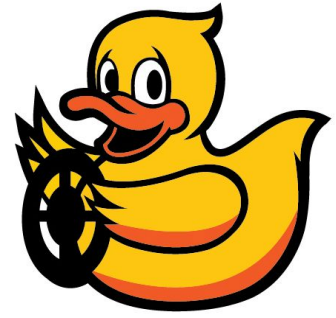


Duckietown Appearance Specification



This document is about the appearance rules for a consistent duckietown. If you build your own duckietown: if it adheres to this rule, we guarantee that our robots will work.

Changes to be made:

- **The specification for the traffic light must be updated**

Layers

Duckietown is built with two layers:

1. The first layer is the floor layer. The floor is built of interconnected exercise mats with tape on them.
2. The second layer is signs and other objects that sit on top of the mats.

Additional note: In the case that Duckietown directly abuts a wall, a perimeter of vertical tiles should be added to reduce visual clutter and false positives.

Layer 1 - Tiles and Tapes

Each tile is a 2ft x 2ft square and is able to interlock with the others.

There are fundamentally three types of tiles as follows:

- 1) StraightTile: Straight road
- 2) CurvedTile90: Curved road
- 3) IntersectionTile: Intersection

An intersection comprises one single tile and MUST be abutted on all sides by a piece of straight road. Curved roads can be linked indefinitely but there MUST be a tag visible at the transition from one tile to the next along the path (see apriltags for more info).

Tapes

There are 3 colors of tapes: white, yellow, and red.

White

We assume that a Duckiebot never collides with Duckietown if it never crosses or touches a white tape strip.

White tapes must be solid

White tapes should be the full thickness of the roll of duck tape.

Width of the tape = ???

It is also true that, during navigation in a StraightTile, the white stripe is always on the right hand side of lanes.

The edge of each tile with a white piece of tape should also have a small curb (use the edge pieces that come with the mats for this). The purpose of this is to clearly delineate the white lanes and make lanes in adjacent roads less visible from the onboard camera.

Moreover, the Duckiebot is not powerful enough to cross the curb, so that failure of one car causes only the current tile to become unavailable.

White lines may also be placed on an intersection tile (in fact this is the only type of a tape that should appear on an intersection tile).

For curved road - the white tape should form a “kitty corner” centered **at $\frac{1}{4}$ of the length of the tile (or 0.5 ft) on each border**.

REVISED SPECIFICATION: For curved road, the white lane marker is formed by five pieces of white tape, while the inner corner is formed by three pieces, placed according to the specifications in the image below, where the edge pieces are matched to adjacent straight or curved tiles:

