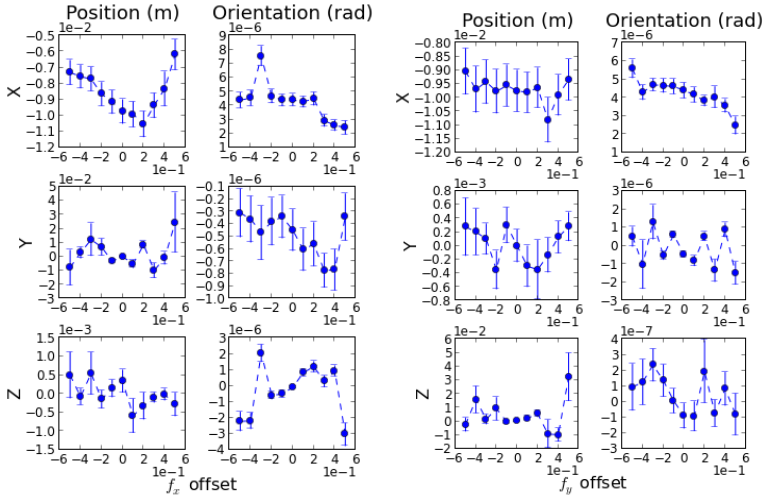

Figure 16: UAS localization error statistic with variable (f_x, f_y)

Feature mapping, on the other hand, is unavoidably affected by the error in (f_x, f_y) since these are the scaling factor that project features from 3D world onto image plane.

Figure 18 shows the error statistic of feature position estimation under variation of (f_x, f_y). The effect on the X axis component is minimum, in the scale of milli meter. Y axis component receive the most impact with unmatched f_x, since this is the scale factor that map feature's Y component in world frame onto U axis on image plane by $u = Y/X * f_x$. Same goes for the Z axis component of the feature position estimate and its relation to f_y.

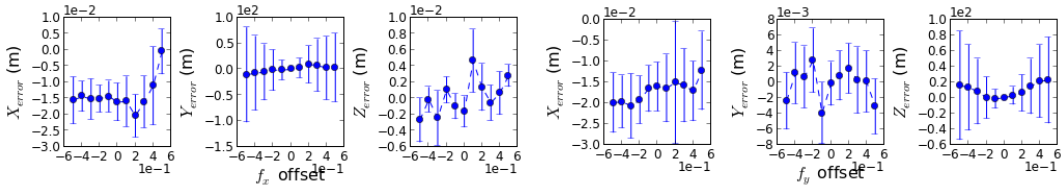

Figure 17: Feature mapping error statistic with variable (f_x, f_y)

Plotting the feature mapping error against feature ground truth position reveals how error in (f_x, f_y) estimate impact on feature position estimate. When f_x estimate contains error, Y component of feature position is determined by the feature's ground truth position on X and Y components. The value of Y_{error} is directly proportional to the Y component of its ground truth position, with the error in f_x determines the function's slope. Same relation can be found for Z_{error} with f_y, and $Z_{groundtruth}$.

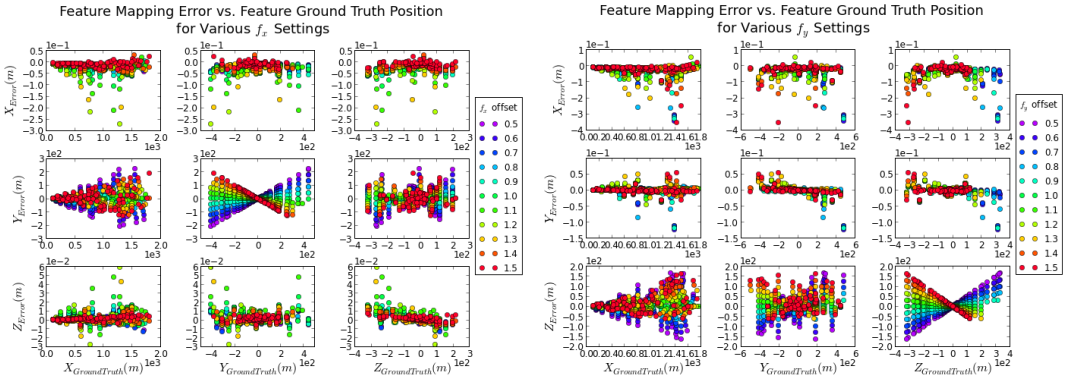

Figure 18: Feature mapping error plotted against feature ground truth for various (f_x, f_y)

2.3.3 Effect from Distortion

The CC_EKF_SLAM algorithm does not consider camera lens distortion at this stage. Therefore, the simulation evaluate the effect of distortion varying from 0% to 150% of the calibrated result to evaluate the amount error resulted by ignoring the lens distortion.

