Imperial College London

Final Presentation

Robotic guidance and localization during endoluminal procedures

Supervisor: Dr Stamatia (Matina) Giannarou, Dr Dan Leff

Group Members:
Demin Li
Ziyan Wang
Yuze Wang
Xinxin Li

The Hamlyn Centre for Robotic Surgery

Imperial College London

Background

A large number of people suffer from bronchial disease and breast disease.

Difficult to detect the lesions' location due to the complexity of bronchial and mammary duct structures

Classical CV algorithms are difficult to apply in vivo positioning

Figure 1 Lung Disease[1]

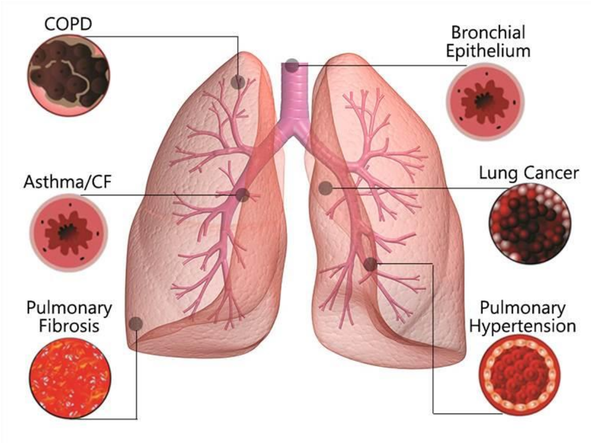
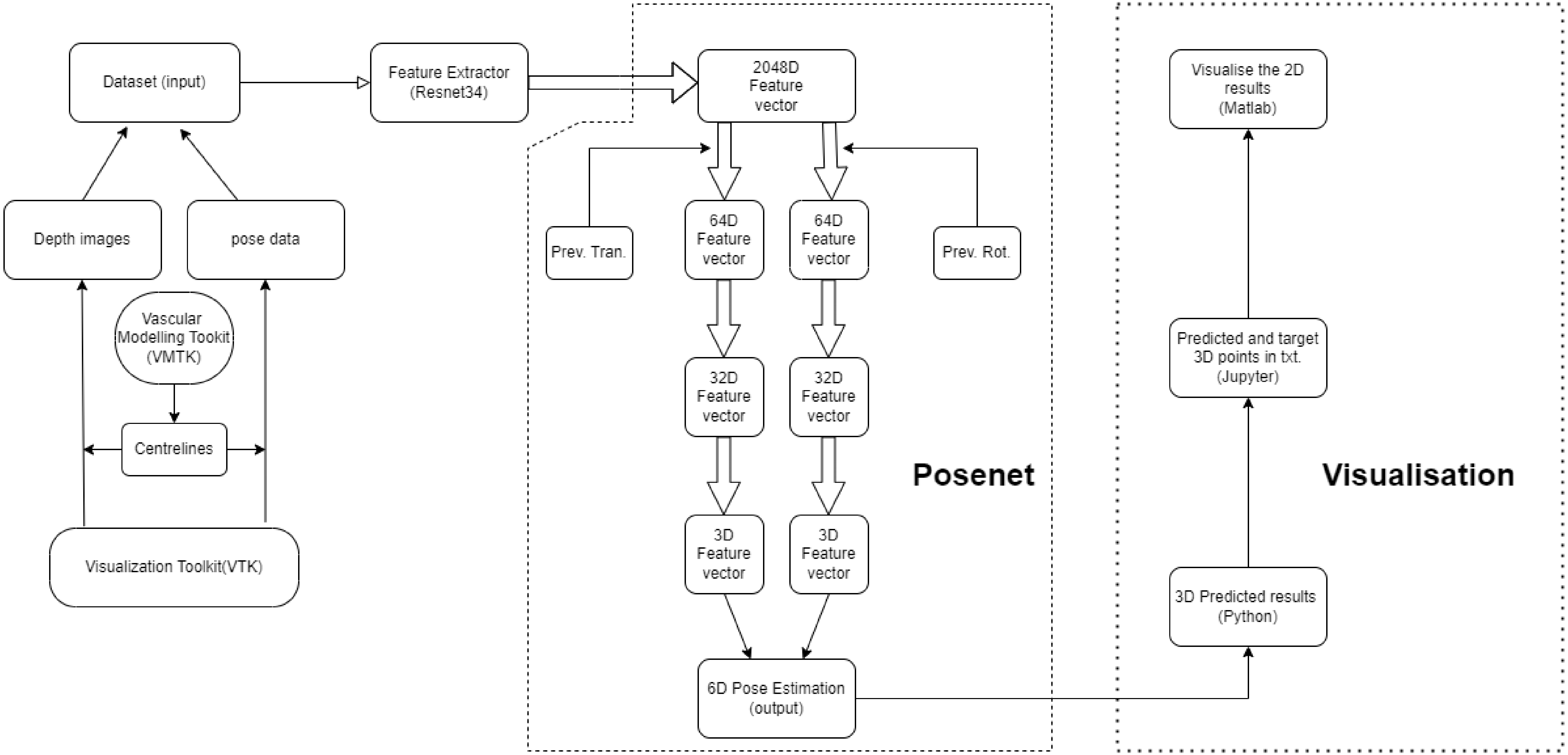
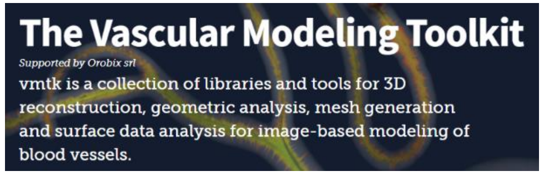


Figure 2 Respiratory Disease[2]

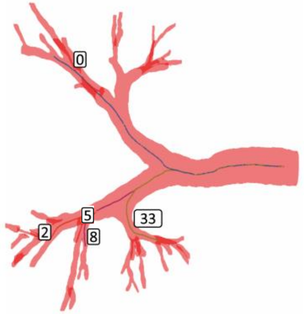
Pipeline


Dataset (input)
Feature Extractor
(Resnet34)
Depth images
pose data
Vascular
Modelling Tookit
(VMTK)
Centrelines
Visualization Toolkit(VTK)
2048D
Feature
vector
Prev. Tran.
64D
Feature
vector
64D
Feature
vector
Prev. Rot.
32D
Feature
vector
32D
Feature
vector
3D
Feature
vector
3D
Feature
vector
6D Pose Estimation
(output)
Posenet
Visualise the 2D
results
(Matlab)
Predicted and target
3D points in txt.
(Jupyter)
3D Predicted results
(Python)
Visualisation

Data source

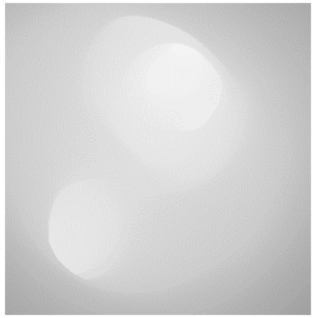


VMTK → Centrelines



VTK → Depth images

VTK → Pose data



Size:224*224

1	0	0	-26.2668
0	1	0	-23.6597
0	0	1	-160.021
0	0	0	1

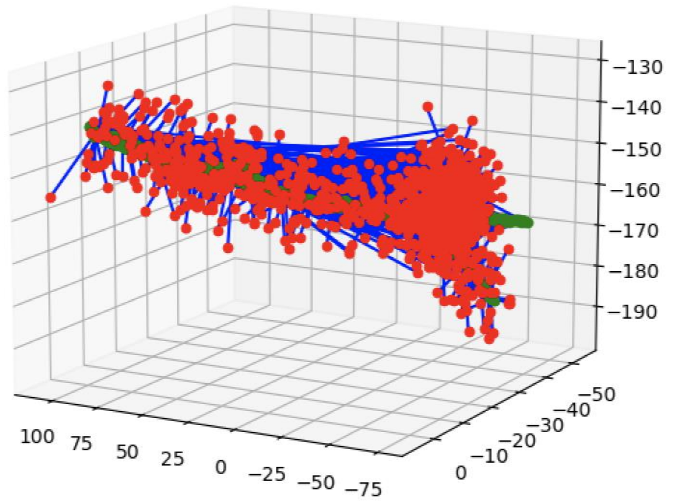
Imperial College London

Visualization

-Evaluate the predicted result in 3D format in python and get pkl files.

-Visualise the result in Jupyter, and save data as required for MATLAB.

-Use matalb to load the generated 2D lung image and visualise the results in 2D format.



PoseNet

- It is designed to obtain an accurate pose estimation with a monocular image input for a natural scene.
- It is constructed based on GoogleNet, 23 layers
- Replace all three softmax classifiers with affine regressors. The softmax layers were removed and each final fully connected layer was modified to output a pose vector of 7-dimensions representing position and orientation.
- Insert another fully connected layer before the final regressor of feature size 2048. This was to form a localization feature vector which may then be explored for generalisation.
- At test time we also normalize the quaternion orientation vector to unit length.

p = [x, q]

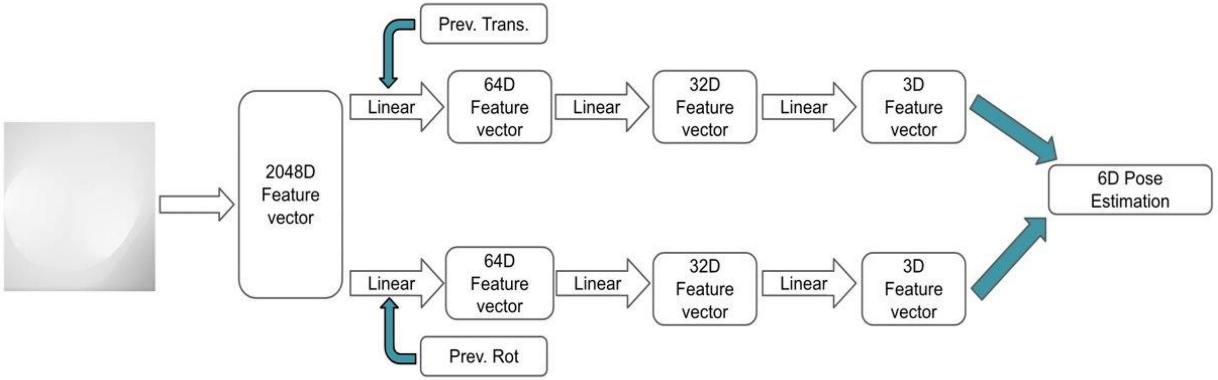
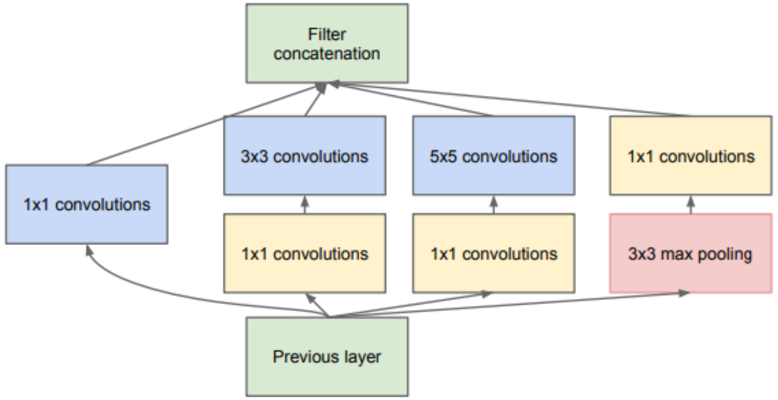


Figure 3: Posenet [3]

GoogleNet

- Inception Module
- Its main function is to reduce the dimensionality of network features and reduce a large amount of calculation without sacrificing the performance of the network model, which is useful for training deeper and wider networks
- 22 layers



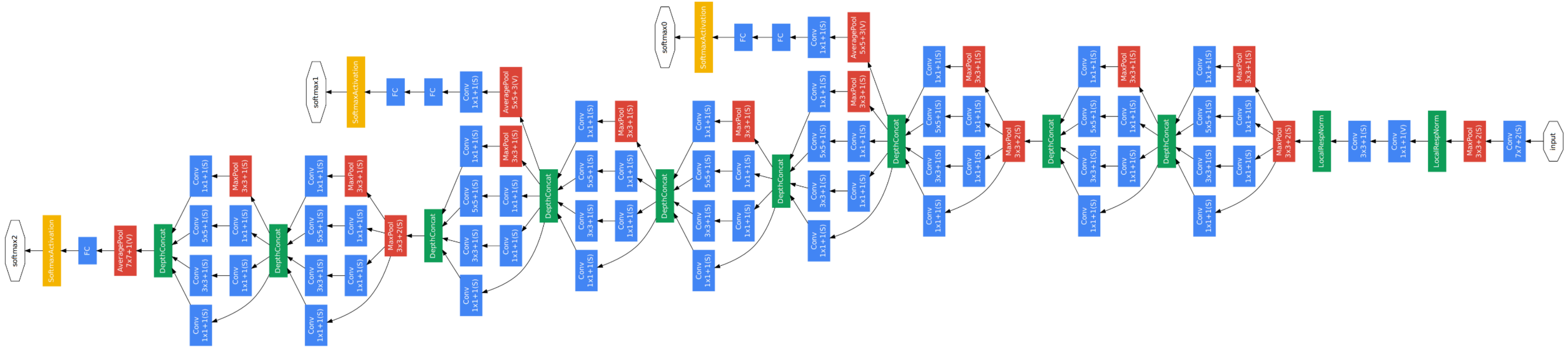


Figure 4: GoogleNet Structure [4]

Loss function

1. Train both transition and orientation together
2. Punish the predicted points out of the given range (bounding box)
3. Backpropagation update weights

$$loss(I) = \|\hat{\mathbf{x}} - \mathbf{x}\|_2 + \beta \left\| \hat{\mathbf{q}} - \frac{\mathbf{q}}{\|\mathbf{q}\|} \right\|_2$$

Get the error of expected value and practical value and update beta

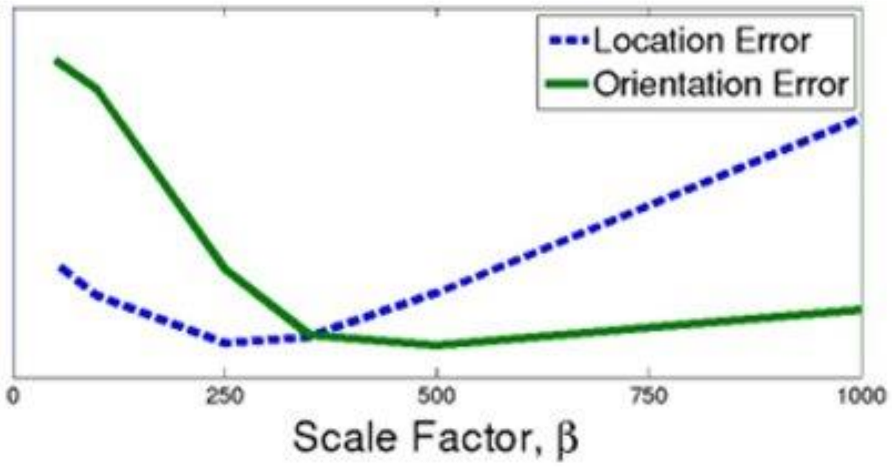


Figure: error with different beta [5]

Compared to training rotation and translation seperately, training them together can obtain a better result; since it is hard for the network to determine the pose by onesided information.

Heaviside Loss

Heaviside loss

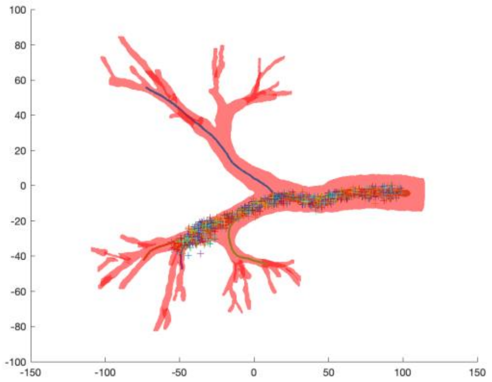
Alpha = 1, beta = 5mm

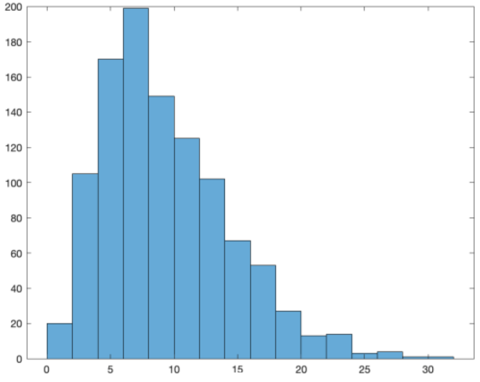
$$Loss_{heavi} = Loss_{orig} + \alpha \mathcal{H}(Norm(target[:3], pred[:3]))$$

$$\mathcal{H}(x) = \begin{cases} 1 & x > \beta \\ 0 & \text{otherwise,} \end{cases}$$

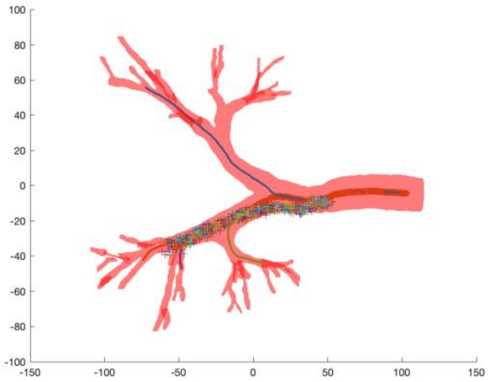
Original poseNet trained with centrelines CL 0,2,5,8,33 and tested on CL5

