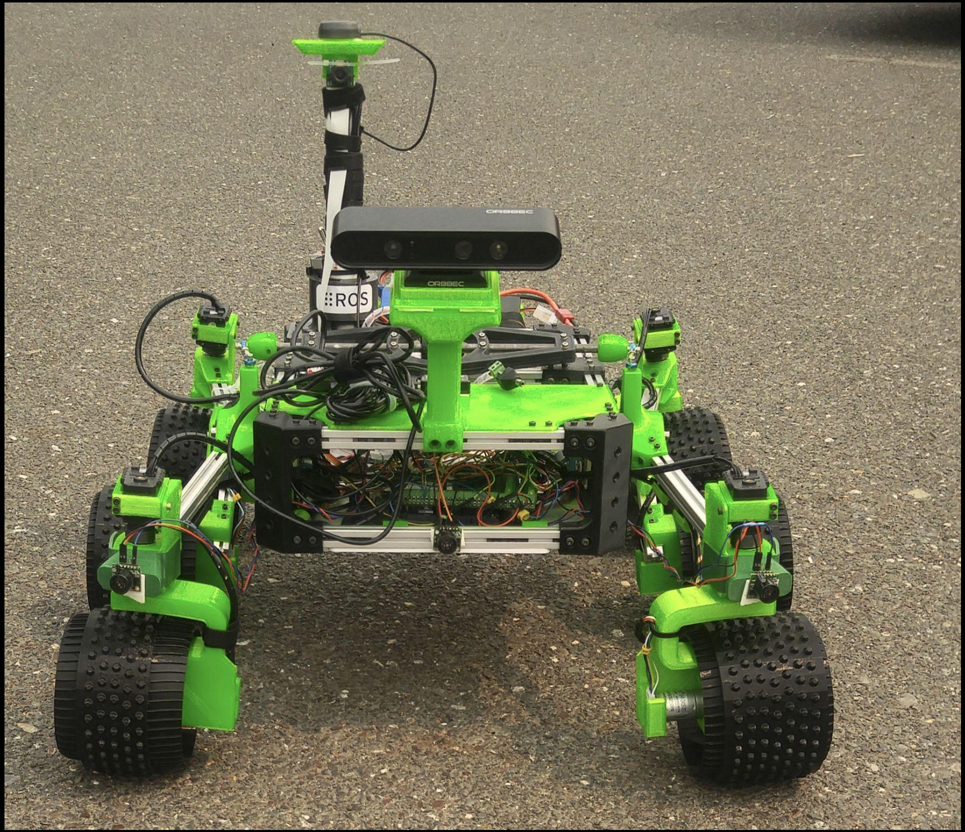
Turning a cheap RoboVac into an Open Source Mapping Tool

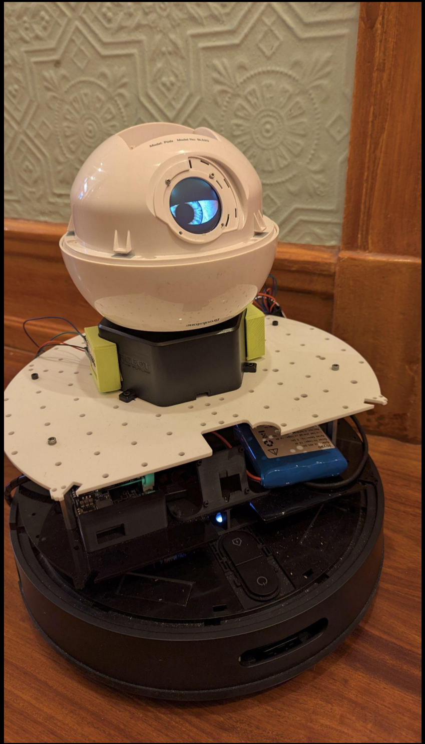
FOSDEM, January 31, 2026
Brussels

Stephen Okay AKA ‘Stef Dillo’
Consulting Roboticist, Familiar Robotics
espressobot@gmail.com

This is my robot...



