

# Evo

October 16, 2022

## 1 Evo

**EVO** is a SLAM evaluation based library for handling, evaluating and comparing the trajectory output of odometry and SLAM algorithms.

```
pip3 install evo --upgrade --no-binary evo
```

See [EVO](#) for more info. EVO doesn't directly support g2o format. We have to first convert g2o to kitti. A small script named g2o\_to\_kitti.py which does exactly this has been provided in misc folder. `python input_file.g2o name_of_output_file.kitti` to convert it to kitti format. Ex: `python gt.g2o gt.kitti`.

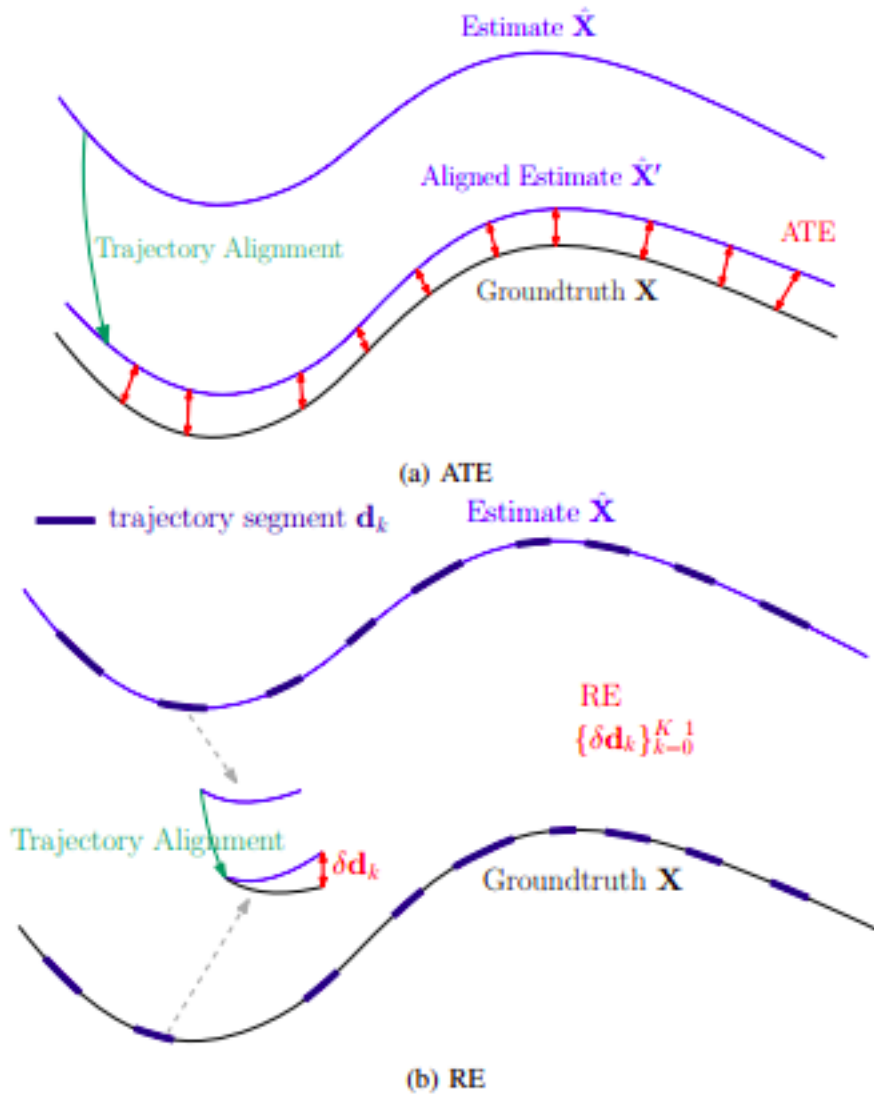
Now that it is in kitti format, run

```
evo_rpe kitti gt.kitti opt.kitti -v --plot --plot_mode xy
evo_ape kitti gt.kitti opt.kitti -v --plot --plot_mode xy
evo_traj kitti gt.kitti opt.kitti -v --plot --plot_mode xy
```

What is APE and RPE?

Topic	APE/ATE	RPE
capture	ATE/APE measures difference between ground truth points and estimated trajectory, obtained by an algorithm using sensor data as input. It is referred as mandatory performance measure.	RPE accounts for measuring SLAM result i.e. estimated trajectory accuracy by comparing reconstructed relative transformation between nearby poses to actual relative transformation i.e. ground truth. It is a recommended performance measure.

Topic	APE/ATE	RPE
calculation	As a pre-processing step, we associate the estimated poses with ground truth poses using the timestamps. Based on this association, we align the true and the estimated trajectory using singular value decomposition. Finally, we compute the difference between each pair of poses, and output the mean/median/standard deviation of these differences. Therefore steps are:- (1) - Align the estimated trajectory. (2) - Calculate the RMSE using the aligned estimation and the groundtruth.	The basic idea of relative error is that the estimation quality can be evaluated by measuring the relative relations between the states at different times. (1) - Select some random sub-trajectory of length $d$ . (2) - Align each sub-trajectory using the first state. (3) - Calculate the error of the end state of each sub-trajectory. (4) - Collect the errors for all the sub-trajectories. (5) - For different lengths $d$ , repeat step 1-4.
difference	(1) - Single number metric, easy for comparison. (2) - Sensitive to the time when the estimation error occurs. For example, a rotation estimation error tends to give a larger ATE when it happens at the beginning of the trajectory than the situation when it occurs at the end. (3) - The ATE more appropriate for measuring the performance of visual SLAM systems.	(1) - Informative statistics can be computed from the errors of all sub-trajectories. By changing the length $d$ , the relative error can reflect both short(local consistency) and long term accuracy (2) - Relatively complicated to compute. (3) - The RPE more appropriate for measuring the drift of a visual odometry system, for example the drift per second.



- Paper using EVO (APE/ATE)(See section VIII. EVALUATION METRIC)
- Kabsch-Umeyama Alignment - Brings the both the trajectory in same frame!