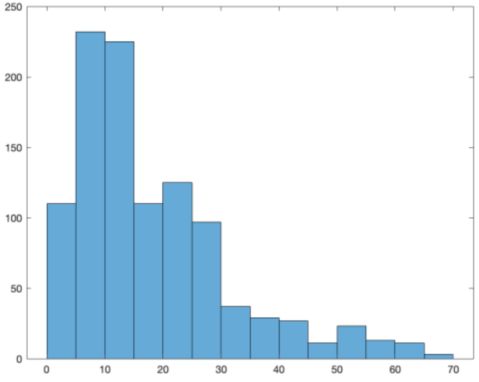
Epoch 20



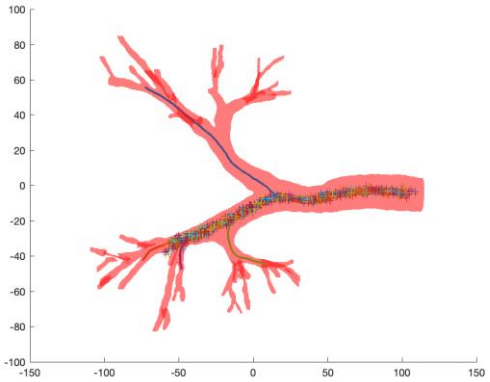


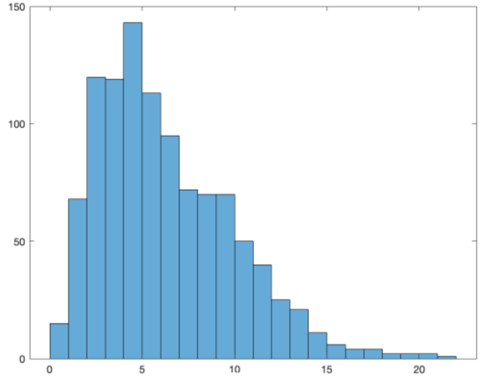
Epoch 40



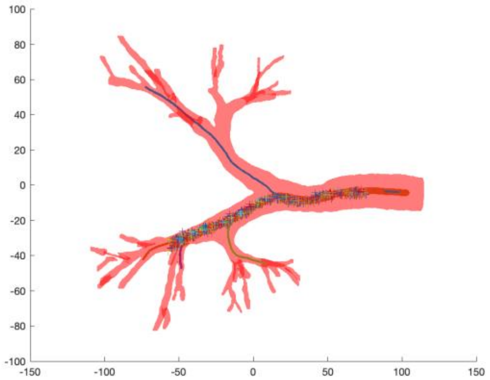


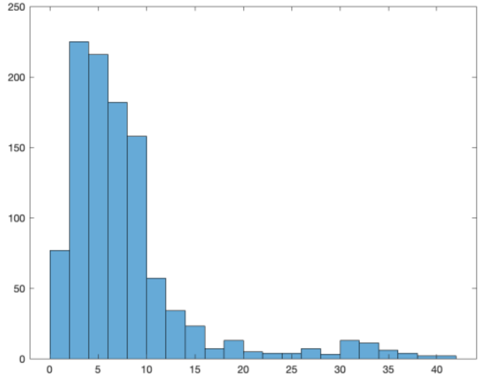
Epoch 100





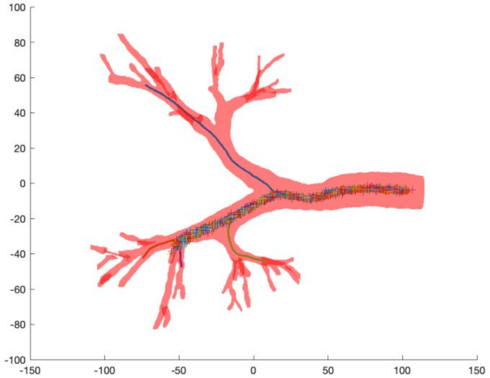
Epoch 200

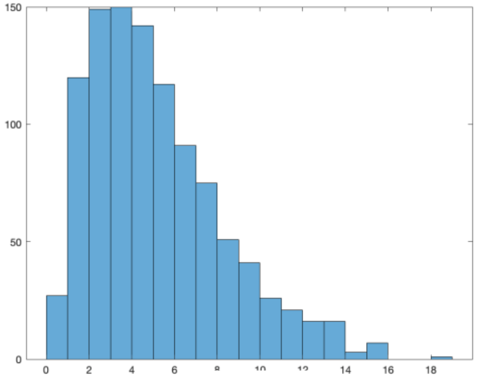




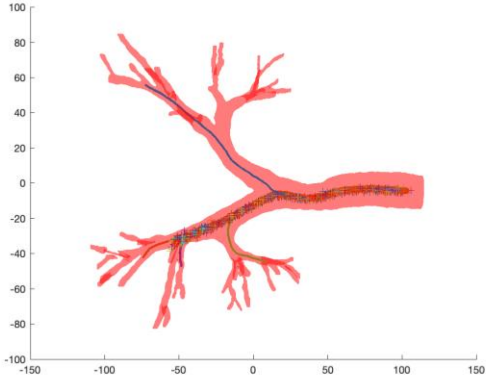
Original poseNet trained with centrelines CL 0,2,5,8,33 and tested on CL5

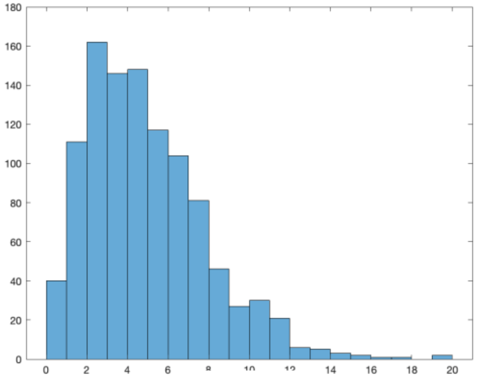
Epoch 300



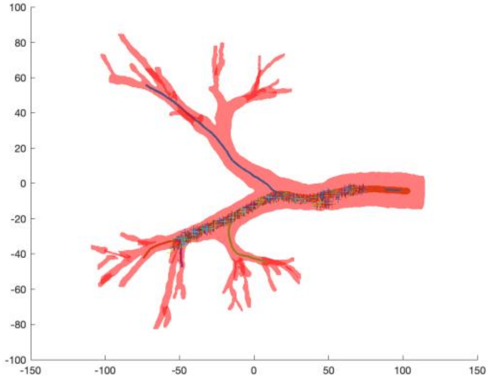


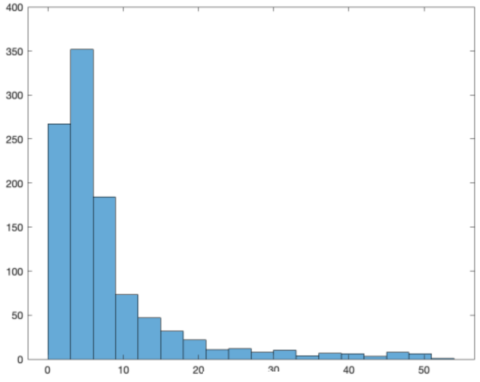
Epoch 400





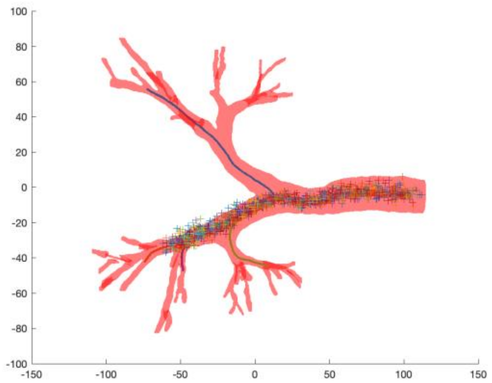
Epoch 500

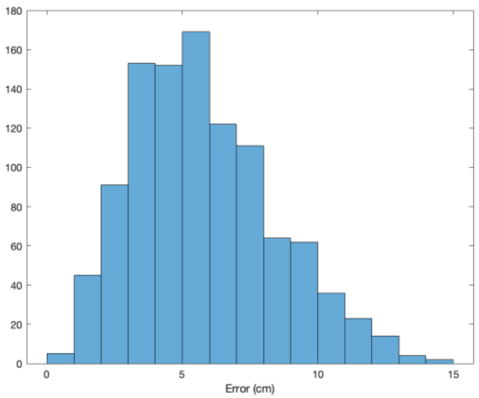




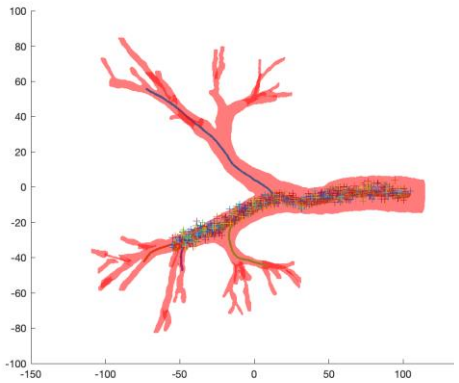
PoseNet using Heaviside loss trained with centrelines CL 0,2,5,8,33 and tested on CL5

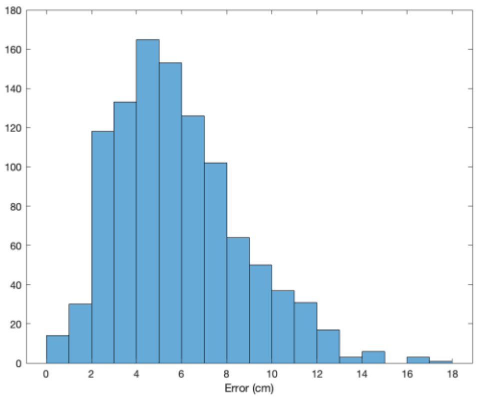
Epoch 20



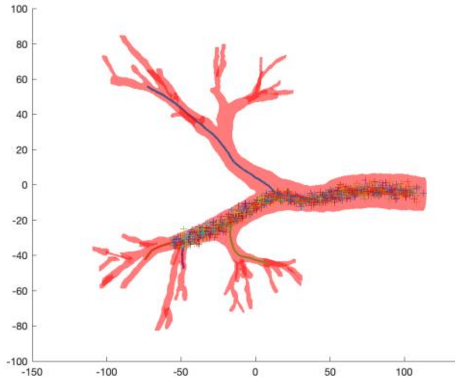


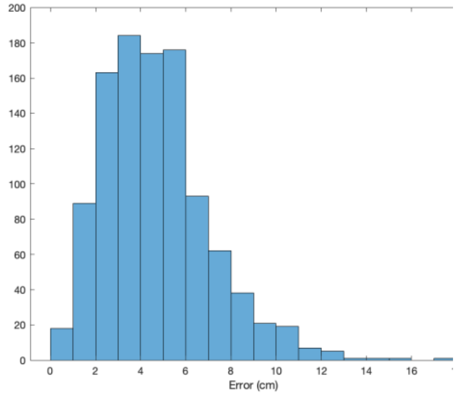
Epoch 40



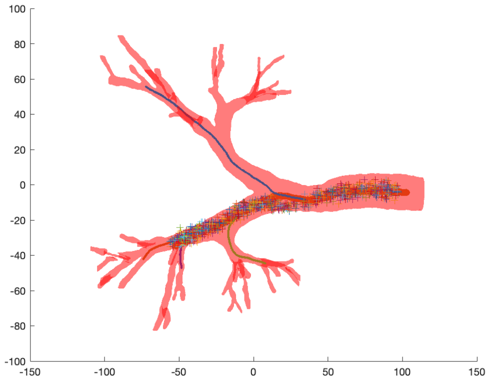


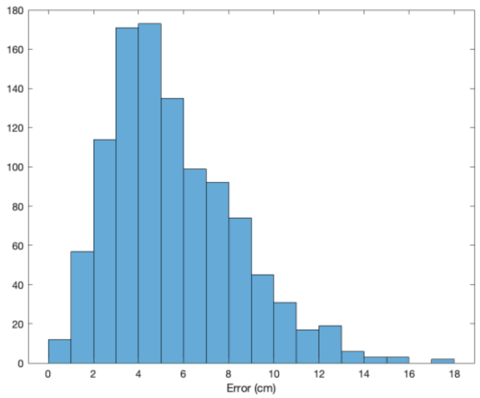
Epoch 100





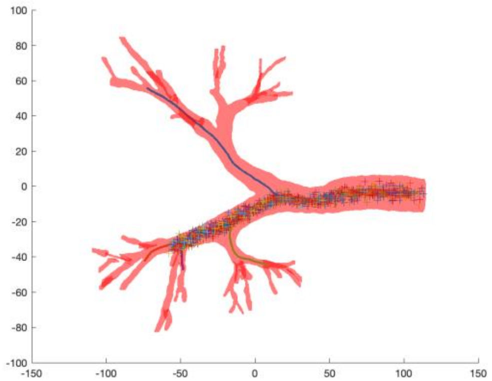
Epoch 200

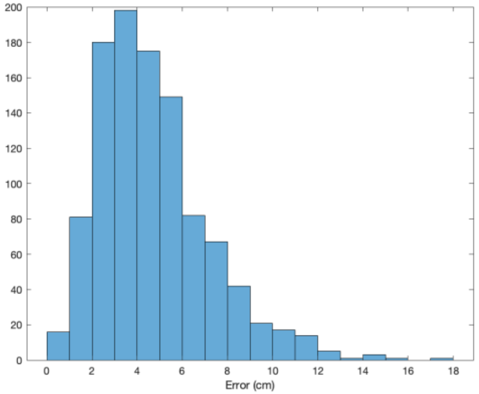




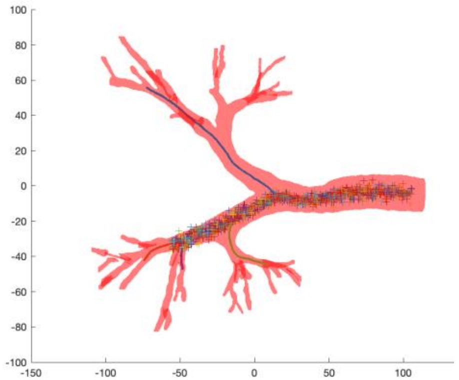
PoseNet using Heaviside loss trained with centrelines CL 0,2,5,8,33 and tested on CL5

