Donkey Car

Make an RC car drive itself with an end-to-end neural network.

Open Source (MIT)
Python
Raspberry Pi
Keras / Tensorflow
OpenCV

Agenda

One year and one day of Donkey Cars.

How the hardware and software works?

Tips to work with neural network autopilots.

What's next for DIY self driving?

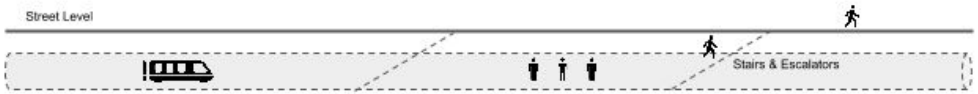
Will Roscoe
Software

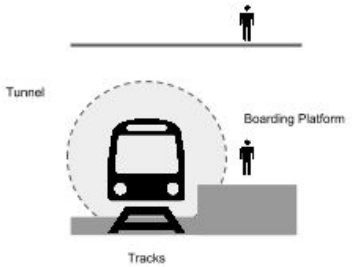
Adam Conway
Hardware

Roscoe makes pitch for competition to demonstrate AutoBART.

October, 2016

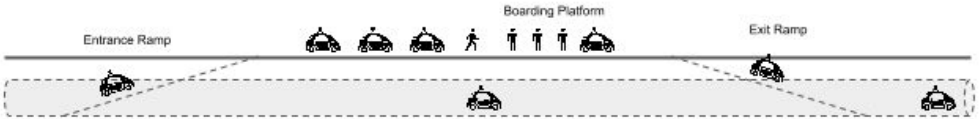
BART Station

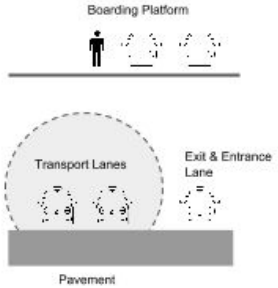




Average Speed: ***33mph (BART)***
Max Throughput: ***30,000 people per hour.***

AutoBART Station

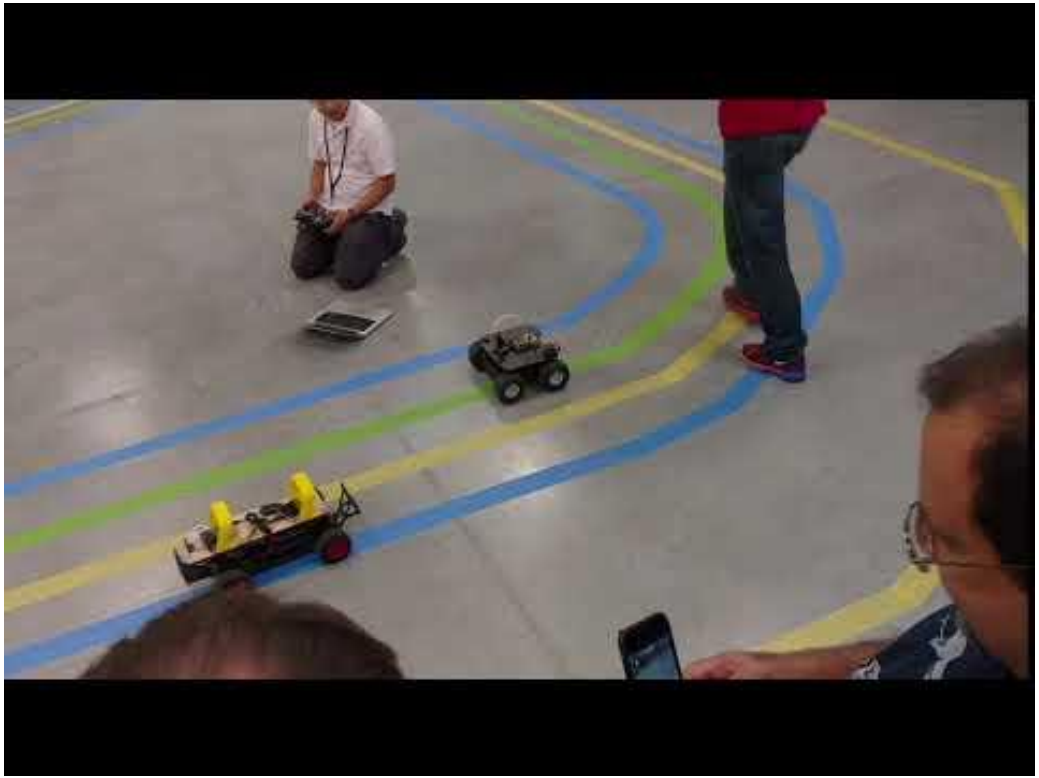




Estimated Average Speed: ***60mph***
Estimated Max Throughput: ***60,000 people per hour***

Chris Anderson hosts first DIYRobocar meetup in Carl Bass' workshop.

November, 2016

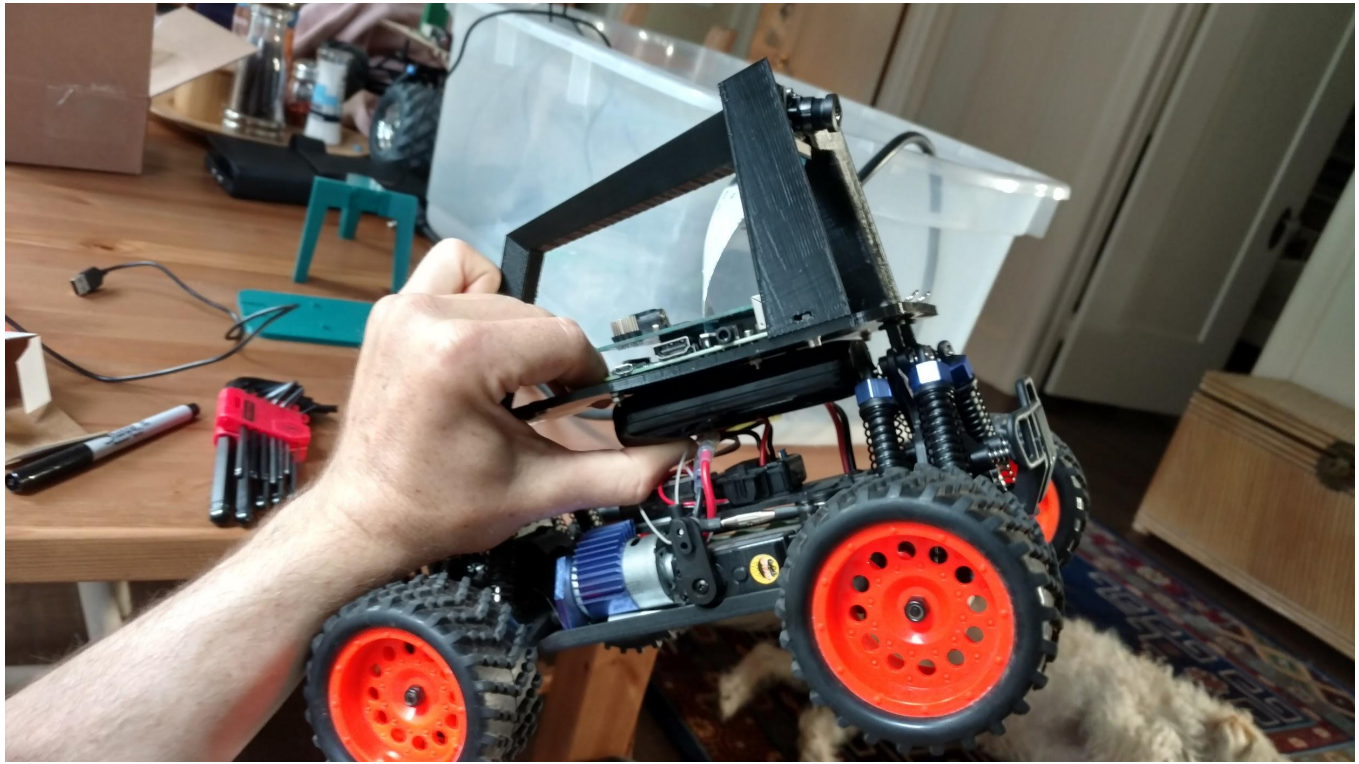


First time DIYRobocars raced in Oakland Pipe Factory.

January, 2017

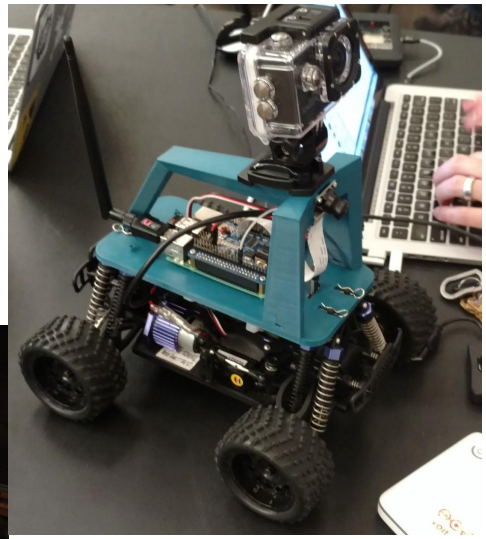
Improved hardware with Donkey2

March, 2017



Many more people build donkey.