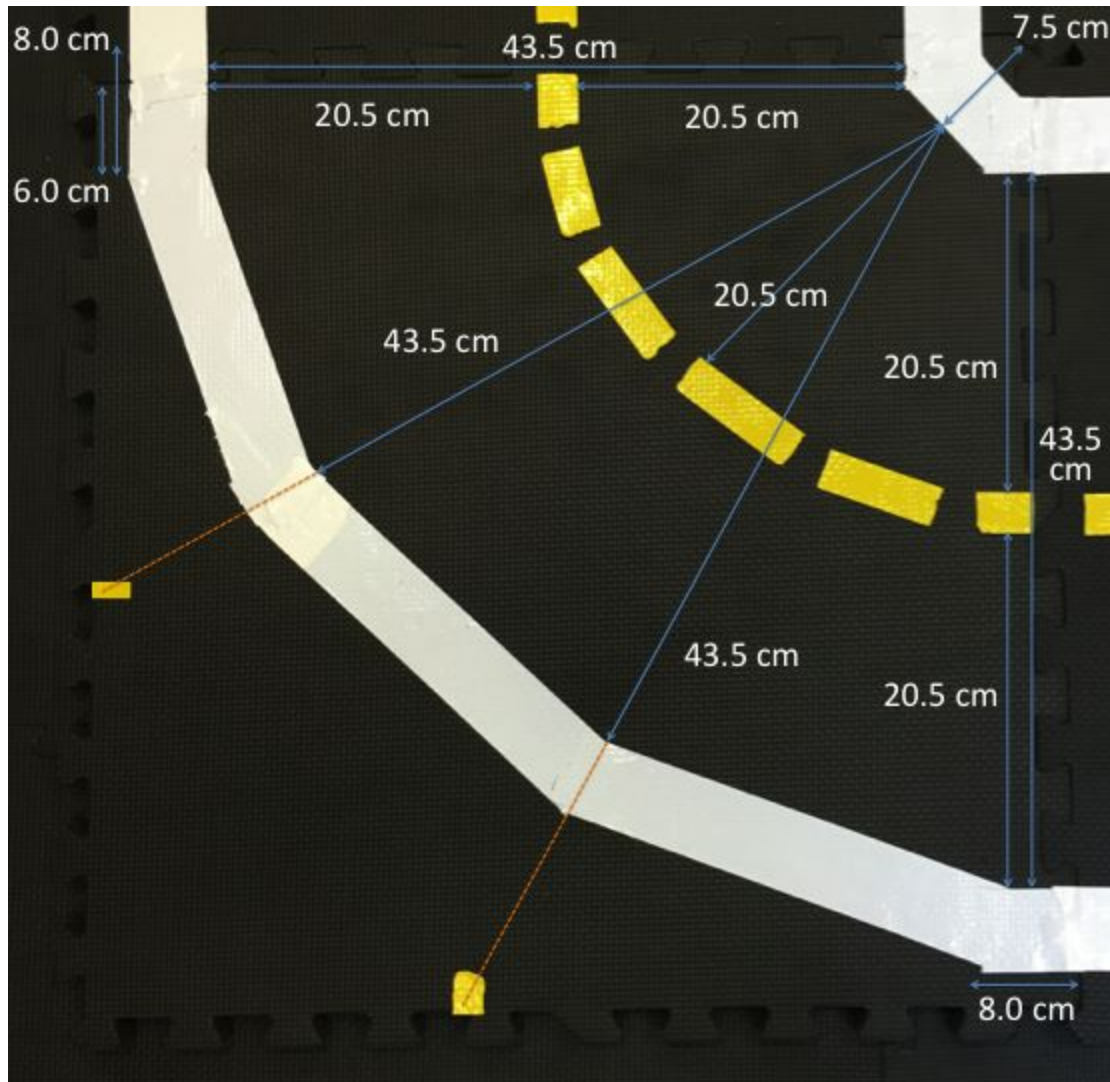


Fig 3: A curved road.

Yellow

Yellow tape should be **half of the thickness** of the roll of duck tape = **???**. On a two-way road, the yellow tape should be dashed.

Each piece should have a length of approximately **4in** with a **2in** gap separating each piece.

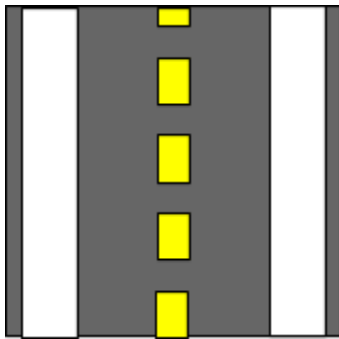


Fig 1: A standard 2-way straight road

A one-way road can have a solid yellow line on the left and solid white on the right:

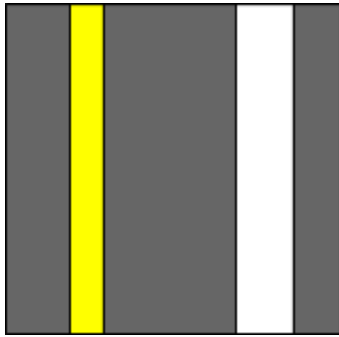
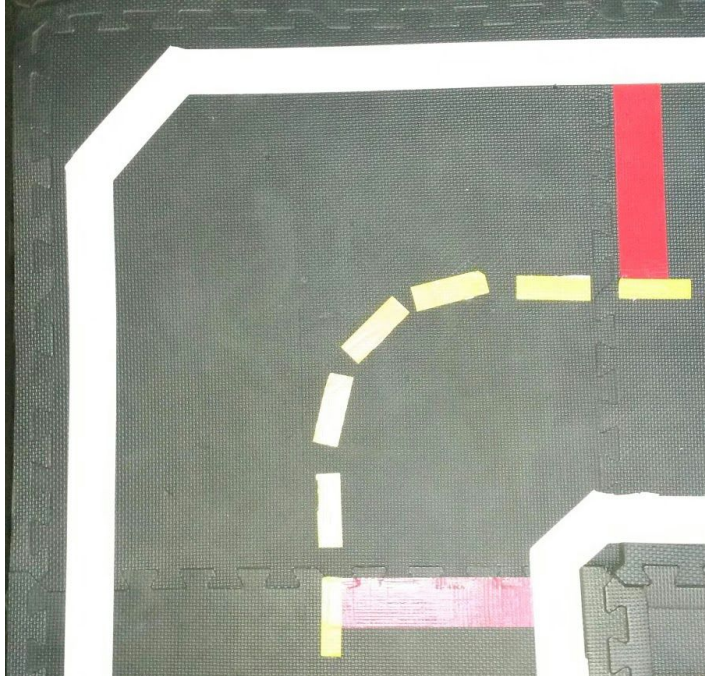


Fig. 2: 1 one-way one lane piece of road - in this case direction of travel is from bottom to top.

example



Yellow tapes on curves: see curved road image in white tape section, pieces at tile edges should be in center of lane, piece at the middle of the curve should be approximately 20.5 cm from middle of inner center white piece of tape, with approximated circular arc in between.

Red

Red tapes MAY **only** appear on **straight road** tiles. A red tape MUST be used whenever the adjacent tile is either a **curved road** or an **intersection** and MUST be placed at the edge of the tile on the edge that abuts the curved road or intersection. The red tape MUST be the full width of the duck tape roll and should cross the entire lane perpendicular to the lane.

The placement of red tape should always be **under** yellow and white tape, as shown in the image above.

The following is true: a Duckiebot can navigate Duckietown by a sequence of:

- Navigating one or more StraightTiles until a Red Strip appears.
- Wait for the coordination signal
- Execute an open-loop motion.
- Relocalize in a StraightTile

Here are a couple of examples:

Fig 4: A straight road tile adjacent to a curved road tile requires a red tape. Note the red strip is on the tile from which the car comes. (can we remove these red strips?)

The invariant is: if you stop before or ON the red strip, no collisions are possible.

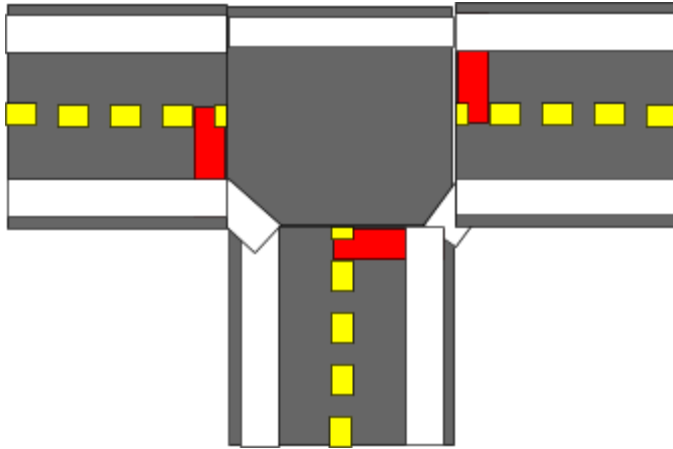


Fig 5: A 'T' intersection

Layer 2 - Signage and Lights

IMPORTANT: All signage should sit on the raised borders of the roads or be placed beside the white street markings. Under no circumstances should the white (any other tape) be obscured.

- Our signs were printed [from this file](#) with these specifications (Mixed B&W and Color, 1-sided, 80# White Cover) at MIT's CopyTech.