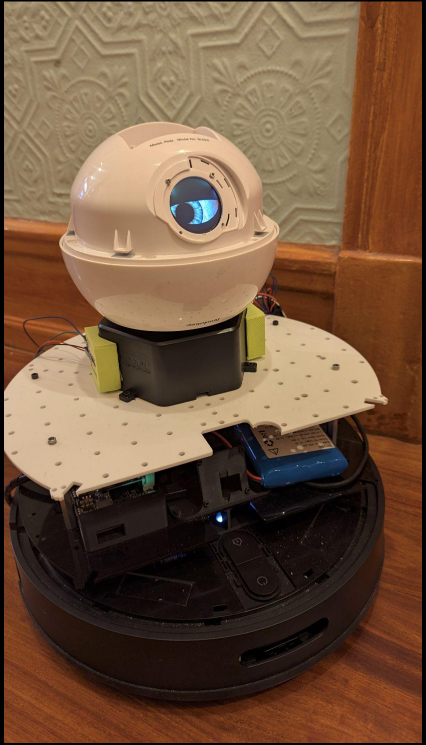
These are some of my other robots...



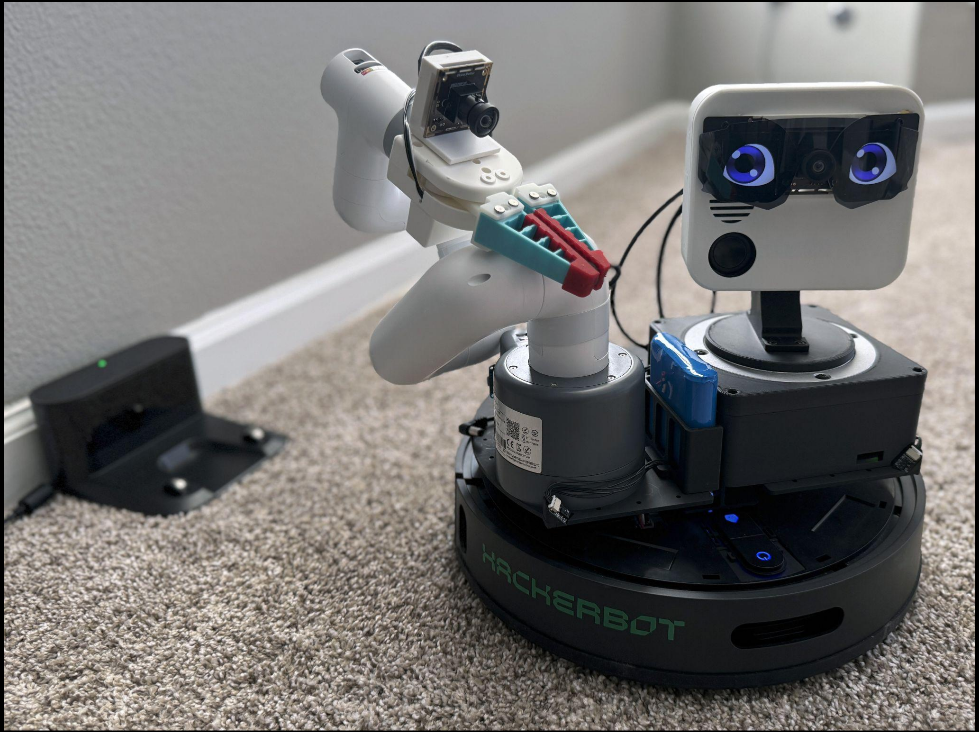
I also get paid to work on Other People's Robots...

...which is like the best job *EVAR*

Let's go back to this robot...



The "Hackerbot"



HACKER BOT INDUSTRIES

Latest Robot from Ian Bernstein (Founder of Misty Robotics & Co-Founder of Sphero)

Applying vision, AI & character/social robotics onto an open "Turtlebot"-like mobile base.

