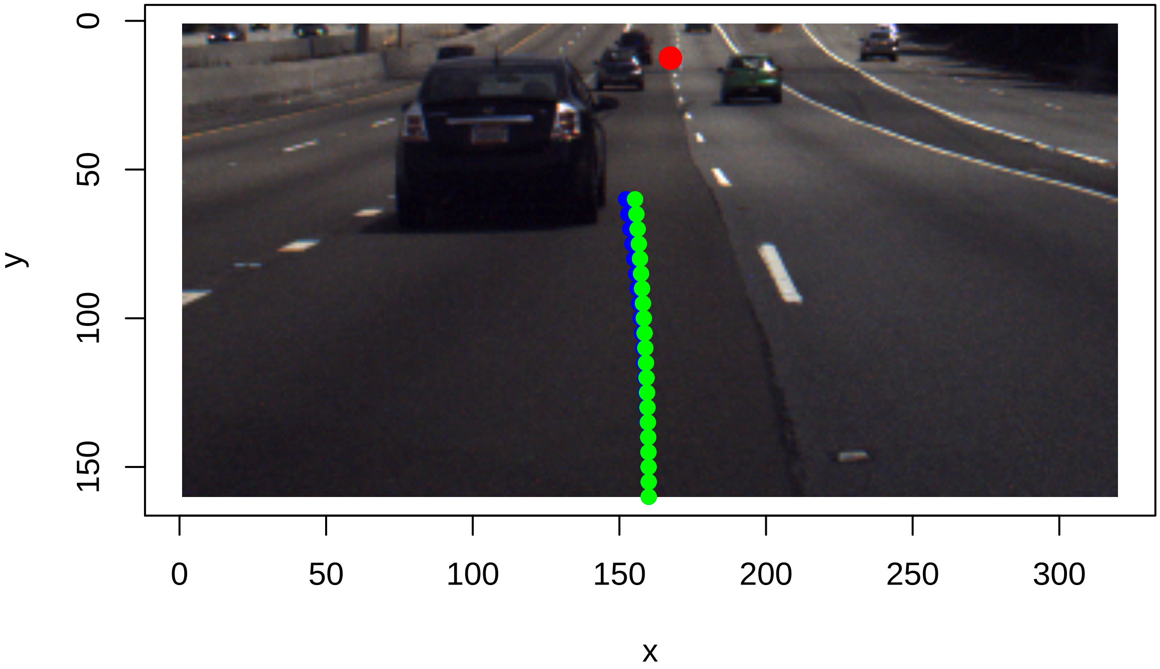
rf_with_vanishing_point

Zeyi Wang

October 21, 2016

0. Scope of the self-driving problem, revisit

- We focus on **image processing**:
  - human drivers don't need lidar
  - gps and lidar adjustment
- Main challenges:
  - vanishing point
  - vehicle detection [for lane changing, heavy traffic]
- Problem of interest:
  - **steering angle** prediction
  - **robust vanishing point detection** as an estimator
  - for lower traffic complexity, highway mainly [lane markings]
- Not of interest:
  - traffic signs [particle filters]
  - lane changing [deep learning or lidar + markings detection + vehicle detection]
  - driving route [gps]



Methods

Data Collection

We use the driving dataset shared by comma.ai on July 2016. We downloaded it on September 17, 2016.

Data was collected by an Acura ILX 2016, including video clips 20 Hz and log information for speed, acceleration, steering angle, etc. at 100Hz. Both were recorded to a same timeline.