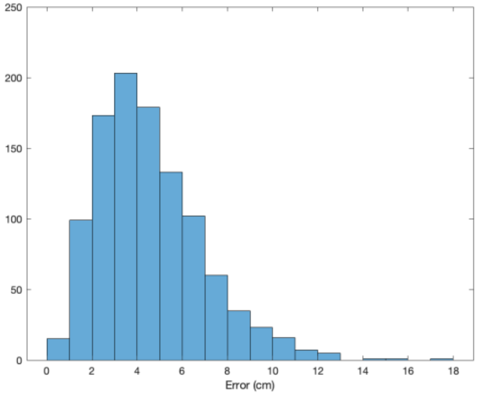
Epoch 300



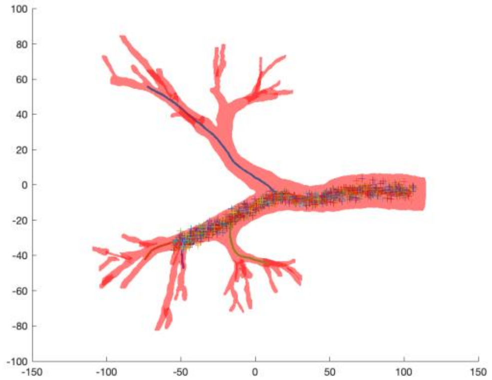


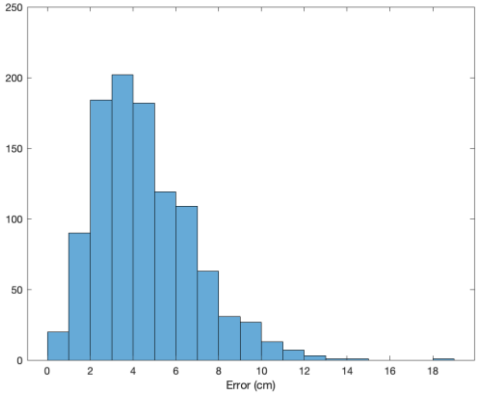
Epoch 400





Epoch 500

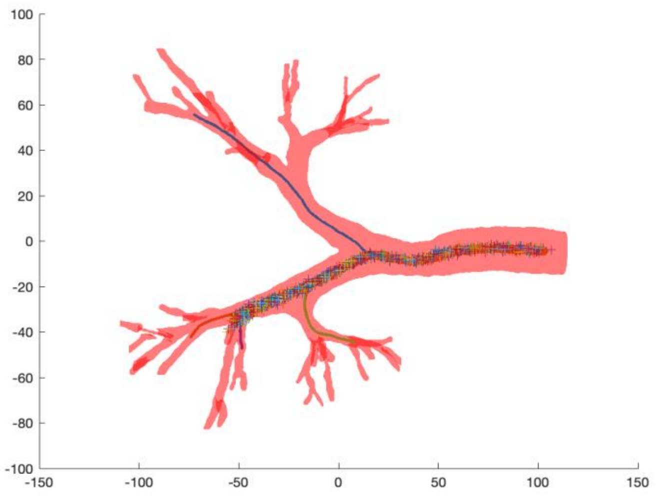


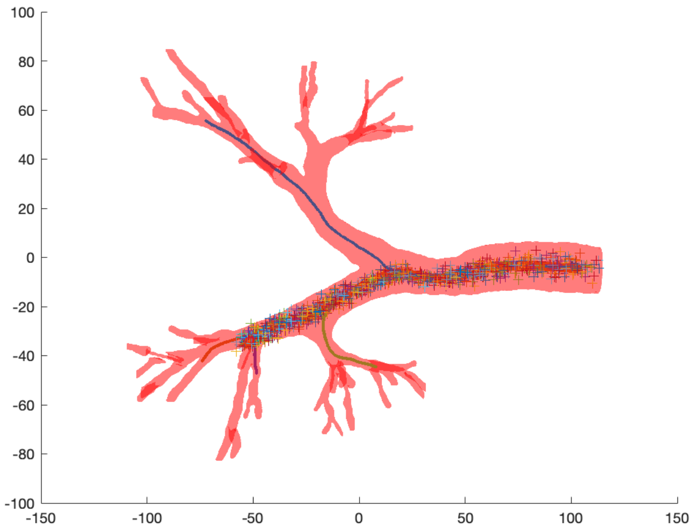


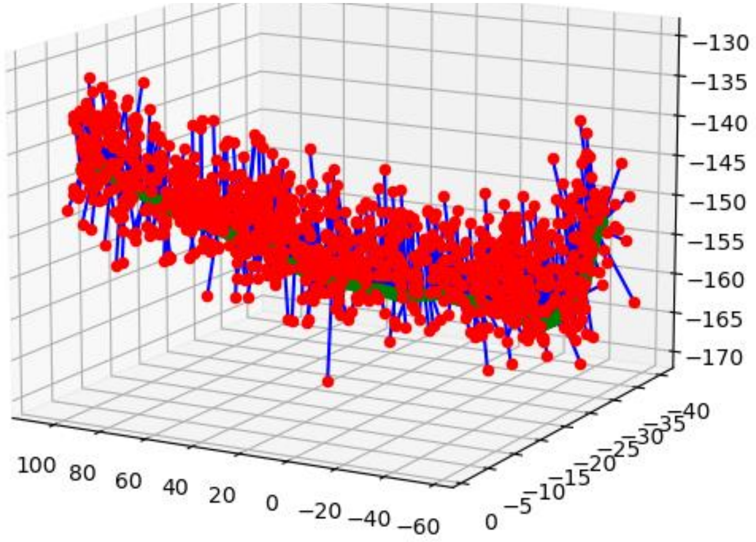
Imperial College London

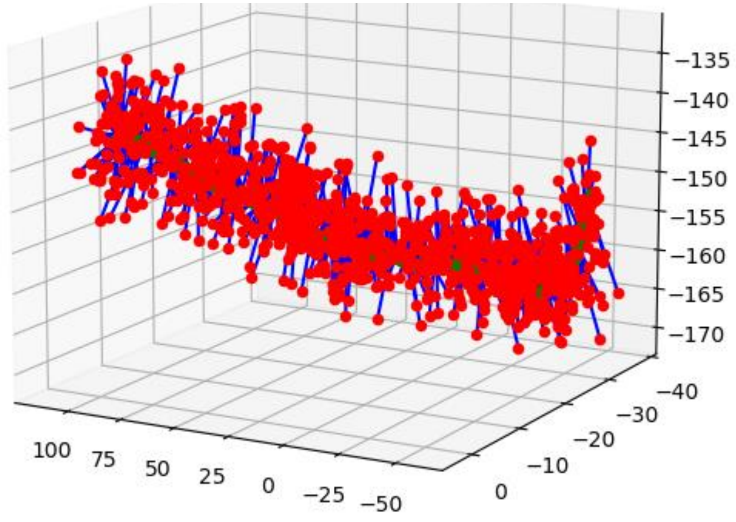
The Hamlyn Centre for Robotic Surgery

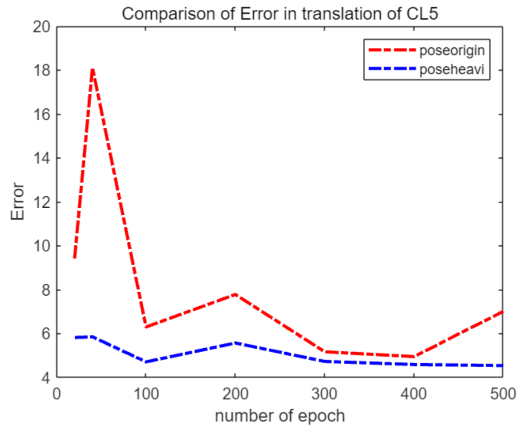
Show 2D and 3D





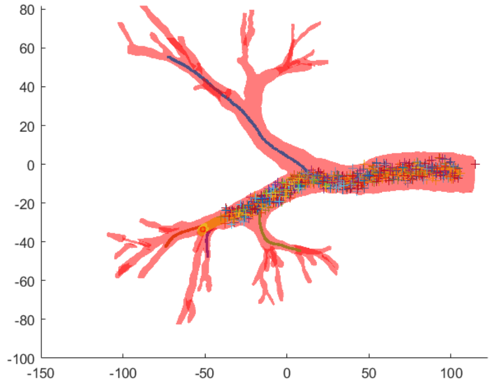


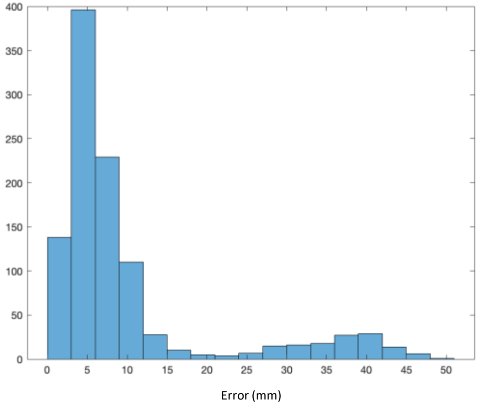




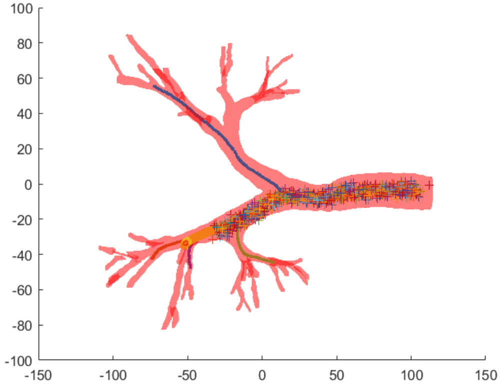
Centreline 5 Results (Train without CL5 via poseheavi)

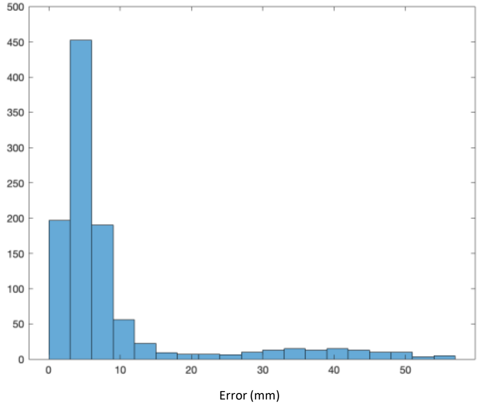
Epoch 20



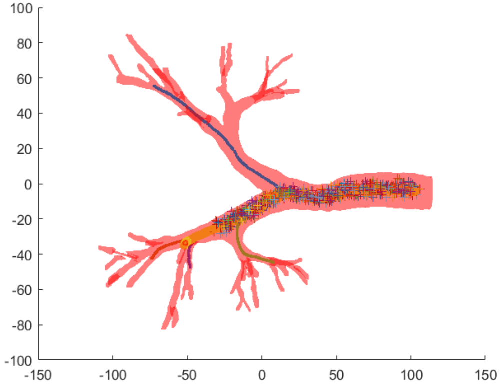


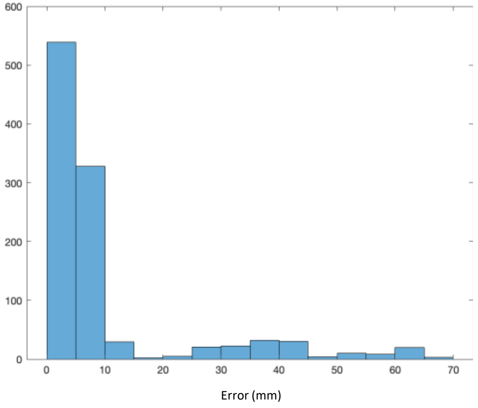
Epoch 40



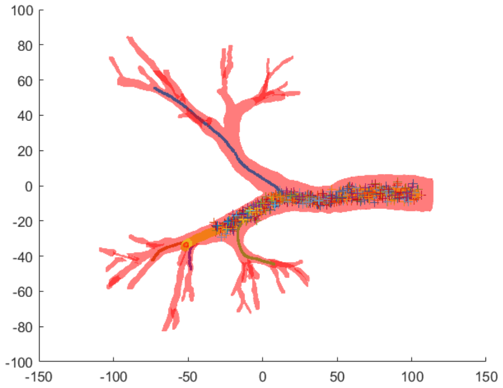


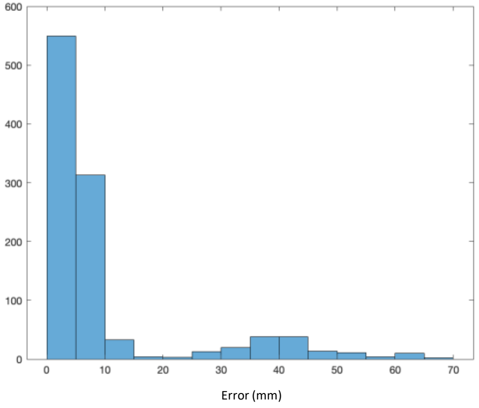
Epoch 100





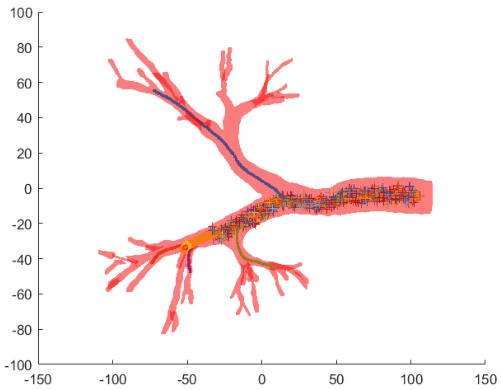
Epoch 200

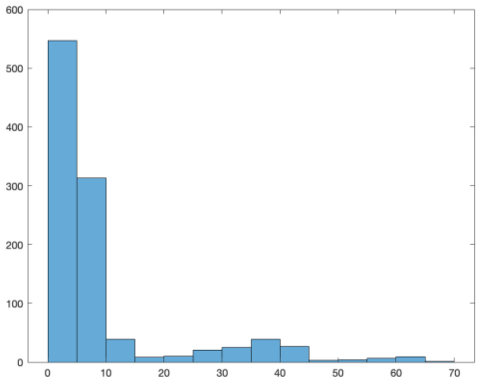




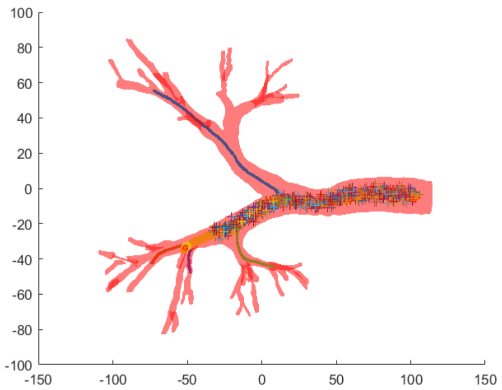
Centreline 5 Results (Train without CL5 via poseheavi)

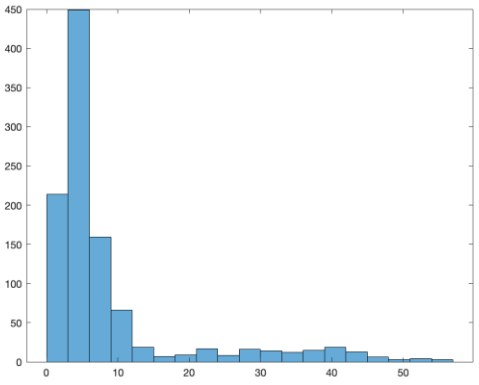
Epoch 300



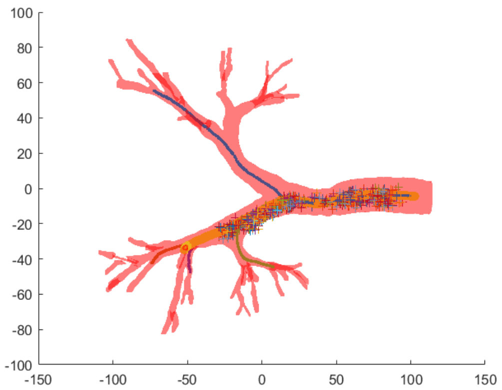


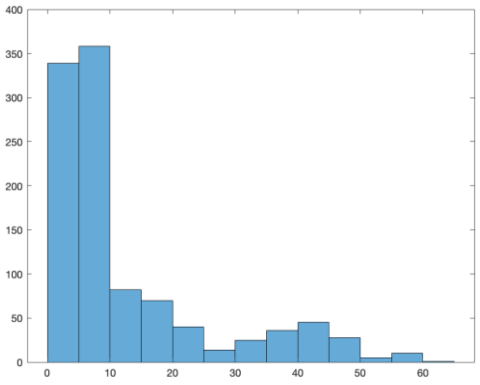
Epoch 400





Epoch 500





Imperial College London

The Hamlyn Centre for Robotic Surgery

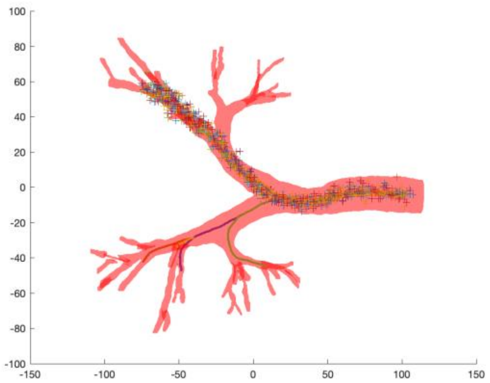
Centreline 0 Results (Train with all centrelines via poseheavi)

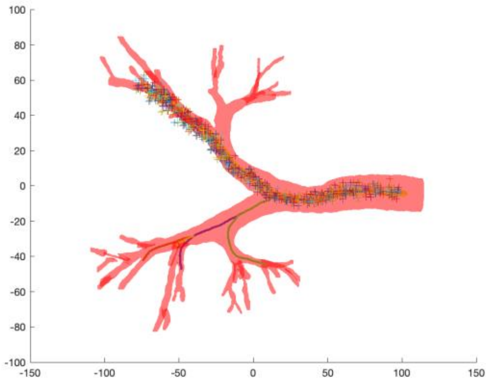
Epoch 20

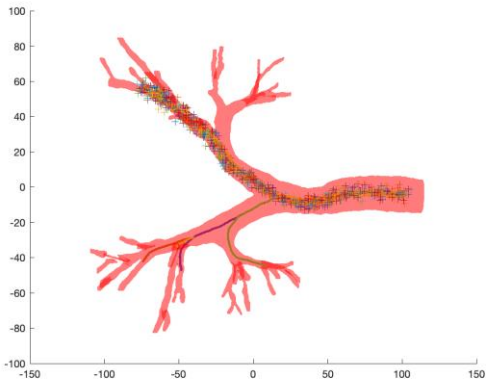
Epoch 40

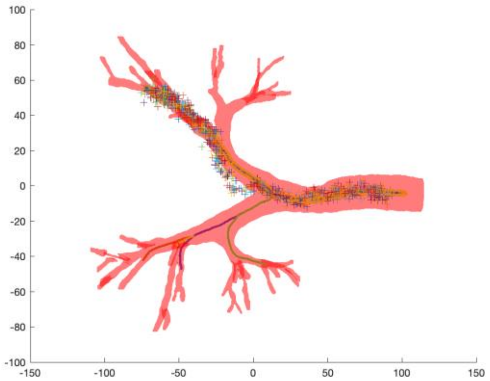
Epoch 100

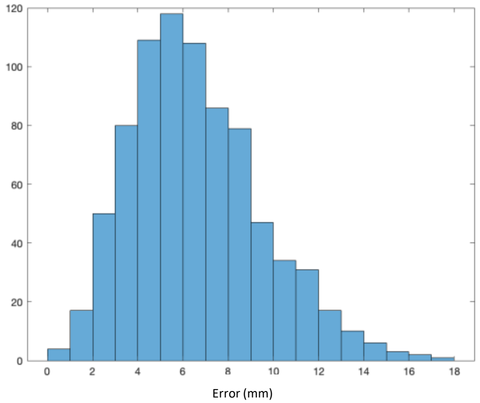
Epoch 200

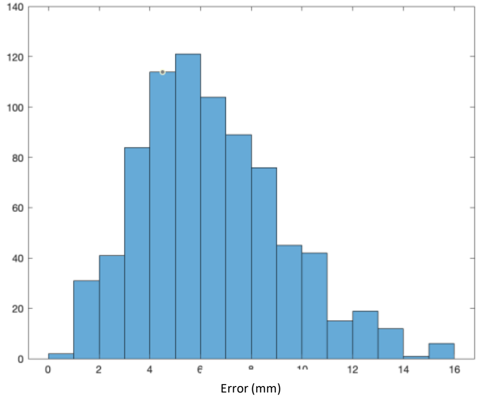


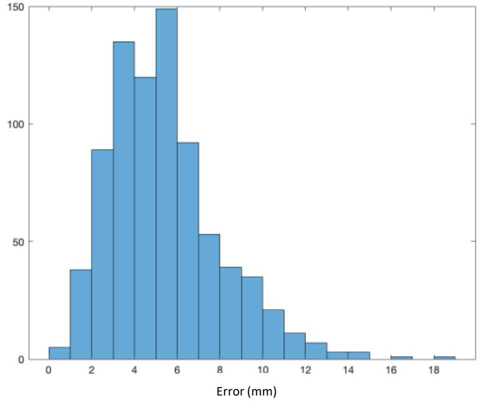


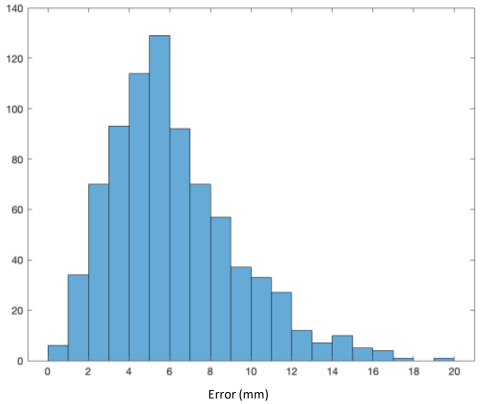










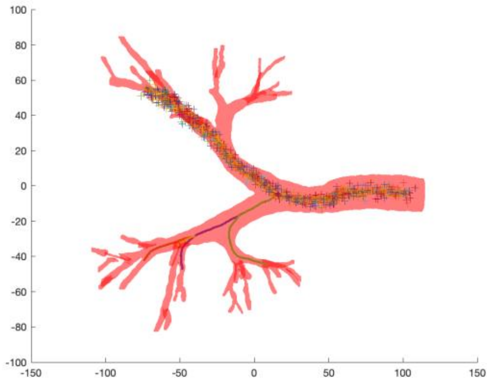


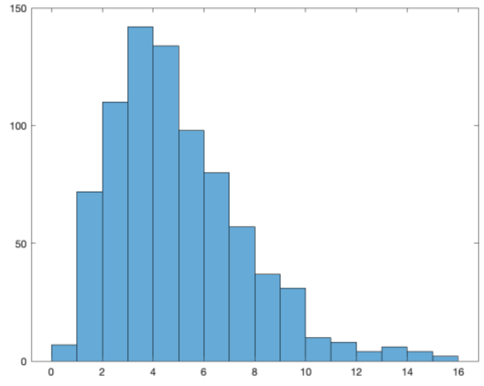
Imperial College London

The Hamlyn Centre for Robotic Surgery

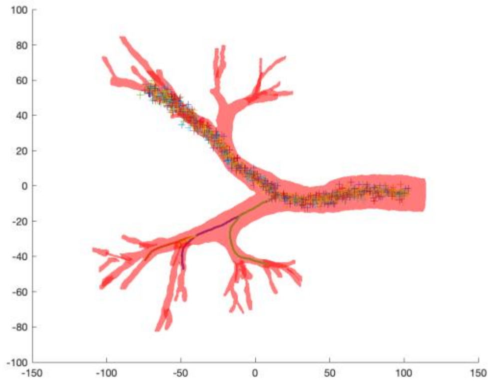
Centreline 0 Results (Train with all centrelines via poseheavi)

