**Step one:**

* run average\_video.m
* path: WiFiMobNet > data > average\_video

(It will create images from video and save the image path in txt file which will be later used in teacher network TeacherNet for testing demo.py file which will give the key points of the images. After running select the folder of the video. The images will be saved in the same folder as video according to this code.)

Change the path:

file\_name = 'your\_dir\WiFiMobNet\TeacherNet\examples\demo\list-coco-demo.txt';

**Step two:**

* run append\_csi\_files.m
* path: WiFiMobNet > CSI > append\_csi\_files

(It will average the csi files and create convert the dat file to mat file. After running select the folder of the CSI. The CSI mat file will be saved in the same folder as CSI.)

**Step three:**

* run demo.py
* path: WiFiMobNet > TeacherNet > demo.py

(This is the test code of teacher network, and it will take input from the images and give output keypoint from the image)

Change the input path:

args.outputpath = 'your\_dir/WiFiMobNet/TeacherNet/examples/res/'

Change the output path:

args.inputlist = 'your\_dir/WiFiMobNet/TeacherNet/examples/demo/list-coco-demo.txt'

Change path from other files as well:

* path: WiFiMobNet > TeacherNet > dataloader.py
* file name: dataloader.py
* function name: DetectionLoader

Change the path:

self.det\_model = Darknet("your\_dir/WiFiMobNet/TeacherNet/yolo/cfg/yolov3-spp.cfg")  
Change the path:

self.det\_model.load\_weights('your\_dir/WiFiMobNet/TeacherNet/models/yolo/yolov3-spp.weights')

**Step four:**

* run preapre\_train\_data.m
* path: WiFiMobNet > preprocess > preapre\_train\_data.m

(It will prepare the data for student network. The program will create a mat file which will consist of CSI data, images and the Json matrix. The mat folder name is train80singleperson.)

Change the path:  
csi\_folder\_path = 'your\_dir\WiFiMobNet\CSI\_data'

Change the path:

output\_folder = 'your\_dir\WiFiMobNet\preprocess\train80singleperson';

Change the path:

image\_folder\_path = 'your\_dir\WiFiMobNet\videos';

Change the path:

fname = ['your\_dir\WiFiMobNet\TeacherNet\examples\res\alphapose-results.json'];

**Step five:**

* run model train.py
* path: WiFiMobNet > model train.py

(This file will train the model taking the mat file from train80singleperson folder and will take the data csi and joint Matrix.)

Change the input path:

mats = glob.glob('your\_dir/WiFiMobNet/preprocess/train80singleperson/\*.mat')

Change the output path:

torch.save(wisppn, 'your\_dir/WiFiMobNet/Pre-Trained/WiFiMobNet\_weights.pkl')

**Step six:**

* run model test.py
* path: WiFiMobNet > model test.py

(This file will test the model taking the mat file from train80singleperson folder and will take the data csi and frame.)

Change the input path:

data = hdf5storage.loadmat('your\_dir/WiFiMobNet/preprocess/train80singleperson/750.mat', variable\_names={'csi\_serial', 'frame'})

Change the weight path:

wisppn = torch.load('your\_dir/WiFiMobNet/Pre-Trained/wisppn-20190226.pkl', map\_location=device)

**Extra information (The paths)**

* Dataset of CSI: WiFiMobNet > data > CSI\_set
* Dataset of video: WiFiMobNet > data > video\_set
* To average video to image: WiFiMobNet > data > average\_video
* To average CSI and convert it to dat to mat: WiFiMobNet > data > append\_csi\_files
* To copy csi and video files to one folder: WiFiMobNet > data > copy\_files\_into\_one\_folder

**The teacher network:**

* To find keypoints from image: WiFiMobNet > TeacherNet > demo
* The path of the image: WiFiMobNet > TeacherNet > examples > demo > list-coco-demo
* The saved keypoints: WiFiMobNet > TeacherNet > examples > res > alphapose-results

**To prepare data of csi, image and keypoints:**

* Prepare data: WiFiMobNet > preprocess > preapre\_train\_data\_
* Saved prepared data: WiFiMobNet > preprocess > train80singleperson

**The student network train:**

* To train the student network: WiFiMobNet > model train.py
* The input path of student network: WiFiMobNet > preprocess > train80singleperson
* The output weight path: WiFiMobNet > Pre-Trained > WiFiMobNet\_weights.pkl

**The student network test:**

* To test the student network: WiFiMobNet > model test.py
* Input data: WiFiMobNet > preprocess > train80singleperson
* Input weight: WiFiMobNet > Pre-Trained > WiFiMobNet\_weights.pkl