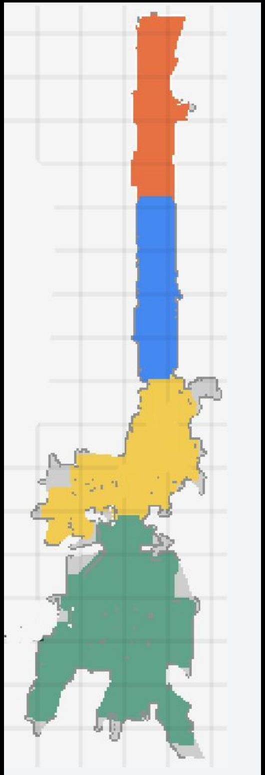
- Commodity robot vacuum base
- On-board LIDAR w/ push-button mapping to map your house for cleaning
- Map saved internally on the robot
  - which just invites hacking...

My goal: one-touch mapping utility robot

-Robot-generated onboard
map, w/ auto-colored "rooms"

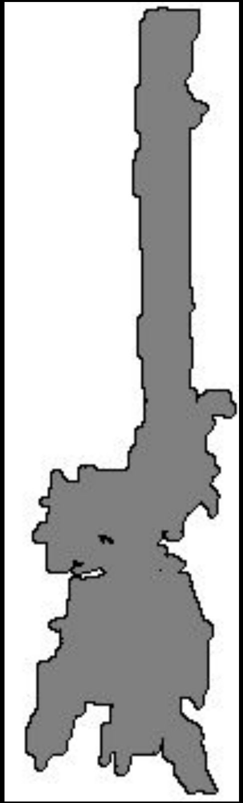
-Stored as bitmap on robot, accessed via "Tuya" app or HA w/ Tuya SDK.