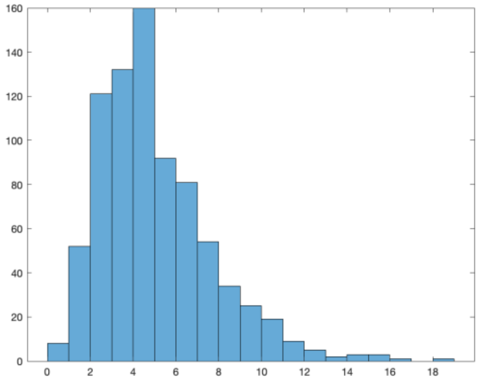
Epoch 300



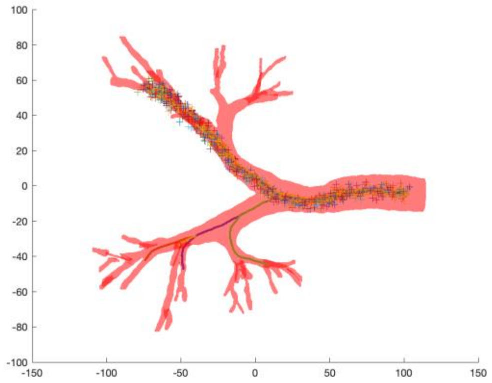


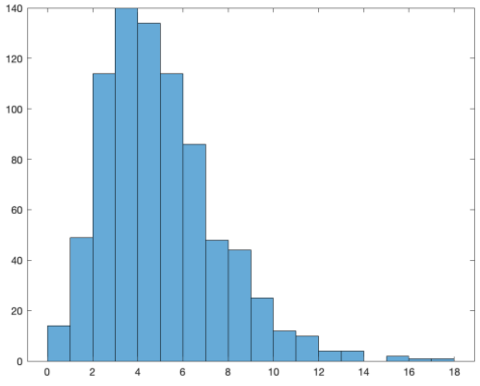
Epoch 400





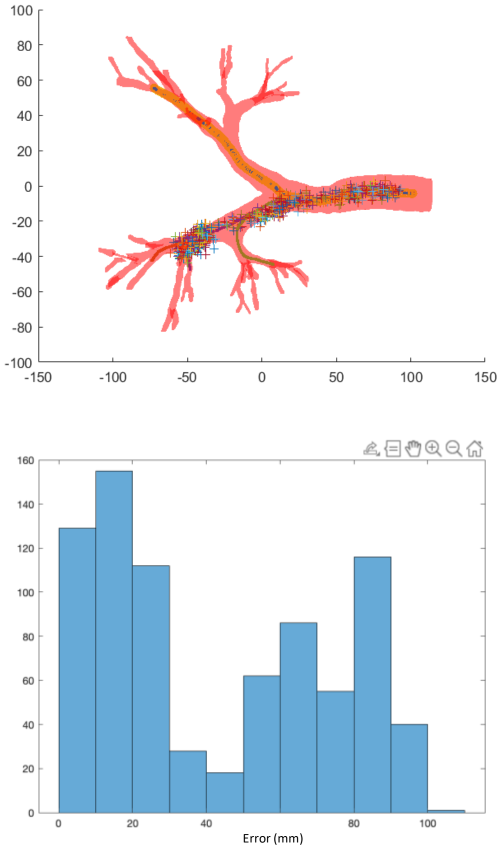
Epoch 500



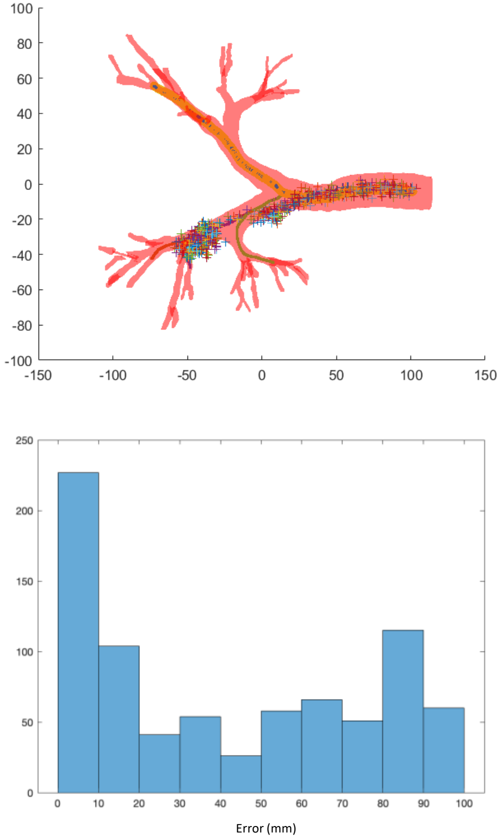


Centreline 0 Results (Train without CL0 via poseheavi)

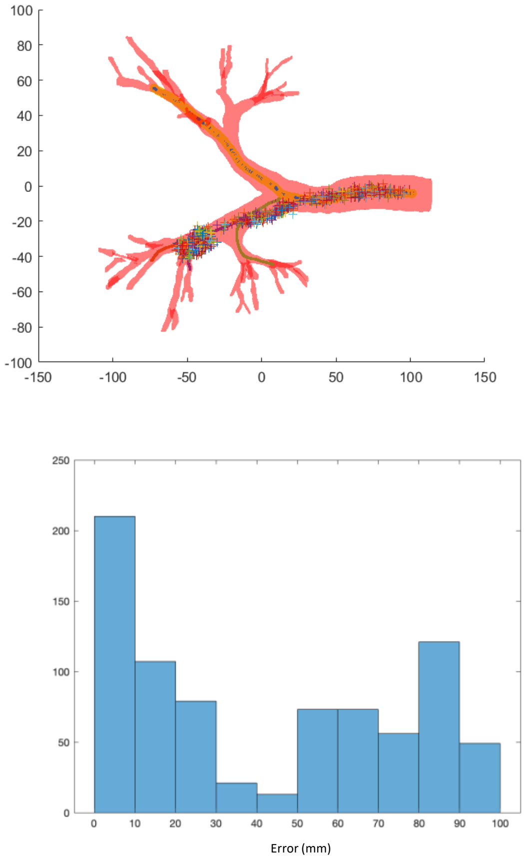
Epoch 20



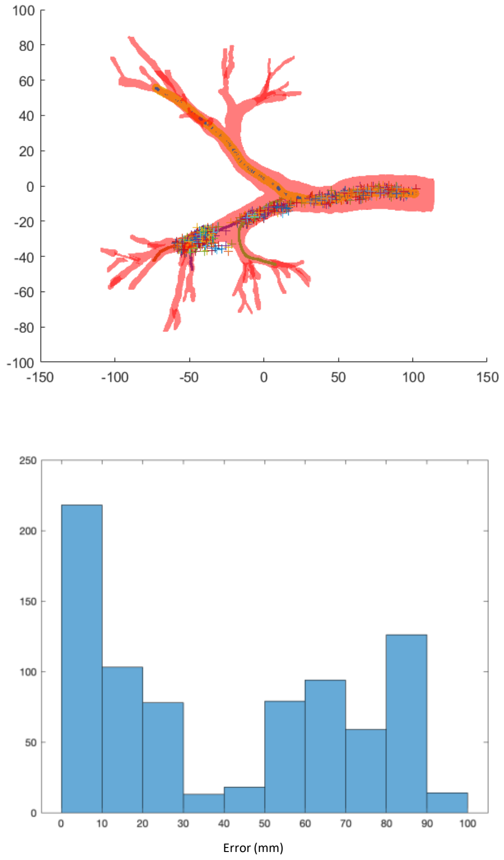
Epoch 40



Epoch 100



Epoch 200

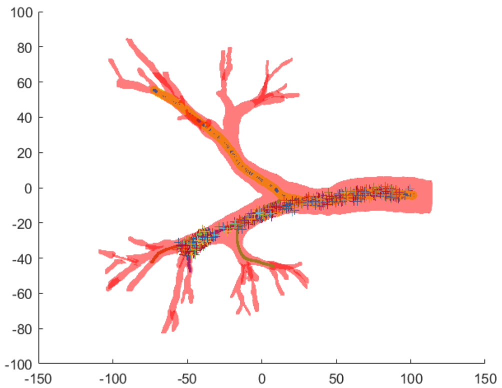


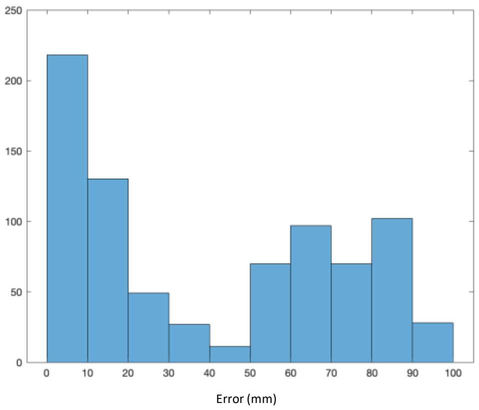
Imperial College London

The Hamlyn Centre for Robotic Surgery

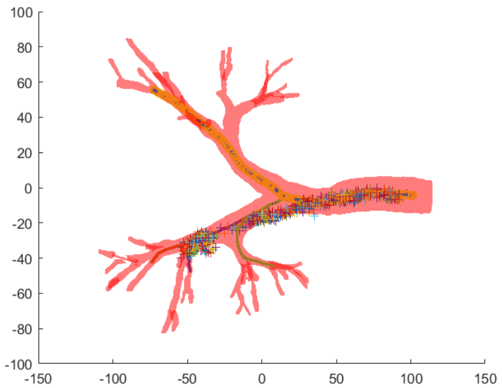
Centreline 0 Results (Train without CL0 via poseheavi)

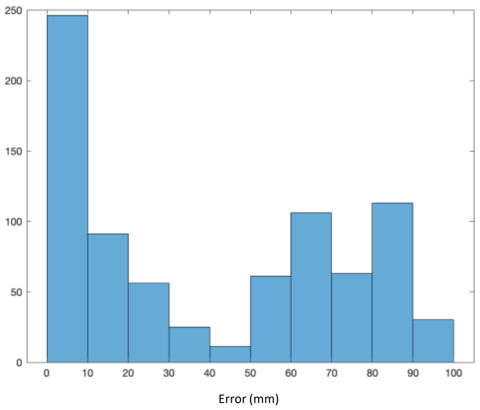
Epoch 300



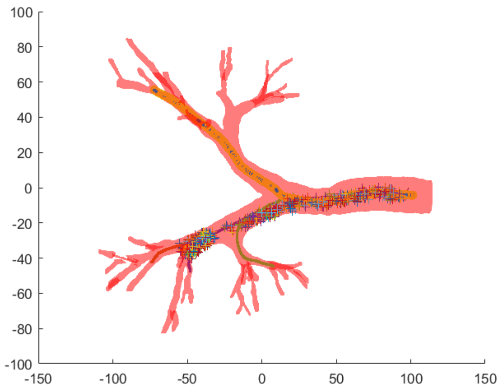


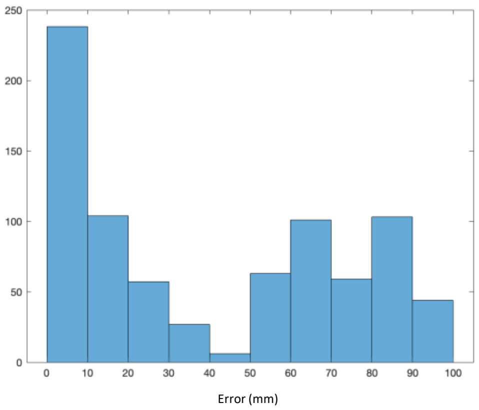
Epoch 400





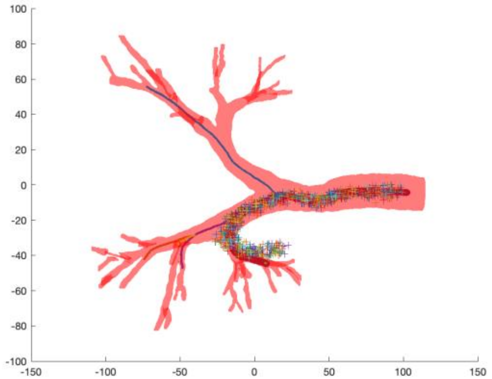
Epoch 500

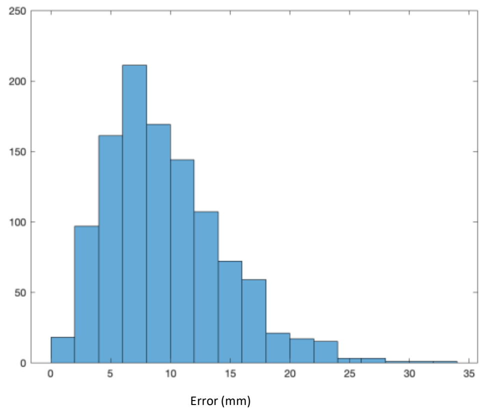




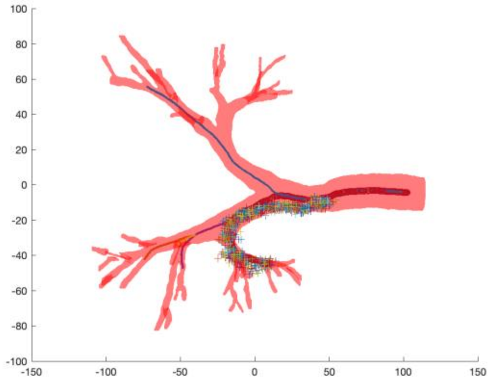
Original poseNet trained with centrelines CL 0,2,5,8,33 and tested on CL33

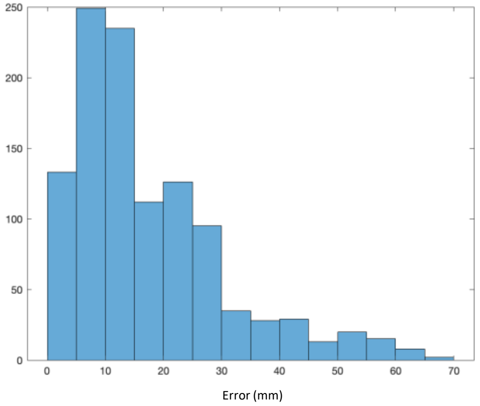
Epoch 20



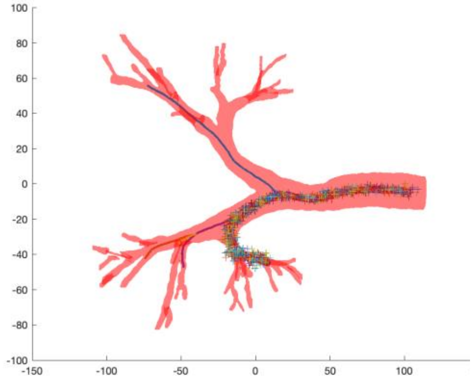


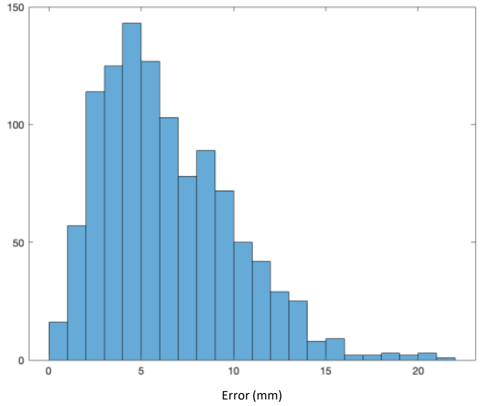
Epoch 40



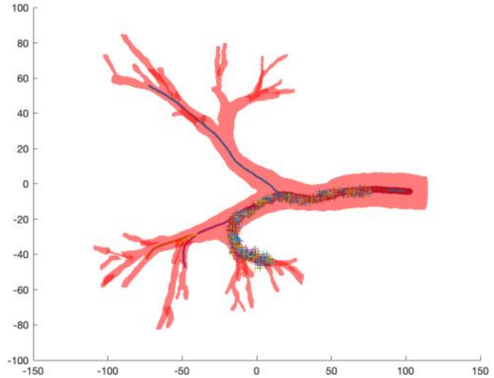


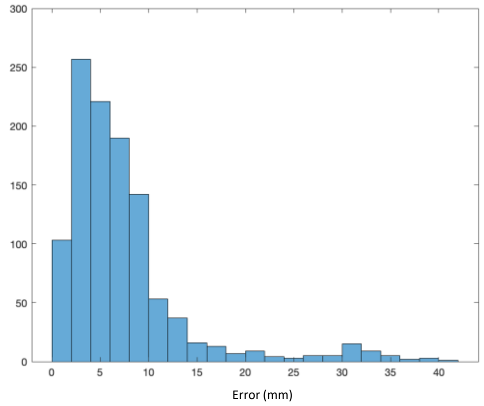
Epoch 100





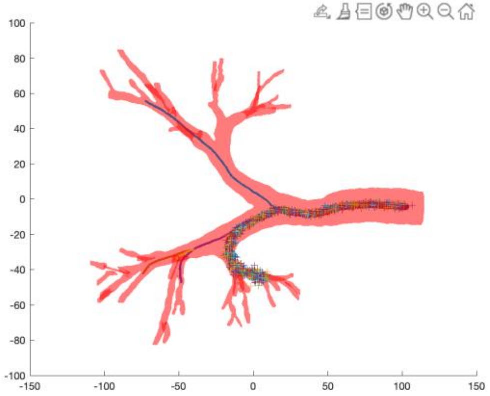
Epoch 200

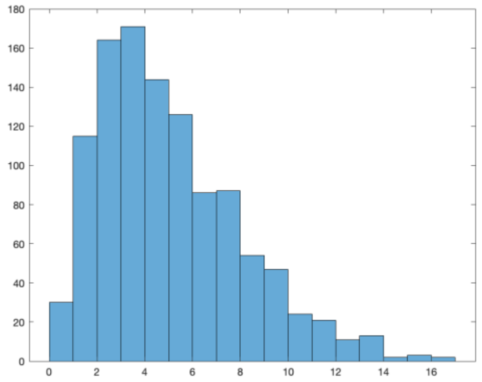




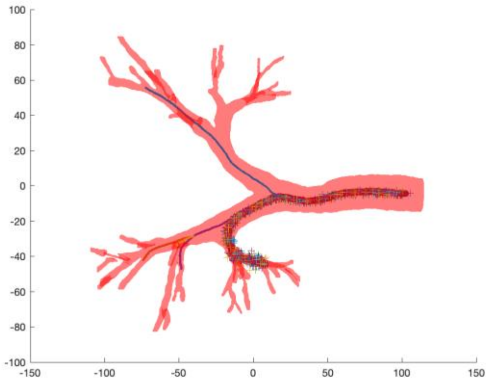
Original poseNet trained with centrelines CL 0,2,5,8,33 and tested on CL33