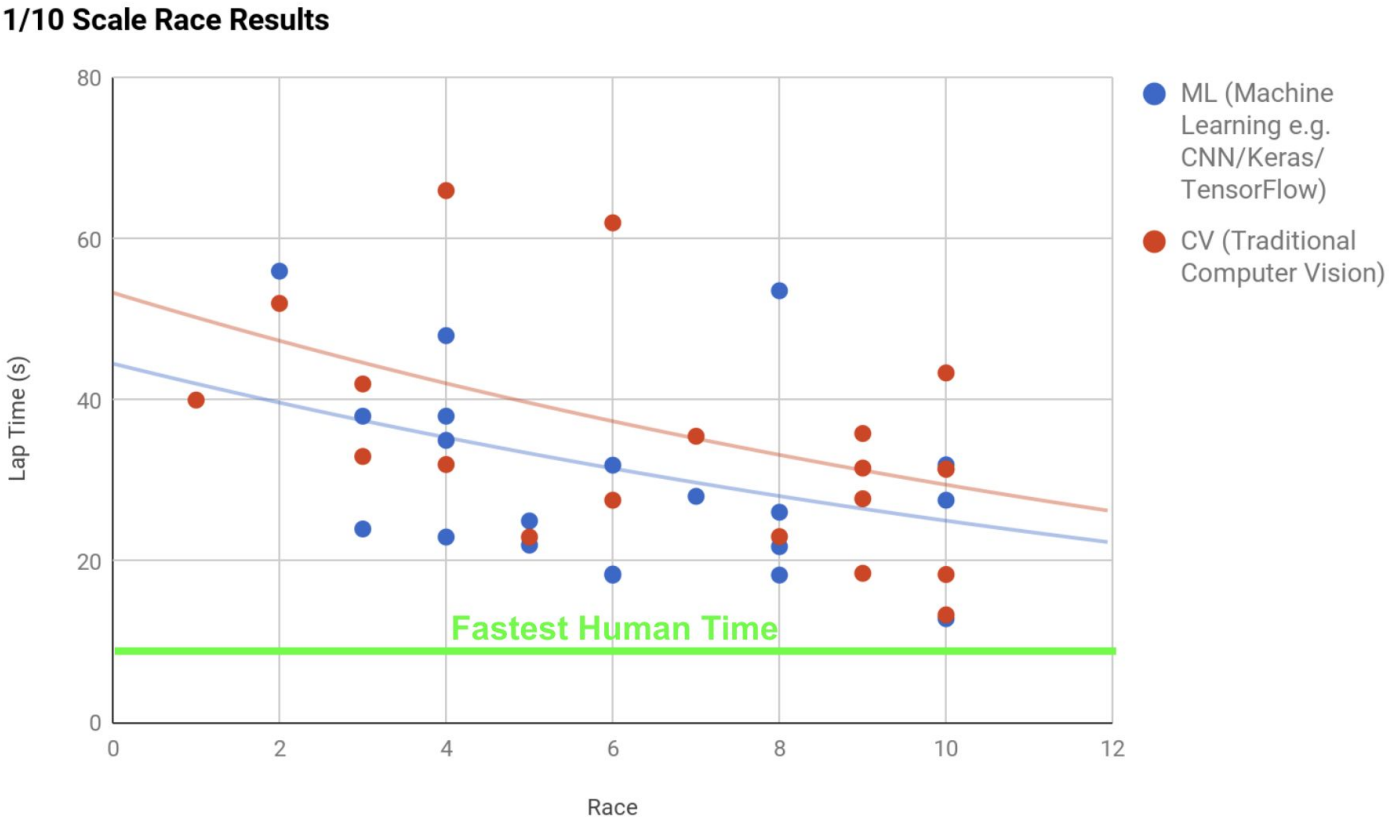
July - August, 2017



Faster and with obstacle avoidance.

October 2017

We are approaching human level racing.



How does it work?

donkeycar.com

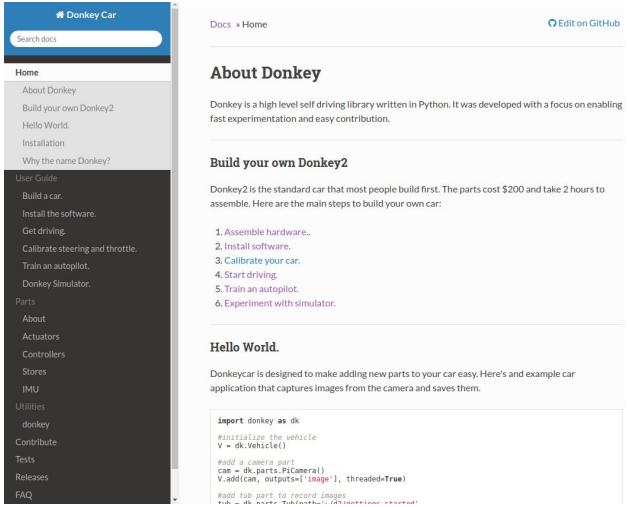
How to Build a Donkey V2

These same instructions can be found in this Make Magazine article. The software has been updated since the article was published so use the instructions found at docs.donkeycar.com after you've built your car.

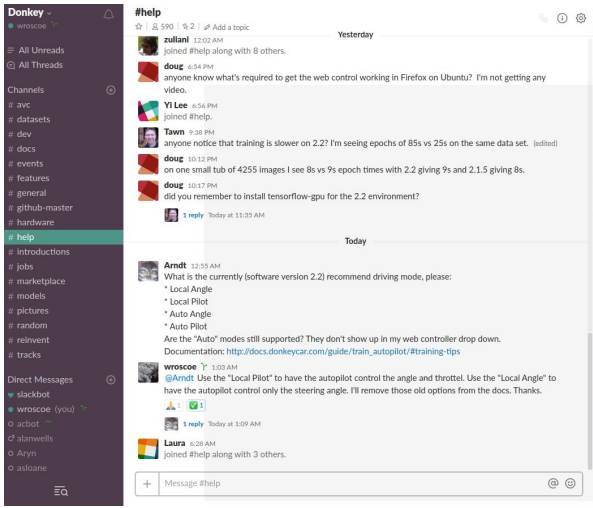
Overview

These instructions will show you how to build this car from start to finish.

Hardware instructions.



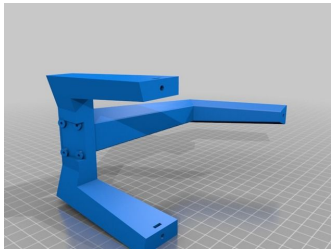
Software Docs

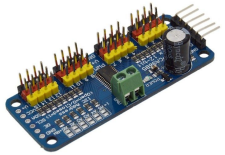


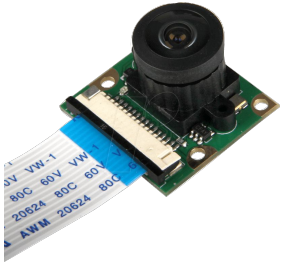
Help Brawl

1. Get Parts ($200)

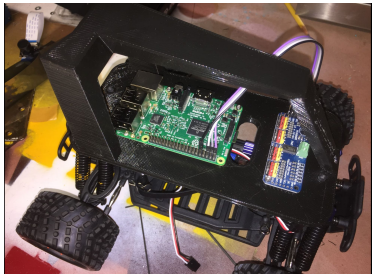
Part Description	Link	Approximate Cost
Magnet Car	https://www.amazon.com/gp/product/9269803775	$92
M2x6 screws (4)	https://www.mcmaster.com/#91292a831/=177k4rp	$6.38 *
M2.5x12 screws (8)	https://www.mcmaster.com/#91292a016/=177k574	$4.80 *
M2.5 nuts (8)	https://www.mcmaster.com/#91828a113/=177k7ex	$5.64 *
M2.5 washers (8)	https://www.mcmaster.com/#93475a196/=177k7x6	$1.58 *
USB Battery with microUSB cable (any battery capable of 2A 5V output is sufficient)	https://www.amazon.com/gp/product/B00P7N0320	$17
Raspberry Pi 3	https://www.amazon.com/gp/product/B01CD5VC92	$38
MicroSD Card (many will work, I like this one because it boots quickly)	https://www.amazon.com/gp/product/B01HUJQ6F2/	$18.99
Wide Angle Raspberry Pi Camera	https://www.amazon.com/gp/product/B00N1YJKFS/	$25
Female to Female Jumper Wire	https://www.amazon.com/gp/product/B010L30SE8	$7 *
Servo Driver PCA 9685	https://www.amazon.com/gp/product/B014KTSMLA	$12 **
3D Printed roll cage and top plate.	CAD Files: http://a360.co/2pf3Dam STL Files: http://www.thingiverse.com/thing:2260575	-

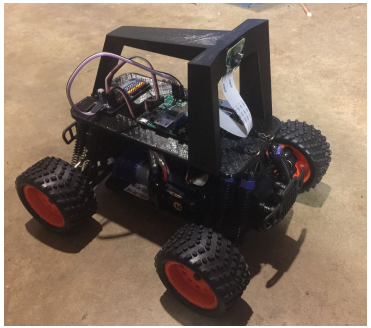


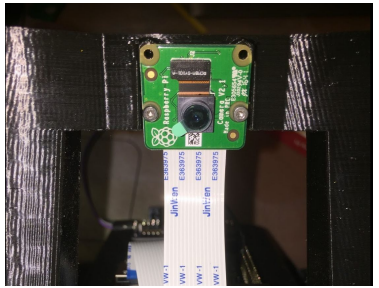




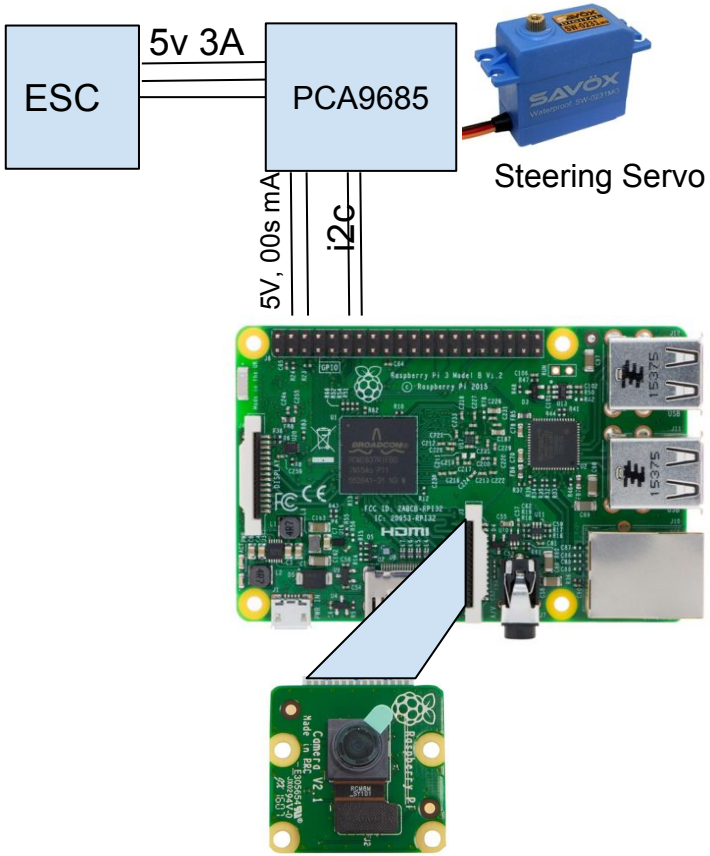
2. Assemble Parts (1 hour)