-HA is its own lifestyle
-Tuya SDK is $$$

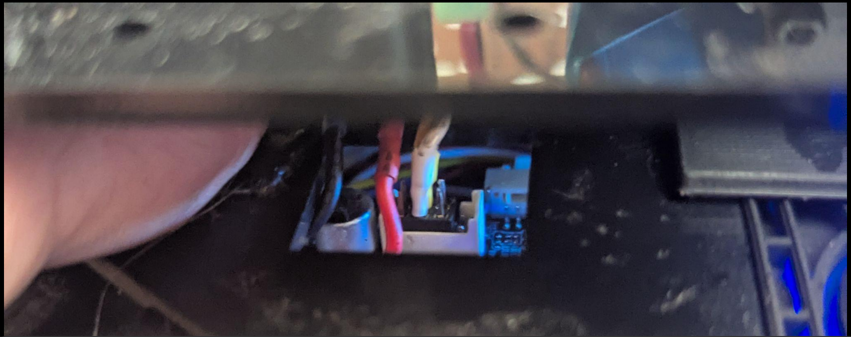


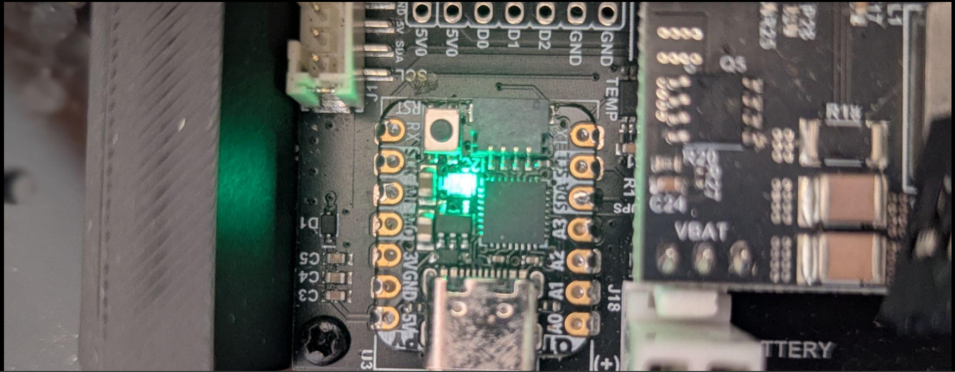
Convert to ROS "Occupancy Grid" map

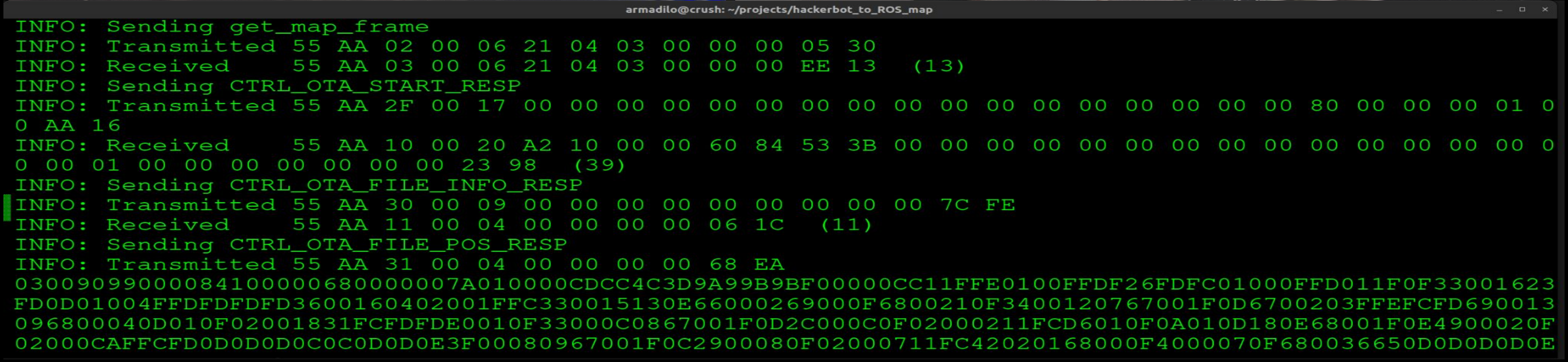
...never have to lug a laptop around again when mapping !



Access to hex commands via serial debug port...

