vio_benchmark

May 10, 2024

Notebook for two-view reconstruction with inertial data.

This code is written upon vo_benchmark.ipynb.

Author: Jongwon Lee

[1]:
```
import symforce
symforce.set_epsilon_to_symbol()

import cv2
import numpy as np
import matplotlib.pyplot as plt
from pathlib import Path
from scipy.spatial.transform import Rotation as R
import time

import vo, vio
# pose_metrics requires evo (see top of pose_metrics.py for install␣
 ↪instructions)
import pose_metrics
import utils

# Note: this notebook requires pandas in addition to all of Prof. Bretl's␣
 ↪dependencies
import pandas as pd
```

0.0.1 Read data

[2]:
```
# Specify the dataset (should be 'kitti' or 'euroc')
chosen_dataset = 'euroc'
# chosen_dataset = 'kitti'

assert(chosen_dataset in ['kitti', 'euroc'])
```

0.0.2 Provide settings

[3]:
```
# When matching (max threshold for ratio test)
if chosen_dataset == 'euroc':
    matching_threshold = 0.5
else:
    matching_threshold = 0.3

# When deciding if triangulated points are invalid
max_reprojection_err = 0.75

# Temporary folder for evo metrics
temporary_folder = Path('./temp')
temporary_folder.mkdir(parents=True, exist_ok=True)
```

[4]:
```
if chosen_dataset == 'euroc': # Note: euroc takes a bit longer to load.
    # Use EuRoC MAV

    # MAV video folder
    mav_video_folder = Path('./data/mav0')


    # Read MAV data
    dataset_info = utils.read_data_mav(mav_video_folder)
    print("Read dataset with keys: {}".format(sorted(list(dataset_info.
 ↪keys()))))

    # Extract relevant data
    cam0_K = dataset_info['cam0_K']
    cam0_distortion = dataset_info['cam0_distortion']
    visual_inertial_data = dataset_info['visual_inertial_data']

    sigma_acc_wn = dataset_info['imu_accelerometer_noise_density']
    sigma_gyr_wn = dataset_info['imu_gyroscope_noise_density']
    sigma_acc_rw = dataset_info['imu_accelerometer_noise_density']
    sigma_gyr_rw = dataset_info['imu_gyroscope_random_walk']

    dt = 1/200 # IMU frequency

    # Get extrinsics
    T_inB_ofC = dataset_info['cam0_extrinsics']
    T_inC_ofB = np.block([[T_inB_ofC[:3,:3].T, (-T_inB_ofC[:3,:3].T @␣
 ↪T_inB_ofC[:3,-1])[:,np.newaxis]], [np.zeros(3), 1]])

    # Collate
    acc_meas, gyr_meas = utils.imu_collate(dataset_info['visual_inertial_data'])
```

```
    R_inR_ofB, v_inR_ofB, p_inR_ofB, b_a, b_w = utils.
↪groundtruth_collate(dataset_info['visual_inertial_data'], True)

    # As EuRoC's ground-truth (MoCap) is not aligned with gravity (i.e., in␣
↪world frame), we identify the orientation of MoCap frame in world frame
    gravity = np.array([0., 0., -9.81])

    g_inB = - np.mean(acc_meas[:10], axis=0)
    g_inW = gravity

    def align_vectors(g_inB, g_inW):
        # Normalize input vectors
        g_inB_unit = g_inB / np.linalg.norm(g_inB)
        g_inW_unit = g_inW / np.linalg.norm(g_inW)

        # Compute the axis of rotation
        v = np.cross(g_inB_unit, g_inW_unit)

        # Compute the angle of rotation
        cos_theta = np.dot(g_inB_unit, g_inW_unit)
        sin_theta = np.linalg.norm(v)
        theta = np.arctan2(sin_theta, cos_theta)

        v /= np.linalg.norm(v)

        # Compute the rotation matrix
        Rot = R.from_rotvec(theta*v)
        return Rot

    # Compute rotation matrix
    R_inW_ofB = align_vectors(g_inB, g_inW)
    print("Rotation Matrix:\n", R_inW_ofB.as_matrix())

    R_inW_ofB.apply(g_inB)

    R_inW_ofR = R_inW_ofB * R_inR_ofB[:10].mean().inv()
    R_inW_ofB = R_inW_ofR * R_inR_ofB
    v_inW_ofB = R_inW_ofR.apply(v_inR_ofB)
    p_inW_ofB = R_inW_ofR.apply(p_inR_ofB)




else:
    # Use KITTI
    kitti_base_path = './data/kitti'
    kitti_date = '2011_09_26'
```

```
    kitti_drive = '0022'

    # Read KITTI data
    dataset_info = utils.read_data_kitti(kitti_base_path, kitti_date,␣
↪kitti_drive)

    # Extract relevant data
    cam0_K = dataset_info['cam0_K']
    cam0_distortion = dataset_info['cam0_distortion']
    visual_inertial_data = dataset_info['visual_inertial_data']

    T_inC_ofB = dataset_info['cam0_extrinsics']
    R_inB_of_C = T_inC_ofB[:3, :3].T
    t_inB_of_C = R_inB_of_C @ T_inC_ofB[:3, 3]
    T_inB_ofC = np.block([[R_inB_of_C, t_inB_of_C[:,np.newaxis]], [np.zeros(3),␣
↪1]])

    sigma_acc_wn = 1e-4  # accelerometer white noise sigma
    sigma_gyr_wn = 1e-6  # gyroscope white noise sigma
    sigma_acc_rw = 1e-5  # accelerometer random walk sigma
    sigma_gyr_rw = 1e-7  # gyroscope random walk sigma

    # KITTI has data at 10 Hz
    dt = 0.1

    # Collate
    acc_meas, gyr_meas = utils.imu_collate(visual_inertial_data)
    R_inW_ofB, v_inW_ofB, p_inW_ofB, b_a, b_w = utils.
↪groundtruth_collate(visual_inertial_data, False)
```

```
Read dataset with keys: ['cam0_K', 'cam0_distortion', 'cam0_extrinsics',
'imu_accelerometer_noise_density', 'imu_accelerometer_random_walk',
'imu_extrinsics', 'imu_gyroscope_noise_density', 'imu_gyroscope_random_walk',
'visual_inertial_data']
Rotation Matrix:
 [[-0.41384846  0.03948562 -0.90948905]
 [ 0.03948562  0.99889726  0.02539999]
 [ 0.90948905 -0.02539999 -0.41495121]]
```

0.0.3 Create random generator

[5]:
```
rng = utils.create_rng(42)
```

```
seeding RNG with 42
```

0.0.4 Create image keypoint feature extractor

[6]:
```
feature_extractor = cv2.SIFT_create() # could also do ORB_create() for ORB␣
↪features
```