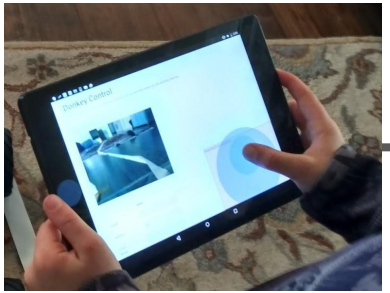






- Simplest possible hardware that we could build for a camera driven car.
- Enables Side-quests
  - Lidar
  - Odometry
  - Joystick controller
  - IMU



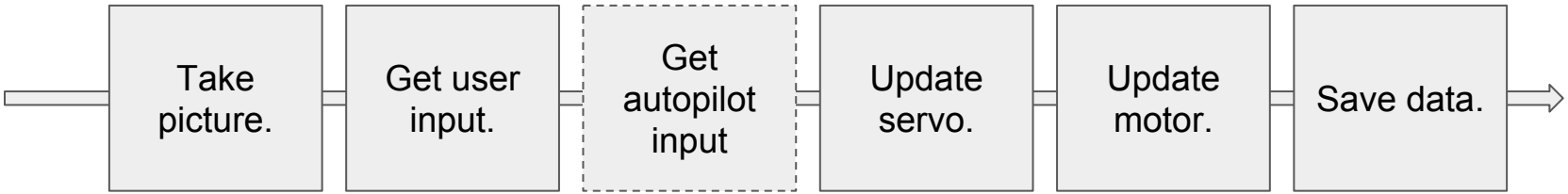


Steering (-1 to 1)
Throttle (-1 to 1)
Drive Mode (manual / auto)

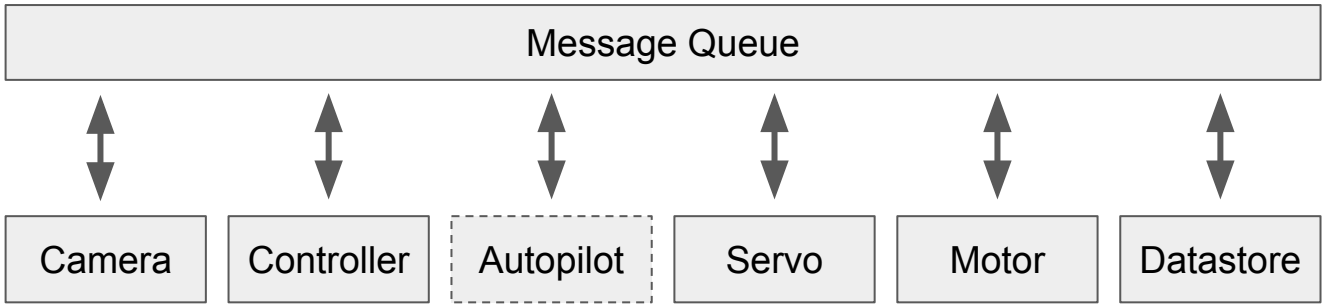
Drive
Perfectly.

Image (120 x 160)

Run the “vehicle loop” 30 times per second.

