Copyright Policy

All content included on the Site or third-party platforms as part of the class, such as text, graphics, logos, button icons, images, audio clips, video clips, live streams, digital downloads, data compilations, and software, is the property of BitTiger or its content suppliers and protected by copyright laws.

Any attempt to redistribute or resell BitTiger content will result in the appropriate legal action being taken.

We thank you in advance for respecting our copyrighted content.For more info see https://www.bittiger.io/termsofuse and https://www.bittiger.io/termsofservice

版权声明

所有太阁官方网站以及在第三方平台课程中所产生的课程内容，如文本，图形，徽标，按钮图标，图像，音频剪辑，视频剪辑，直播流，数字下载，数据编辑和软件均属于太阁所有并受版权法保护。

对于任何尝试散播或转售BitTiger的所属资料的行为，太阁将采取适当的法律行动。

我们非常感谢您尊重我们的版权内容。

有关详情，请参阅
https://www.bittiger.io/termsofuse
https://www.bittiger.io/termsofservice

Introduction to Straight Lane Finding

A Gentle CV Introduction for Self-Driving Car

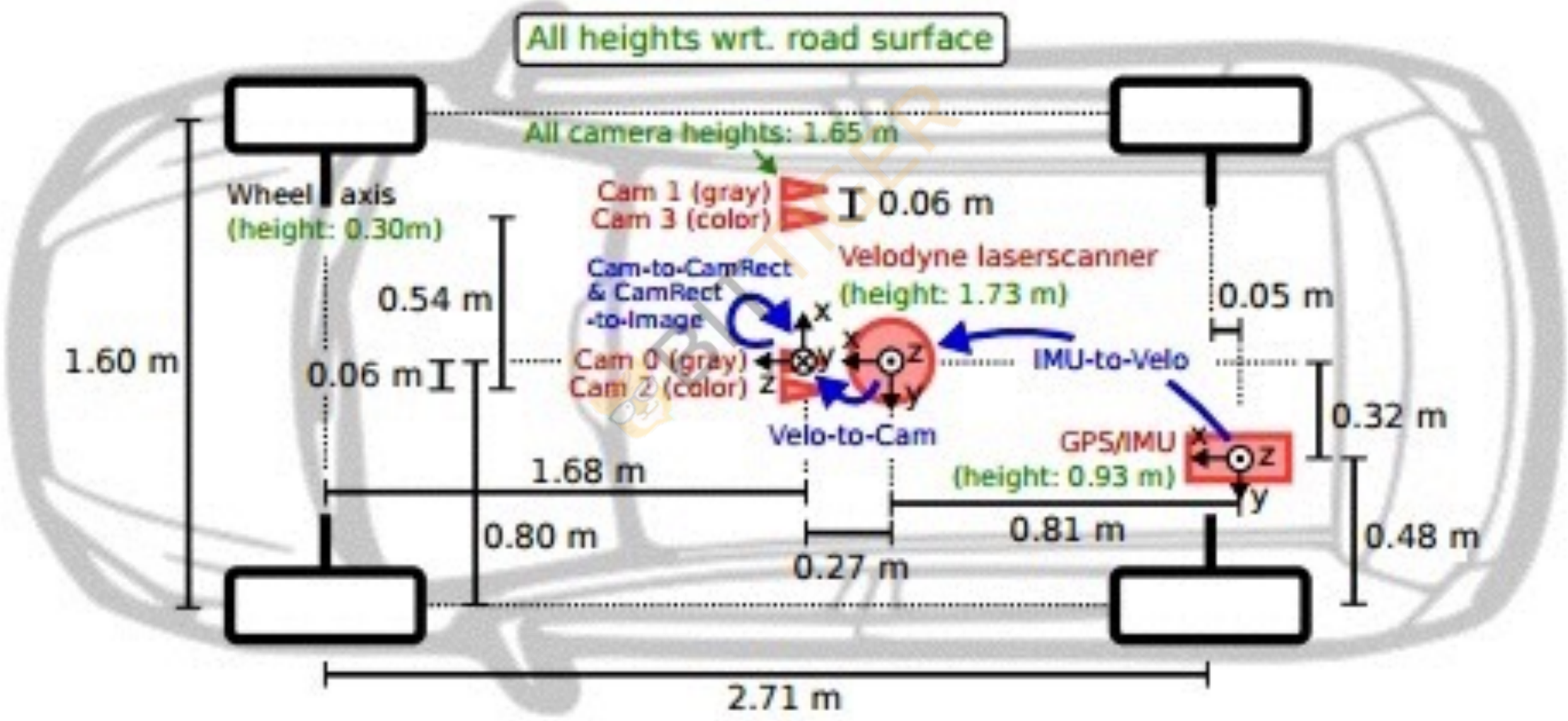
Question to start

Why are we doing lane finding ?

What are we going to do after lane finding?

How are we going to achieve it ?

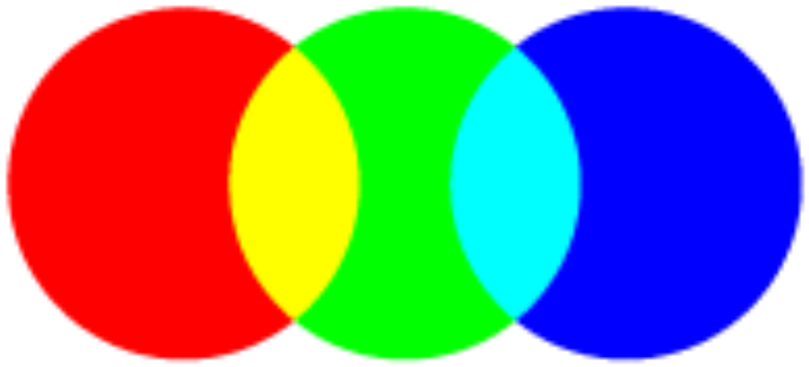
Sensors in Self-driving car



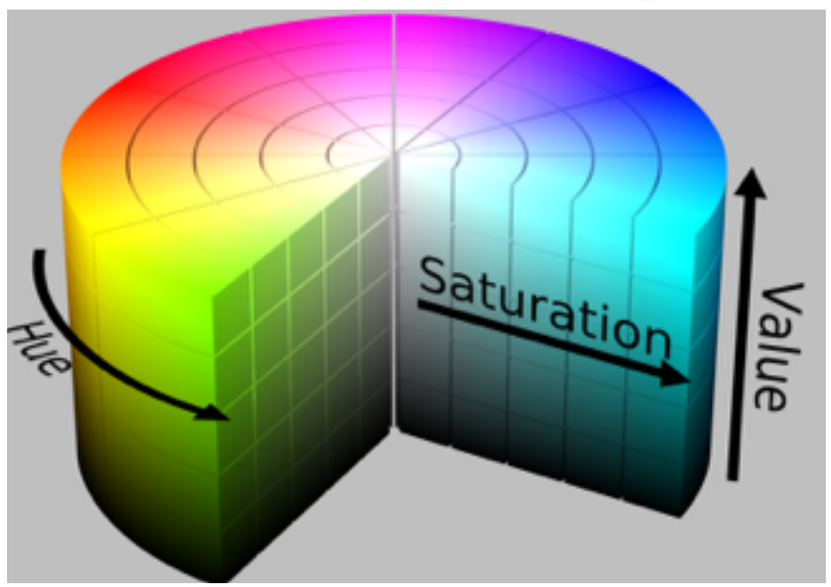
The Difficult Part of Building Driverless Cars