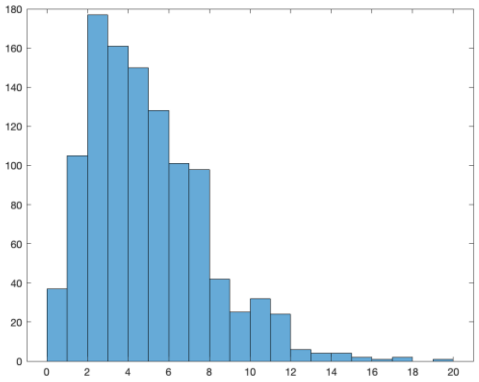
Epoch 300



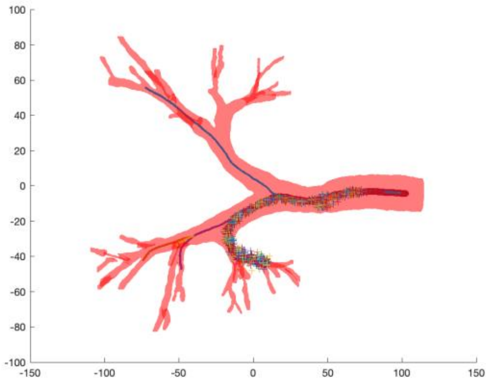


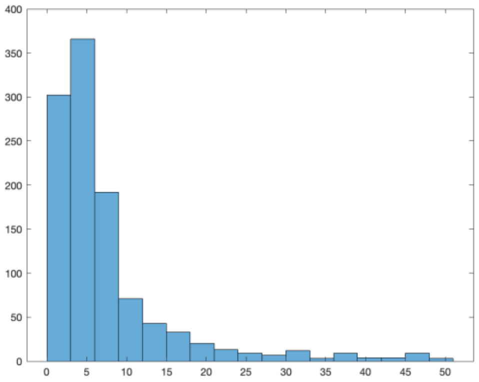
Epoch 400





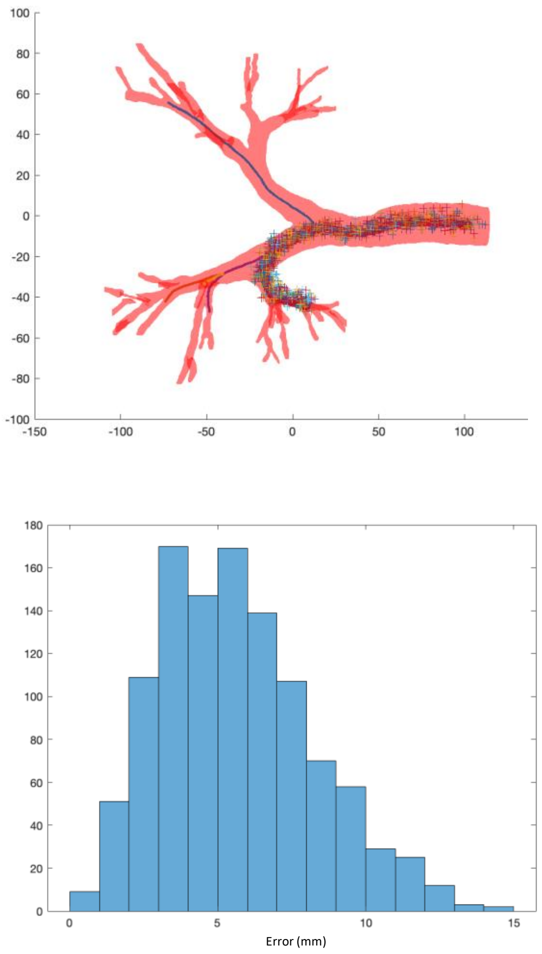
Epoch 500



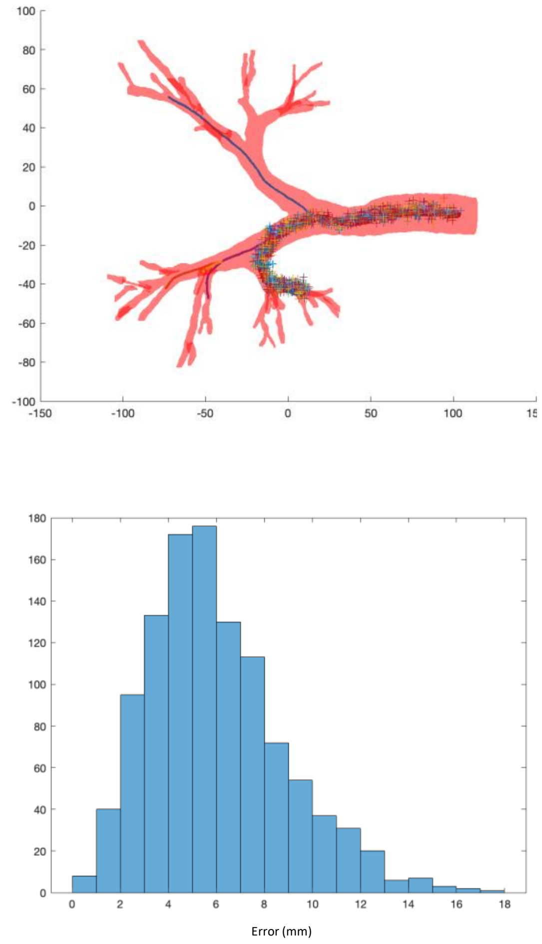


Centreline 33 Results (Train with all centrelines via poseheavi)

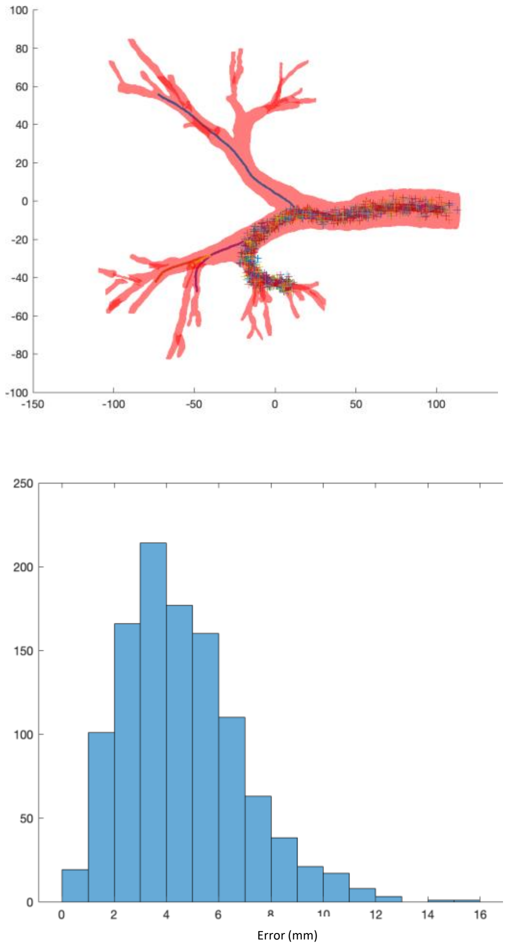
Epoch 20



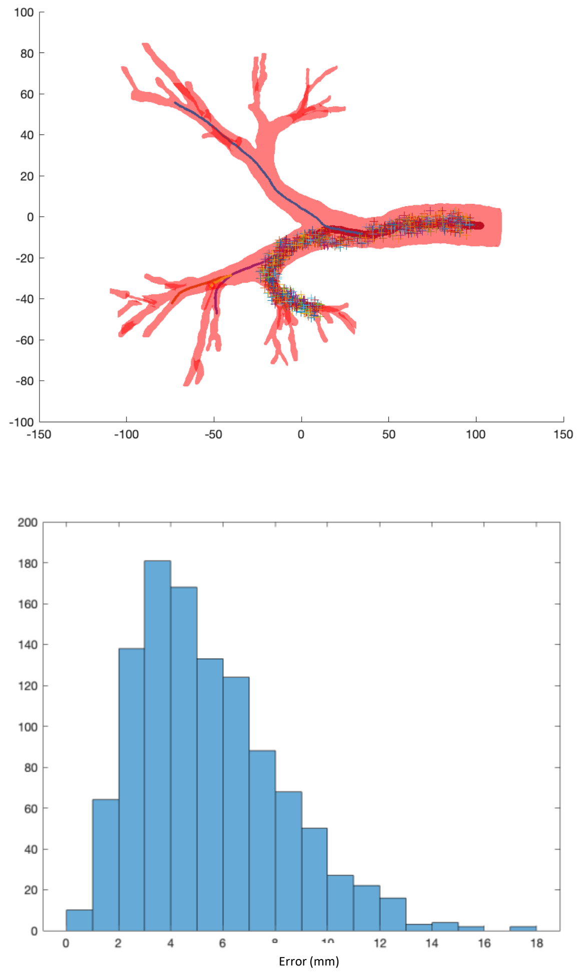
Epoch 40



Epoch 100

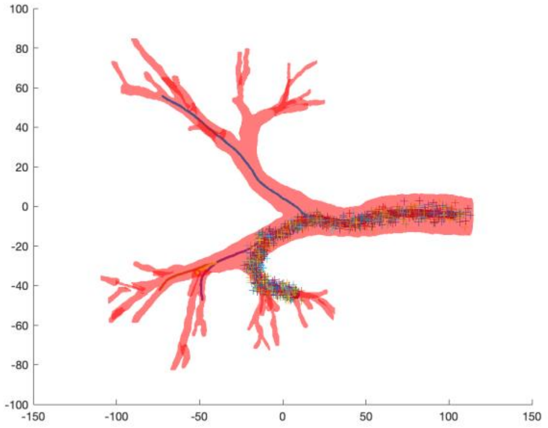


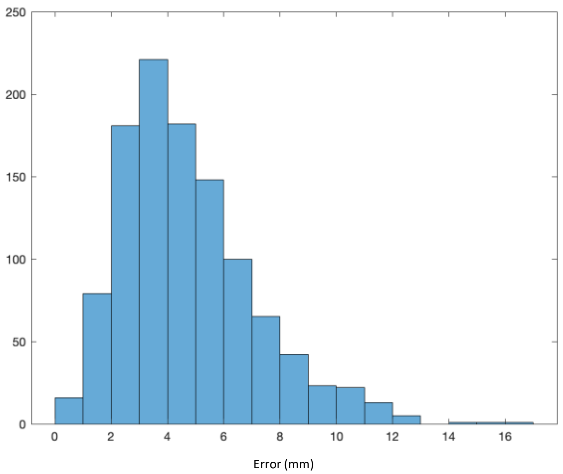
Epoch 200



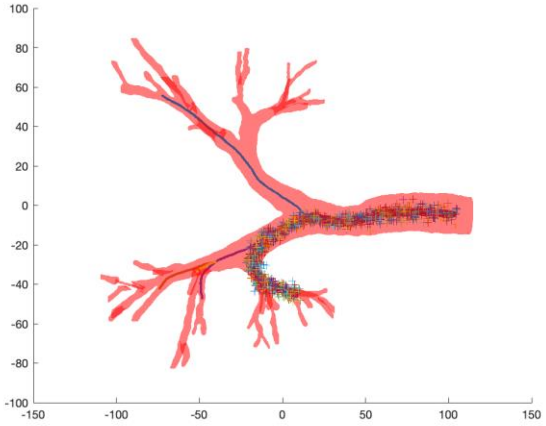
Centreline 33 Results (Train with all centrelines via poseheavi)

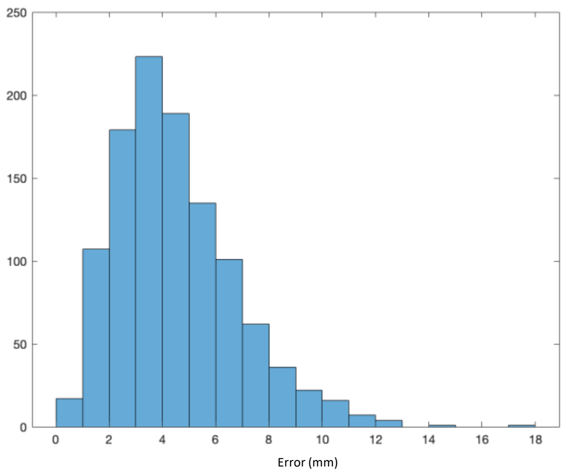
Epoch 300



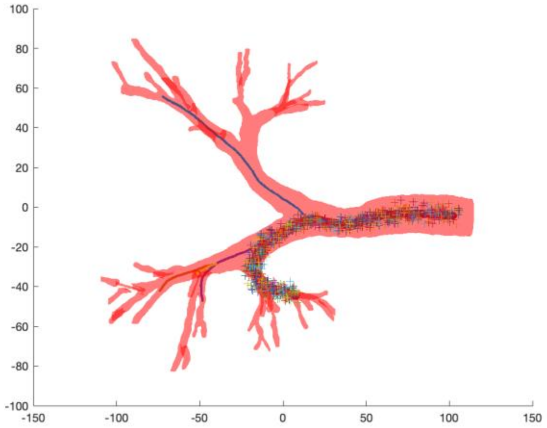


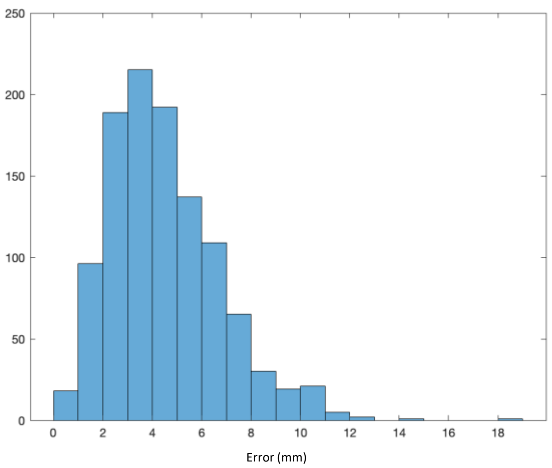
Epoch 400



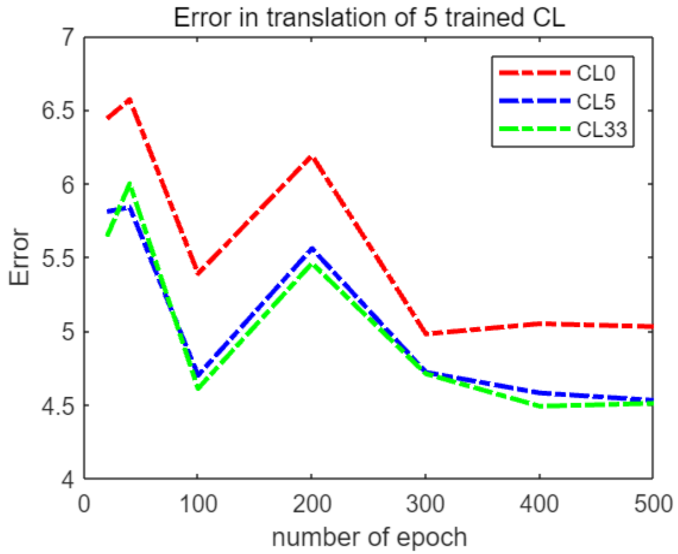


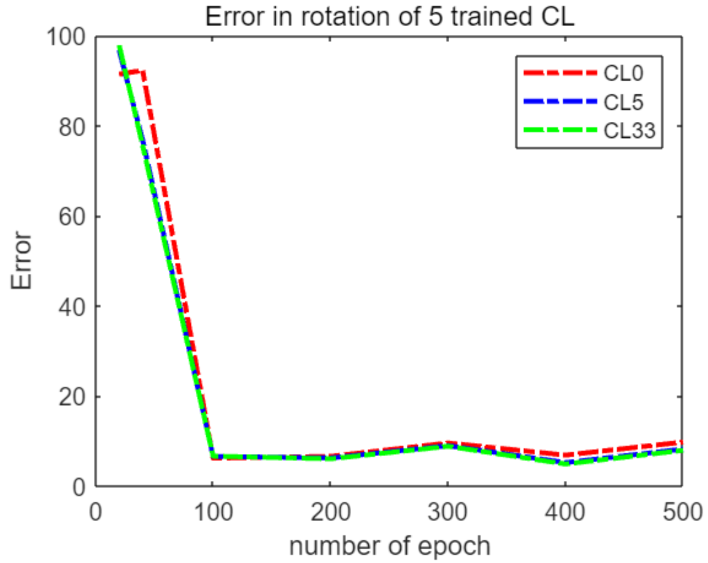
Epoch 500

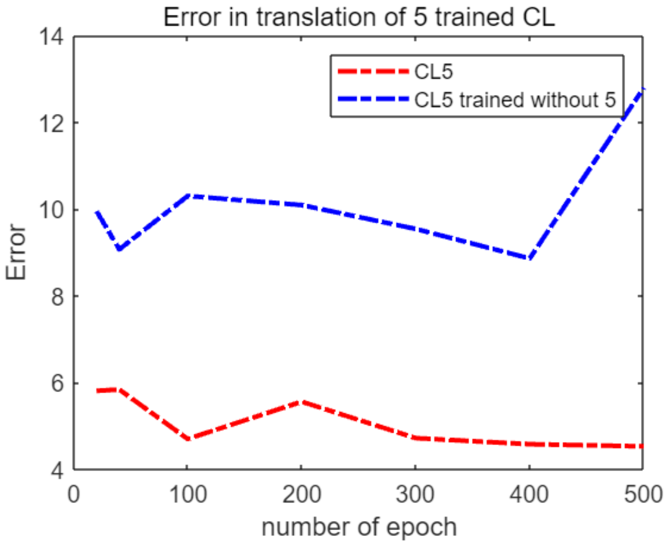


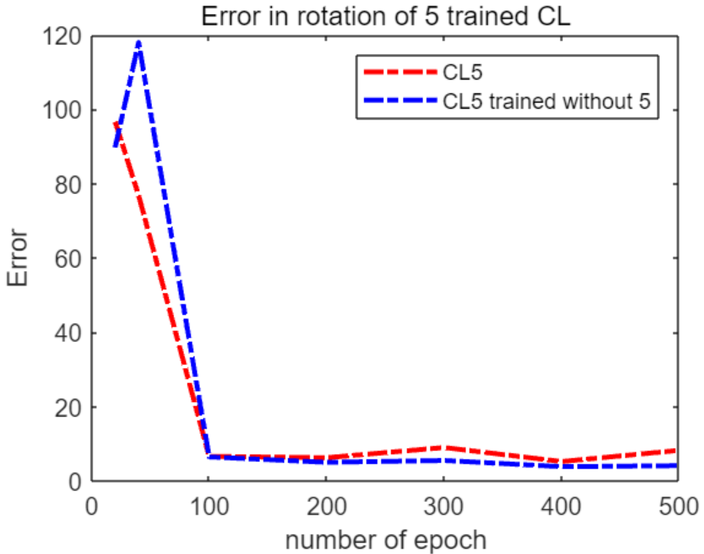


Error Comparison of different chosen CL Posenet with Heaviside loss









Mapnet

Similar to PoseNet, MapNet also trains a DNN that estimates the 6-DoF camera pose. The main difference is that MapNet considers both the loss of the per-image absolute pose and the loss of the relative pose between image pairs. Which is the visual odometry in this case.