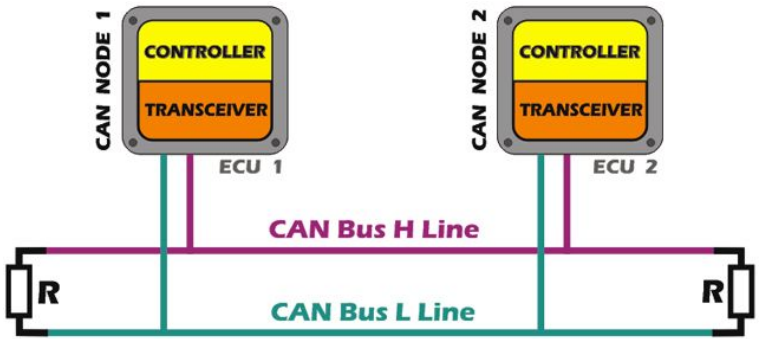


Publisher / Subscriber

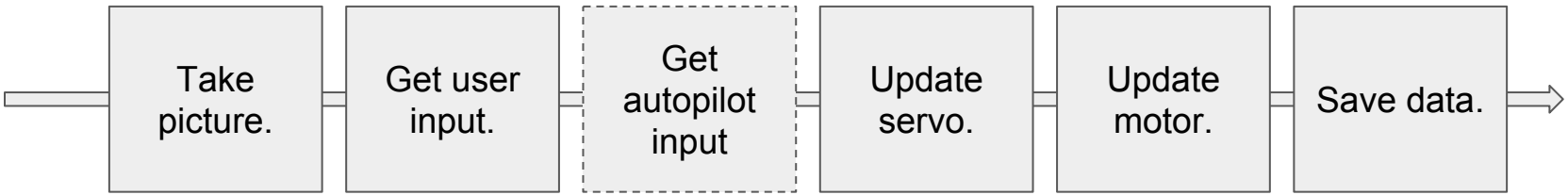


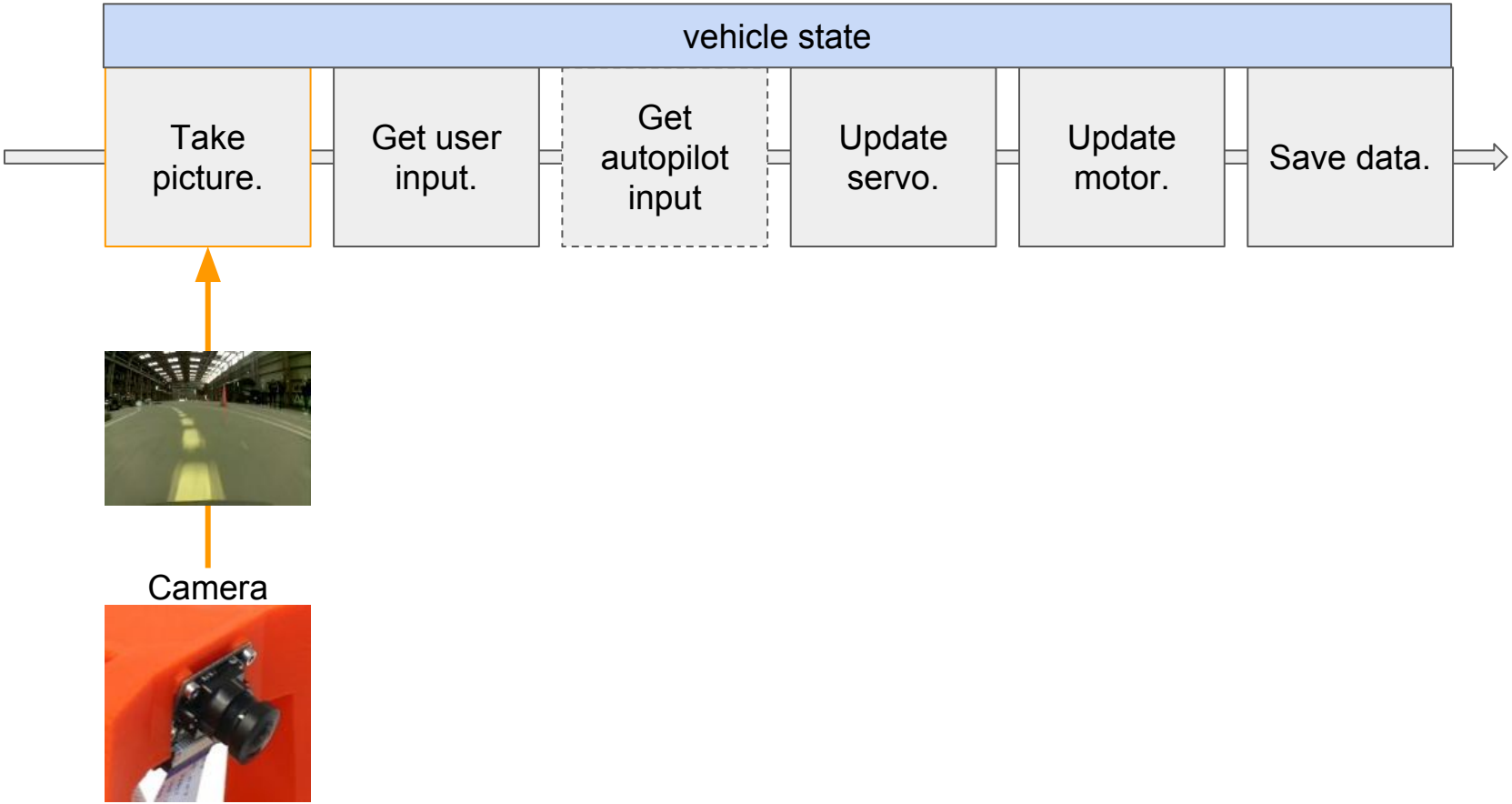
CAN Bus

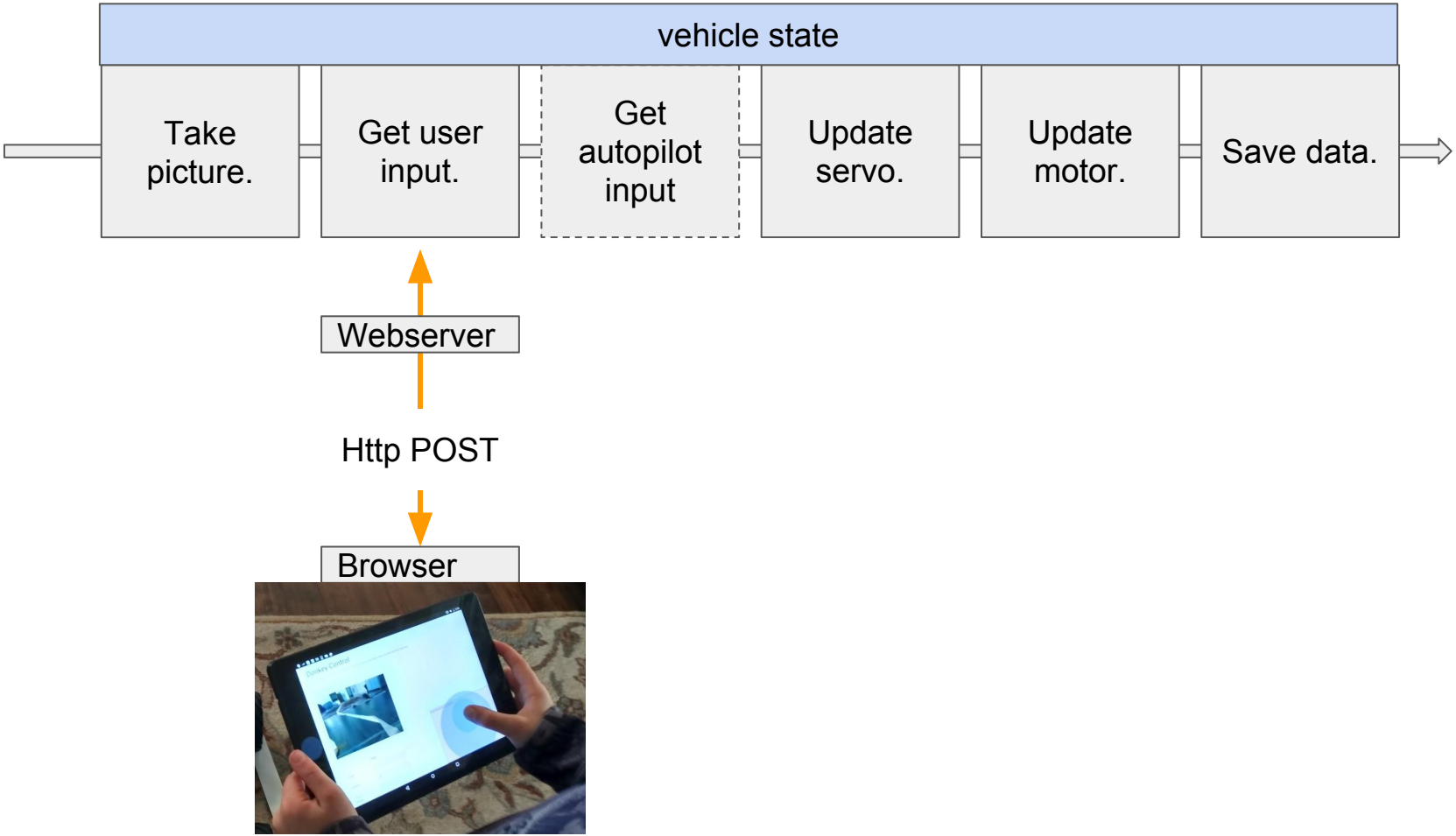
Cars

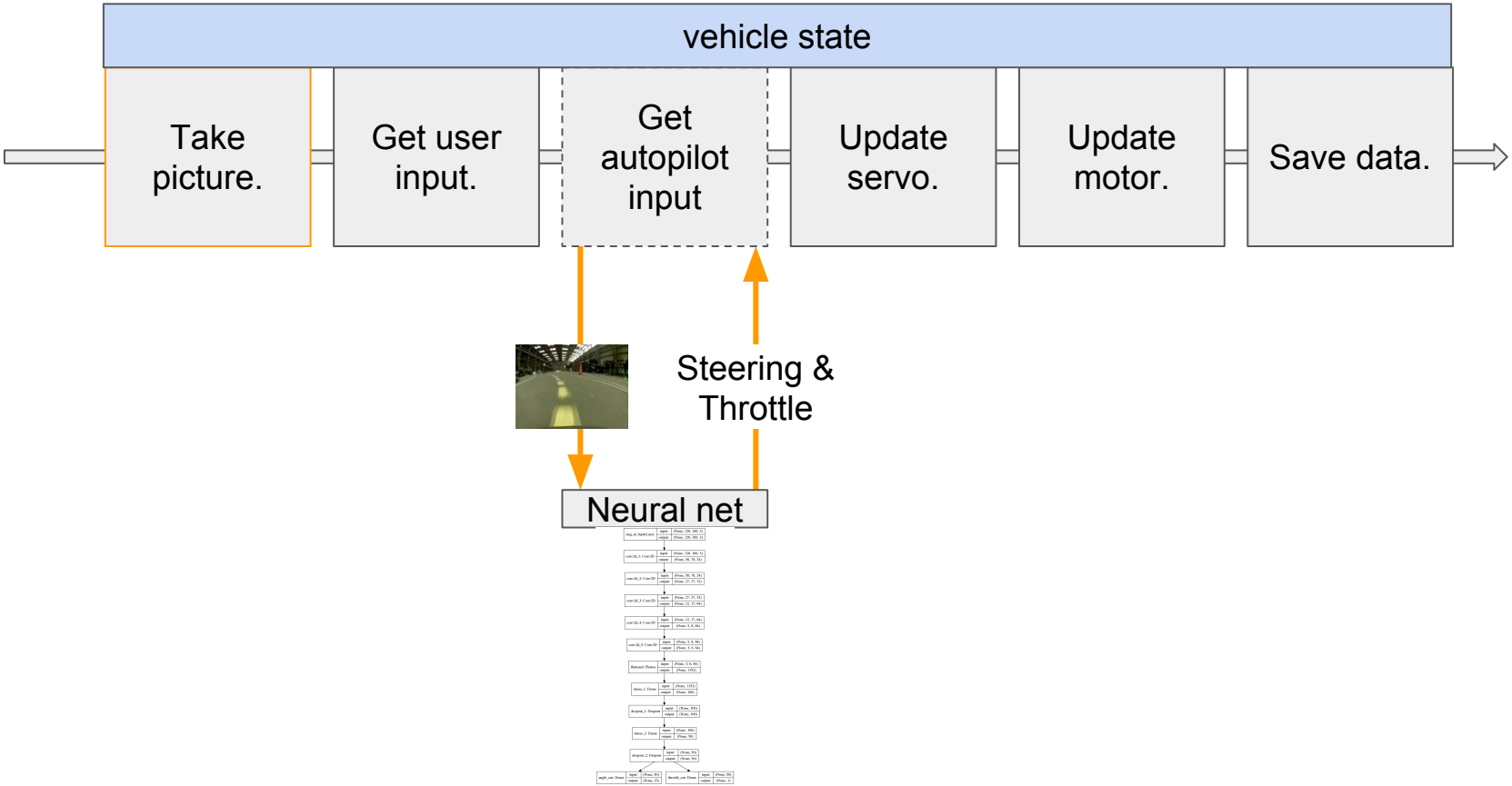


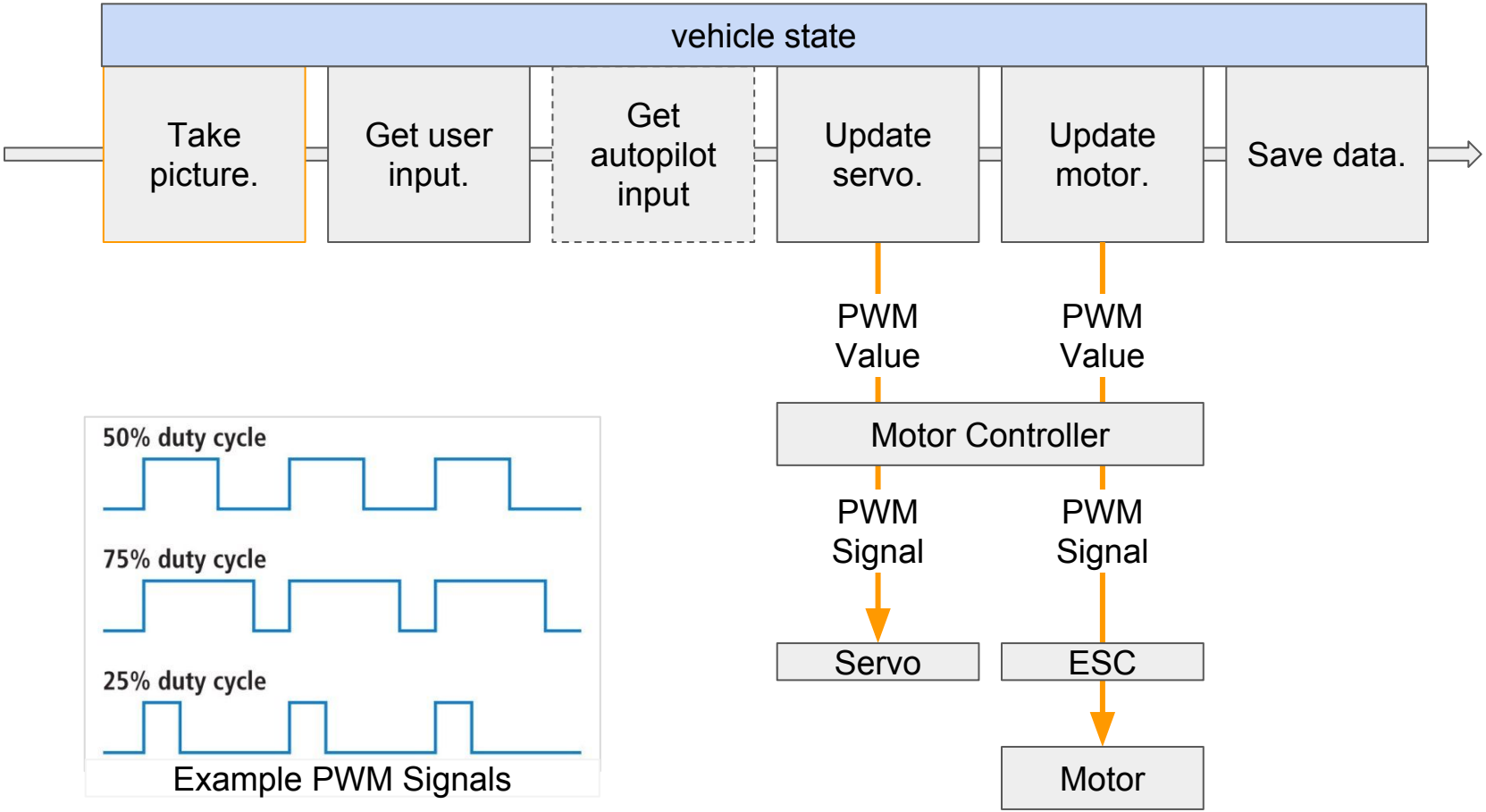
But.... KISS

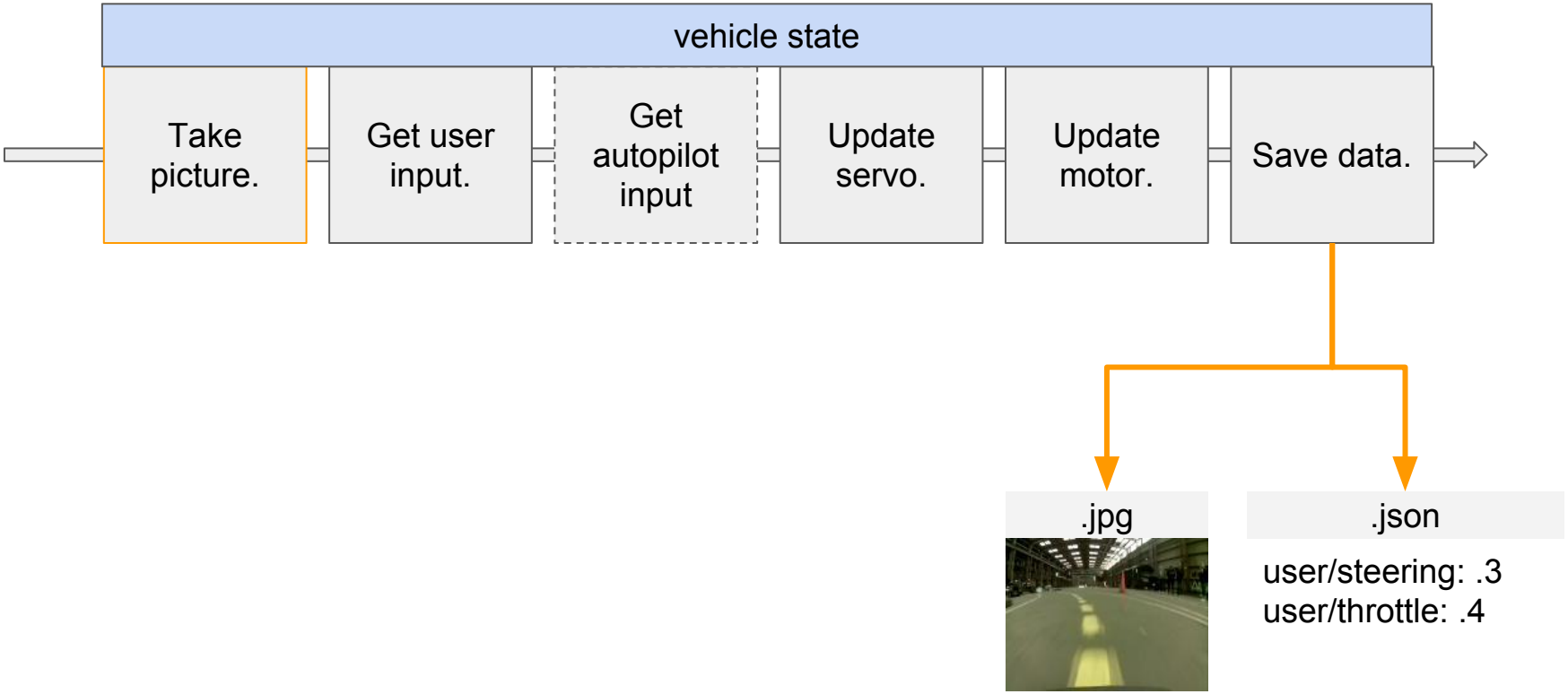

Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.


vehicle state
Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.
Camera


vehicle state
Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.
