

Running Test and Train in the instance

(Please use this page only after completing the setup of the Instance as per the Launch and Setup guide.)

This page is a guide to train and test of KPConv Model described as the following architecture:

```
'simple',
'resnetb',
'resnetb_strided',
'resnetb',
'resnetb_strided',
'resnetb_deformable',
'resnetb_deformable_strided',
'resnetb_deformable',
'resnetb_deformable_strided',
'resnetb_deformable',
'nearest_upsample',
'unary',
'nearest_upsample',
'unary',
'nearest_upsample',
'unary',
'nearest_upsample',
'unary'
```

1. Training the Model

Training the model takes around 32 GB RAM and required 16 GB Graphics for the Dales dataset (This requirement is dataset dependent). Following are the steps: (If the dataset have difference in the classes compared to DALES, change it in the `dataset/DALES.py` at line #97) 1. Connect to the ssh terminal. 2. Utilise tmux to create a new session using the following command `tmux attach` 3. Run the following commands:

```
cd {Location of the KPConv}/
conda activate aerotronic
python -u training_DALES.py > filename.txt
```

filename.txt will contain the output log of the training.

4. Detach from the tmux session using the following * Press Ctrl-A and then press D or (if this doesn't work) * Press Ctrl-B and then press D 5. Now the session is running and the current ssh session can be safely closed.

2. Testing the trained model

Testing the model takes around 16 GB RAM and required 16 GB Graphics for the Dales dataset (This requirement is dataset dependent). Following are the steps:

1. Edit the `test_any_model.py` file at line #235 as per the training log of your choice
`chosen_log = 'Log_YYYY-MM-DD_HH-MM-SS'`
This sets the testing script to look for the trained model in 'results/Log_YYYY-MM-DD_HH-MM-SS'
2. Connect to the ssh terminal.
3. Utilise tmux to create a new session using the following command
`tmux attach`
4. Run the following commands:

```
cd {Location of the KPConv}/
conda activate aerotronic
python -u test_any_model.py > output_filename.txt
```

output_filename.txt will contain the output log of the training.

4. Detach from the tmux session using the following * Press Ctrl-A and then press D or (if this doesn't work) * Press Ctrl-B and then press D 5. The session is running. The current ssh session can be safely closed.

3. Expected running times for test and train.

Training is expected to take 3 Days. It can be stopped at any time by using the `Ctrl-C break`.

Testing is expected to take a maximum of 1 hour. (The default number of votes is 100, this may lead to longer test time> 2Hr) To reduce the running time you can edit 'test_any_model.py' at line #197 to

```
tester.test_cloud_segmentation(model, dataset, num_votes=1)
```

4. Model Artifacts and Results

Training model artifacts are stored in `results/Log_YYYY-MM-DD_HH-MM-SS`

Testing results are stored at `test/Log_YYYY-MM-DD_HH-MM-SS/predictions`

To check the accuracy of the testing `test_accuracy.py` can be utilized. Edit line #19 to the model predicted files which are by default in `test/Log_YYYY-MM-DD_HH-MM-SS/predictions`. `test_accuracy.py` compares the files from the prediction files to ground truth files which are stored in `Data/test_bin/`