

Evaluate current Localization accuracy of current NDT scan matching against Urban Localization dataset #5135

meliketanrikulu started this conversation in Show and tell



meliketanrikulu on Aug 28 Collaborator

edited ▾

Introduction

Related Links

The Issue --> [autowarefoundation/autoware.universe#7652](#)

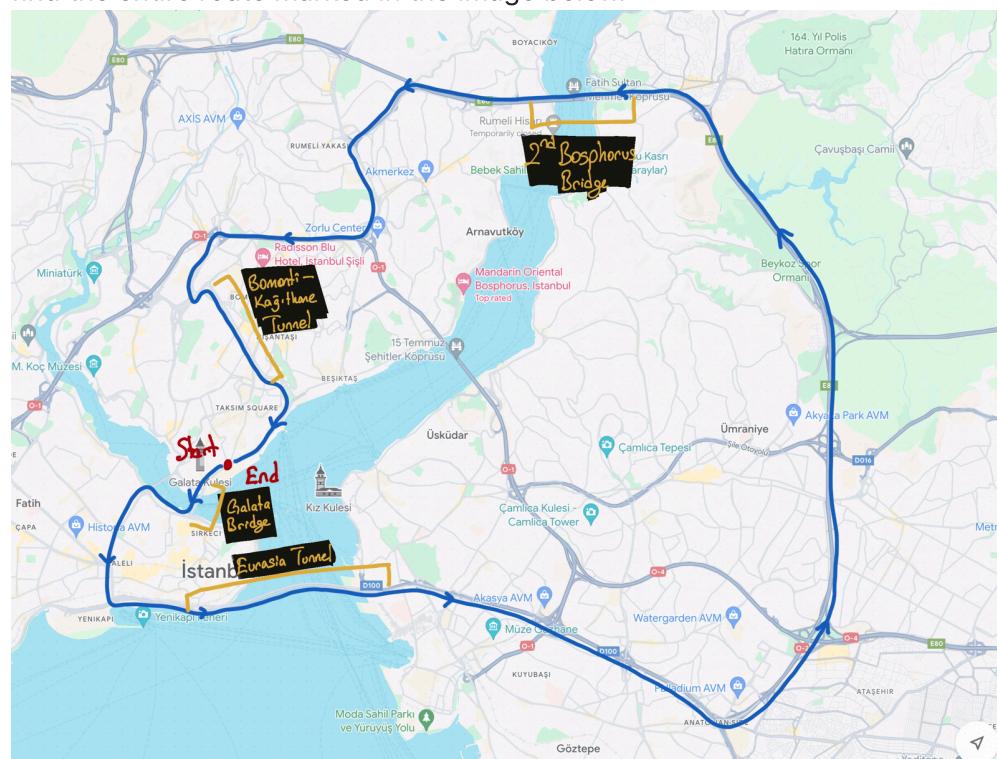
Data Collection Documentation PR --> [autowarefoundation/autoware-documentation#597](#)

Purpose

The purpose of this test is to see the performance of the current NDT based default autoware localization system in an urban environment and evaluate its results. Our expectation is to see the deficiencies of the localization and understand which situations we need to improve.

Test Environment

The test data was collected in Istanbul. It also includes some scenarios that may be challenging for localization, such as tunnels and bridges. You can find the entire route marked in the image below.



Category



Show and tell

Labels

meeting:localizatio...

1 participant



Test Dataset & Map

Test and Mapping datasets are same and the dataset contains data from the portable mapping kit used for general mapping purposes.

The data contains data from the following sensors:

- 1 x Applanix POS LVX GNSS/INS System
- 1 x Hesai Pandar XT32 LiDAR

You can find the data collected for testing and mapping in this [PR document](#).

Note

Since there was no velocity source coming from the vehicle during all these tests, the twist message coming from GNSS/INS was given to ekf_localizer as the linear&angular velocity source.

In order to understand whether this increases the error in cases where the GNSS/INS error increases in the tunnel and how it affects the system, localization in the tunnel was tested by giving only the pose from the NDT, without giving this velocity to ekf_localizer. The video of this test is [here](#).

As seen in the video, when velocity is not given, localization in the tunnel deteriorates more quickly.

It is also predicted that if the IMU Twist message combined (/localization/twist_estimator/twist_with_covariance) with the linear velocity from the vehicle is given instead of the GNSS/INS Twist message, the performance in the tunnel will increase. However, this test cannot be done with the current data.