Webserver
Http POST
Browser

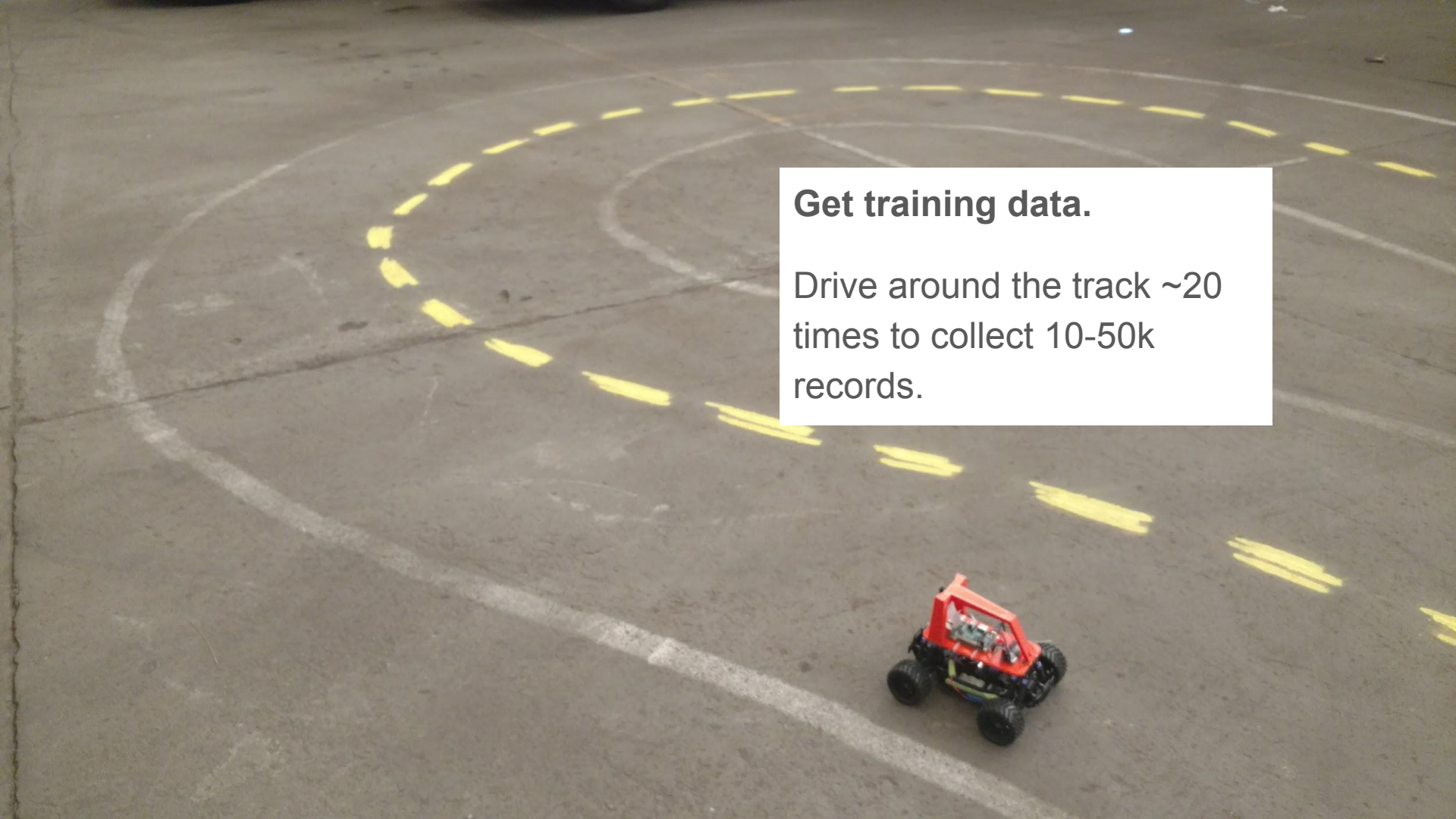

vehicle state
Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.
Steering & Throttle
Neural net


vehicle state
Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.
PWM Value
PWM Value
Motor Controller
PWM Signal
PWM Signal
Servo
ESC
Motor
50% duty cycle
75% duty cycle
25% duty cycle
Example PWM Signals


vehicle state
Take picture.
Get user input.
Get autopilot input
Update servo.
Update motor.
Save data.
.jpg
.json
user/steering: .3
user/throttle: .4

Get your donkey driving.