Traffic Signs

Specs

Center of signs are 13cm height with apriltags of 6.5cm sq. and a white border pasted below them.

Type

The allowable signs are:

				
stop (1)	yield (2)	no-right-turn (3)	no-left-turn (4)	do-not-enter (5)
				
oneway-right (6)	oneway-left (7)	4-way-intersect (8)	right-T-intersect (9)	left-T-intersect (10)
				
T-intersection (11)	pedestrian (12)	t-light-ahead(13)	duck-crossing (14)	

Each sign is printed from the [signs and tags doc](#)

Placement

Signs MAY appear on the opposite side and at the corner of the adjacent tile from which they are viewed. In the absence of any signs, it is assumed that all network flows are allowed so a sign MUST be placed and visible whenever this is not the case.

- **to help localization/SLAM:** avoid having 2 signs which are similar and too close to each other (issue with data association). For instance: two consecutive tiles cannot be marked with the same sign

For example, for the 'T' intersection the placement of the sign MUST be as indicated:

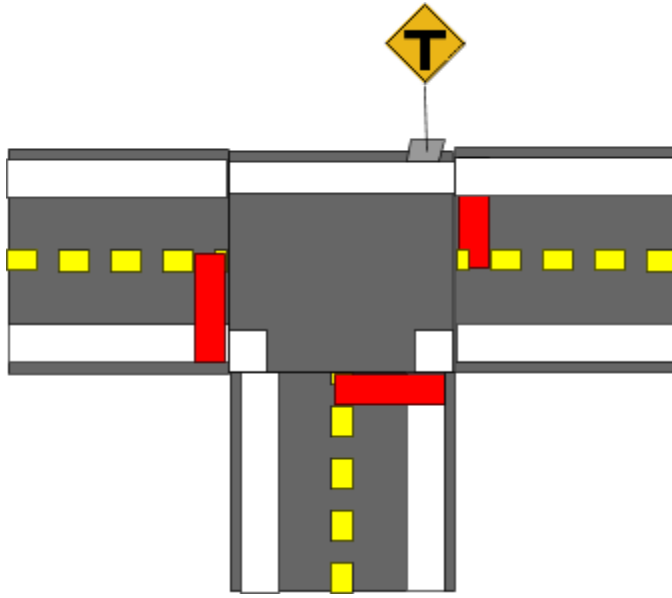
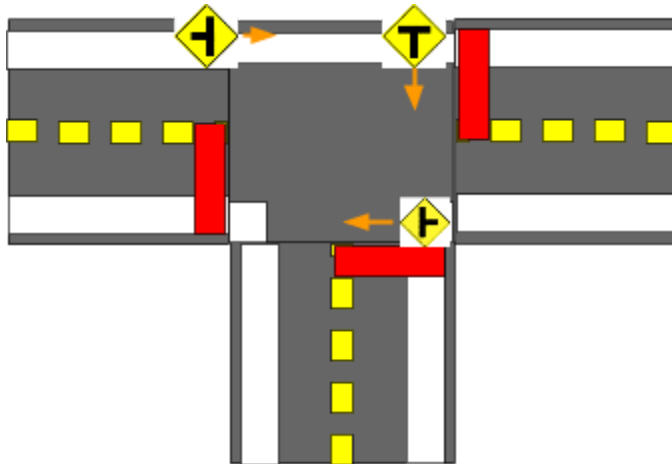
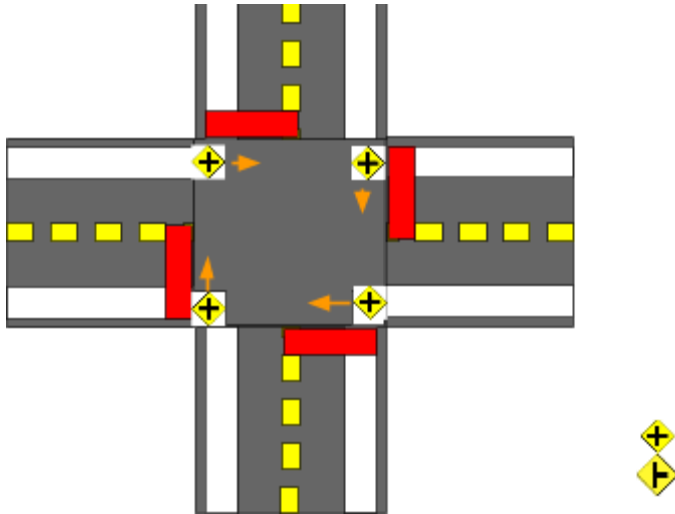