Expected Tests

- 1.) Firstly, it is aimed to detect the points where the localization is completely broken and to roughly control the localization.
- 2.) By extracting the metrics, it is aimed to see concretely how much the localization error has increased and what the level of performance is.

How to Reproduce Tests

Test With Raw Data

If you test with raw data, you need to follow these [Test With Raw Data](#) instructions. Since you need to repeat all preprocessing operations on the input data in this test, you need to test by switching to the test branches in the sensor kit and individual params repos. You can perform these steps by following the instructions below.

Installation

- 1.) Download and unpack a test map files.
 - You can also download [the map](#) manually.

```
mkdir ~/autoware_ista_map  
gdown --id 1WPWmFCjV7eQee4kyBpmGNlX7awerCPxc -O  
~/autoware_ista_map/
```



Note

You also need to add `lanelet2_map.osm` file to `autoware_ista_map` folder. Since no lanelet file is created for this map at the moment, you can run any `lanelet2_map.osm` file by placing it in this folder.

2.) Download the test rosbag files.

- You can also download [the rosbag file](#) manually.

```
mkdir ~/autoware_ista_data  
gdown --id 1uta5Xr_ftV4jERxPNVqooDvWerK0dn89 -O  
~/autoware_ista_data/
```



Prepare Autoware to Test:

1.) Checkout autoware_launch:

```
cd ~/autoware/src/launcher/autoware_launch/  
git remote add autoware_launch  
https://github.com/meliketanrikulu/autoware\_launch.git  
git remote update  
git checkout evaluate_localization_issue_7652
```



2.) Checkout individual_params:

```
cd ~/autoware/src/param/autoware_individual_params/  
git remote add autoware_individual_params  
https://github.com/meliketanrikulu/autoware\_individual\_params.git  
git remote update  
git checkout evaluate_localization_issue_7652
```



3.) Checkout sample_sensor_kit_launch:

```
cd ~/autoware/src/sensor_kit/sample_sensor_kit_launch/  
git remote add sample_sensor_kit_launch  
https://github.com/meliketanrikulu/sample\_sensor\_kit\_launch.git  
git remote update  
git checkout evaluate_localization_issue_7652
```



4.) Compile updated packages:

```
cd ~/autoware  
colcon build --symlink-install --packages-select
```



```
sample_sensor_kit_launch autoware_individual_params  
autoware_launch common_sensor_launch
```

Launch Autoware:

```
source ~/autoware/install/setup.bash  
ros2 launch autoware_launch logging_simulator.launch.xml  
map_path:=~/autoware_ista_map/ vehicle_model:=sample_vehicle  
sensor_model:=sample_sensor_kit
```



Run Rosbag:

```
source ~/autoware/install/setup.bash  
ros2 bag play ~/autoware_ista_data/rosbag2_2024_09_11-  
17_53_54_0.db3
```



Localization Test Only

If you only want to see the localization performance, follow the [Localization Test Only](#) instructions. For those who only want to perform localization tests, a second test bag file and a separate launch file have been created for this test. You can perform this test by following the instructions below.

Installation

1.) Download and unpack a test map files.

- You can also download [the map](#) manually.

```
mkdir ~/autoware_ista_map  
gdown --id 1WPWmFCjV7eQee4kyBpmGNlX7awerCPxc -O  
~/autoware_ista_map/
```



Note

You also need to add `lanelet2_map.osm` file to `autoware_ista_map` folder. Since no lanelet file is created for this map at the moment, you can run any `lanelet2_map.osm` file by placing it in this folder.

2.)Download the test rosbag files.

- You can also download [the localization rosbag file](#) manually.

```
mkdir ~/autoware_ista_data  
gdown --id 1yEB5j74gPLLbkkf87cuCxUgHXTkgSZbn -O  
~/autoware_ista_data/
```



Prepare Autoware to Test:

1.) Checkout autoware_launch:

```
cd ~/autoware/src/launcher/autoware_launch/  
git remote add autoware_launch  
https://github.com/meliketanrikulu/autoware\_launch.git  
git remote update  
git checkout evaluate_localization_issue_7652
```



2.) Compile updated packages:

```
cd ~/autoware  
colcon build --symlink-install --packages-select autoware_laun...
```



Launch Autoware:

```
source ~/autoware/install/setup.bash  
ros2 launch autoware_launch  
urban_environment_localization_test.launch.xml  
map_path:=~/autoware_ista_map/
```



Run Rosbag:

```
source ~/autoware/install/setup.bash  
ros2 bag play ~/autoware_ista_data/rosbag2_2024_09_12-  
14_59_58_0.db3
```



Test Results

Test 1: Simple Test of Localization

Simply put, how localization works in an urban environment was tested and the results were recorded on video.