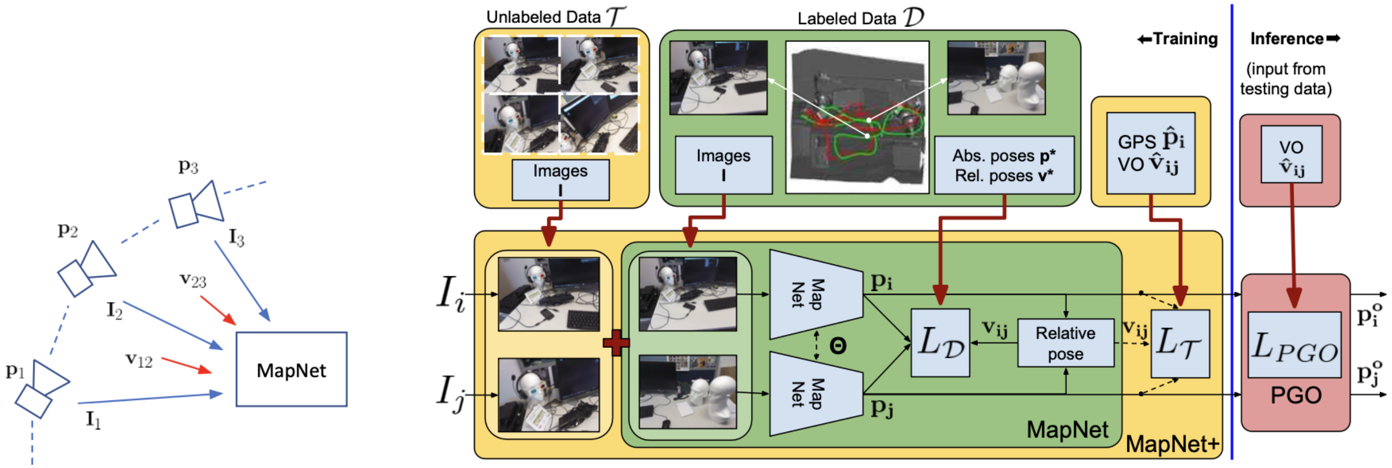
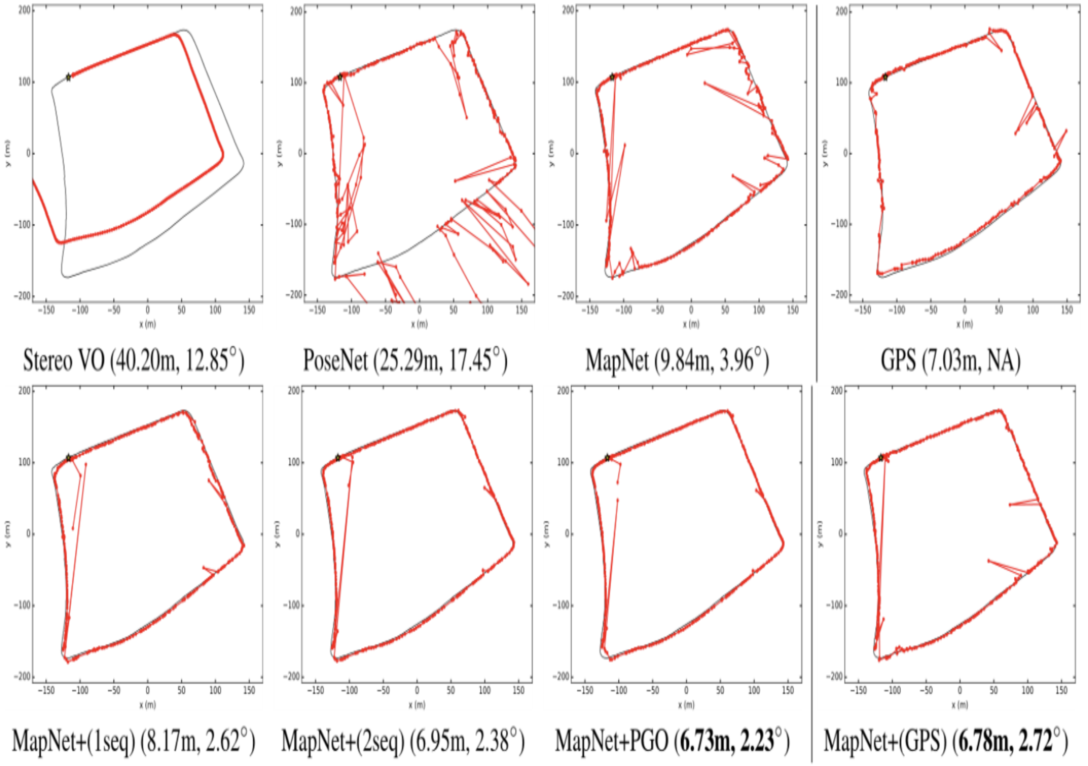


Figure 2: **Left**: MapNet learns a general map representation directly from input data, including images, visual odometry (VO), and other sensory inputs. **Right**: Data flow for our proposed algorithms. MapNet enforces geometric constraints between relative poses and absolute poses in network training. MapNet+ fuses other inputs such as visual odometry to update maps with self-supervised learning. MapNet+PGO performs PGO at testing time to further improve accuracy.

[6] Brahmbhatt et al.

Mapnet



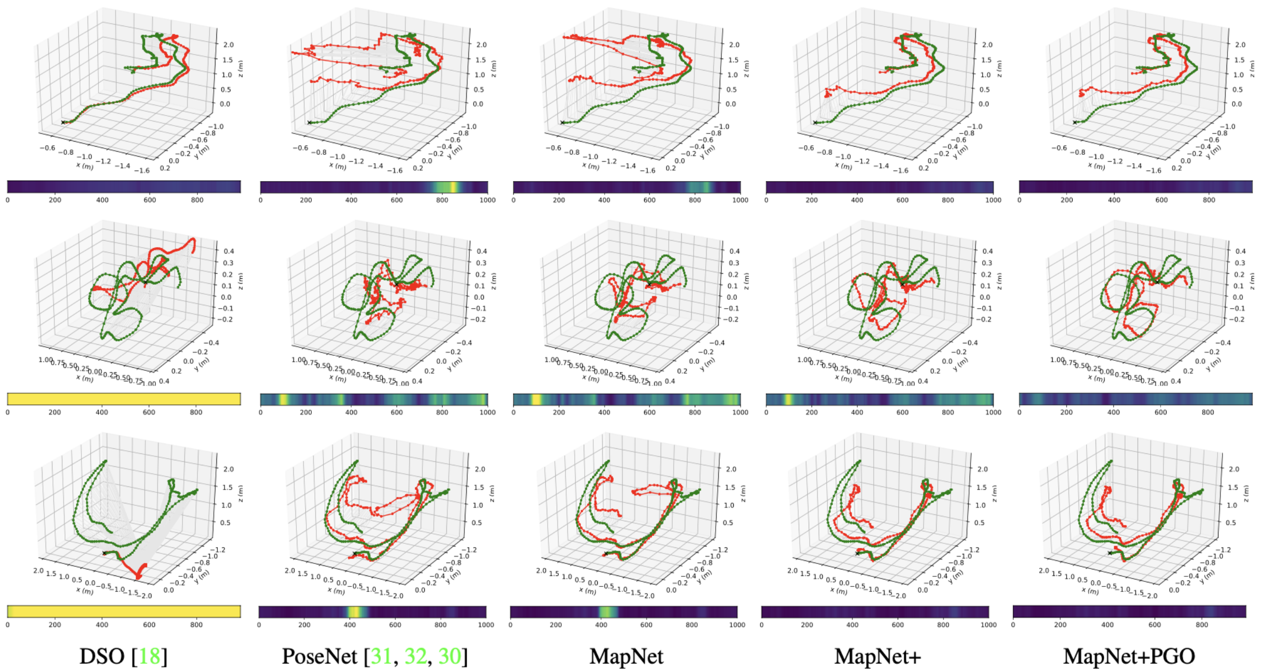
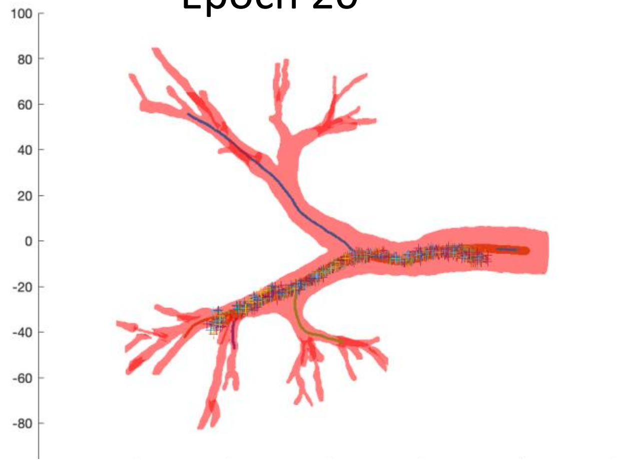


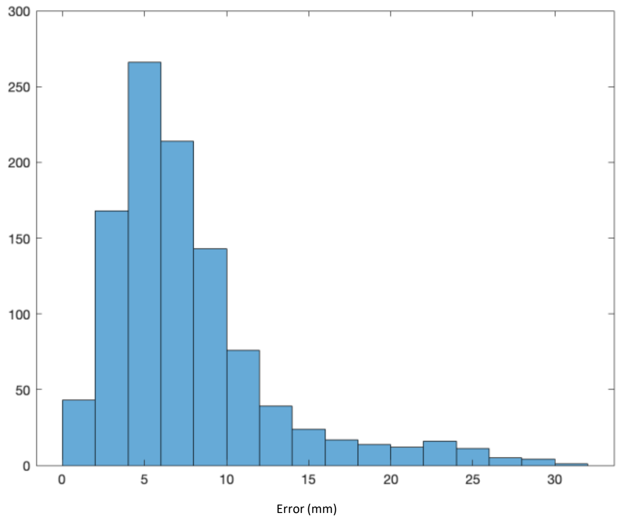
Figure 4: **Camera localization results on 7-Scenes dataset [48]**. For each subfigure, the top 3D plot shows the camera trajectory (green for the ground truth and red for the prediction), and the bottom color bar shows rotation error for all the frames. From top to bottom, the three testing sequences are: Redkitchen-seq-03, Heads-seq-01, and Redkitchen-seq-12. See Table 3 for quantitative comparison.

[6] Brahmbhatt et al.

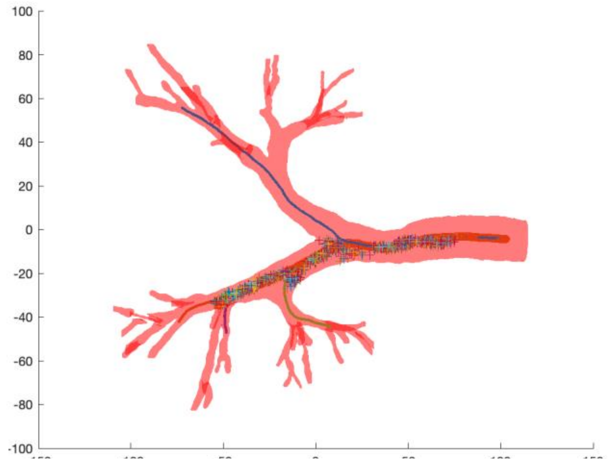
Original MapNet, Train on CL 0,2,5,8,33, Test on CL5

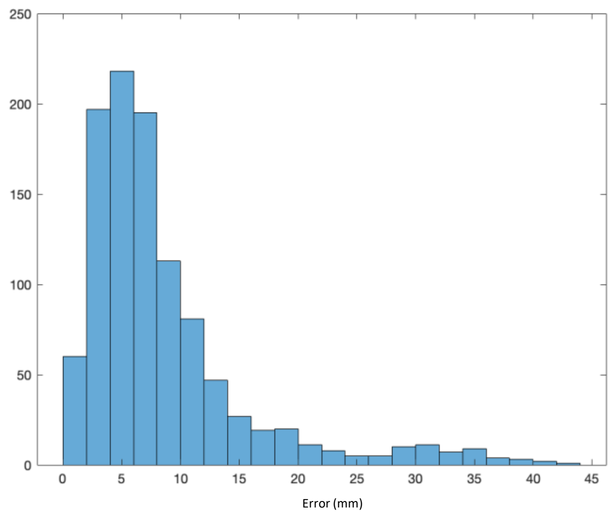
Epoch 20



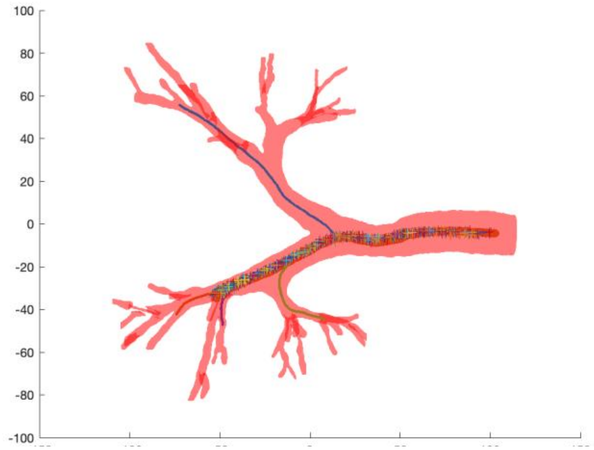


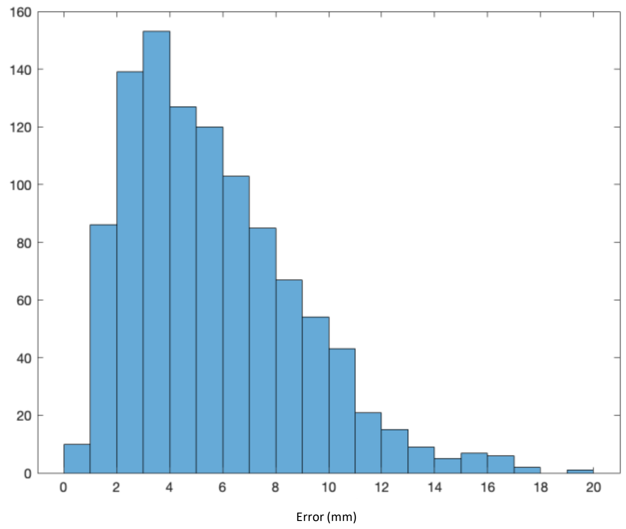
Epoch 40





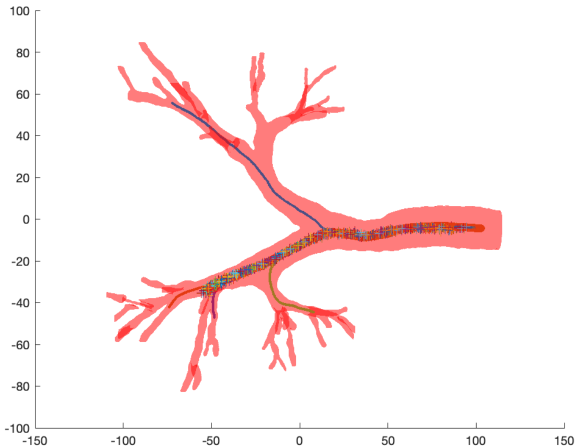
Epoch 100

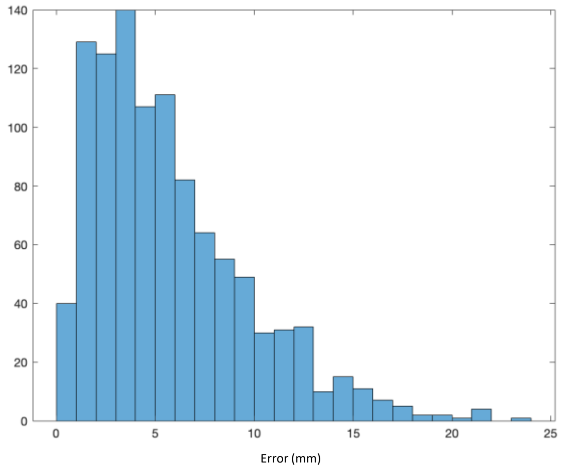




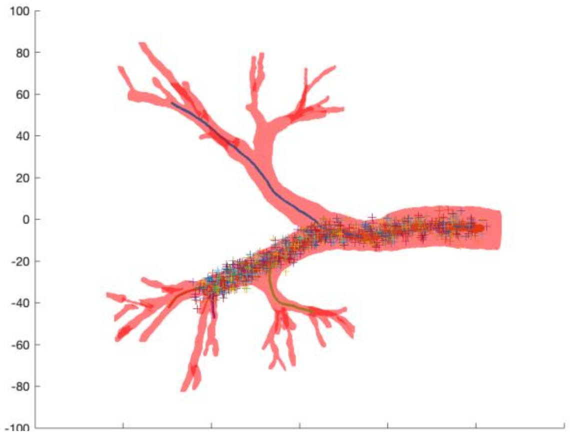
Original MapNet, Train on CL 0,2,5,8,33, Test on CL5

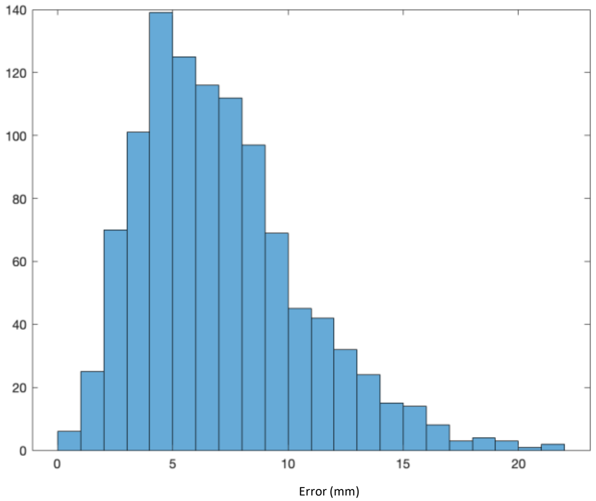
Epoch 200





Epoch 300

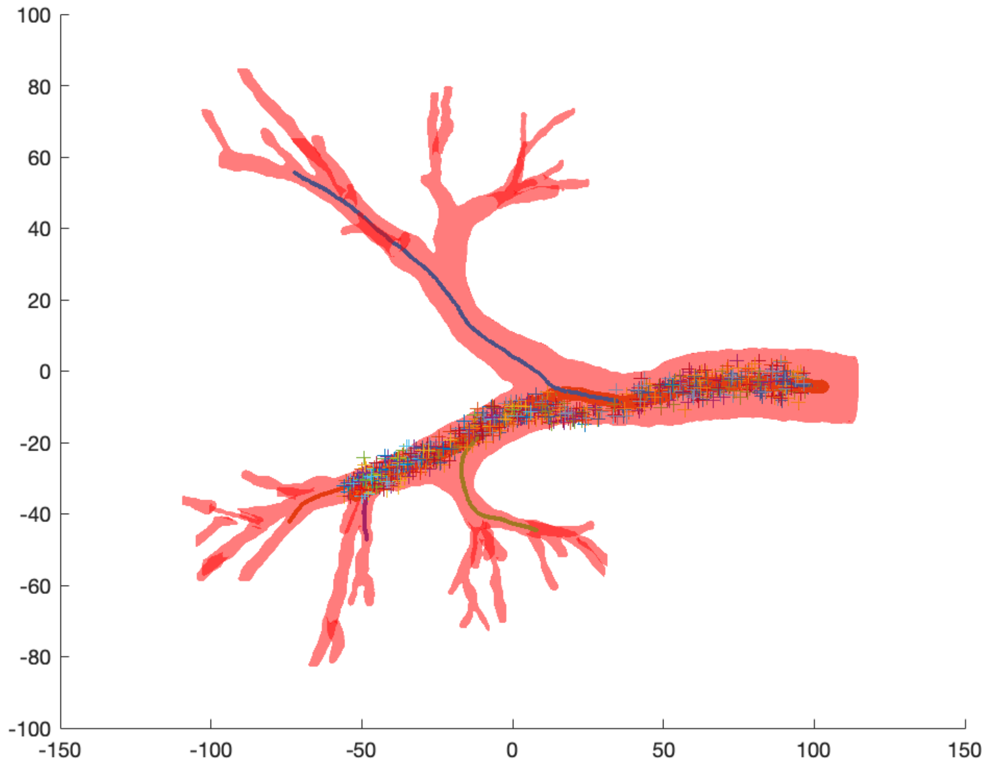




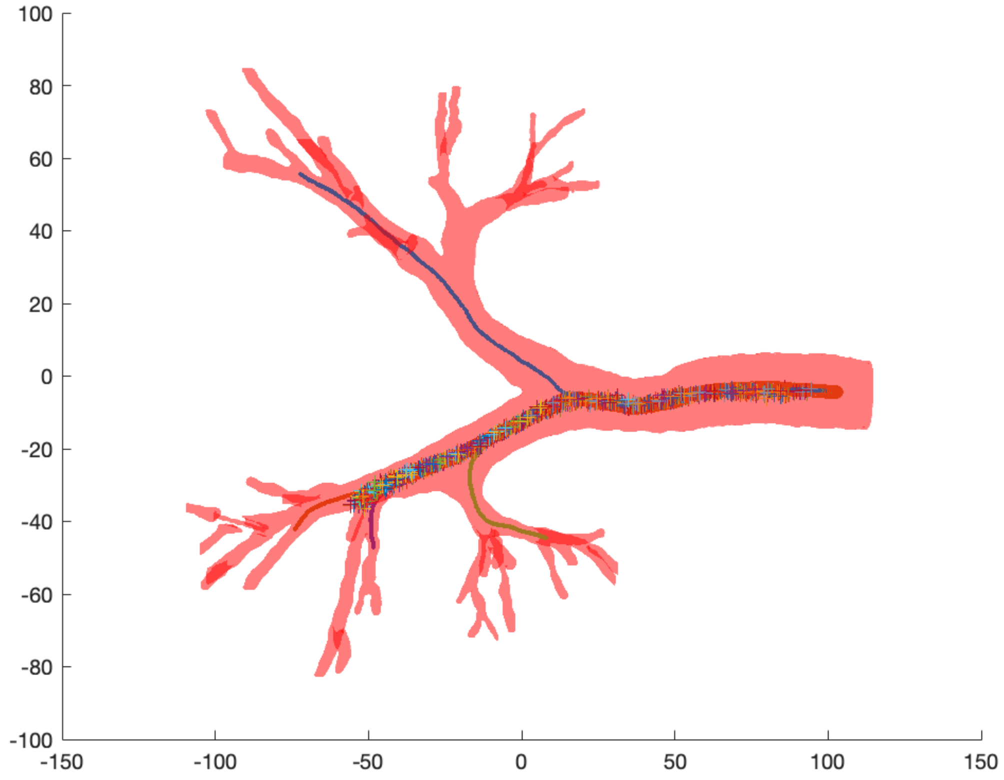
Imperial College London

The Hamlyn Centre for Robotic Surgery

Mapnet compaired with Posenet+heavi



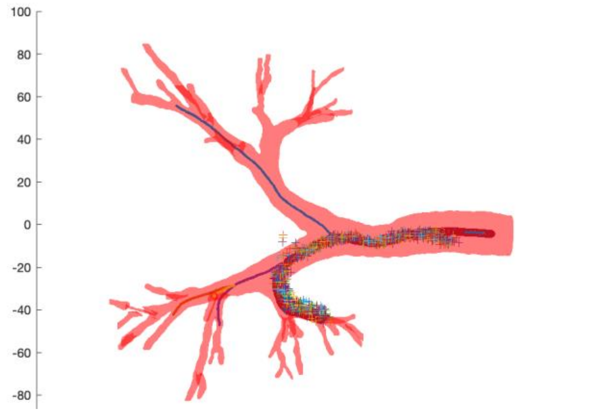
Posenet with Heaviside loss CL5 200 epoch

