Control robot with voice – documentation

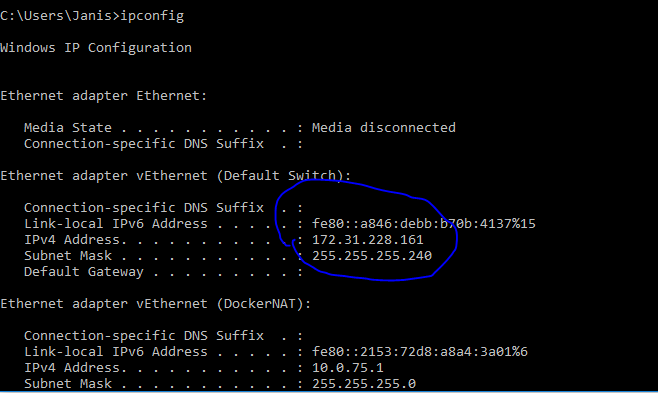
Interface to NRP – how does it work

The interface to the NRP is realized through a service (see *CommandService.py*) using flask framework which is repeatedly called upon by a transfer function from within the NRP (see *transfer\_function.py*). The current command is maintained in a text file (*command\_file.txt*). Upon a call of the service from within the NRP the service provides the NRP with the current voice command which can then be further processed.

Alternatively one could have run the service on the NRP side. This way one would avoid the need to constantly call the service on the local machine since the service just needs to be called once a new command is available. But this makes it necessary to adjust the NRP environment. Since everybody of our team struggled with the installation of the NRP we decided to alter this environment as little as possible.

Interface to NRP – setting it up

First one has to insert the IP of the PC running the service in the transfer function used in the NRP (*transfer\_function.py*). To obtain IP address (on windows) open command line and type



Then navigate to folder containing *CommandService.py* and run the service via *python CommandService.py.*

The service is now running and can be used to mediate between local system and NRP.

User-Interface – how does it work

We implemented a very simple user interface using the library tkinter in python (*BasicInterface.py*). It consists of two buttons – record and quit. Quit ends the interface. Record starts recording via microphone by utilizing functionality from *SpeechInterface.py* which resultingly writes the command to *command\_file.txt*.

Further does *BasicInterface.py* initialize the command to “none” which is the default command for no action.

User-Interface – setting it up

Navigate to the folder containing *BasicInterface.py* via command line and run *python* *BasicInterface.py.* The User-Interface should now be visible.