Here is the [test video](#)



We can see from video that there is a localization error in the longitudinal axis along the Eurasia tunnel. As expected, NDT based localization does not work properly here. However, since the NDT score cannot detect the distortion here, localization is not completely broken until the end of the tunnel. Localization is completely broken at the exit of the tunnel. I re-initialized the localization after vehicle exit the tunnel.

From this point on, we move on to the bridge scenario. This is one of the bridges connecting the Bosphorus and was the longest bridge on our route. Here too, we thought that NDT-based localization might be disrupted. However, I did not observe any disruption.

After this part there is another tunnel(Kagithane - Bomonti) and localization behaves similar to Eurasia tunnel . However, at the exit of this tunnel, it recovers localization on its own without the need for re-initialization.

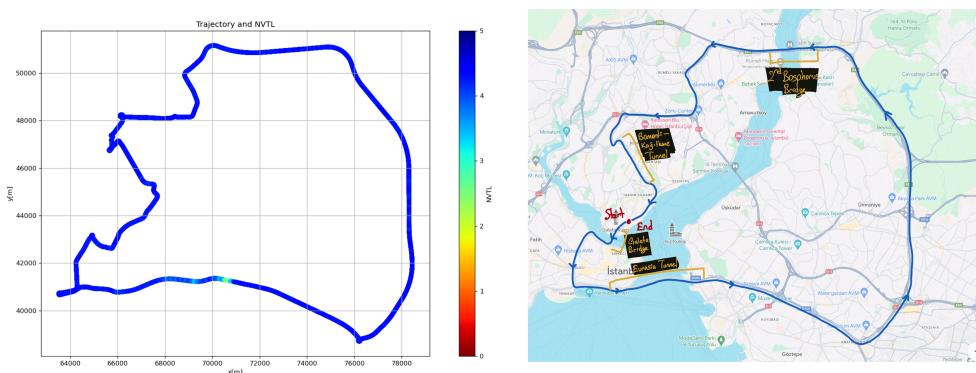
Test 1 Summary :

In summary, the places where there was visible deterioration with our test route were the tunnels. This was already an expected situation. Apart from this, we anticipated that we could have problems on the bridges, but I could not observe any deterioration in the localization on the bridges. Of course, it is useful to remind you at this point that these tests were conducted with a single data.

Test 2: Comparing Results with Ground Truth

1.) How NDT score changed ?

Along the route, only in a small section of the longest tunnel, the Eurasia Tunnel, did the NDT score remain below the expected values. However, there is a visible deterioration in localization in both tunnels on the route. As a result of this test, we saw that we cannot test the problems with the NDT score. You can see how the NDT score changes along the route in the image below. While looking at the visual, let's remember that the NDT score threshold is accepted as 2.3.



NDT Score (Nearest Voxel Transformation Likelihood) Threshold = 2.3

2.) Compare with Ground Truth

Ground Truth : In these tests, the post-processed GNSS / INS data was used as ground truth. Since the error of this ground truth data also decreases in the tunnel environment, it is necessary to evaluate these regions by taking into account the Ground Truth error.

During these tests, I compared the NDT and EKF exposures with Ground Truth and presented the results. I am sharing the test results below as png. However, if you want to examine this data in more detail, I have created an executable file for you to visualize and take a closer look at. You can access this executable file from [here](#). Currently there is only a version that works on Ubuntu at this link, but I plan to add it for Windows as well.

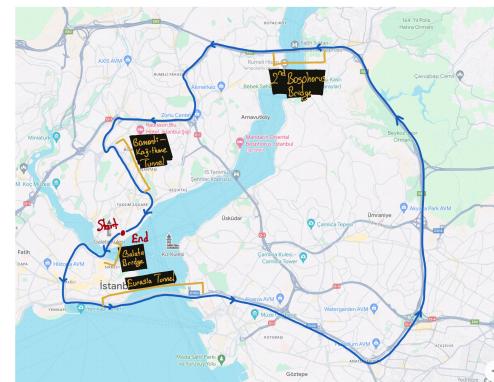
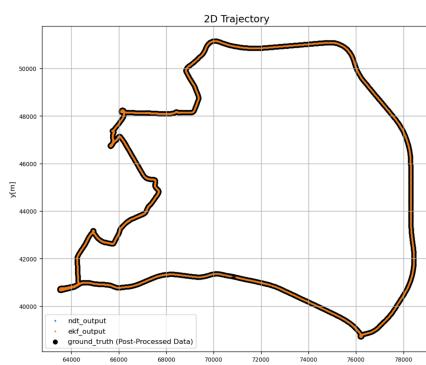
You need to follow these steps:

```
cd /your/path/show_evaluation_ubuntu  
.pose_main
```