Taken from Wall Street Journal

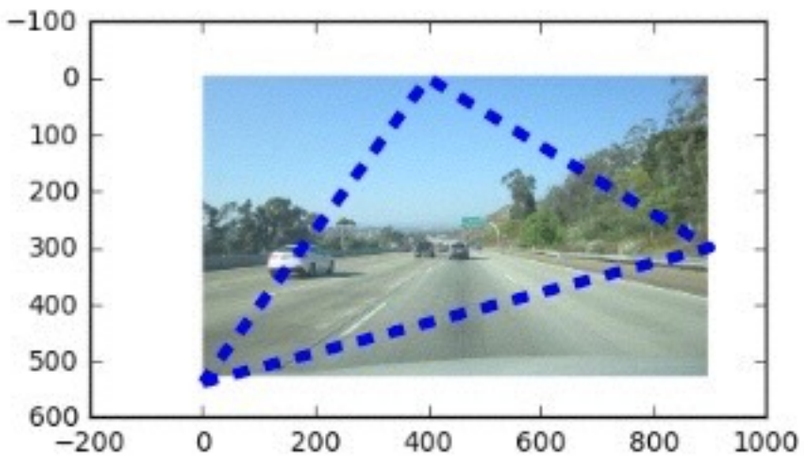
Most Common Color Selection

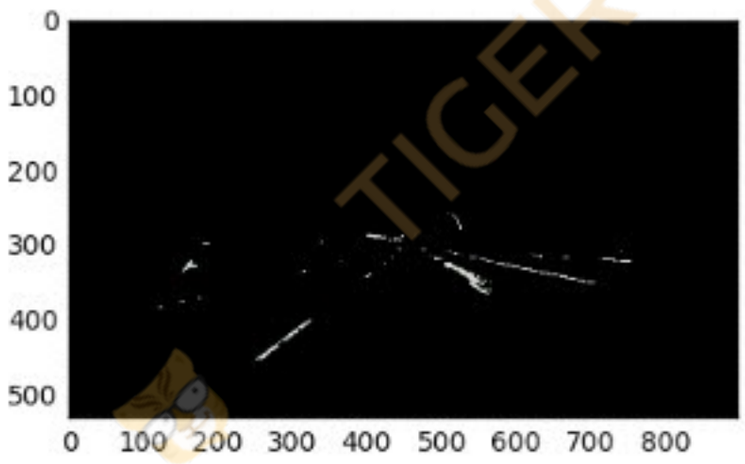


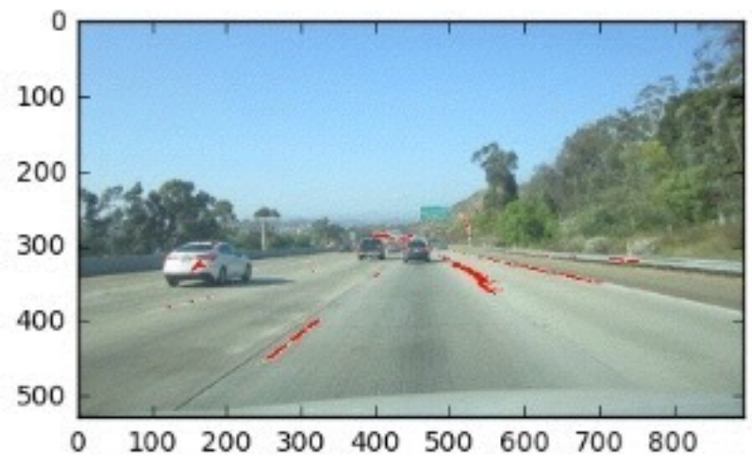
R=Red, **G** = Green, **B** = Blue



Color Region Mixed



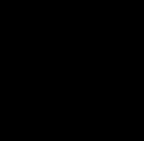




Color Selection

I want to detect a specific color say, **black**, from a front face camera

RGB: 0, 0, 0
CMYK: 0, 0, 0, 100
HSV: ??, ??, ??