1. Clone donkey disk image.
2. donkey createcar --path ~/d2
3. python d2/manage.py drive
4. Go to d2.localhost:8887 in your browser.

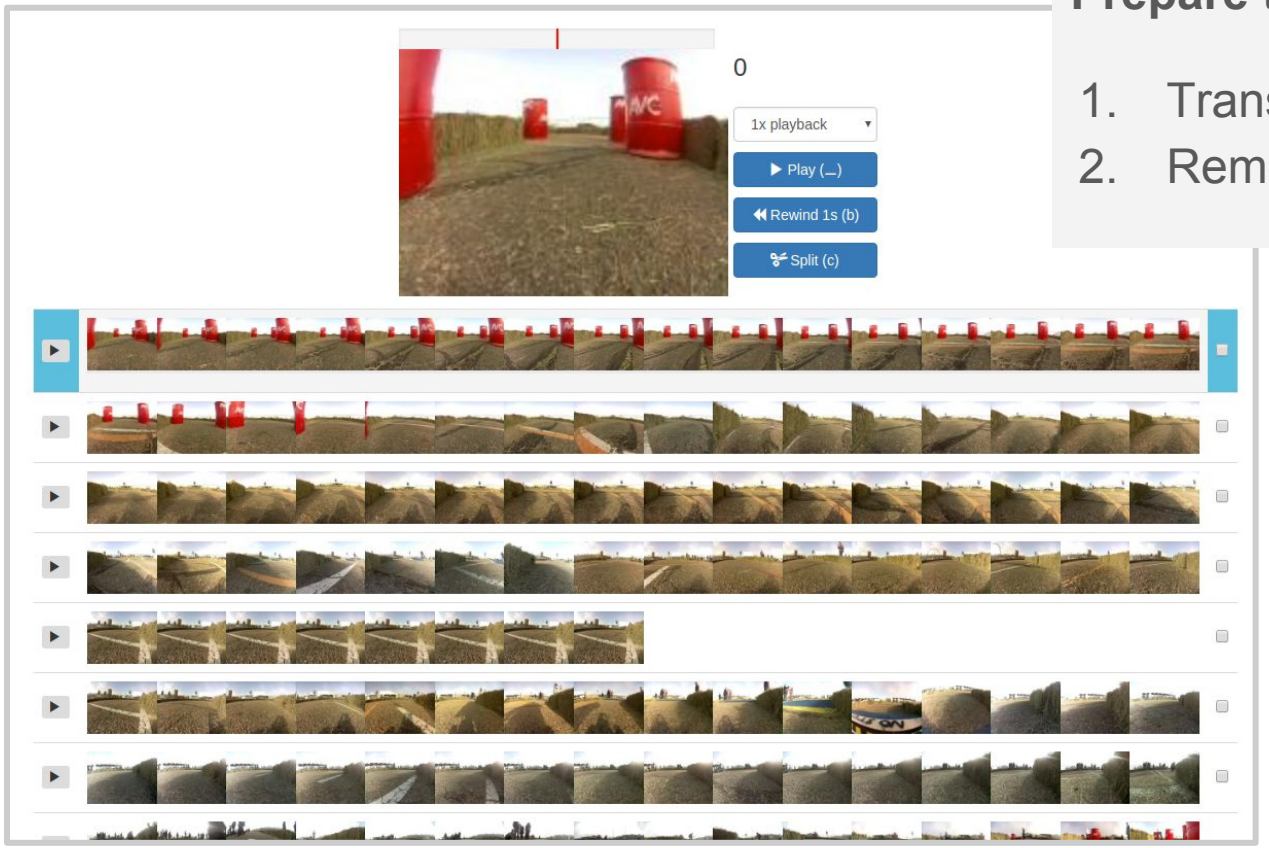


Get training data.

Drive around the track ~20 times to collect 10-50k records.

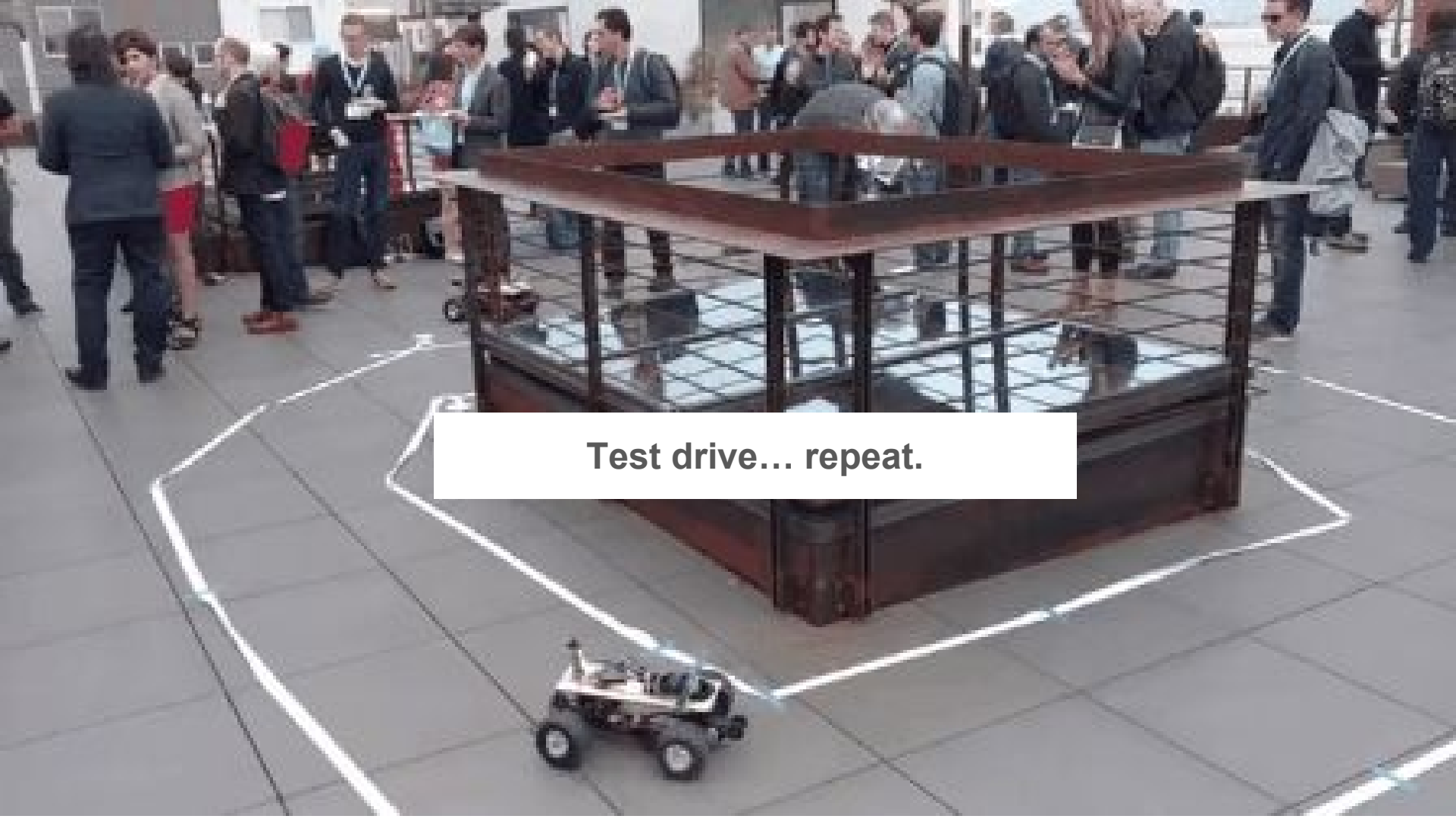
Prepare training data.