Fig. : Placement of signs to indicate intersection flows should be viewable from the stop lines

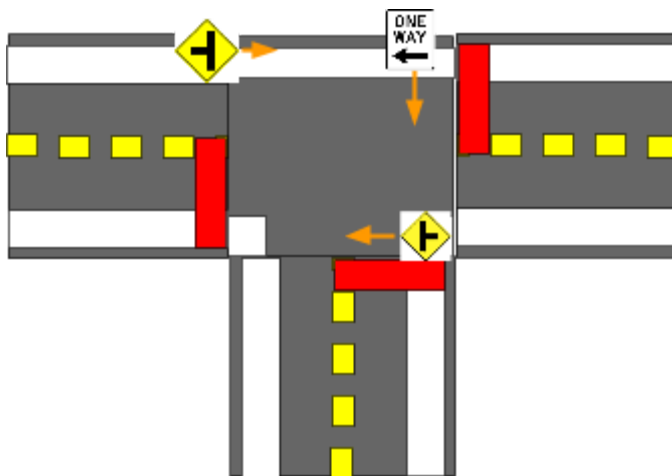
Examples of conforming intersections

In these figures the arrow is the direction of the sign.

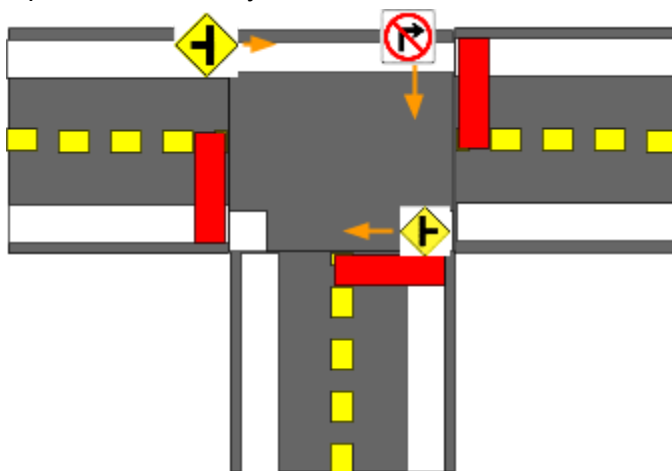




Is this:



equivalent to this? yes



Assembly



Fig : Placement of an apriltag on a sign.

- each street sign must have an april tag according to the [April Tags DB](#)

Street Name Signs

Specs



.....
april.tag.Tag36h11, id = 503

- Font: arial.
 - Color: Perhaps we could start with real-world settings: white as foreground and green as background.
 - Border: currently no additional borders
 - The rounded corners are modified into 90 degrees.
 - Height: sign board height is 4.5 in. (**2.1 in**),
 - Width: Currently 4.5 in for id 500-511. (**6.1 in + 1.1 in “ST” or 5.5 in + 1.7 in “AVE”**)
 - Alphabet = English upper case. Different writing systems may need different algorithms.
 - Text direction: Horizontal for alphabetical languages
-
- The choice of street name: The names are MIT roboticists' names (see further below) and popular street names arranged alphabetically. The Word file is [here](#)
 -

Placement

- **Similar to traffic light:** The street name should sit on a pole that is based at the corner of the tile outside of the allowable driving region. The bottom of the street name should be at a height of 7in, and allow a duckiebot to pass through. The street names should be visible from both sides of the road.
- **to help localization/SLAM:** ~~have a street name every 2 tiles on a straight lane.~~ **There should be a tag at least every 2 tiles but it should not be a street name sign because they are too big**