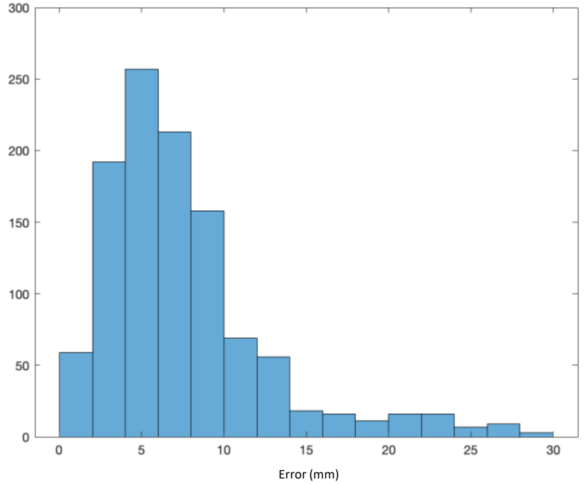


Mapnet CL5 200 epoch

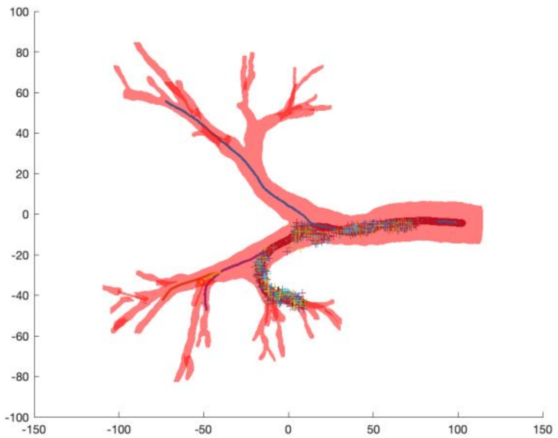
MapNet, Train on CL 0,2,5,8,33, Test on CL33

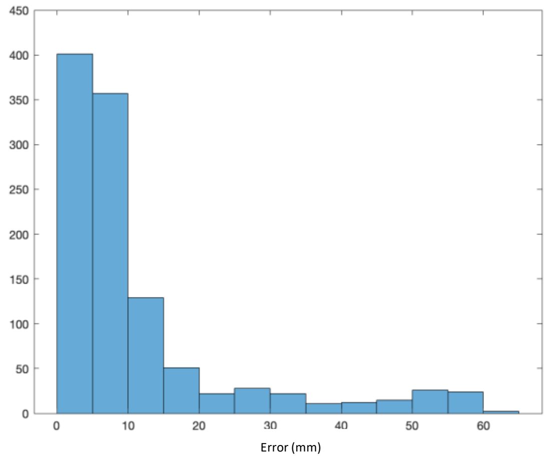
Epoch 20



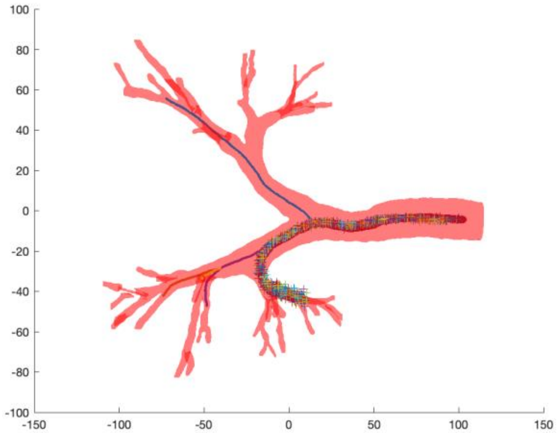


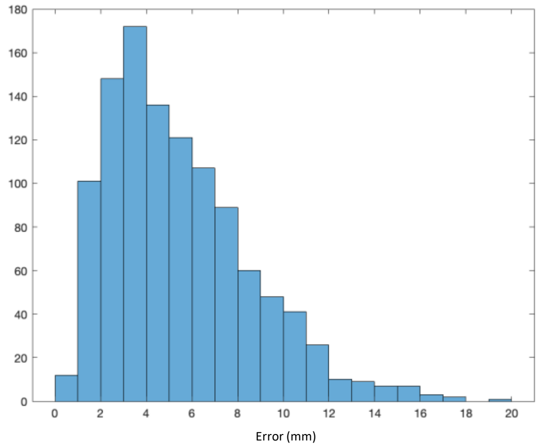
Epoch 40





Epoch 100



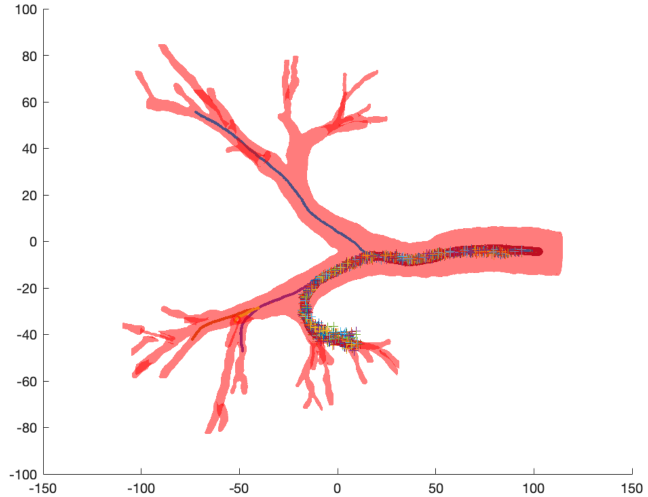


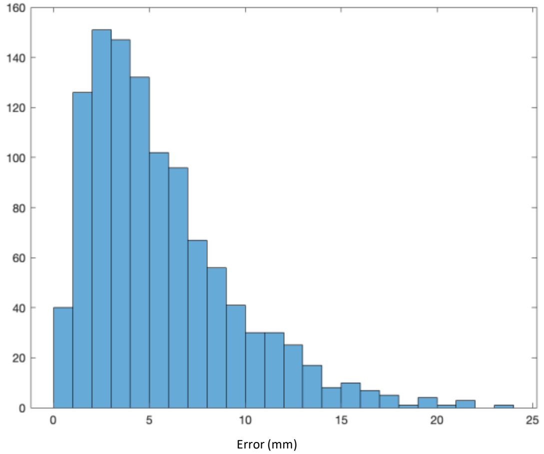
Imperial College London

The Hamlyn Centre for Robotic Surgery

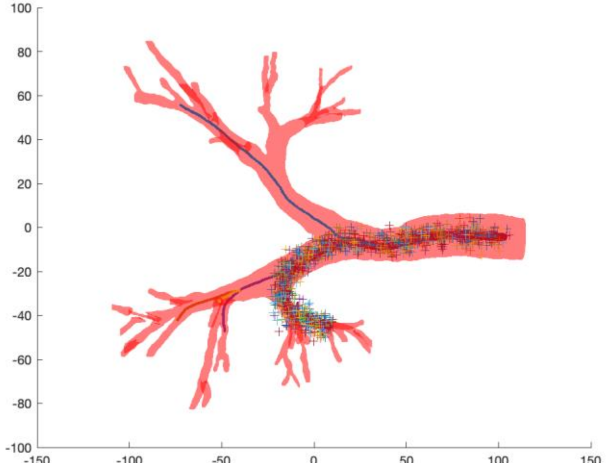
MapNet, Train on CL 0,2,5,8,33, Test on CL5

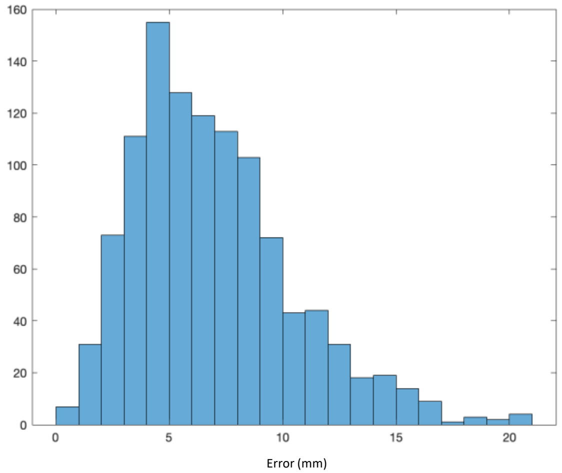
Epoch 200



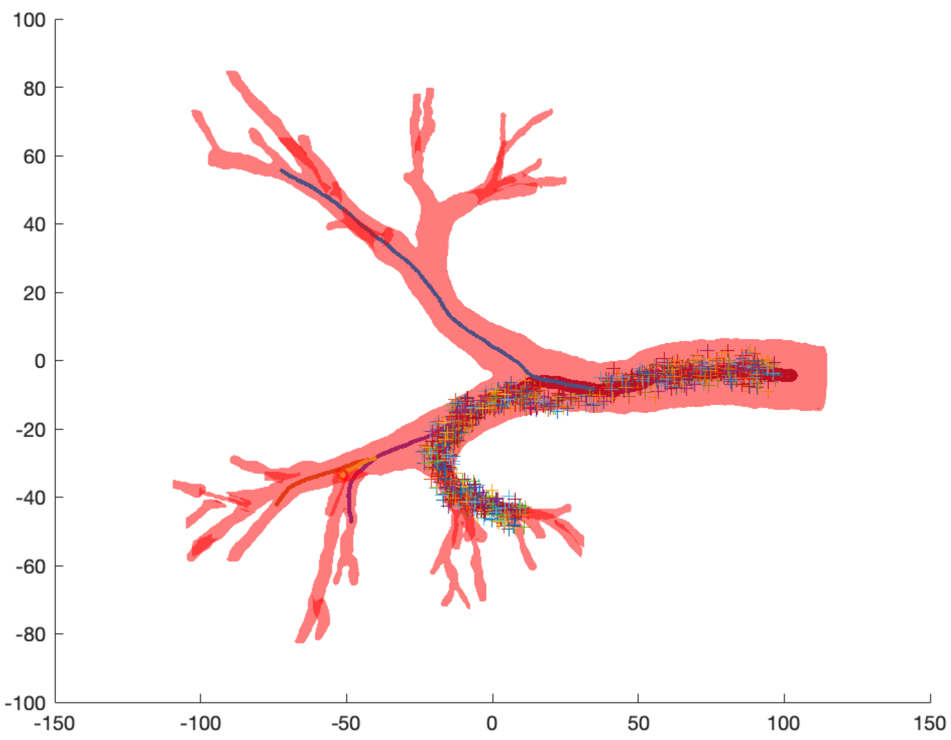


Epoch 300

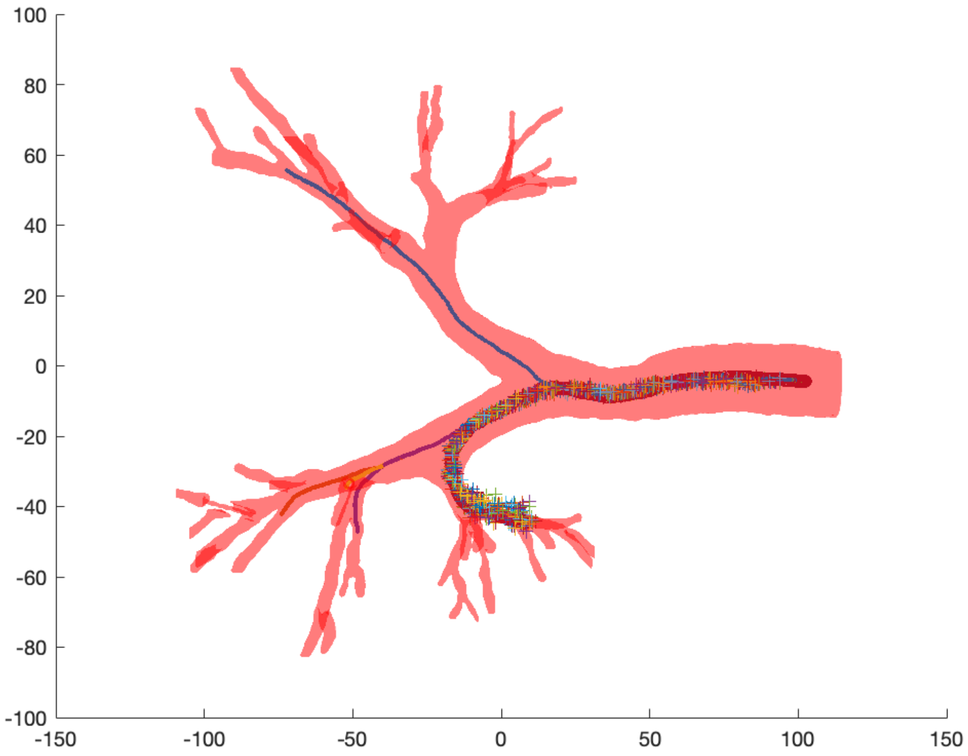




Mapnet compaired with Posenet+heavi



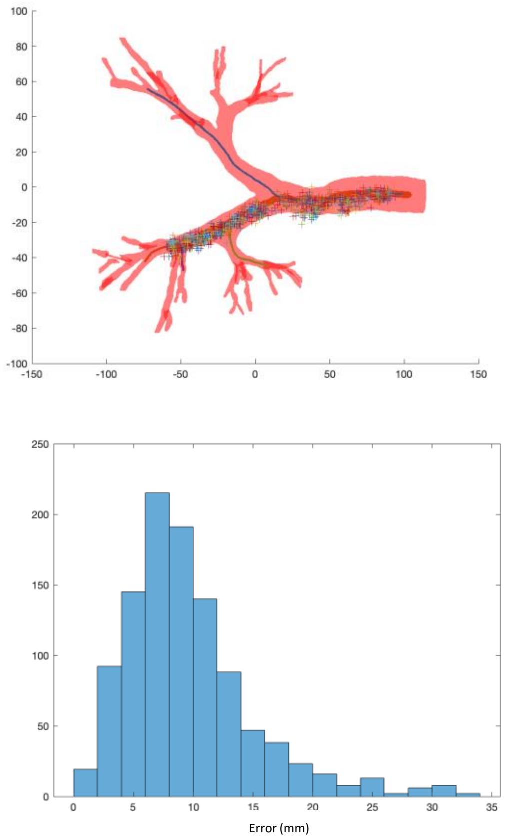
Posenet with heaviside loss CL33 epoch 200