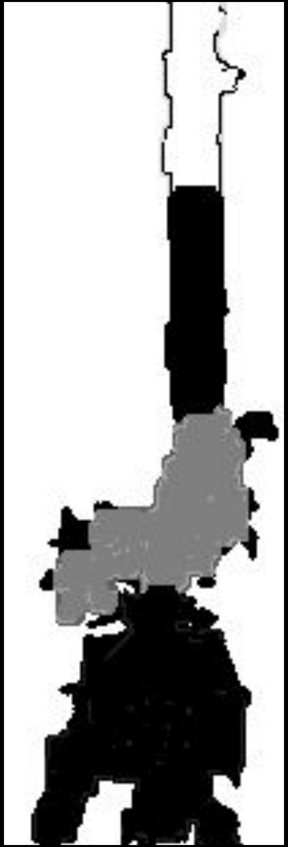


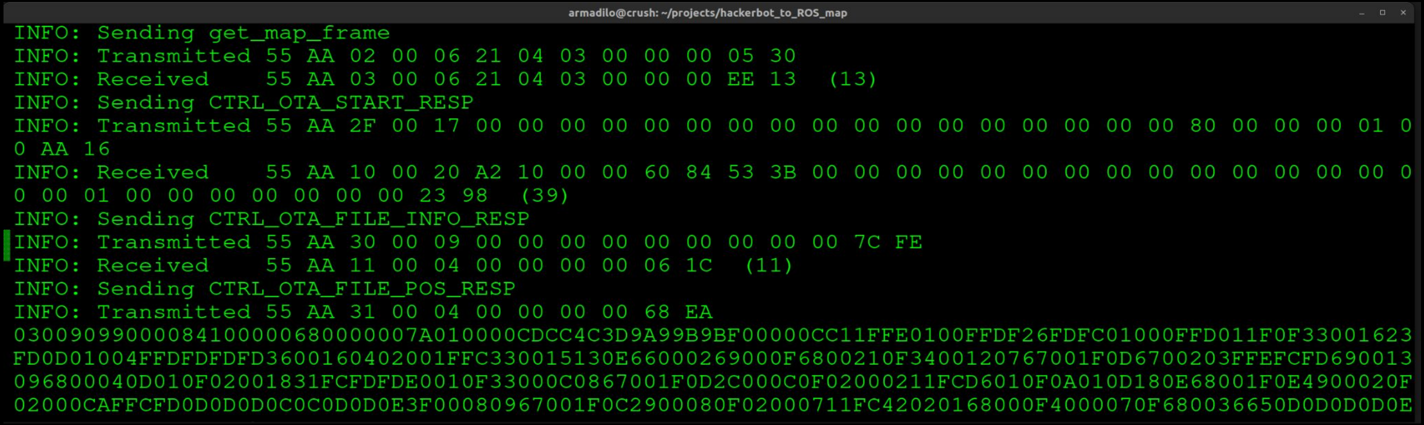




```
INFO: Sending get_map_frame
INFO: Transmitted 55 AA 02 00 06 21 04 03 00 00 00 05 30
INFO: Received    55 AA 03 00 06 21 04 03 00 00 00 EE 13  (13)
INFO: Sending CTRL_OTA_START_RESP
INFO: Transmitted 55 AA 2F 00 17 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 00 01 0
0 AA 16
INFO: Received    55 AA 10 00 20 A2 10 00 00 60 84 53 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0
0 00 01 00 00 00 00 00 00 00 23 98  (39)
INFO: Sending CTRL_OTA_FILE_INFO_RESP
INFO: Transmitted 55 AA 30 00 09 00 00 00 00 00 00 00 00 00 7C FE
INFO: Received    55 AA 11 00 04 00 00 00 00 06 1C  (11)
INFO: Sending CTRL_OTA_FILE_POS_RESP
INFO: Transmitted 55 AA 31 00 04 00 00 00 00 68 EA
0300909900008410000068000007A010000CDCC4C3D9A99B9BF00000CC11FFE0100FFDF26FDFC01000FFD011F0F33001623
FD0D01004FFDFDFDFD3600160402001FFC330015130E66000269000F6800210F3400120767001F0D6700203FFEFCFD690013
096800040D010F02001831FCFDFDE0010F33000C0867001F0D2C000C0F02000211FCD6010F0A010D180E68001F0E4900020F
02000CAFFCFD0D0D0D0C0C0D0D0E3F00080967001F0C2900080F02000711FC42020168000F4000070F680036650D0D0D0D0E
```

Grab the map as a byte stream...





```
INFO: Sending get_map_frame
INFO: Transmitted 55 AA 02 00 06 21 04 03 00 00 00 05 30
INFO: Received    55 AA 03 00 06 21 04 03 00 00 00 EE 13  (13)
INFO: Sending CTRL_OTA_START_RESP
INFO: Transmitted 55 AA 2F 00 17 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 00 00 00 01 0
0 AA 16
INFO: Received    55 AA 10 00 20 A2 10 00 00 60 84 53 3B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0
0 00 01 00 00 00 00 00 00 00 23 98  (39)
INFO: Sending CTRL_OTA_FILE_INFO_RESP
INFO: Transmitted 55 AA 30 00 09 00 00 00 00 00 00 00 00 00 7C FE
INFO: Received    55 AA 11 00 04 00 00 00 00 06 1C  (11)
INFO: Sending CTRL_OTA_FILE_POS_RESP
INFO: Transmitted 55 AA 31 00 04 00 00 00 00 68 EA
0300909900008410000068000000 7A010000CDCC4C3D9A99B9BF00000CC11FFE0100FFDF26FDFC01000FFD011F0F33001623
FD0D01004FFDFDFDFD3600160402001FFC330015130E66000269000F6800210F3400120767001F0D6700203FFEFCFD690013
096800040D010F02001831FCFDFDE0010F33000C0867001F0D2C000C0F02000211FCD6010F0A010D180E68001F0E4900020F
02000CAFFCFD0D0D0D0C0C0D0D0E3F00080967001F0C2900080F02000711FC42020168000F4000070F680036650D0D0D0D0E
```

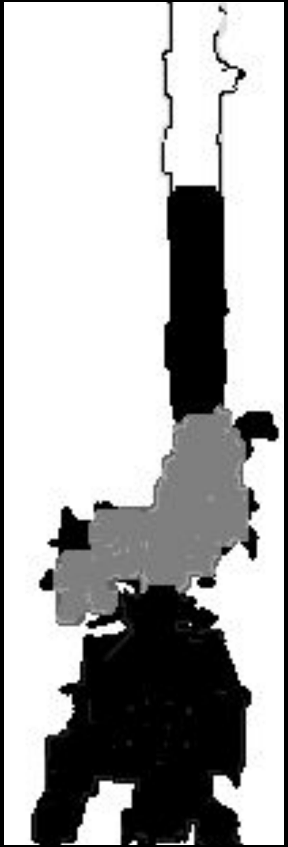
-Occupancy/Obstacle Grid(like most robots)