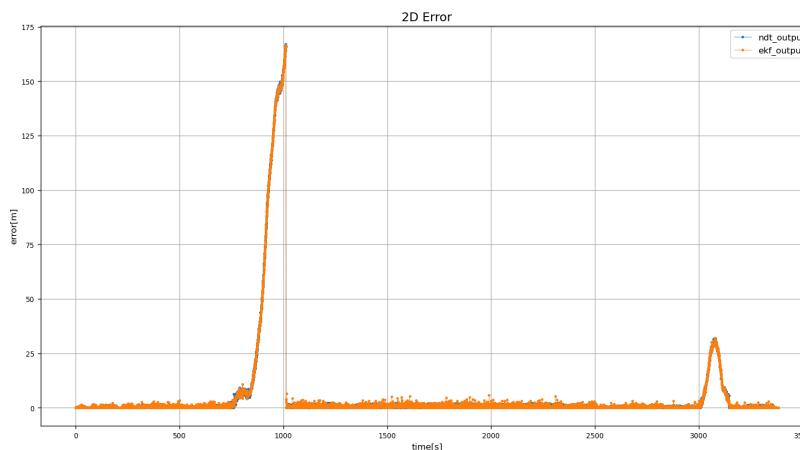


You can also update configurations with changing
/configs/evaluation_pose.yaml

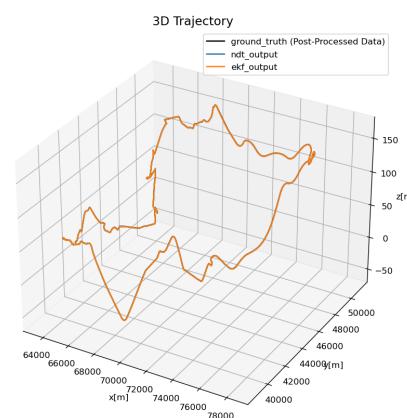
2d Trajectory :



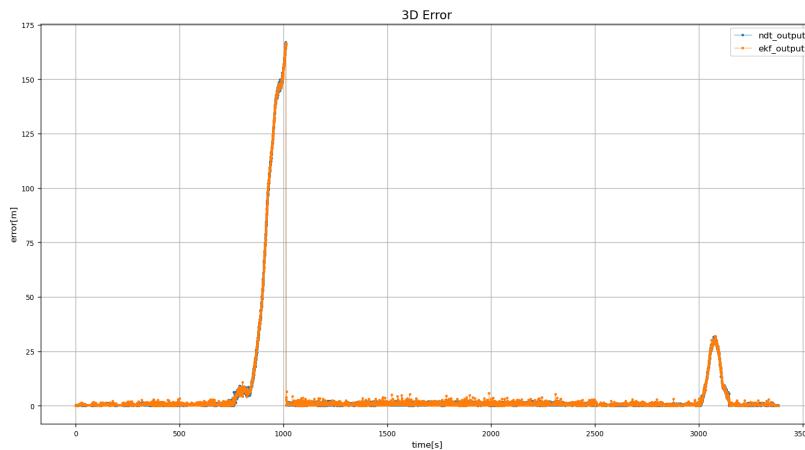
2D Error :



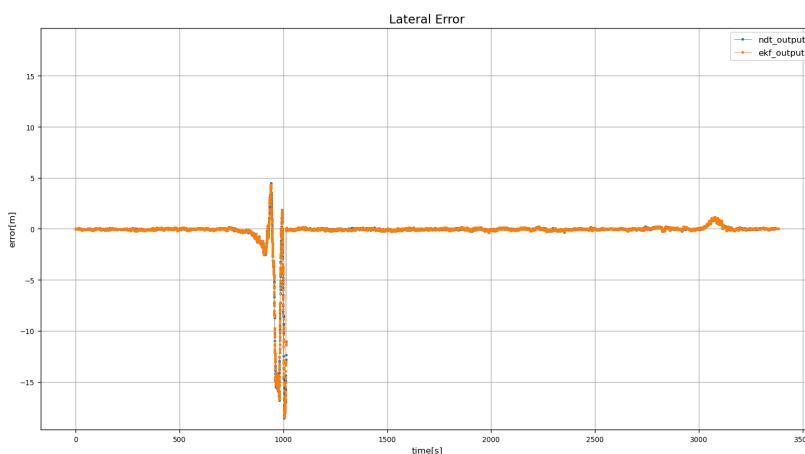
3D Trajectory:



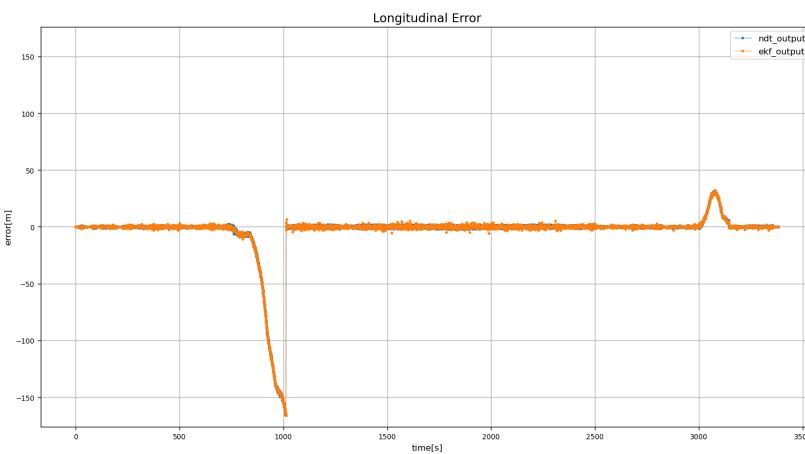
3D Error:



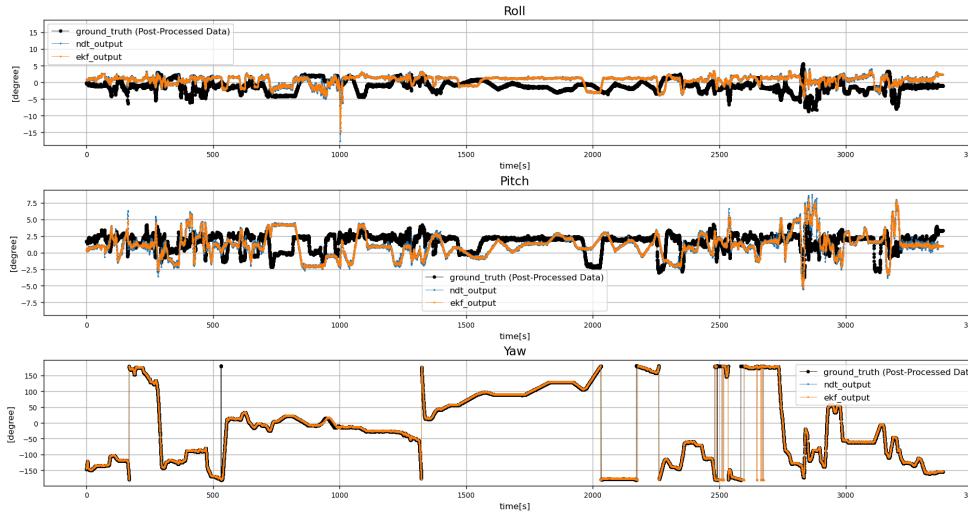
Lateral Error:



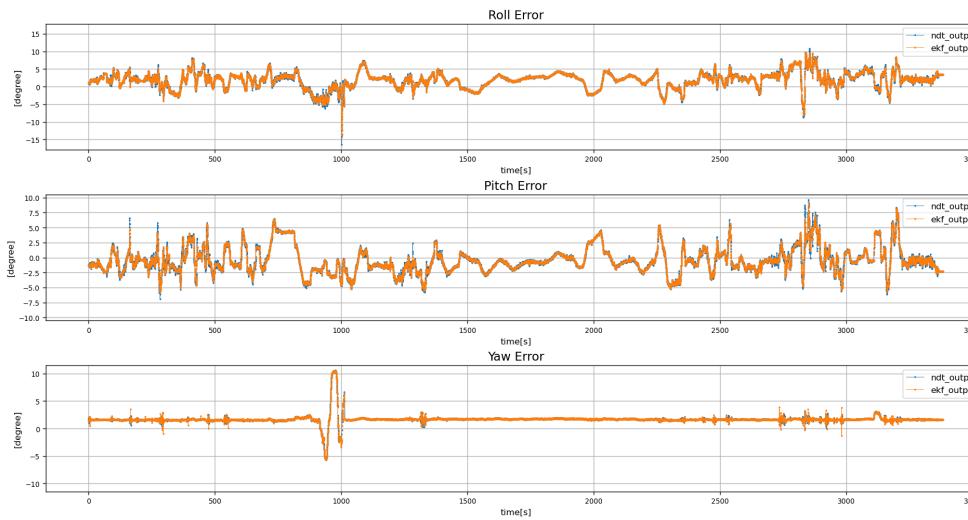
Longitudinal Error:



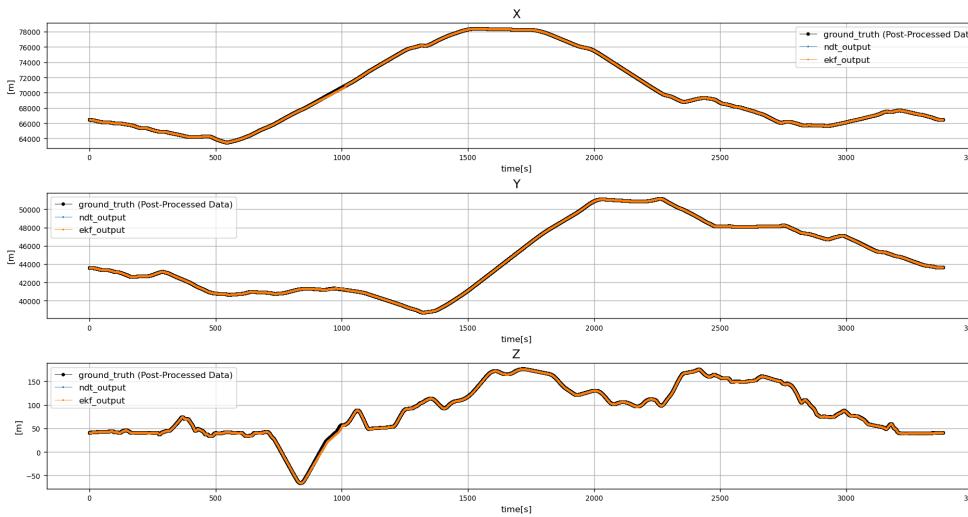
Roll-Pitch-Yaw:



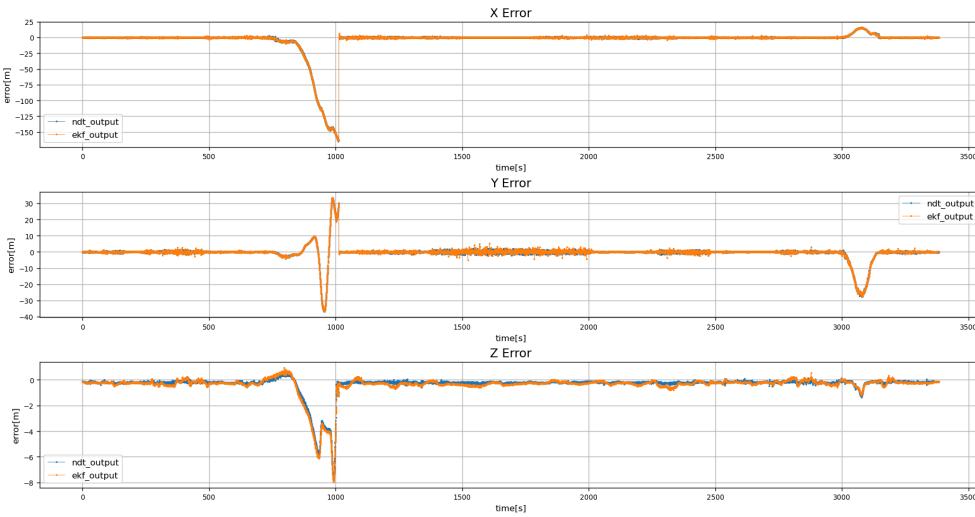
Roll-Pitch-Yaw Error:



X-Y-Z:



X-Y-Z Error:



↑ 1

👍 6

0 comments