**1) Need Jupyter Notebook or Anaconda is the best practices. We done it on Google Colab Notebook and google drive for holding data**

**2) Install all necessary python packages. Here are some common packages-**

torch, torchvision, opencv-python, numpy, tqdm, pathlib, torchsummary, tensorboardX, albumentations, argparse, pickle, plyfile, pyyaml (< 6), datetime, shutil, matplotlib, tensorflow-gpu.

**3) Generate training data from training videos using Structure from Motion (SfM) or Simultaneous Localization and Mapping (SLAM).** In terms of the format, please refer to one training data example in this repository. We use SfM to generate training data in this work. Color images with the format of "{:08d}.jpg" are extracted from the video sequence where SfM is applied. camer\_intrinsics\_per\_view stores the estimated camera intrinsic matrices for all registered views.

**4) Run train.ipynb with proper arguments for self-supervised learning.**

One example is:

/path/to/python /path/to/train.py --id\_range 1 2 --input\_downsampling 4.0 --network\_downsampling 64 --adjacent\_range 5 30 --input\_size 256 320 --batch\_size 8 --num\_workers 8 --num\_pre\_workers 8 --validation\_interval 1 --display\_interval 50 --dcl\_weight 5.0 --sfl\_weight 20.0 --max\_lr 1.0e-3 --min\_lr 1.0e-4 --inlier\_percentage 0.99 --visibility\_overlap 30 --training\_patient\_id 1 --testing\_patient\_id 1 --validation\_patient\_id 1 --number\_epoch 100 --num\_iter 2000 --architecture\_summary --training\_result\_root "/path/to/training/directory" --training\_data\_root "/path/to/training/data"

**5) Run test.ipynb to evaluate results.** Apply a registration algorithm that is able to estimate a similarity transformation to register the predicted point clouds to the corresponding CT model to calculate residual errors (this step may require manual point cloud initialization).

One example is:

/path/to/python /path/to/evaluate.py --id\_range 1 2 --input\_downsampling 4.0 --network\_downsampling 64 --adjacent\_range 5 30 --input\_size 256 320 --batch\_size 1 --num\_workers 2 --num\_pre\_workers 8 --load\_all\_frames --inlier\_percentage 0.99 --visibility\_overlap 30 --testing\_patient\_id 1 --load\_intermediate\_data --architecture\_summary --trained\_model\_path "/path/to/trained/model" --sequence\_root "/path/to/sequence/path" --evaluation\_result\_root "/path/to/testing/result" --evaluation\_data\_root "/path/to/testing/data" --phase "test”