0.0.5 Two view reconstruction

Get initial solution

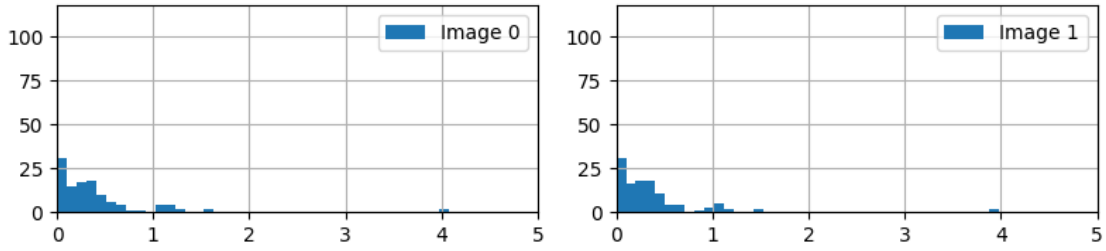
[7]:
```
if chosen_dataset == 'euroc': # Note: euroc takes a bit longer to load.
    # Use EuRoC MAV
    chosen_index = 500
    advance = 100
else:
    chosen_index = 50
    advance = 5


# Get first index closest to chosen index
first_frame_idx = utils.get_index_of_next_image(visual_inertial_data,␣
↪chosen_index)
# Get second index
second_frame_idx = utils.get_index_of_next_image(visual_inertial_data,␣
↪first_frame_idx+advance)

# Create two views
views = [
    vio.create_view_data(utils.
↪read_image(visual_inertial_data[first_frame_idx]['image_file']),
                         first_frame_idx, feature_extractor, cam0_K,␣
↪cam0_distortion),
    vio.create_view_data(utils.
↪read_image(visual_inertial_data[second_frame_idx]['image_file']),
                         second_frame_idx, feature_extractor, cam0_K,␣
↪cam0_distortion)
]

# Perform two-view reconstruction
R_inC_ofB = T_inC_ofB[:3,:3]
p_inC_ofB = T_inC_ofB[:3,-1]

R_inC_ofW = R_inC_ofB @ R_inW_ofB[first_frame_idx].as_matrix().T
p_inC_ofW = - R_inC_ofB @ R_inW_ofB[first_frame_idx].as_matrix().T @␣
↪p_inW_ofB[first_frame_idx] + p_inC_ofB

tic = time.time()
```

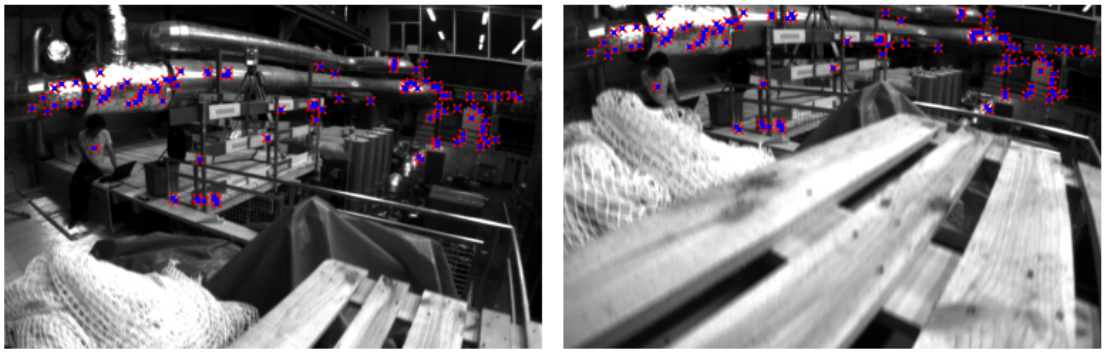
```
tracks = vio.vo_2view(views, matching_threshold, cam0_K, R_inC_ofW, p_inC_ofW,␣
 ↪rng, use_opencv=False)
toc = time.time()

analyctical_guess = toc - tic
print(f"Analytical guess: {analyctical_guess:.2f} [s]")
```

```
found 118 good matches
found 87 inliers
Analytical guess: 2.38 [s]
```

[8]:
```
vio.show_reproj_results(views, tracks, cam0_K, cam0_distortion,␣
 ↪print_raw_reproj=True, show_reproj_histogram=True)
vio.visualize_predictions(views, tracks, cam0_K, cam0_distortion)
```

```
REPROJECTION ERRORS
 Image  0 (  118 points) : (mean, std, max, min) = (0.4617, 0.7235, 5.1047,
0.0069)
 Image (raw reprojection)  0 (  118 points) : (mean, std, max, min) = (0.3597,
0.4605, 2.9180, 0.0043)
 Image  1 (  118 points) : (mean, std, max, min) = (0.4405, 0.7032, 5.0185,
0.0064)
 Image (raw reprojection)  1 (  118 points) : (mean, std, max, min) = (0.2953,
0.3940, 2.6551, 0.0177)
```