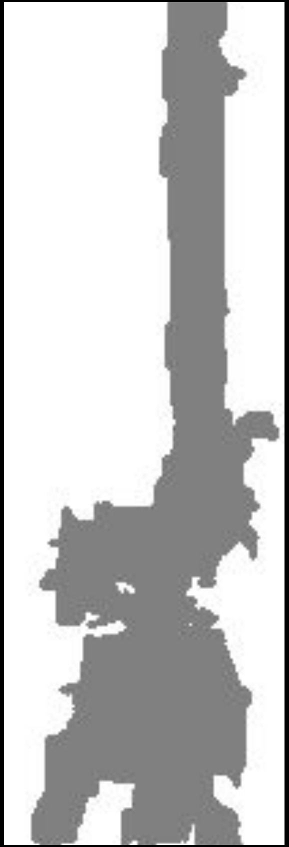
-Short(ish) range LIDAR

Greyscale, but still encoding some color

-White region is part of the image

Clearly, my work was cut out for me...



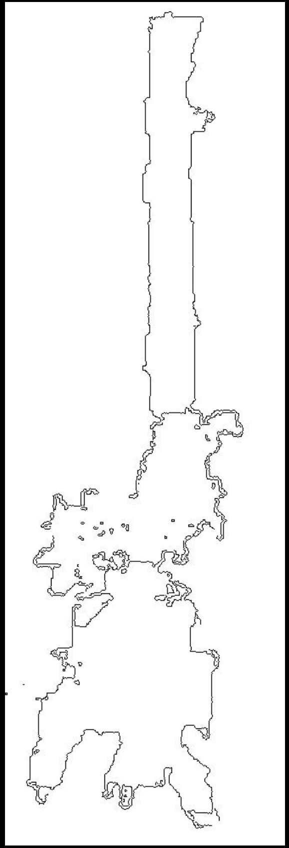


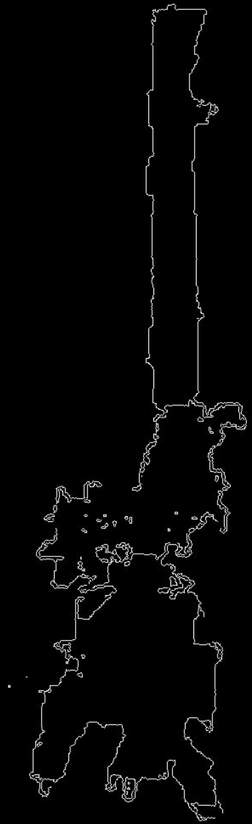
- Onboard map
  - encodes color for user convenience, but ignored in navigation
  - White is just one more color
  - Single layer

- ROS abstracts things like “rooms” to waypoints or other layers.
  - White is “explored and open” space
  - Grey is “unexplored”

Hey, I know, I’ll use OpenCV!

...yeah, no

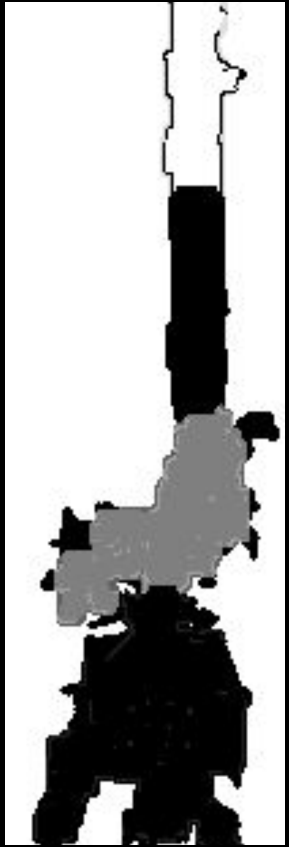




Edge detection with cv2.Canny()

