Autonomy Backpack

Operation

Turn robot on with backpack attached, wait for "Boot complete" to show on robot screen

Connect to WiFi network "Micromelon{4 digit bot id}" password: melon{last 4 digits in SSID}

ssh pi@192.168.4.1

password: raspberry

Once logged in there are some examples in the home folder:

- All .py files must be run with python3
- python module for communicating with the robot is the micromelon folder
  - from micromelon import * # easiest way to import for python usage
- sometimes sudo is required to access the serial gpio
- Some example scripts and their usage below:

```
python3 show_bot_message.py {arg} # will display {arg} on the robot screen and leave
python3 melontest.py # goes through some example usage of the api

# OpenCV example that shows how to capture an image form the pi camera and
# control motor speeds based on a simple follow the light algorithm
python3 cvtest.py
```

For getting images off the pi you can use the raspistill command

- https://www.raspberrypi.org/documentation/usage/camera/raspicam/raspistill.md

```
raspistill -o image.jpg
```

- and then scp that back to your local machine

```
# From local machine
scp pi@192.168.4.1:~/image.jpg ~/Desktop/
```

The robot can be stopped while running code by pressing the right button (this won't necessarily stop the python script on the pi)

Network and Matlab operation

Procedure to log into the pi same as above.

Once logged into pi, run:

```
python3 micromelon_server.py
```

Once it shows as waiting for connection, open a local terminal on your computer or open matlab

Python network operation

```
python3
>>> from micromelon import *
>>> rc = RoverController()
>>> rc.connectIP('192.168.4.1')
>>> # all set
>>> Ultrasonic.read()
```

Small script to display an image from a network backpack

```
import numpy
import cv2

from micromelon import *
rc = RoverController()
rc.connectIP() # default is 192.168.4.1

#image = Robot.getImageCapture(IMRES.R640x480)
#image = Robot.getImageCapture(IMRES.R1280x720)
image = Robot.getImageCapture(IMRES.R1920x1088)

image = image.astype(numpy.uint8)
cv2.imshow('image', image)
cv2.waitKey()
```

Matlab network operation

```
>> pyversion('/usr/local/bin/python3') % your python 3 binary may be different
>> py.importlib.import_module('micromelon')
>> rc = py.micromelon.RoverController();
>> rc.connectIP('192.168.4.1') % this will timeout and show a warning with no robot
>> image = py.micromelon.robot.getImageCapture(py.micromelon.IMRES(2));
## Note py.micromelon.IMRES options are 1, 2, 3 for (640, 480) (1280, 720) (1920, 1088)
>> matlabImage = uint8(image);
>> imshow(matlabImage)
```

Note that the images are returned from the backpack in BGR format (correct for opencv) but will need to be converted to RGB for matlab