Get post-optimization solution

Run below to keep the initial views and tracks

[9]:
```
views_ini = views.copy()
tracks_ini = tracks.copy()
```

Run two-view reconstruction with inertial data

[10]:
```
T_inB0_ofW = np.block([[R_inW_ofB[first_frame_idx].as_matrix().T, -␣
↪R_inW_ofB[first_frame_idx].as_matrix().T @ p_inW_ofB[first_frame_idx][:,np.
↪newaxis]],
                       [np.zeros(3), 1]])
T_inC0_ofW = T_inC_ofB @ T_inB0_ofW

T_inB1_ofW = np.block([[R_inW_ofB[second_frame_idx].as_matrix().T, -␣
↪R_inW_ofB[second_frame_idx].as_matrix().T @ p_inW_ofB[second_frame_idx][:,np.
↪newaxis]],
                       [np.zeros(3), 1]])
T_inC1_ofW = T_inC_ofB @ T_inB1_ofW

T_inC0_ofW = None   # comment this out when you'd like to use ground truth as␣
↪initial guess
T_inC1_ofW = None   # comment this out when you'd like to use ground truth as␣
↪initial guess

tic = time.time()



views, tracks, initial_values, results = vio.vio_nonlinear_optimize(views_ini,␣
↪tracks_ini, acc_meas[first_frame_idx:second_frame_idx],␣
↪gyr_meas[first_frame_idx:second_frame_idx],
                                                                    cam0_K,␣
↪T_inC_ofB, max_reprojection_err,
                                                                    ␣
↪sigma_acc_wn, sigma_gyr_wn, sigma_acc_rw, sigma_gyr_rw, dt,
                                                                    np.
↪mean(b_a, axis=0), np.mean(b_a, axis=0), np.mean(b_w, axis=0), np.mean(b_w,␣
↪axis=0),
                                                                    ␣
↪v_inW_ofB[first_frame_idx], v_inW_ofB[second_frame_idx],
                                                                    T_inC0_ofW,␣
↪T_inC1_ofW)
```

```python
toc = time.time()

nonlinear = toc - tic
print(f"{nonlinear:.2f} [s]")
```

```
[2024-05-10 16:48:44.656] [info] LM<sym::Optimize> [iter    0] lambda:
1.000e+00, error prev/linear/new: 5.635e+06/0.000e+00/2.100e+05, rel reduction:
9.62736e-01
[2024-05-10 16:48:44.673] [info] LM<sym::Optimize> [iter    1] lambda:
1.000e-01, error prev/linear/new: 2.100e+05/0.000e+00/4.042e+04, rel reduction:
8.07514e-01
[2024-05-10 16:48:44.688] [info] LM<sym::Optimize> [iter    2] lambda:
1.000e-02, error prev/linear/new: 4.042e+04/0.000e+00/1.305e+04, rel reduction:
6.77072e-01
[2024-05-10 16:48:44.704] [info] LM<sym::Optimize> [iter    3] lambda:
1.000e-03, error prev/linear/new: 1.305e+04/0.000e+00/7.286e+04, rel reduction:
-4.58155e+00
[2024-05-10 16:48:44.721] [info] LM<sym::Optimize> [iter    4] lambda:
5.000e-03, error prev/linear/new: 1.305e+04/0.000e+00/4.580e+04, rel reduction:
-2.50890e+00
[2024-05-10 16:48:44.737] [info] LM<sym::Optimize> [iter    5] lambda:
2.500e-02, error prev/linear/new: 1.305e+04/0.000e+00/2.498e+04, rel reduction:
-9.13349e-01
[2024-05-10 16:48:44.753] [info] LM<sym::Optimize> [iter    6] lambda:
1.250e-01, error prev/linear/new: 1.305e+04/0.000e+00/1.296e+04, rel reduction:
7.38155e-03
[2024-05-10 16:48:44.768] [info] LM<sym::Optimize> [iter    7] lambda:
1.250e-02, error prev/linear/new: 1.296e+04/0.000e+00/1.320e+04, rel reduction:
-1.90201e-02
[2024-05-10 16:48:44.784] [info] LM<sym::Optimize> [iter    8] lambda:
6.250e-02, error prev/linear/new: 1.296e+04/0.000e+00/1.063e+04, rel reduction:
1.79309e-01
[2024-05-10 16:48:44.800] [info] LM<sym::Optimize> [iter    9] lambda:
6.250e-03, error prev/linear/new: 1.063e+04/0.000e+00/1.810e+04, rel reduction:
-7.02188e-01
[2024-05-10 16:48:44.816] [info] LM<sym::Optimize> [iter   10] lambda:
3.125e-02, error prev/linear/new: 1.063e+04/0.000e+00/1.092e+04, rel reduction:
-2.73505e-02
[2024-05-10 16:48:44.832] [info] LM<sym::Optimize> [iter   11] lambda:
1.563e-01, error prev/linear/new: 1.063e+04/0.000e+00/6.799e+03, rel reduction:
3.60591e-01
[2024-05-10 16:48:44.847] [info] LM<sym::Optimize> [iter   12] lambda:
1.563e-02, error prev/linear/new: 6.799e+03/0.000e+00/1.303e+04, rel reduction:
-9.16905e-01
[2024-05-10 16:48:44.863] [info] LM<sym::Optimize> [iter   13] lambda:
7.813e-02, error prev/linear/new: 6.799e+03/0.000e+00/7.848e+03, rel reduction:
-1.54300e-01
```

```
[2024-05-10 16:48:44.879] [info] LM<sym::Optimize> [iter   14] lambda:
3.906e-01, error prev/linear/new: 6.799e+03/0.000e+00/4.632e+03, rel reduction:
3.18794e-01
[2024-05-10 16:48:44.895] [info] LM<sym::Optimize> [iter   15] lambda:
3.906e-02, error prev/linear/new: 4.632e+03/0.000e+00/4.357e+03, rel reduction:
5.92324e-02
[2024-05-10 16:48:44.911] [info] LM<sym::Optimize> [iter   16] lambda:
3.906e-03, error prev/linear/new: 4.357e+03/0.000e+00/1.322e+03, rel reduction:
6.96547e-01
[2024-05-10 16:48:44.927] [info] LM<sym::Optimize> [iter   17] lambda:
3.906e-04, error prev/linear/new: 1.322e+03/0.000e+00/8.521e+03, rel reduction:
-5.44438e+00
[2024-05-10 16:48:44.943] [info] LM<sym::Optimize> [iter   18] lambda:
1.953e-03, error prev/linear/new: 1.322e+03/0.000e+00/7.022e+02, rel reduction:
4.68942e-01
[2024-05-10 16:48:44.959] [info] LM<sym::Optimize> [iter   19] lambda:
1.953e-04, error prev/linear/new: 7.022e+02/0.000e+00/1.592e+03, rel reduction:
-1.26682e+00
[2024-05-10 16:48:44.974] [info] LM<sym::Optimize> [iter   20] lambda:
9.766e-04, error prev/linear/new: 7.022e+02/0.000e+00/4.152e+02, rel reduction:
4.08716e-01
[2024-05-10 16:48:44.991] [info] LM<sym::Optimize> [iter   21] lambda:
9.766e-05, error prev/linear/new: 4.152e+02/0.000e+00/6.835e+02, rel reduction:
-6.46353e-01
[2024-05-10 16:48:45.007] [info] LM<sym::Optimize> [iter   22] lambda:
4.883e-04, error prev/linear/new: 4.152e+02/0.000e+00/3.562e+02, rel reduction:
1.42079e-01
[2024-05-10 16:48:45.022] [info] LM<sym::Optimize> [iter   23] lambda:
4.883e-05, error prev/linear/new: 3.562e+02/0.000e+00/2.116e+03, rel reduction:
-4.93974e+00
[2024-05-10 16:48:45.038] [info] LM<sym::Optimize> [iter   24] lambda:
2.441e-04, error prev/linear/new: 3.562e+02/0.000e+00/5.382e+02, rel reduction:
-5.10878e-01
[2024-05-10 16:48:45.054] [info] LM<sym::Optimize> [iter   25] lambda:
1.221e-03, error prev/linear/new: 3.562e+02/0.000e+00/3.413e+02, rel reduction:
4.18491e-02
[2024-05-10 16:48:45.071] [info] LM<sym::Optimize> [iter   26] lambda:
1.221e-04, error prev/linear/new: 3.413e+02/0.000e+00/9.442e+02, rel reduction:
-1.76667e+00
[2024-05-10 16:48:45.087] [info] LM<sym::Optimize> [iter   27] lambda:
6.104e-04, error prev/linear/new: 3.413e+02/0.000e+00/3.310e+02, rel reduction:
3.00818e-02
[2024-05-10 16:48:45.103] [info] LM<sym::Optimize> [iter   28] lambda:
6.104e-05, error prev/linear/new: 3.310e+02/0.000e+00/1.413e+03, rel reduction:
-3.26863e+00
[2024-05-10 16:48:45.118] [info] LM<sym::Optimize> [iter   29] lambda:
3.052e-04, error prev/linear/new: 3.310e+02/0.000e+00/4.305e+02, rel reduction:
-3.00378e-01
```

```
[2024-05-10 16:48:45.134] [info] LM<sym::Optimize> [iter   30] lambda:
1.526e-03, error prev/linear/new: 3.310e+02/0.000e+00/2.949e+02, rel reduction:
1.09054e-01
[2024-05-10 16:48:45.150] [info] LM<sym::Optimize> [iter   31] lambda:
1.526e-04, error prev/linear/new: 2.949e+02/0.000e+00/3.516e+02, rel reduction:
-1.92248e-01
[2024-05-10 16:48:45.166] [info] LM<sym::Optimize> [iter   32] lambda:
7.629e-04, error prev/linear/new: 2.949e+02/0.000e+00/2.652e+02, rel reduction:
1.00833e-01
[2024-05-10 16:48:45.181] [info] LM<sym::Optimize> [iter   33] lambda:
7.629e-05, error prev/linear/new: 2.652e+02/0.000e+00/2.332e+02, rel reduction:
1.20532e-01
[2024-05-10 16:48:45.198] [info] LM<sym::Optimize> [iter   34] lambda:
7.629e-06, error prev/linear/new: 2.332e+02/0.000e+00/2.304e+02, rel reduction:
1.22069e-02
[2024-05-10 16:48:45.213] [info] LM<sym::Optimize> [iter   35] lambda:
7.629e-07, error prev/linear/new: 2.304e+02/0.000e+00/3.396e+02, rel reduction:
-4.74315e-01
[2024-05-10 16:48:45.229] [info] LM<sym::Optimize> [iter   36] lambda:
3.815e-06, error prev/linear/new: 2.304e+02/0.000e+00/2.602e+02, rel reduction:
-1.29379e-01
[2024-05-10 16:48:45.244] [info] LM<sym::Optimize> [iter   37] lambda:
1.907e-05, error prev/linear/new: 2.304e+02/0.000e+00/2.001e+02, rel reduction:
1.31438e-01
[2024-05-10 16:48:45.261] [info] LM<sym::Optimize> [iter   38] lambda:
1.907e-06, error prev/linear/new: 2.001e+02/0.000e+00/2.622e+02, rel reduction:
-3.10222e-01
[2024-05-10 16:48:45.277] [info] LM<sym::Optimize> [iter   39] lambda:
9.537e-06, error prev/linear/new: 2.001e+02/0.000e+00/2.117e+02, rel reduction:
-5.80824e-02
[2024-05-10 16:48:45.294] [info] LM<sym::Optimize> [iter   40] lambda:
4.768e-05, error prev/linear/new: 2.001e+02/0.000e+00/1.784e+02, rel reduction:
1.08306e-01
[2024-05-10 16:48:45.310] [info] LM<sym::Optimize> [iter   41] lambda:
4.768e-06, error prev/linear/new: 1.784e+02/0.000e+00/2.156e+02, rel reduction:
-2.08162e-01
[2024-05-10 16:48:45.326] [info] LM<sym::Optimize> [iter   42] lambda:
2.384e-05, error prev/linear/new: 1.784e+02/0.000e+00/1.704e+02, rel reduction:
4.48715e-02
[2024-05-10 16:48:45.342] [info] LM<sym::Optimize> [iter   43] lambda:
2.384e-06, error prev/linear/new: 1.704e+02/0.000e+00/2.065e+02, rel reduction:
-2.11994e-01
[2024-05-10 16:48:45.358] [info] LM<sym::Optimize> [iter   44] lambda:
1.192e-05, error prev/linear/new: 1.704e+02/0.000e+00/1.658e+02, rel reduction:
2.73532e-02
[2024-05-10 16:48:45.374] [info] LM<sym::Optimize> [iter   45] lambda:
1.192e-06, error prev/linear/new: 1.658e+02/0.000e+00/1.664e+02, rel reduction:
-3.83397e-03
```

```
[2024-05-10 16:48:45.390] [info] LM<sym::Optimize> [iter   46] lambda:
5.960e-06, error prev/linear/new: 1.658e+02/0.000e+00/1.598e+02, rel reduction:
3.61191e-02
[2024-05-10 16:48:45.406] [info] LM<sym::Optimize> [iter   47] lambda:
5.960e-07, error prev/linear/new: 1.598e+02/0.000e+00/1.643e+02, rel reduction:
-2.83672e-02
[2024-05-10 16:48:45.423] [info] LM<sym::Optimize> [iter   48] lambda:
2.980e-06, error prev/linear/new: 1.598e+02/0.000e+00/1.632e+02, rel reduction:
-2.12361e-02
[2024-05-10 16:48:45.439] [info] LM<sym::Optimize> [iter   49] lambda:
1.490e-05, error prev/linear/new: 1.598e+02/0.000e+00/1.583e+02, rel reduction:
9.37296e-03
[2024-05-10 16:48:45.455] [info] LM<sym::Optimize> [iter   50] lambda:
1.490e-06, error prev/linear/new: 1.583e+02/0.000e+00/1.639e+02, rel reduction:
-3.57172e-02
[2024-05-10 16:48:45.470] [info] LM<sym::Optimize> [iter   51] lambda:
7.451e-06, error prev/linear/new: 1.583e+02/0.000e+00/1.595e+02, rel reduction:
-7.79826e-03
[2024-05-10 16:48:45.487] [info] LM<sym::Optimize> [iter   52] lambda:
3.725e-05, error prev/linear/new: 1.583e+02/0.000e+00/1.585e+02, rel reduction:
-1.74784e-03
[2024-05-10 16:48:45.503] [info] LM<sym::Optimize> [iter   53] lambda:
1.863e-04, error prev/linear/new: 1.583e+02/0.000e+00/1.556e+02, rel reduction:
1.68761e-02
[2024-05-10 16:48:45.519] [info] LM<sym::Optimize> [iter   54] lambda:
1.863e-05, error prev/linear/new: 1.556e+02/0.000e+00/1.552e+02, rel reduction:
2.33023e-03
[2024-05-10 16:48:45.535] [info] LM<sym::Optimize> [iter   55] lambda:
1.863e-06, error prev/linear/new: 1.552e+02/0.000e+00/1.597e+02, rel reduction:
-2.89658e-02
[2024-05-10 16:48:45.550] [info] LM<sym::Optimize> [iter   56] lambda:
9.313e-06, error prev/linear/new: 1.552e+02/0.000e+00/1.541e+02, rel reduction:
7.22724e-03
[2024-05-10 16:48:45.566] [info] LM<sym::Optimize> [iter   57] lambda:
9.313e-07, error prev/linear/new: 1.541e+02/0.000e+00/1.588e+02, rel reduction:
-3.03125e-02
[2024-05-10 16:48:45.582] [info] LM<sym::Optimize> [iter   58] lambda:
4.657e-06, error prev/linear/new: 1.541e+02/0.000e+00/1.547e+02, rel reduction:
-4.02983e-03
[2024-05-10 16:48:45.599] [info] LM<sym::Optimize> [iter   59] lambda:
2.328e-05, error prev/linear/new: 1.541e+02/0.000e+00/1.555e+02, rel reduction:
-8.91069e-03
[2024-05-10 16:48:45.615] [info] LM<sym::Optimize> [iter   60] lambda:
1.164e-04, error prev/linear/new: 1.541e+02/0.000e+00/1.551e+02, rel reduction:
-6.16305e-03
[2024-05-10 16:48:45.631] [info] LM<sym::Optimize> [iter   61] lambda:
5.821e-04, error prev/linear/new: 1.541e+02/0.000e+00/1.537e+02, rel reduction:
2.93937e-03
```

```
[2024-05-10 16:48:45.647] [info] LM<sym::Optimize> [iter   62] lambda:
5.821e-05, error prev/linear/new: 1.537e+02/0.000e+00/1.533e+02, rel reduction:
2.38572e-03
[2024-05-10 16:48:45.664] [info] LM<sym::Optimize> [iter   63] lambda:
5.821e-06, error prev/linear/new: 1.533e+02/0.000e+00/1.524e+02, rel reduction:
6.05848e-03
[2024-05-10 16:48:45.680] [info] LM<sym::Optimize> [iter   64] lambda:
5.821e-07, error prev/linear/new: 1.524e+02/0.000e+00/1.584e+02, rel reduction:
-3.98386e-02
[2024-05-10 16:48:45.697] [info] LM<sym::Optimize> [iter   65] lambda:
2.910e-06, error prev/linear/new: 1.524e+02/0.000e+00/1.584e+02, rel reduction:
-3.92776e-02
[2024-05-10 16:48:45.713] [info] LM<sym::Optimize> [iter   66] lambda:
1.455e-05, error prev/linear/new: 1.524e+02/0.000e+00/1.519e+02, rel reduction:
3.37002e-03
[2024-05-10 16:48:45.729] [info] LM<sym::Optimize> [iter   67] lambda:
1.455e-06, error prev/linear/new: 1.519e+02/0.000e+00/1.541e+02, rel reduction:
-1.46348e-02
[2024-05-10 16:48:45.746] [info] LM<sym::Optimize> [iter   68] lambda:
7.276e-06, error prev/linear/new: 1.519e+02/0.000e+00/1.520e+02, rel reduction:
-7.35252e-04
[2024-05-10 16:48:45.763] [info] LM<sym::Optimize> [iter   69] lambda:
3.638e-05, error prev/linear/new: 1.519e+02/0.000e+00/1.522e+02, rel reduction:
-1.94649e-03
[2024-05-10 16:48:45.780] [info] LM<sym::Optimize> [iter   70] lambda:
1.819e-04, error prev/linear/new: 1.519e+02/0.000e+00/1.514e+02, rel reduction:
2.95127e-03
[2024-05-10 16:48:45.796] [info] LM<sym::Optimize> [iter   71] lambda:
1.819e-05, error prev/linear/new: 1.514e+02/0.000e+00/1.523e+02, rel reduction:
-5.87373e-03
[2024-05-10 16:48:45.812] [info] LM<sym::Optimize> [iter   72] lambda:
9.095e-05, error prev/linear/new: 1.514e+02/0.000e+00/1.523e+02, rel reduction:
-6.22079e-03
[2024-05-10 16:48:45.828] [info] LM<sym::Optimize> [iter   73] lambda:
4.547e-04, error prev/linear/new: 1.514e+02/0.000e+00/1.503e+02, rel reduction:
7.07919e-03
[2024-05-10 16:48:45.845] [info] LM<sym::Optimize> [iter   74] lambda:
4.547e-05, error prev/linear/new: 1.503e+02/0.000e+00/1.493e+02, rel reduction:
6.79303e-03
[2024-05-10 16:48:45.861] [info] LM<sym::Optimize> [iter   75] lambda:
4.547e-06, error prev/linear/new: 1.493e+02/0.000e+00/1.496e+02, rel reduction:
-2.22871e-03
[2024-05-10 16:48:45.877] [info] LM<sym::Optimize> [iter   76] lambda:
2.274e-05, error prev/linear/new: 1.493e+02/0.000e+00/1.477e+02, rel reduction:
1.08799e-02
[2024-05-10 16:48:45.893] [info] LM<sym::Optimize> [iter   77] lambda:
2.274e-06, error prev/linear/new: 1.477e+02/0.000e+00/1.465e+02, rel reduction:
8.00436e-03
```