Figure 19 (left) shows the UAS localization error for all lens distortion setting. Ignoring the distortion brings significant error into the UAS localization. UAS X position receives the most impact with error up to 100 meters with increasing standard deviation. Y and Z position shows less error, but standard deviation grows larger with distortion offset.Figure 19 (right) reveals the cause of increasing standard deviation. The UAS position error was growing larger with time. The UAS orientation estimates also experience error increase going from maximum mean error of 5e-6 rad in low noise simulation to 2.5e-4 rad. The standard deviation of orientation estimates also increases with lens distortion offset. Similarly, the orientation error vs. frame number plots show that the error amplitude increase with time, although it fluctuates around zero.

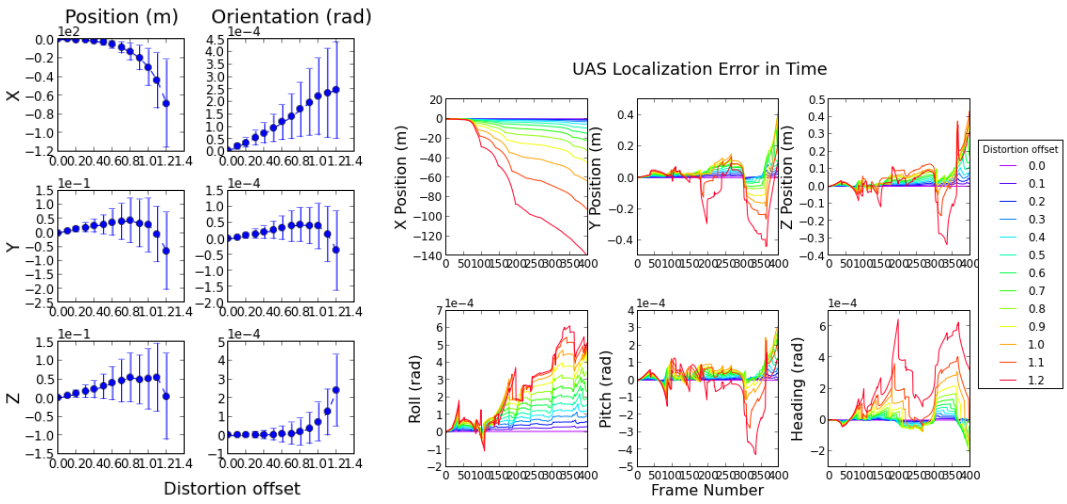

Figure 19: UAS localization error with lens distortion

The feature mapping error statistic is shown in figure 20. The X axis of feature position shows the most error, with mean value reading -400 meters with distortion offset at %150. Y, and Z axis position has less mean error, but the standard deviation is bigger. Plotting feature position error agains feature ground truth coordinate (figure 21) shows that the increasing standard deviation is due to the first order linear relation between the Y (or Z) and the Y (or Z) axis of feature ground truth coordinate, where distortion offset determine the slope of the line.