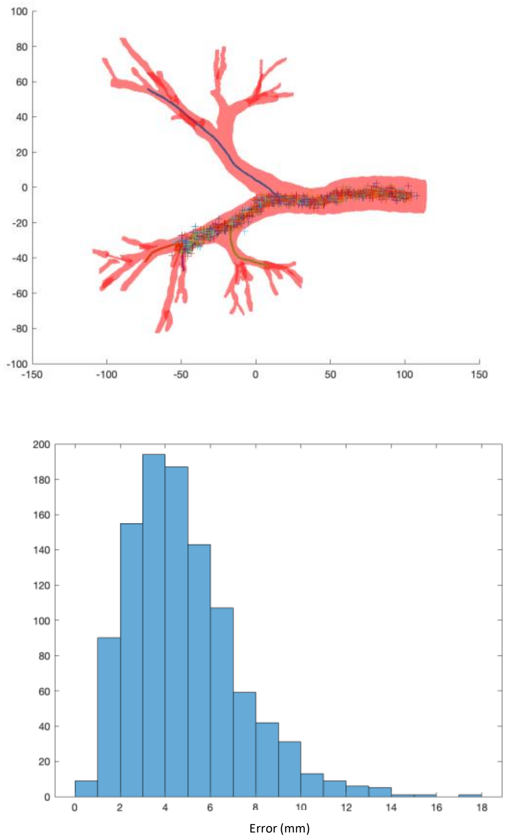
Mapnet CL33 epoch 200

MapNet using Hevaiside loss, Train on CL 0,2,5,8,33, Test on CL5

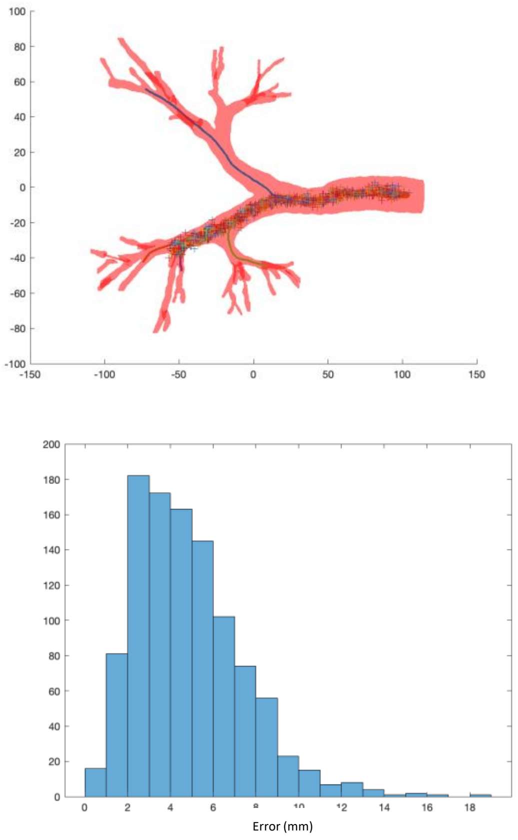
Epoch 20



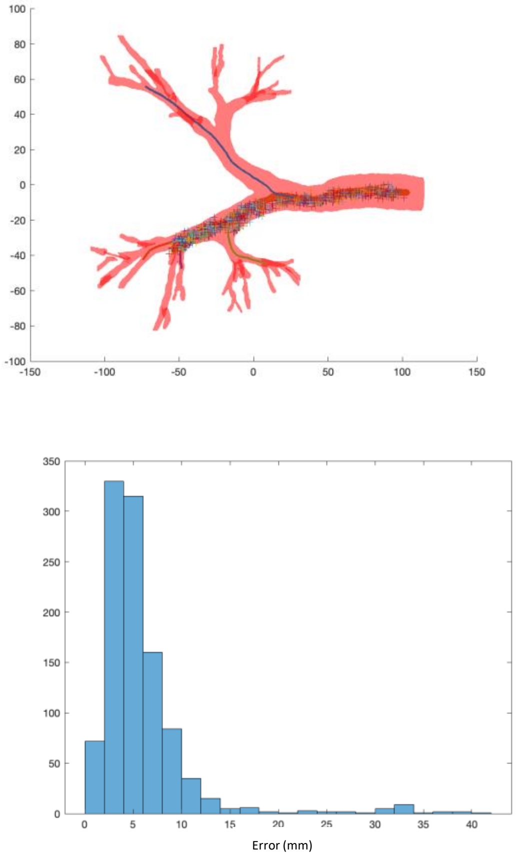
Epoch 40



Epoch 100

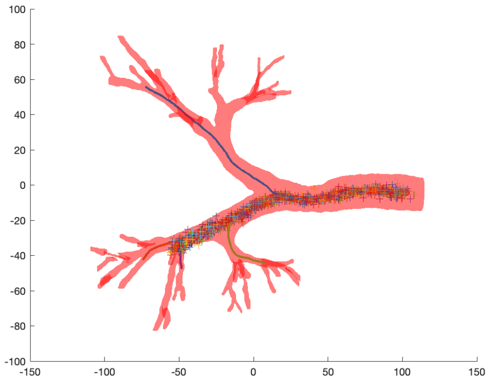


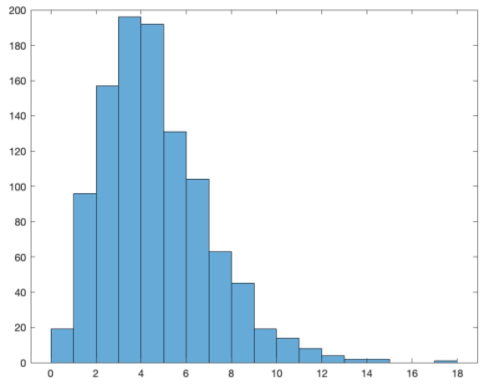
Epoch 200



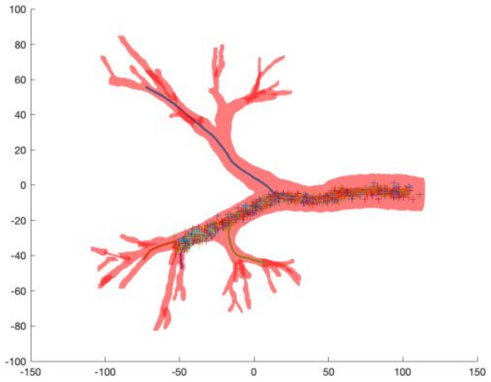
MapNet using Hevaiside loss, Train on CL 0,2,5,8,33, Test on CL5

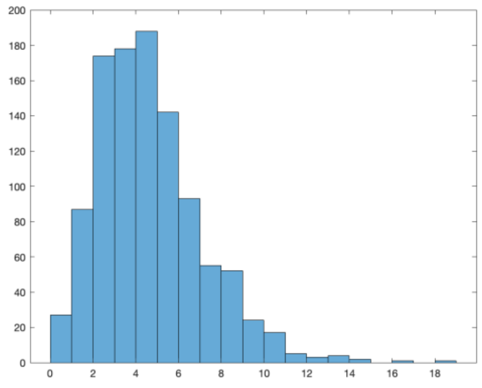
Epoch 300



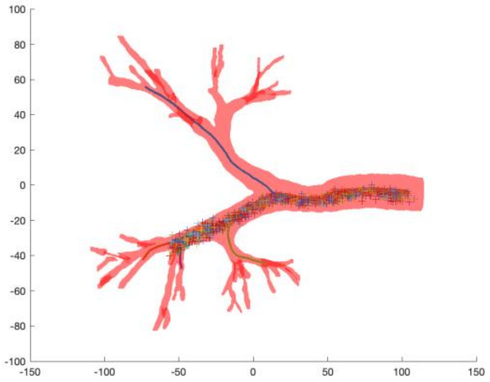


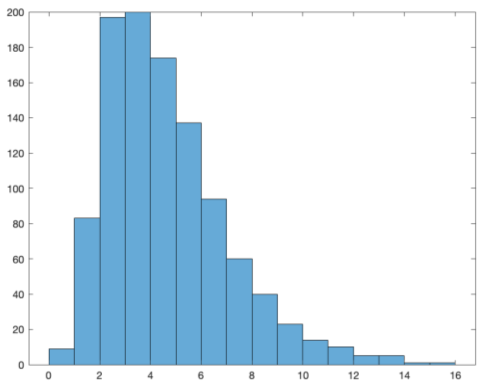
Epoch 400



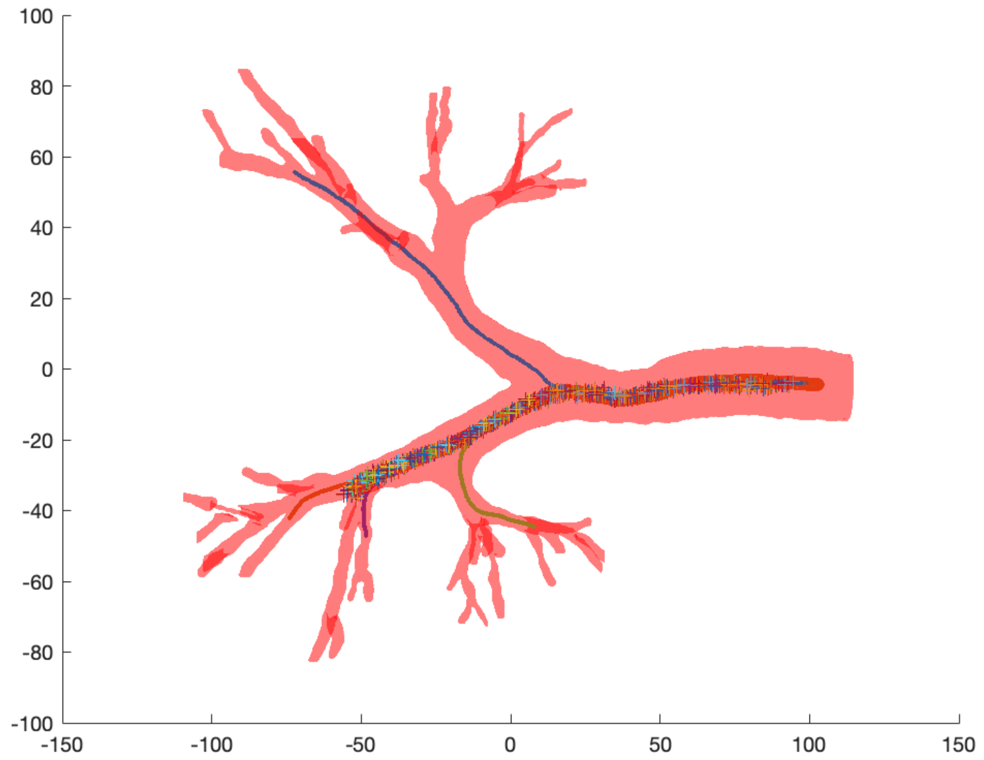


Epoch 500

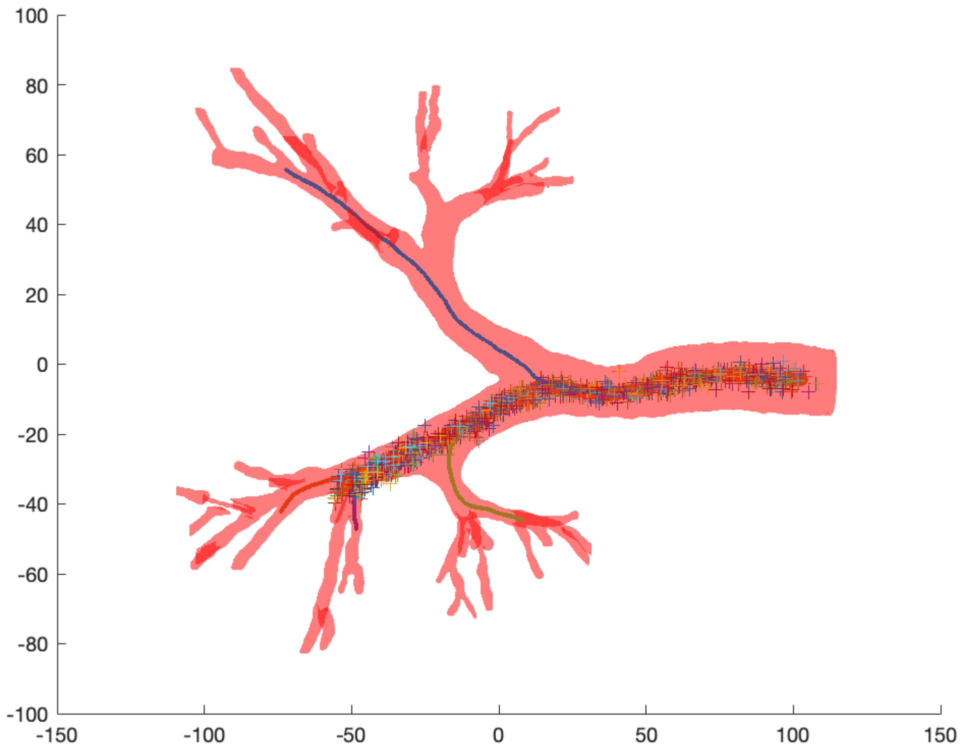




Mapnet with heavi compaired with Mapnet

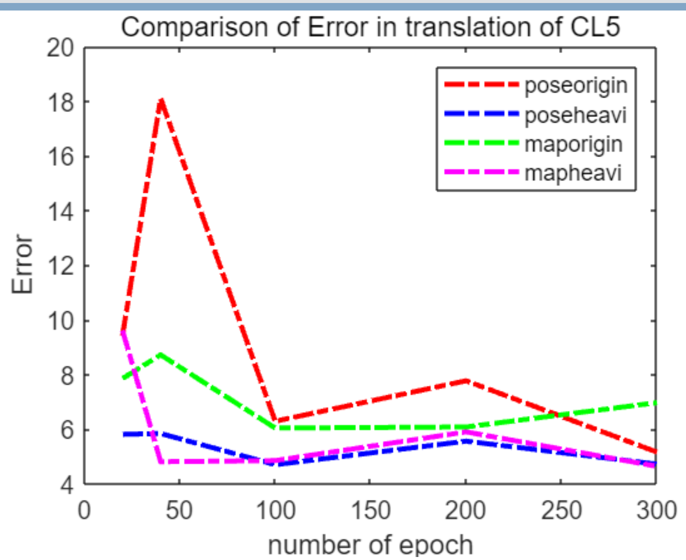


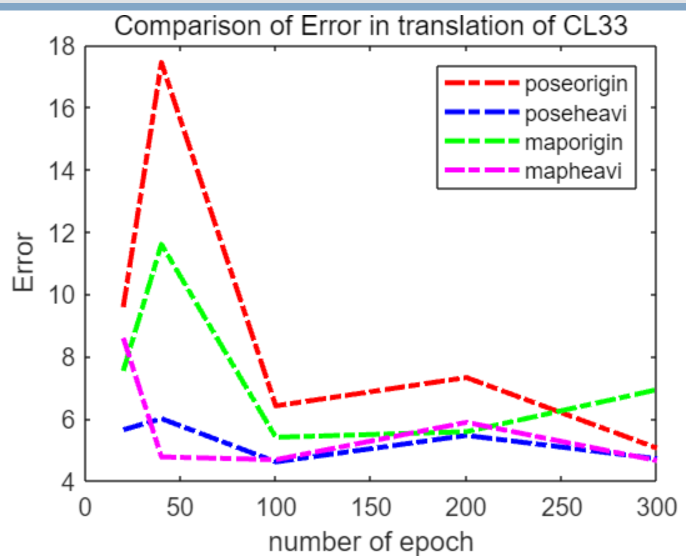
Mapnet CL5 200 epoch

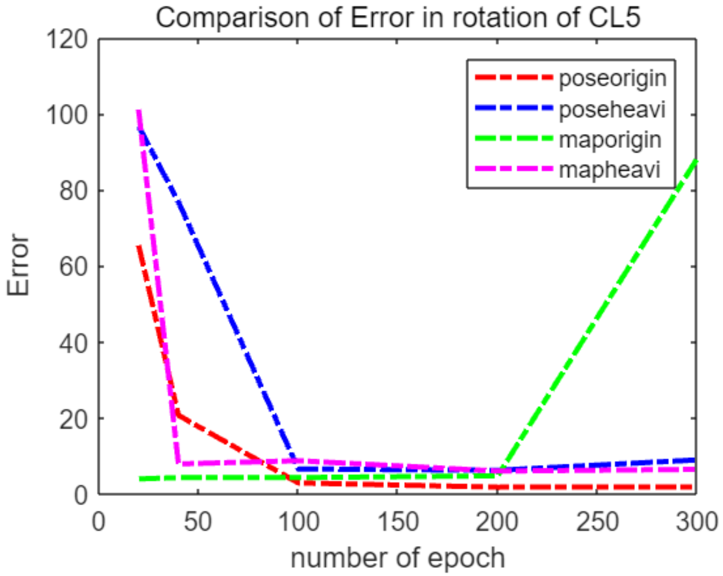


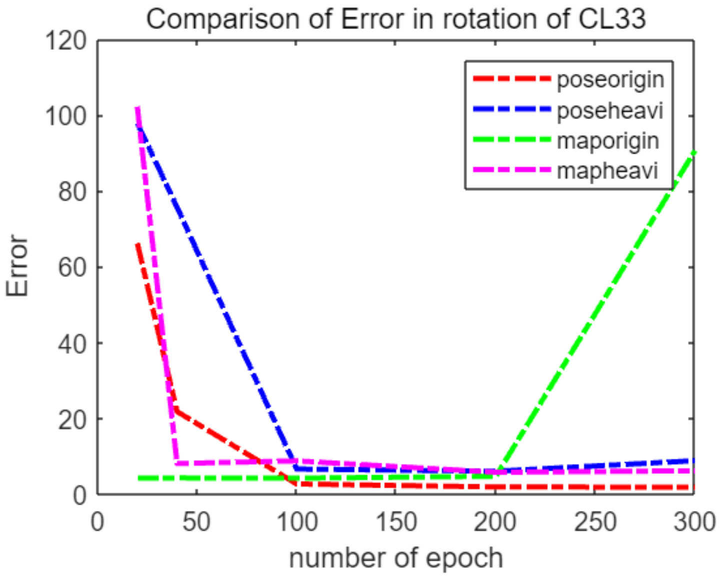
Mapnet with heaviside loss CL5 300 epoch

Error Comparison









Future Work

(1): As we have access to the centreline labels, we could train the model to predict which centreline the camera is currently on, and how far along it is into the lung. This would reduce the number of parameters required to learn (6D pose) and more efficiently use the gathered training data.

(2): Improve the heaviside loss function, choose better hyperparameters (Bayesian hyperparameter optimisation), or choose different hyperparam- eters as the camera progresses (smaller thresholds for thinner airways)

(3): Apply temporal consistency loss to enforce temporal consistency by heavily punishing predictions which were a certain distance away from the previously predicted location.

(4): Other methods could be used to improve the internal consistency of the camera's path. Examples could be the use of a Kalman filter to maintain a constant state estimate that is continually updated at each frame to reduce large jumps.

Technological issues

Computer environment unsuitable → Configure the environment

Training is too slow on personal computers → Call GPU for training/Training on our own and lab computers

Original codes are lost → Update the code from Rema.

Conclusion

(1): We give an overview of the reasons for accurate, safe, and reliable endoscopic camera localisation in a medical setting.

(2): We implemented the model PoseNet, which takes in the depth map and returns a 6D pose estimate

(3): We altered the loss function of PoseNet, useed a heaviside loss to punish estimates that were outside of a bounding box of the target

(4): Measure the performance of these two models (including split the test dataset from train dataset) and compare

(5): Implement a new model Mapnet and evaluate its performance

(6): Experimentally add the Heaviside loss to Mapnet

Acknowledgment

Thanks to the assistance of Matina, Maxim and Haozheng, we complete this project. Although we encountered some difficulties in the middle of the process, we solved most of them through constant communication and access to information. This was the first time the four of us had been exposed to a project in deep learning and we got a lot out of it, thank you!

Group contributions

	Contributions
Demin, Li	Posenet implementation, environment configuration, data training and data evaluation
Ziyan, Wang	Model implementation and optimization
Yuze, Wang	Data Operating, slides design and presentation of introduction and pipeline
Xinxin, Li	Data operating and visualize the results, error histogram plot

Reference

[1]Lung disease - Xpress Pathlabs | Liver Function Test, Kidney Function Test, Full Body Checkup, Kidney Home Test, Urinalysis Urine Test, [online available]: "https://www.dailypioneer.com/2018/pioneer-health/chronic-obstructive-lung-disease-cases-in-india-up-from-28-to-55-million-between-1990-2016--study.html"

[2]Respiratory Disease (icloudhospital.com), [online available]: “https://icloudhospital.com/specialties/respiratory-disease”

[3] Maxim J. Ramsay King, "Endoscopic Robot Localisation in the Human Body," 2022.8.31

[4] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

[5] Kendall A, Grimes M, Cipolla R. Convolutional networks for real-time 6-DOF camera relocalization. CoRR abs/1505.07427 (2015)[J]. arXiv preprint arxiv:1505.07427, 2015.

[6] S. Brahmbhatt, J. Gu, K. Kim, J. Hays, and J. Kautz, “Geometry-aware learning of maps for camera localization,” 2017. 2.2.2

Thank you
Any Questions?