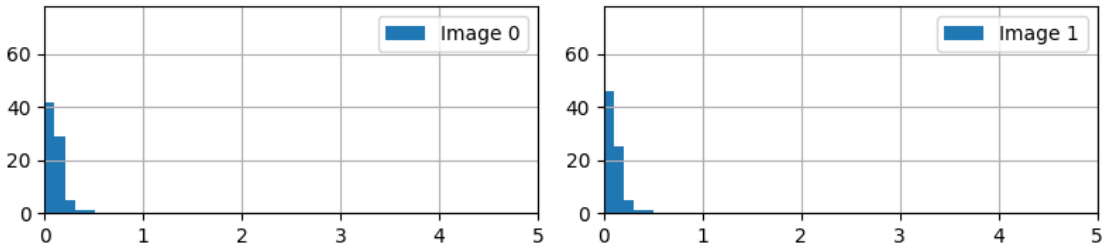
```
[2024-05-10 16:48:45.910] [info] LM<sym::Optimize> [iter   78] lambda:
2.274e-07, error prev/linear/new: 1.465e+02/0.000e+00/1.505e+02, rel reduction:
-2.74900e-02
[2024-05-10 16:48:45.926] [info] LM<sym::Optimize> [iter   79] lambda:
1.137e-06, error prev/linear/new: 1.465e+02/0.000e+00/1.493e+02, rel reduction:
-1.88615e-02
[2024-05-10 16:48:45.942] [info] LM<sym::Optimize> [iter   80] lambda:
5.684e-06, error prev/linear/new: 1.465e+02/0.000e+00/1.452e+02, rel reduction:
8.69882e-03
[2024-05-10 16:48:45.959] [info] LM<sym::Optimize> [iter   81] lambda:
5.684e-07, error prev/linear/new: 1.452e+02/0.000e+00/1.458e+02, rel reduction:
-3.67452e-03
[2024-05-10 16:48:45.975] [info] LM<sym::Optimize> [iter   82] lambda:
2.842e-06, error prev/linear/new: 1.452e+02/0.000e+00/1.468e+02, rel reduction:
-1.07073e-02
[2024-05-10 16:48:45.992] [info] LM<sym::Optimize> [iter   83] lambda:
1.421e-05, error prev/linear/new: 1.452e+02/0.000e+00/1.447e+02, rel reduction:
3.36871e-03
[2024-05-10 16:48:46.008] [info] LM<sym::Optimize> [iter   84] lambda:
1.421e-06, error prev/linear/new: 1.447e+02/0.000e+00/1.445e+02, rel reduction:
2.02639e-03
[2024-05-10 16:48:46.025] [info] LM<sym::Optimize> [iter   85] lambda:
1.421e-07, error prev/linear/new: 1.445e+02/0.000e+00/1.511e+02, rel reduction:
-4.60516e-02
[2024-05-10 16:48:46.042] [info] LM<sym::Optimize> [iter   86] lambda:
7.105e-07, error prev/linear/new: 1.445e+02/0.000e+00/1.444e+02, rel reduction:
2.50158e-04
[2024-05-10 16:48:46.058] [info] LM<sym::Optimize> [iter   87] lambda:
7.105e-08, error prev/linear/new: 1.444e+02/0.000e+00/1.594e+02, rel reduction:
-1.03818e-01
[2024-05-10 16:48:46.074] [info] LM<sym::Optimize> [iter   88] lambda:
3.553e-07, error prev/linear/new: 1.444e+02/0.000e+00/1.443e+02, rel reduction:
7.98166e-04
[2024-05-10 16:48:46.091] [info] LM<sym::Optimize> [iter   89] lambda:
3.553e-08, error prev/linear/new: 1.443e+02/0.000e+00/1.550e+02, rel reduction:
-7.39613e-02
[2024-05-10 16:48:46.108] [info] LM<sym::Optimize> [iter   90] lambda:
1.776e-07, error prev/linear/new: 1.443e+02/0.000e+00/1.468e+02, rel reduction:
-1.70498e-02
[2024-05-10 16:48:46.126] [info] LM<sym::Optimize> [iter   91] lambda:
8.882e-07, error prev/linear/new: 1.443e+02/0.000e+00/1.462e+02, rel reduction:
-1.35039e-02
[2024-05-10 16:48:46.142] [info] LM<sym::Optimize> [iter   92] lambda:
4.441e-06, error prev/linear/new: 1.443e+02/0.000e+00/1.462e+02, rel reduction:
-1.34294e-02
[2024-05-10 16:48:46.159] [info] LM<sym::Optimize> [iter   93] lambda:
2.220e-05, error prev/linear/new: 1.443e+02/0.000e+00/1.430e+02, rel reduction:
9.15717e-03
```