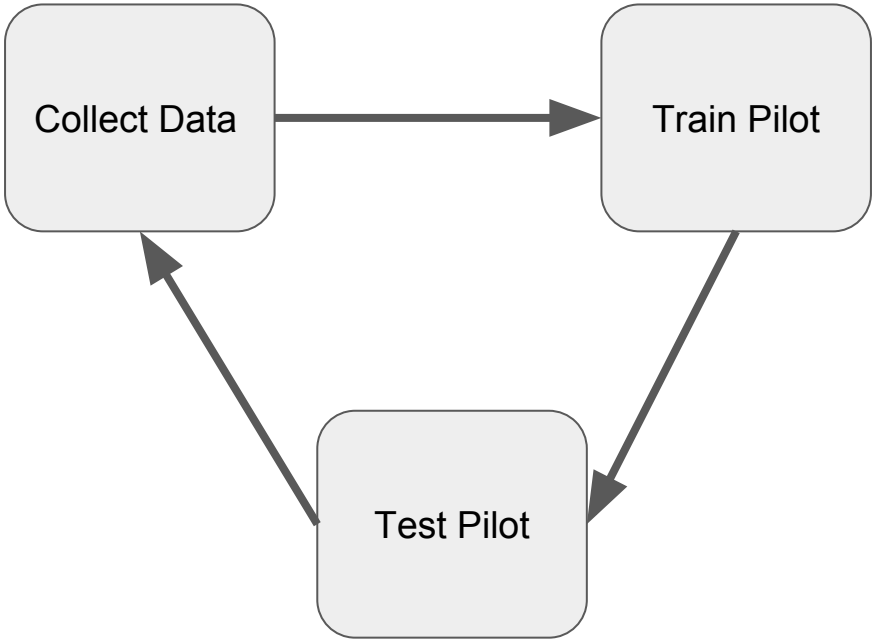
1. Transfer records to computer.
2. Remove bad data.



```
26/398 [>.............................] - ETA: 579s - loss: 3.3032 - angle_out_loss: 3.6691 - throttle_out_loss:
27/398 [=>............................] - ETA: 584s - loss: 3.2644 - angle_out_loss: 3.6259 - throttle_out_loss:
28/398 [=>............................] - ETA: 592s - loss: 3.2208 - angle_out_loss: 3.5774 - throttle_out_loss:
29/398 [=>............................] - ETA: 588s - loss: 3.1875 - angle_out_loss: 3.5405 - throttle_out_loss:
30/398 [=>............................] - ETA: 585s - loss: 3.1518 - angle_out_loss: 3.5009 - throttle_out_loss:
31/398 [=>............................] - ETA: 582s - loss: 3.1174 - angle_out_loss: 3.4627 - throttle_out_loss:
32/398 [=>............................] - ETA: 583s - loss: 3.0913 - angle_out_loss: 3.4337 - throttle_out_loss:
33/398 [=>............................] - ETA: 584s - loss: 3.0608 - angle_out_loss: 3.3998 - throttle_out_loss:
34/398 [=>............................] - ETA: 583s - loss: 3.0319 - angle_out_loss: 3.3677 - throttle_out_loss:
35/398 [=>............................] - ETA: 582s - loss: 3.0075 - angle_out_loss: 3.3406 - throttle_out_loss:
36/398 [=>............................] - ETA: 582s - loss: 2.9815 - angle_out_loss: 3.3118 - throttle_out_loss:
37/398 [=>............................] - ETA: 583s - loss: 2.9570 - angle_out_loss: 3.2845 - throttle_out_loss:
38/398 [=>............................] - ETA: 583s - loss: 2.9359 - angle_out_loss: 3.2611 - throttle_out_loss:
39/398 [=>............................] - ETA: 582s - loss: 2.9147 - angle_out_loss: 3.2376 - throttle_out_loss:
40/398 [==>...........................] - ETA: 581s - loss: 2.8930 - angle_out_loss: 3.2134 - throttle_out_loss:
41/398 [==>...........                                                                    throttle_out_loss:
42/398 [==>...........                                                                    throttle_out_loss:
43/398 [==>...........                                                                    throttle_out_loss:
44/398 [==>...........                                                                    throttle_out_loss:
45/398 [==>...........................] - ETA: 575s - loss: 2.8039 - angle_out_loss: 3.1146 - throttle_out_loss:
46/398 [==>...........................] - ETA: 574s - loss: 2.7900 - angle_out_loss: 3.0990 - throttle_out_loss:
47/398 [==>...........................] - ETA: 574s - loss: 2.7762 - angle_out_loss: 3.0837 - throttle_out_loss:
48/398 [==>...........................] - ETA: 572s - loss: 2.7618 - angle_out_loss: 3.0678 - throttle_out_loss:
49/398 [==>...........................] - ETA: 571s - loss: 2.7468 - angle_out_loss: 3.0512 - throttle_out_loss:
50/398 [==>...........................] - ETA: 570s - loss: 2.7333 - angle_out_loss: 3.0361 - throttle_out_loss:
51/398 [==>...........................] - ETA: 568s - loss: 2.7180 - angle_out_loss: 3.0191 - throttle_out_loss:
52/398 [==>...........................] - ETA: 567s - loss: 2.7046 - angle_out_loss: 3.0043 - throttle_out_loss:
53/398 [==>...........................] - ETA: 565s - loss: 2.6937 - angle_out_loss: 2.9921 - throttle_out_loss:
54/398 [===>..........................] - ETA: 564s - loss: 2.6812 - angle_out_loss: 2.9783 - throttle_out_loss:
55/398 [===>..........................] - ETA: 563s - loss: 2.6666 - angle_out_loss: 2.9620 - throttle_out_loss:
56/398 [===>..........................] - ETA: 561s - loss: 2.6543 - angle_out_loss: 2.9484 - throttle_out_loss:
57/398 [===>..........................] - ETA: 559s - loss: 2.6441 - angle_out_loss: 2.9371 - throttle_out_loss:
58/398 [===>..........................] - ETA: 559s - loss: 2.6325 - angle_out_loss: 2.9241 - throttle_out_loss:
```

Train an autopilot using Keras / Tensorflow.


Test drive… repeat.


Collect Data
Train Pilot
Test Pilot

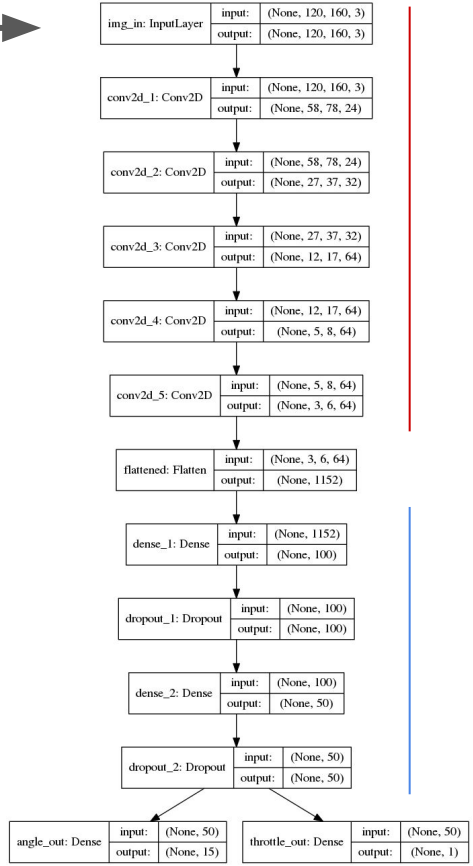
Keras / Tensorflow Autopilots

```
img_in = Input(shape=(120, 160, 3), name='img_in')
x = img_in
x = Convolution2D(24, (5,5), strides=(2,2), activation='relu')(x)
x = Convolution2D(32, (5,5), strides=(2,2), activation='relu')(x)
x = Convolution2D(64, (5,5), strides=(2,2), activation='relu')(x)
x = Convolution2D(64, (3,3), strides=(2,2), activation='relu')(x)
x = Convolution2D(64, (3,3), strides=(1,1), activation='relu')(x)

x = Flatten(name='flattened')(x)

x = Dense(100, activation='relu')(x)
x = Dropout(.1)(x)
x = Dense(50, activation='relu')(x)
x = Dropout(.1)(x)

#categorical output of the angle
angle_out = Dense(15, activation='softmax', name='angle_out')(x)

#continous output of throttle
throttle_out = Dense(1, activation='relu', name='throttle_out')(x)
```

Image Array

120 pixels high
160 pixels wide
3 RGB channels

Neural network.

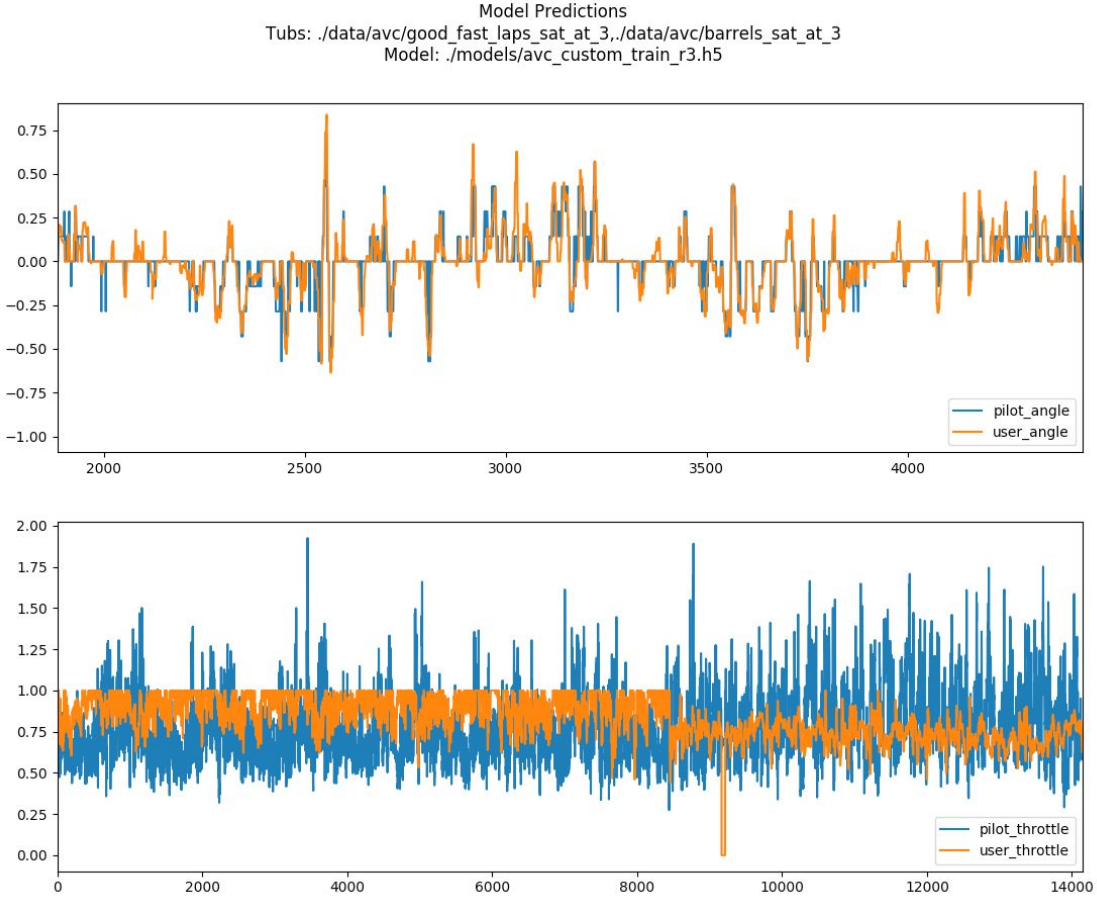


Convolution layers

Fully connected layers

Steering + Throttle

Can we test an autopilot without driving?