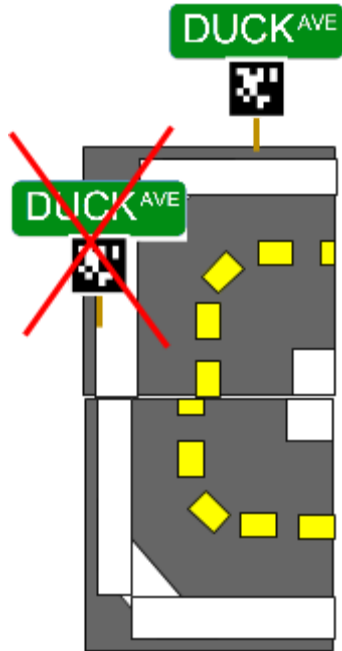


Fig: The placement of road signs on the roads. There should not be any road signs that are perpendicular to roads.

NEW UPDATE:

- Street name signs should never be perpendicular to the road - they are too big and obtrusive.
 - **NOTE:** If a tag is required for the SLAM 1 tag per 2 tiles rule it should be one of the superfluous traffic sign tags (such as pedestrian crossing)
- Tag and sign relative position: around XXX; printed from the [signs and tags doc](#)
- April tag ids and size are specified in the [April Tags DB](#)

Localization Signs

Specs

Localization signs contain only a tag and no traffic or road name sign. The id of these tags is specified in the [April Tags DB](#). These signs can be printed from [this document](#).

Placement

A localization tag should be placed at each corner.

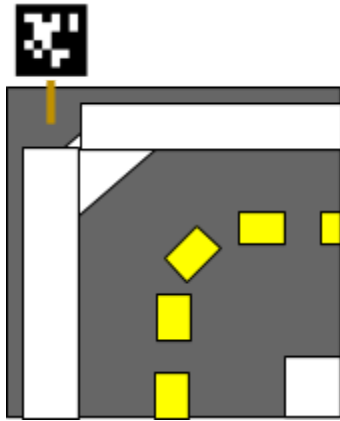
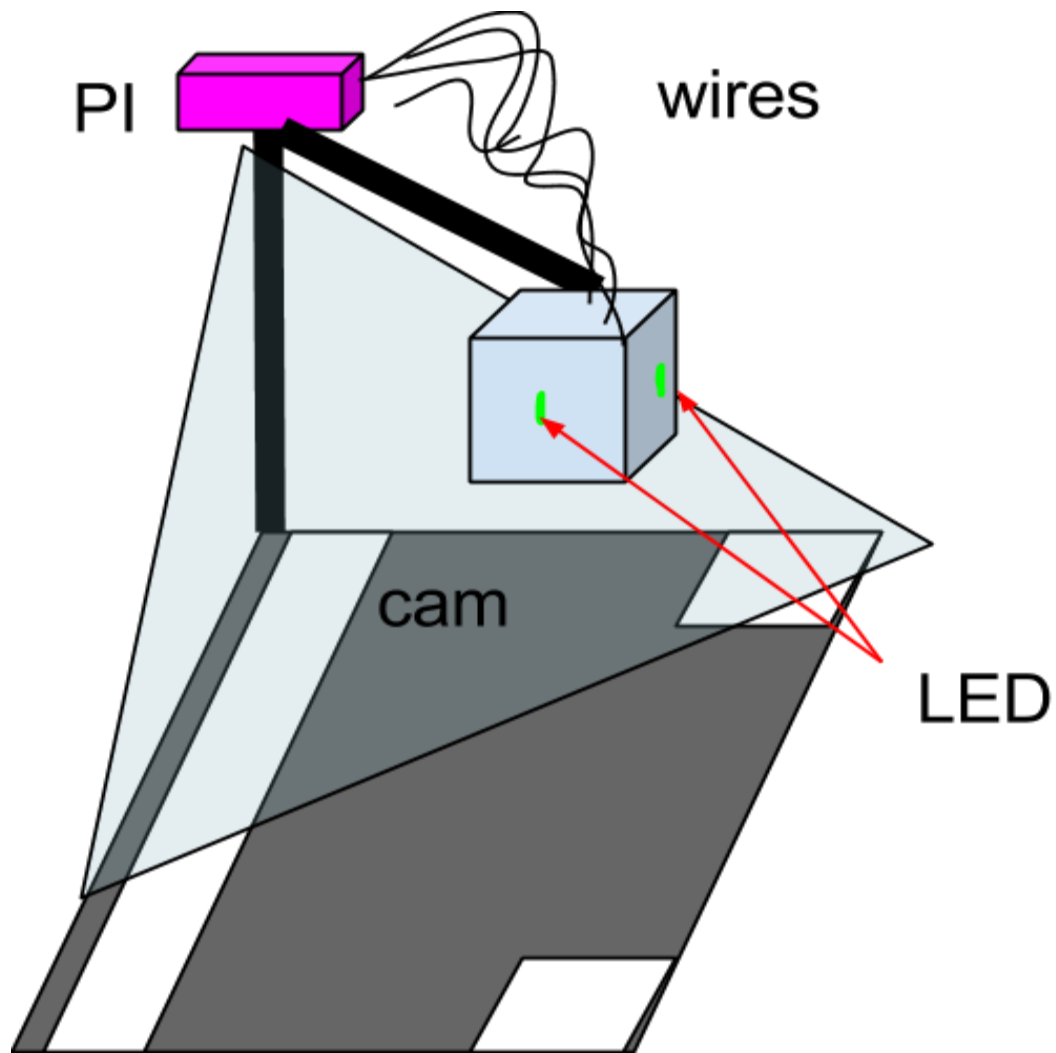