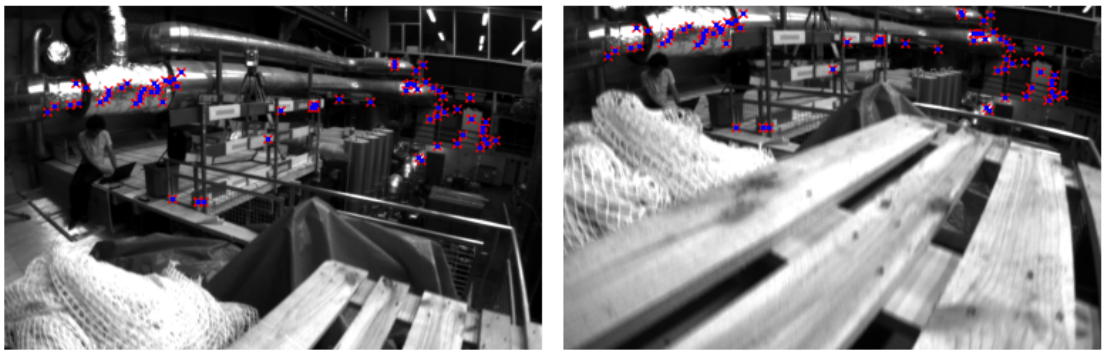
```
[2024-05-10 16:48:46.175] [info] LM<sym::Optimize> [iter   94] lambda:
2.220e-06, error prev/linear/new: 1.430e+02/0.000e+00/1.423e+02, rel reduction:
4.42526e-03
[2024-05-10 16:48:46.192] [info] LM<sym::Optimize> [iter   95] lambda:
2.220e-07, error prev/linear/new: 1.423e+02/0.000e+00/1.427e+02, rel reduction:
-2.28454e-03
[2024-05-10 16:48:46.208] [info] LM<sym::Optimize> [iter   96] lambda:
1.110e-06, error prev/linear/new: 1.423e+02/0.000e+00/1.422e+02, rel reduction:
7.08041e-04
[2024-05-10 16:48:46.225] [info] LM<sym::Optimize> [iter   97] lambda:
1.110e-07, error prev/linear/new: 1.422e+02/0.000e+00/1.438e+02, rel reduction:
-1.12029e-02
[2024-05-10 16:48:46.241] [info] LM<sym::Optimize> [iter   98] lambda:
5.551e-07, error prev/linear/new: 1.422e+02/0.000e+00/1.422e+02, rel reduction:
2.47968e-04
[2024-05-10 16:48:46.258] [info] LM<sym::Optimize> [iter   99] lambda:
5.551e-08, error prev/linear/new: 1.422e+02/0.000e+00/1.442e+02, rel reduction:
-1.42431e-02
[2024-05-10 16:48:46.275] [info] LM<sym::Optimize> [iter  100] lambda:
2.776e-07, error prev/linear/new: 1.422e+02/0.000e+00/1.422e+02, rel reduction:
1.65998e-04
[2024-05-10 16:48:46.291] [info] LM<sym::Optimize> [iter  101] lambda:
2.776e-08, error prev/linear/new: 1.422e+02/0.000e+00/1.423e+02, rel reduction:
-1.15373e-03
[2024-05-10 16:48:46.307] [info] LM<sym::Optimize> [iter  102] lambda:
1.388e-07, error prev/linear/new: 1.422e+02/0.000e+00/1.422e+02, rel reduction:
2.05853e-04
[2024-05-10 16:48:46.324] [info] LM<sym::Optimize> [iter  103] lambda:
1.388e-08, error prev/linear/new: 1.422e+02/0.000e+00/1.441e+02, rel reduction:
-1.35800e-02
[2024-05-10 16:48:46.342] [info] LM<sym::Optimize> [iter  104] lambda:
6.939e-08, error prev/linear/new: 1.422e+02/0.000e+00/1.441e+02, rel reduction:
-1.35101e-02
[2024-05-10 16:48:46.359] [info] LM<sym::Optimize> [iter  105] lambda:
3.469e-07, error prev/linear/new: 1.422e+02/0.000e+00/1.420e+02, rel reduction:
8.17935e-04
[2024-05-10 16:48:46.375] [info] LM<sym::Optimize> [iter  106] lambda:
3.469e-08, error prev/linear/new: 1.420e+02/0.000e+00/1.441e+02, rel reduction:
-1.42828e-02
[2024-05-10 16:48:46.391] [info] LM<sym::Optimize> [iter  107] lambda:
1.735e-07, error prev/linear/new: 1.420e+02/0.000e+00/1.441e+02, rel reduction:
-1.41725e-02
[2024-05-10 16:48:46.408] [info] LM<sym::Optimize> [iter  108] lambda:
8.674e-07, error prev/linear/new: 1.420e+02/0.000e+00/1.431e+02, rel reduction:
-7.18330e-03
[2024-05-10 16:48:46.424] [info] LM<sym::Optimize> [iter  109] lambda:
4.337e-06, error prev/linear/new: 1.420e+02/0.000e+00/1.420e+02, rel reduction:
8.29883e-05
```

```
9.95 [s]
```