What’s next for Donkey.

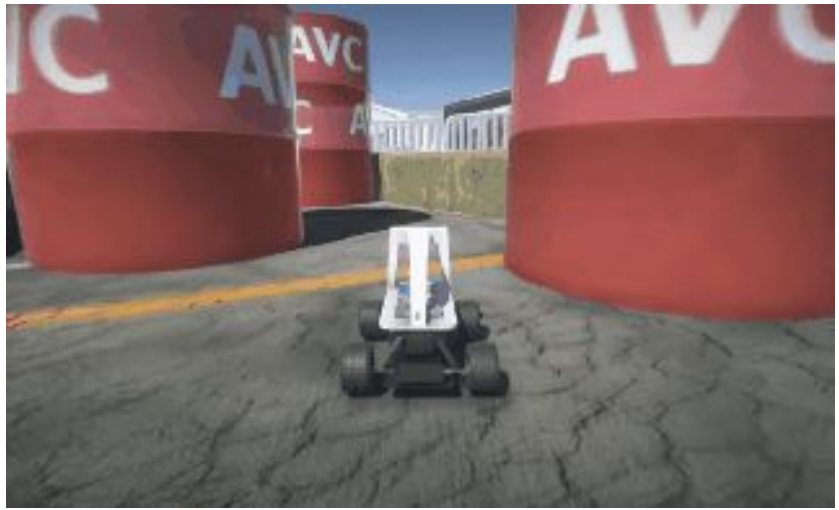
More competitions.

diyrobocars.com

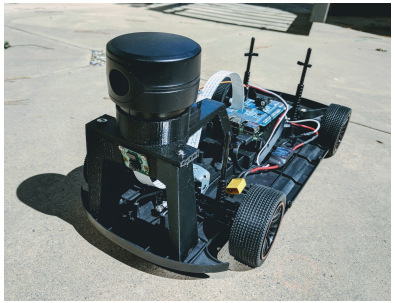
Next race is this weekend in Oakland 10am - 2pm.

2 day Hackathon in Vegas Nov 26-27

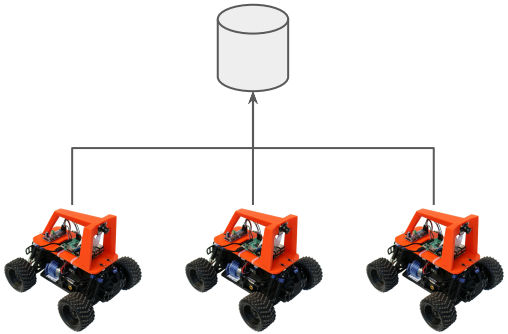
Simulators



Additional data.



More sensors.

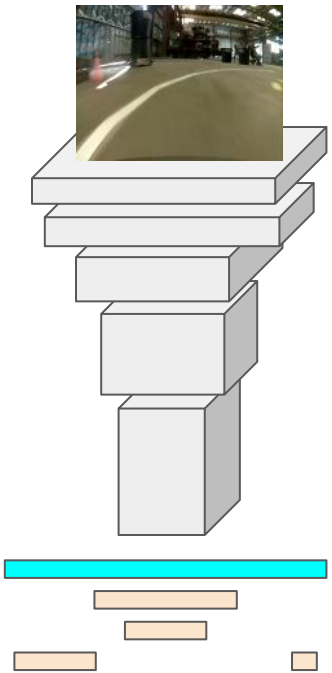


Pooling data from multiple cars..

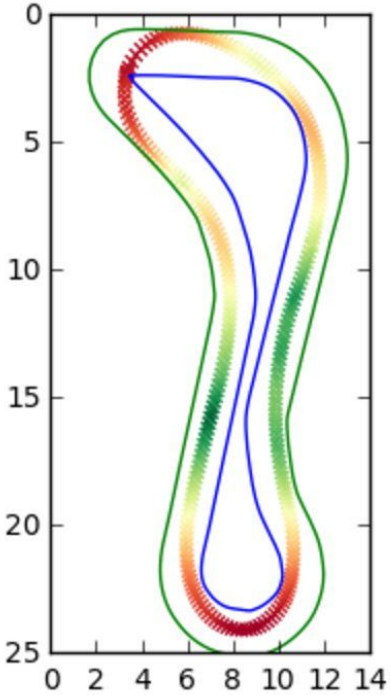
New Environments

Better Autopilots

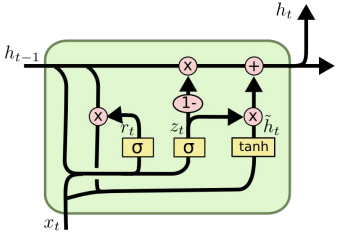
Modular Nets

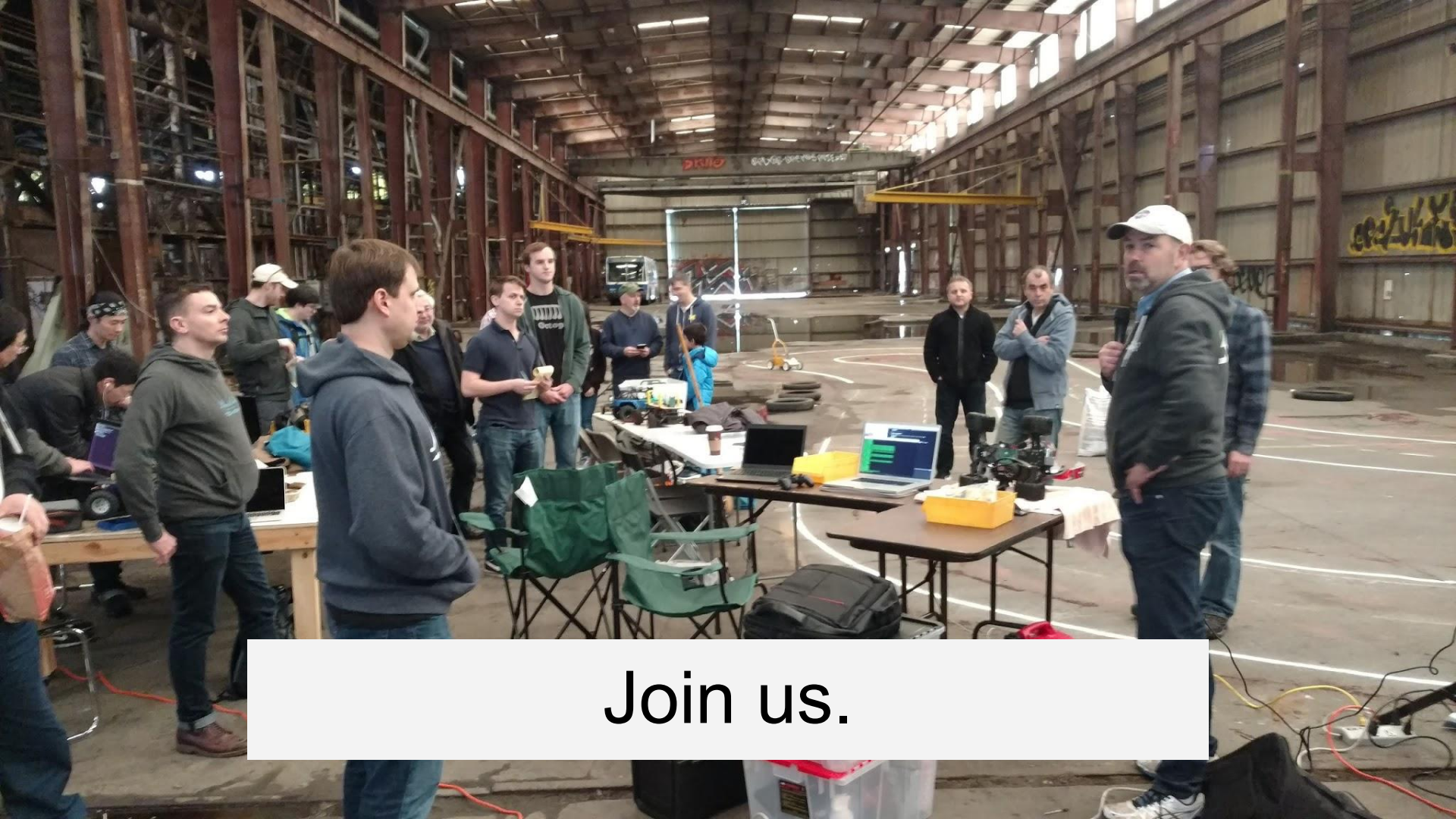


Maps



Stateful

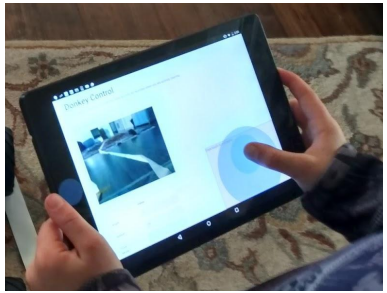



Join us.

1 - Build your own car or try to drive the simulator.

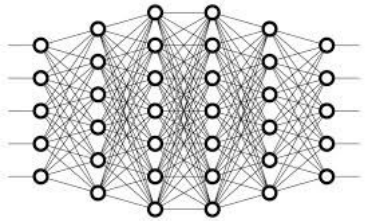
2 - Hack on any part that interests you.

Web

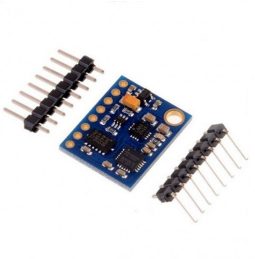


Computer Vision

Deep Learning



Hardware



Instructions to build your own car.
donkeycar.com

Race events / meetups.
diyrobocars.com

Stay in touch:

Will Roscoe: **@dataduce**

Adam Conway **@acb0t** (with a zero)