Threshold

```
red_threshold = 200
green_threshold = 200
blue_threshold = 200
```

Here’s what we know so far

Color Region

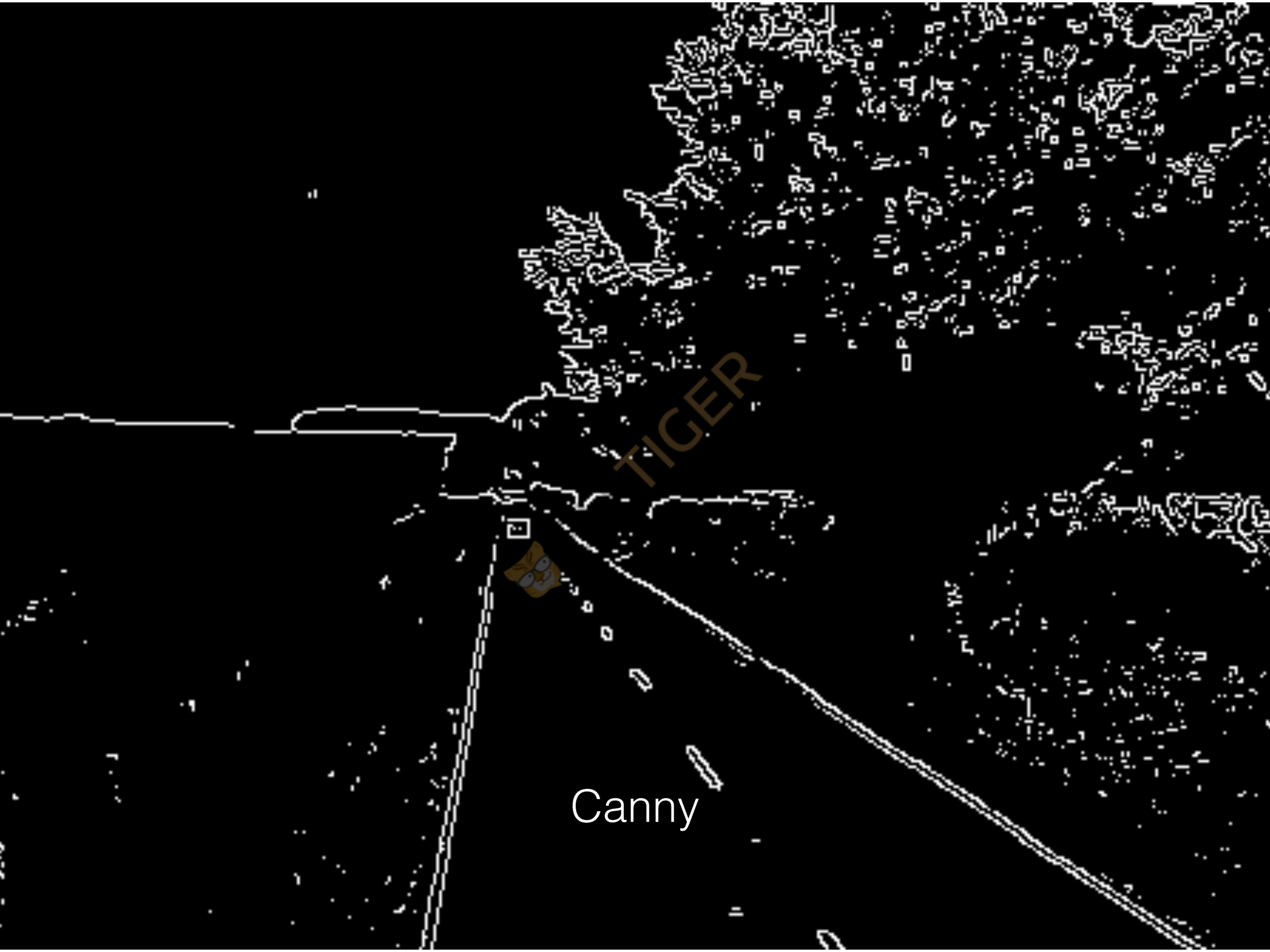
We did triangulation , mask the threshold, find region and mask the selection. Manual process concept.