[11]:
```
vio.show_reproj_results(views, tracks, cam0_K, cam0_distortion,␣
↪print_raw_reproj=True, show_reproj_histogram=True)
vio.visualize_predictions(views, tracks, cam0_K, cam0_distortion)
```

```
REPROJECTION ERRORS
 Image  0 (   78 points) : (mean, std, max, min) = (0.1119, 0.0871, 0.4758,
0.0013)
 Image (raw reprojection)  0 (   78 points) : (mean, std, max, min) = (0.0934,
0.0720, 0.3628, 0.0024)
 Image  1 (   78 points) : (mean, std, max, min) = (0.1050, 0.0808, 0.4308,
0.0013)
 Image (raw reprojection)  1 (   78 points) : (mean, std, max, min) = (0.1242,
0.0753, 0.3723, 0.0023)
```





[12]:
```
print(f"\nTiming\n{'='*100}")
print(f"Analytical guess: {analyctical_guess:.2f} [s]")
print(f"Non linear (VO): {nonlinear:.2f} [s]")
```

```
Timing
====================================================================================
```

```
====================
Analytical guess: 2.38 [s]
Non linear (VO): 9.95 [s]
```

[13]:
```python
# This highlights the scale ambiguity seen in two-view reconstruction
print("Metric pose difference norm (gt) = {:.3f}".format(np.linalg.norm(
    # Note: while this is in world frame, the frames do not have any scaling,
    # so measuring displacement this way still should provide the correct␣
 ↪distance.
    np.linalg.norm(p_inW_ofB[first_frame_idx] - p_inW_ofB[second_frame_idx])
)))
print("Metric pose difference norm (ini) = {:.3f}".format(np.linalg.
 ↪norm(results.initial_values['T_inC0_ofW'].inverse().t - results.
 ↪initial_values['T_inC1_ofW'].inverse().t)))
print("Metric pose difference norm (sf) = {:.3f}".format(np.linalg.norm(results.
 ↪optimized_values['T_inC0_ofW'].inverse().t - results.
 ↪optimized_values['T_inC1_ofW'].inverse().t)))
```

```
Metric pose difference norm (gt) = 0.323
Metric pose difference norm (ini) = 1.000
Metric pose difference norm (sf) = 0.319
```

(ALL: IGNORE BELOW UNLESS IT'S GOING TO BE REPORTED) Evaluate change in position between frames

[14]:
```python
# Symforce - Initial values
R_inW_ofC0_ini = results.initial_values['T_inC0_ofW'].R.to_rotation_matrix().T
p_inW_ofC0_ini = - R_inW_ofC0_ini @ results.initial_values['T_inC0_ofW'].t

R_inW_ofB0_ini = R_inW_ofC0_ini @ R_inC_ofB
p_inW_ofB0_ini = R_inW_ofC0_ini @ p_inC_ofB + p_inW_ofC0_ini

R_inW_ofC1_ini = results.initial_values['T_inC1_ofW'].R.to_rotation_matrix().T
p_inW_ofC1_ini = - R_inW_ofC1_ini @ results.initial_values['T_inC1_ofW'].t

R_inW_ofB1_ini = R_inW_ofC1_ini @ R_inC_ofB
p_inW_ofB1_ini = R_inW_ofC1_ini @ p_inC_ofB + p_inW_ofC1_ini

# Symforce - Optimized values
R_inW_ofC0_sf = results.optimized_values['T_inC0_ofW'].R.to_rotation_matrix().T
p_inW_ofC0_sf = - R_inW_ofC0_sf @ results.optimized_values['T_inC0_ofW'].t

R_inW_ofB0_sf = R_inW_ofC0_sf @ R_inC_ofB
p_inW_ofB0_sf = R_inW_ofC0_sf @ p_inC_ofB + p_inW_ofC0_sf

R_inW_ofC1_sf = results.optimized_values['T_inC1_ofW'].R.to_rotation_matrix().T
p_inW_ofC1_sf = - R_inW_ofC1_sf @ results.optimized_values['T_inC1_ofW'].t
```

```
R_inW_ofB1_sf = R_inW_ofC1_sf @ R_inC_ofB
p_inW_ofB1_sf = R_inW_ofC1_sf @ p_inC_ofB + p_inW_ofC1_sf

# ground truth
R_inW_ofB0_gt = R_inW_ofB[first_frame_idx].as_matrix()
p_inW_ofB0_gt = p_inW_ofB[first_frame_idx]
v_inW_ofB0_gt = v_inW_ofB[first_frame_idx]

R_inW_ofB1_gt = R_inW_ofB[second_frame_idx].as_matrix()
p_inW_ofB1_gt = p_inW_ofB[second_frame_idx]
v_inW_ofB1_gt = v_inW_ofB[second_frame_idx]
```