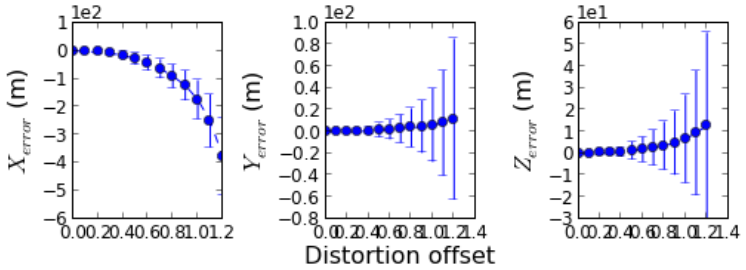

Figure 20: UAS localization error with lens distortion

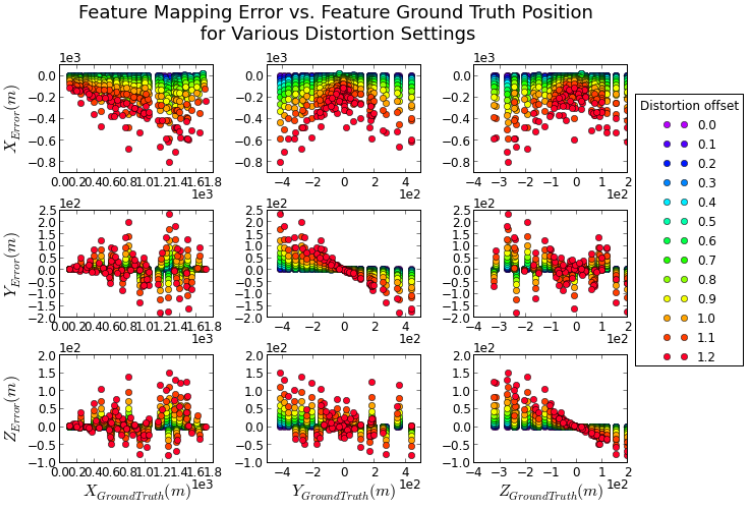

Figure 21: UAS localization error with lens distortion

2.3.4 Effect from Image Resolution

It is well known that higher resolution sensor will give more accuracy to the estimate. To know how high a resolution is good enough for the distance range that this work is targeting, a quantitive analysis is necessary. The ideal case simulation is ran with various image resolution setting, and the result is below (figure 22).

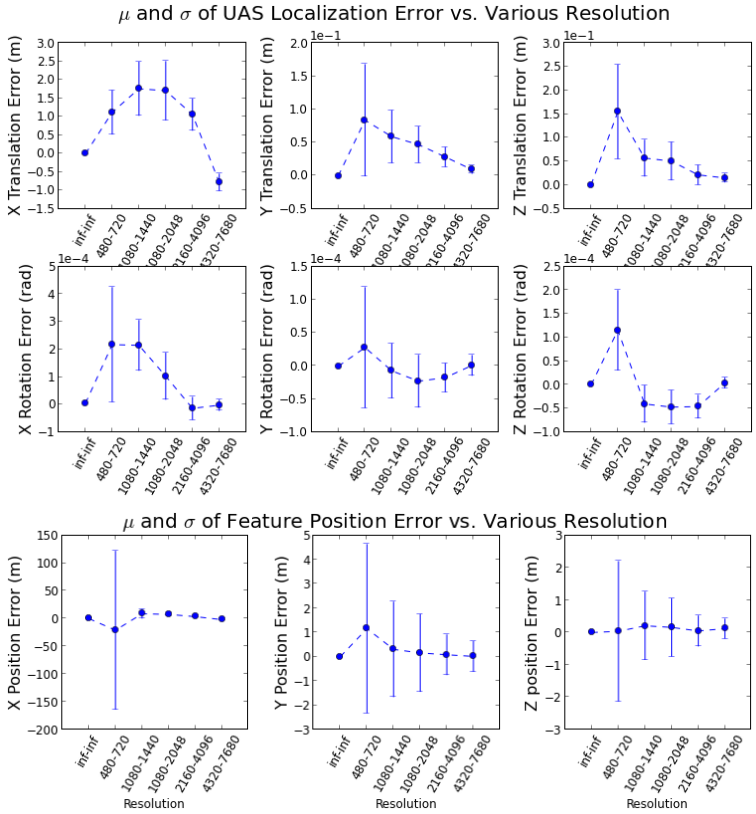

Figure 22: Error statistic for various image resolution

This test confirmed that higher resolution the image sensor is, the more accuracy it will bring. The most significant error is seen at resolution 480x720 where the X axis of feature position error is +/- 150m. At one 720x1080, which is one step up of 480x720, the X axis feature position error is hugely reduced to a few meters. For all other parameters, improvement at each level of resolution increase is nearly linear. This result suggests that to achieve reasonably good accuracy for obstacle detection,

a resolution of 720x1080 or higher is preferred.

List of References

[1] "Athena 111m integrated flight control system." http://www.rockwellcollins.com/sitecore/content/Data/Products/Controls/Flight_Controls/At

[2] "CGIAR-CSI SRTM 90m DEM digital elevation database." http://srtm.csi.cgiar.org/.

[3] "World geodetic system - wikipedia, the free encyclopedia." https://en.wikipedia.org/wiki/World_Geodetic_System.

[4] "pyproj - python interface to PROJ.4 library - google project hosting." http://code.google.com/p/pyproj/.