-Not completely enclosing the map
-Creates false edges
-Canny algorithm notorious w/ respect to small features
  -Tuning this can be quite fussy -cv2.add() results in obscuring/losing features
-OpenCV functions apply to the whole image/region

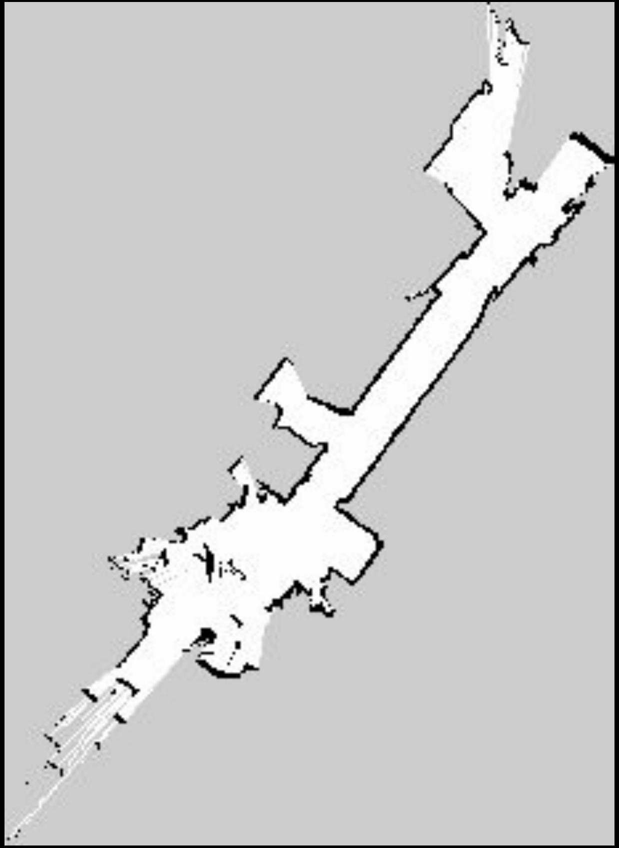
I (partially) blame LIDAR...



-LIDAR inherently produces tiny edges and artifacts, just having a "better one" doesn't help