Error evaluation on optimized results *w.r.t to Ground Truth*

[15]:
```
p_inW_ofB0toB1_ini = p_inW_ofB1_ini - p_inW_ofB0_ini
R_inB1_ofB0_ini = R_inW_ofB1_ini.T @ R_inW_ofB0_ini

p_inW_ofB0toB1_sf = p_inW_ofB1_sf - p_inW_ofB0_sf
R_inB1_ofB0_sf = R_inW_ofB1_sf.T @ R_inW_ofB0_sf

v_inW_ofB0toB1_gt = v_inW_ofB1_gt - v_inW_ofB0_gt
p_inW_ofB0toB1_gt = p_inW_ofB1_gt - p_inW_ofB0_gt
R_inB1_ofB0_gt = R_inW_ofB1_gt.T @ R_inW_ofB0_gt

print(f"ERROR WRT TO GROUND TRUTH!")
dR_err_ini = R.from_matrix(R_inB1_ofB0_ini.T @ R_inB1_ofB0_gt).as_euler('xyz',␣
 ↪degrees=True)
dR_err_sf = R.from_matrix(R_inB1_ofB0_sf.T @ R_inB1_ofB0_gt).as_euler('xyz',␣
 ↪degrees=True)
print(f'dR: {dR_err_ini} (init.) --> {dR_err_sf} (optm.) [deg] (xyz)')
print('dR scalar: {:.5f} (init.) --> {:.5f} (optm.) [deg]'.format(
    pose_metrics.rotational_error(R_inB1_ofB0_ini, R_inB1_ofB0_gt),
    pose_metrics.rotational_error(R_inB1_ofB0_sf, R_inB1_ofB0_gt),
))

dp_err_ini = p_inW_ofB0toB1_ini - p_inW_ofB0toB1_gt
dp_err_sf = p_inW_ofB0toB1_sf - p_inW_ofB0toB1_gt
print(f'dp: {np.linalg.norm(dp_err_ini):.2f} (init.) --> {np.linalg.
 ↪norm(dp_err_sf):.2f} (optm.) [m]')
```

```
ERROR WRT TO GROUND TRUTH!
dR: [-0.20324676  0.90124231 -0.25614932] (init.) --> [ 0.03977598 -0.18582176
0.1081884 ] (optm.) [deg] (xyz)
dR scalar: 0.95830 (init.) --> 0.21870 (optm.) [deg]
dp: 0.68 (init.) --> 0.00 (optm.) [m]
```

Relative change in position (Δp) and rotation (ΔR) between frames (**NOT ERROR w.r.t Ground Truth**)

[16]:
```
print(f"RELATIVE CHANGE BETWEEN FRAMES (NOT ERROR)\n")
print(f'dp\n{"="*50}')
print('(Analytical guess) dp: {:.2f} [m]'.format( np.linalg.norm(␣
 ↪p_inW_ofB0_ini -  p_inW_ofB1_ini)))
print('(Non linear - VIO) dp: {:.2f} [m]'.format( np.linalg.norm( p_inW_ofB0_sf␣
 ↪ -  p_inW_ofB1_sf)))
print('(Ground Truth)     dp: {:.2f} [m]'.format( np.linalg.norm( p_inW_ofB0_gt␣
 ↪ -  p_inW_ofB1_gt)))

print(f'\ndR\n{"="*50}')
print('(Analytical guess) dR scalar: {:.5f} [deg]'.format(pose_metrics.
 ↪rotational_error( R_inW_ofB0_ini, R_inW_ofB1_ini)))
print('(Non linear - VIO) dR scalar: {:.5f} [deg]'.format(pose_metrics.
 ↪rotational_error( R_inW_ofB0_sf,  R_inW_ofB1_sf)))
print('(Ground Truth)     dR scalar: {:.5f} [deg]'.format(pose_metrics.
 ↪rotational_error( R_inW_ofB0_gt,  R_inW_ofB1_gt)))
```

```
RELATIVE CHANGE BETWEEN FRAMES (NOT ERROR)

dp
==================================================
(Analytical guess) dp: 1.00 [m]
(Non linear - VIO) dp: 0.32 [m]
(Ground Truth)     dp: 0.32 [m]

dR
==================================================
(Analytical guess) dR scalar: 6.88214 [deg]
(Non linear - VIO) dR scalar: 8.02001 [deg]
(Ground Truth)     dR scalar: 7.82089 [deg]
```

(Optional) Error evaluation on preintegrated results

[17]:
```
gravity = results.initial_values['gravity']
dt_01 = results.initial_values['dt_01']
dR_01 = results.initial_values['dR_01']
dv_01 = results.initial_values['dv_01']
dp_01 = results.initial_values['dp_01']

print(f"Results\n{'='*100}")
dR_err_pi = R.from_matrix(dR_01 @ R_inB1_ofB0_gt).as_euler('xyz', degrees=True)
print(f'dR_preintegrated: {dR_err_pi} [deg] (xyz)')
print('dR_preintegrated scalar: {:.5f} [deg]'.format(
    pose_metrics.rotational_error(dR_01, R_inB1_ofB0_gt)
))

v_inW_ofB0toB1_pi = R_inW_ofB0_gt @ dv_01 + gravity * dt_01
```

```
dv_err_pi = v_inW_ofB0toB1_pi - v_inW_ofB0toB1_gt

print(f'dv_preintegrated: {np.linalg.norm(v_inW_ofB0toB1_pi):.2f} [m/s] (abs␣
 ↪err: {np.linalg.norm(dv_err_pi):.2f} [m/s], rel err: {np.linalg.
 ↪norm(dv_err_pi)/np.linalg.norm(v_inW_ofB0toB1_gt)*100:.2f} [%])')

p_inW_ofB0toB1_pi = R_inW_ofB0_gt @ dp_01 + v_inW_ofB0_gt * dt_01 + 0.5 *␣
 ↪gravity * dt_01 ** 2
dp_err_pi = p_inW_ofB0toB1_pi - p_inW_ofB0toB1_gt

print(f'dp_preintegrated: {np.linalg.norm(p_inW_ofB0toB1_pi):.2f} [m] (abs err:␣
 ↪{np.linalg.norm(dp_err_pi):.2f} [m], rel err: {np.linalg.norm(dp_err_pi)/np.
 ↪linalg.norm(p_inW_ofB0toB1_gt)*100:.2f} [%])')


print(f"Timing\n{'='*100}")
print(f"Non linear (VIO): {nonlinear:.2f} [s]")
```

```
Results
================================================================================
====================
dR_preintegrated: [0.01200351 0.01672942 0.01971249] [deg] (xyz)
dR_preintegrated scalar: 15.62900 [deg]
dv_preintegrated: 0.12 [m/s] (abs err: 0.04 [m/s], rel err: 34.08 [%])
dp_preintegrated: 0.32 [m] (abs err: 0.01 [m], rel err: 2.63 [%])
Timing
================================================================================
====================
Non linear (VIO): 9.95 [s]
```