What is
`cv2.Canny ?`

cv2.Canny(gray, low_threshold, high_threshold)

What is
`cv2.HoughLines ?`

cv2.HoughLinesP(edges, rho, theta, threshold, np.array([]),
min_line_length, max_line_gap)


Canny

Hough lines

Here’s what we know so far

Color Region

We did triangulation , mask the threshold, find region and mask the selection. Manual process concept.

Canny Edge

Canny detection goal is to identify the boundaries of an object in a image. Convert to grayscale then take the gradient. After that, trace out the edges with strongest gradient.

Hough Lines

The Hough transform is a feature extraction technique used in image analysis, computer vision, and image processing The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.

Steps For Finding Lanes

1. Determine the Region of Interest (ROI) by slicing the image (focus the middle part)

2. Grayscale the ROI

3. Equalized the grayscaled ROI with **`cv2.equalizeHist`**

4. Apply Gaussian blur to (3)

5. Threshold (4) by using **`cv2.adaptiveThreshold`**

6. Skeletonize (5) by using skimage.morphology.skeletonize

7. Apply the **`cv2.HoughLines`** on (6)

But wait, what about

Sobel, Harris, and Probabilistic Hough?

Sobel Edge Detection

Canny Edge Detection

Sobel detection refers to computing the gradient magnitude of an image using 3x3 filters. Where "gradient magnitude" is, for each a pixel, a number giving the greatest rate of change in light intensity in the direction where intensity is changing fastest.

Harris

Harris Corner Detection looks for corners because corners are translation invariant and rotation invariant while distinguishable, unlike edges. These properties make corners good feature candidates.

First it computes the horizontal and vertical derivatives (edges) of an image, then it performs cross correlation on these edge images to highlight corners, and then it performs non-maximum suppression to get rid of the edge features.

Probabilistic Hough

Use line polar coordinate

$$\rho = x \cos\theta + y \sin\theta$$

where:

ρ (rho) = distance from origin to the line. [-max_dist to max_dist].
max_dist is the diagonal length of the image.
θ = angle from origin to the line. [-90° to 90°]

Probabilistic Hough

Algorithm:

1. Perform Corner or Edge Detection as input image

2. Rho range and Theta range creation

3. Setup Hough accumulator in 2D army with numbers equal from p and theta

4. Find the total points/pixels contributed for potential line lanes

5. Perform Peak finding by applying threshold

Quiz

Let’s say your project manager asked; what are some very fast, less computation edge detection techniques?

A. Canny B. Harris C. Sobel D. Marr-Hildreth

Functions

Some OpenCV functions (beyond those introduced in the lesson) that might be useful for this project are:

`cv2.inRange()` for color selection
`cv2.fillPoly()` for regions selection
`cv2.line()` to draw lines on an image given endpoints
`cv2.addWeighted()` to coadd / overlay two images
`cv2.cvtColor()` to grayscale or change color
`cv2.imwrite()` to output images to file
`cv2.bitwise_and()` to apply a mask to an image

Moving to the Project

I would recommend considering using the Probabilistic Hough Line Transform for our application. In OpenCV's Python API, it's implemented in the function, cv2.HoughLinesP.

Lane detection is an essential component of many intelligent vehicle applications, including Lane Following (LF), Lane Keeping Assistance (LKA), Lane Departure Warning (LDW), lateral control, Intelligent Cruise Control (ICC), Collision Warning (CW) and eventually autonomous vehicle guidance.

Let’s Prototype

Here’s what we learn so far

Color Region

We did triangulation , mask the threshold, find region and mask the selection. Manual process concept.

Canny Edge

Canny detection goal is to identify the boundaries of an object in a image. Convert to grayscale then take the gradient. After that, trace out the edges with strongest gradient.

Hough Lines

The Hough transform is a feature extraction technique used in image analysis, computer vision, and image processing The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.

Harris and Sobel

Harris Corner Detection looks for corners because corners are translation invariant and rotation invariant while distinguishable, unlike edges. Sobel detection refers to computing the gradient magnitude of an image