Fig: The placement of localization signs. The should be placed at each corner pointing diagonally towards the center of the tile (not as shown in the figure)

Traffic Lights

Specs

???



Placement

The lights **MUST** be at a height of **EXACTLY 20cm** above the center of the tile.
The Pi **SHOULD** sit on a pole that is based at the corner of the tile outside of the allowable driving region.

Test Scenario

```

$ roscore
$ rosbag play
~/Dropbox/duckietown-data/logs/20160122-logs/160122_intersection1/160122-intersection1_ferrari.bag -l
$ roslaunch image_transport republish theora in:=ferrari/camera_node/image _image_transport:=compressed out:=rosberry_pi_cam/image_raw
$ roslaunch street_name_detector street_name_detector.launch
$ rviz
subscribe
/ferrari/camera_node/image
/street_name_detector/tags_image

```

street_name_detector_node

- use MSER to detect region proposal
- use FAST keypoints to detect the corners of text strokes.
- Need some hacking for parameters

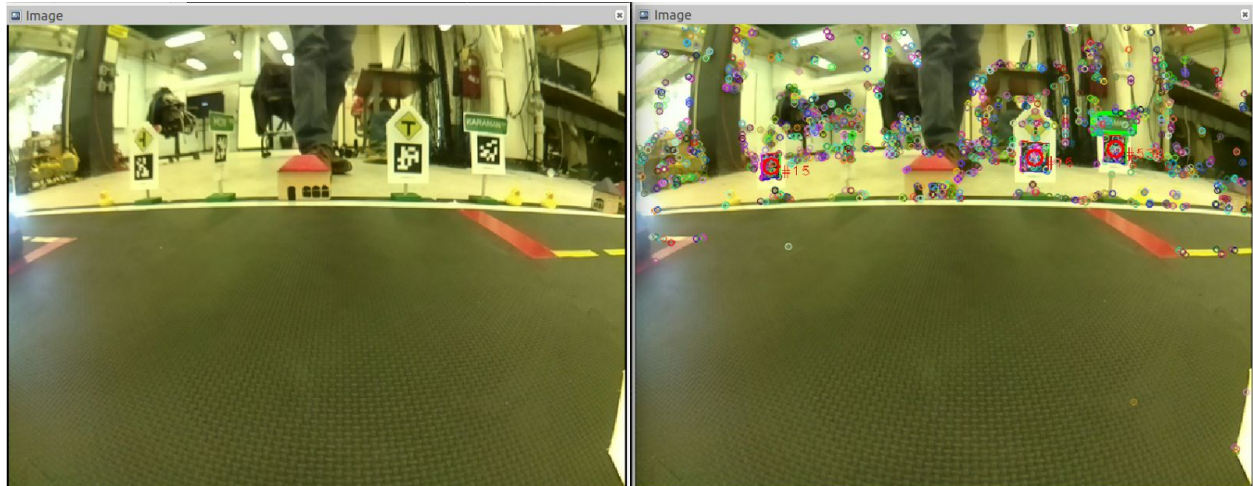
Estimation:

1. Color (green plate) is not robust, but still useful. Currently using L*a*b color space
2. Text is too small; we need keypoints being detected inside the letters -> suggest to increase the size to 6cm x 18cm (currently 3.81 x 11.43)
3. all "ST" or "AVE" are too small to be detected

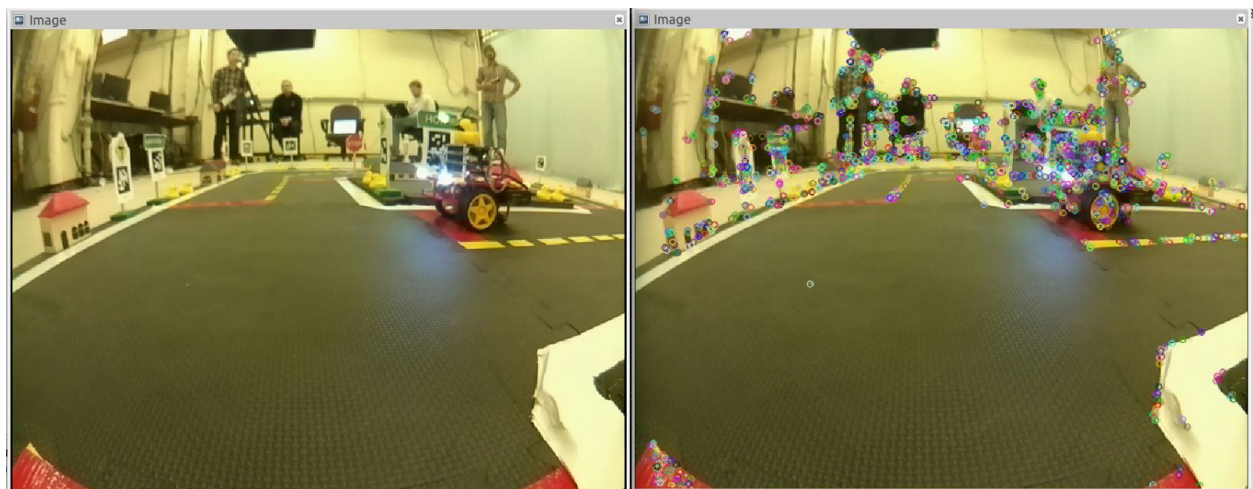


HOW (right) -> should be detectable and decodable (readable)

HOW (left) -> may be detectable but not decodable



KARAMAN -> region is detected but text is not (too few corners)



Occlusion -> not detected

Semantics of LEDS

headlights: white, constant

Assumption:

- 20 fps to do LED detection
- 1s to decide
- 3 frequencies to detect

-> ≤ 5 hz

tail lights: red, **6 hz square wave**

traffic light "GO" = green, **1 hz square wave**

traffic light "STOP" = red, **1.5 Hz square wave**

duckie light on top, state 0 = off

duckie light on top, state 1 = blue, **3 Hz, square wave**

duckie light on top, state 2 = ?, **2.5 Hz square wave**

duckie light on top, state 3 = ?, **2 Hz square wave**