<-Hackerbot LIDAR

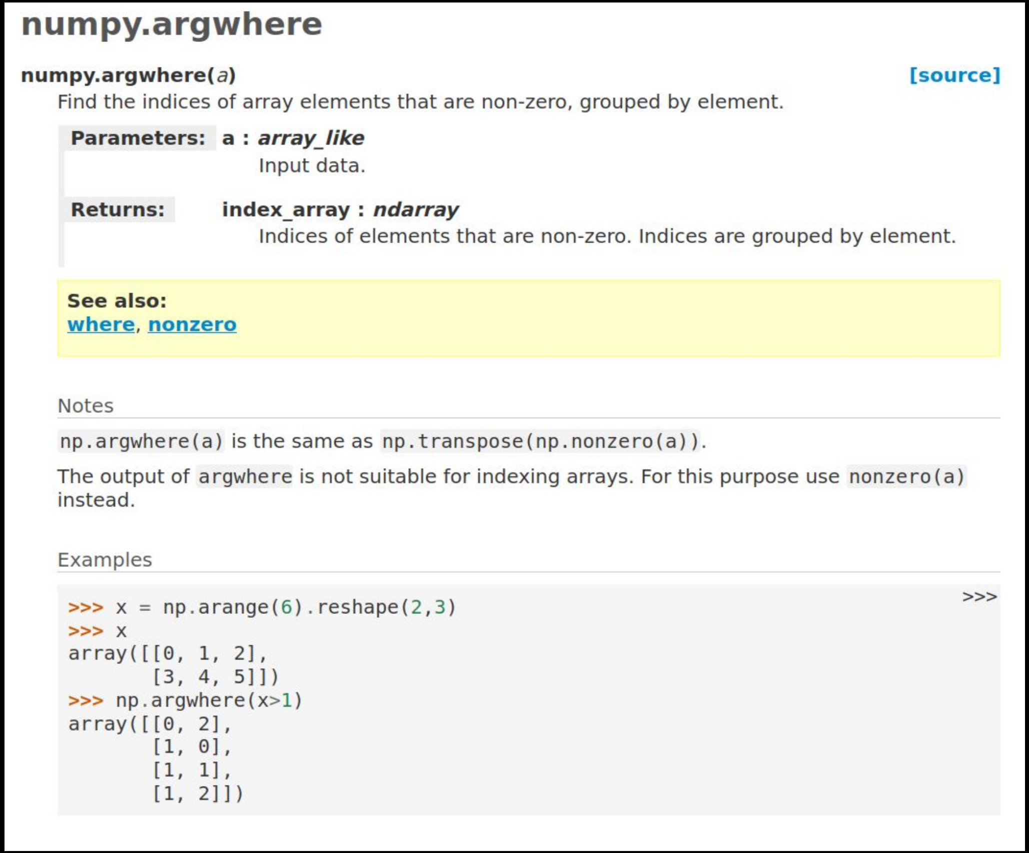
Nova Carter w/ Velodyne ->



If OpenCV says no, go ask numPy!

- Images are just array.shape(2,3,*N*) to numPy
- So, everything you could do to some training data, you can do to an image.
- Has a rich library with which to slice, dice & chop arrays, into vectors, cells

this looks promising...



numpy.argwhere

numpy.argwhere(*a*) [source]

Find the indices of array elements that are non-zero, grouped by element.

Parameters: **a :** ***array_like***

Input data.

Returns: **index_array :** ***ndarray***

Indices of elements that are non-zero. Indices are grouped by element.

See also:
where, nonzero

Notes

`np.argwhere(a)` is the same as `np.transpose(np.nonzero(a))`.

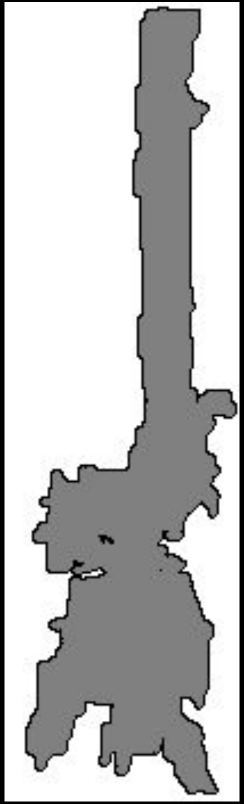
The output of `argwhere` is not suitable for indexing arrays. For this purpose use `nonzero(a)` instead.

Examples

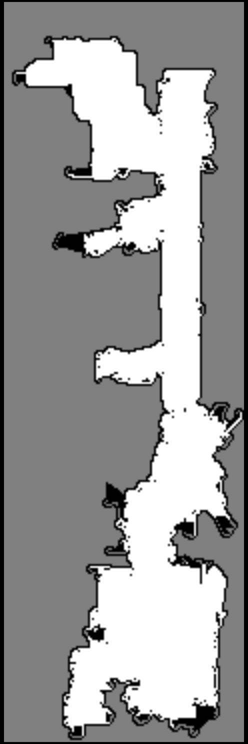
```
>>> x = np.arange(6).reshape(2,3)
>>> x
array([[0, 1, 2],
       [3, 4, 5]])
>>> np.argwhere(x>1)
array([[0, 2],
       [1, 0],
       [1, 1],
       [1, 2]])
```

<Code walkthrough and demo>

Success!



Cleaning up also helps...



Code

Hackerbot Github:
https://github.com/hackerbotindustries

Mapping Tool:
https://github.com/jetdillo/hackerbot-maptools

Hire me!

- Currently looking for work, contract or FT(for the right fit)
  - Github: https://github.com/jetdillo
  - Email: armadilo@special-circumstanc.es
  - Web: https://www.familiarrobotics.com

Thanks to....

- FOSDEM for inviting me here...
-Andra, Dan, Patrick, Alex & the rest of my CircuitLaunch family for having me there as their Roboticist in Residence in Oakland, CA. and giving me the space to build robots

-The Homebrew Robotics Club of Silicon Valley for inspiration to build robots.
-The [`alnum`] chars C,V, 2, R,O & S.

...and also to folks like you...