Exploratory Analysis

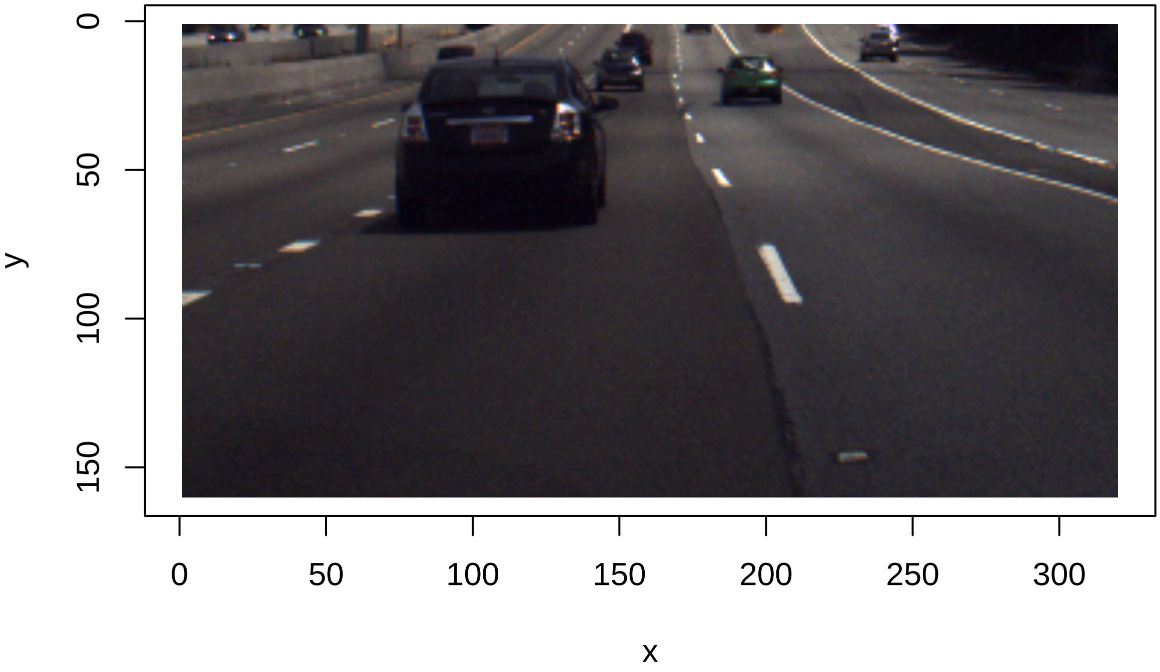
Statistical Methods

1. Edge detection and greyscale

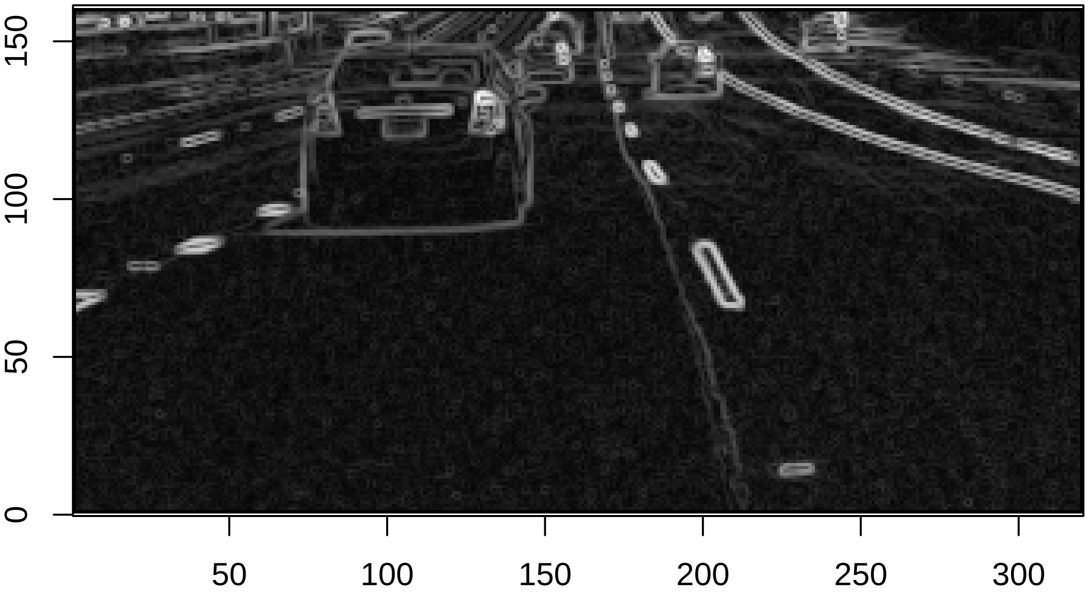
For each frame from the original video, a Sobel filter is applied as the first step. The benefit of Sobel is simplicity and fast computing.

The image is then tranformed into greyscale so that we could filter for markings.

```
i <- 595
k <- index_train[i]
image_use <- (as.cimg(aperm(image[image_names][k,,,], c(4,3,1,2))))
result <- edges( make.image(image_use[,160:1,1,]), type = "Sobel")
result <- rgb2grey(result)
plot(image_use)
```



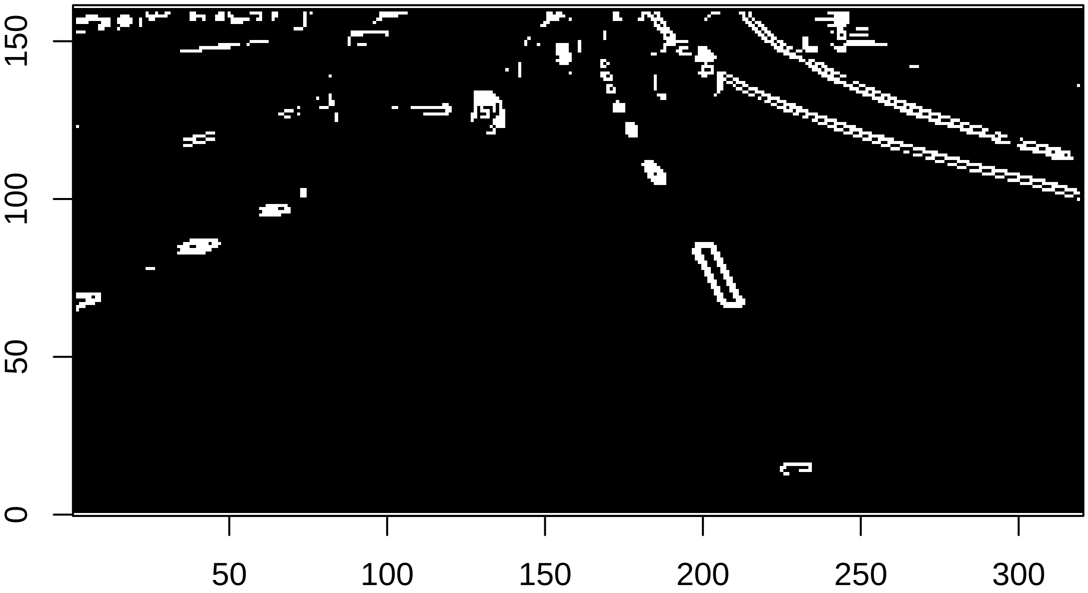
```
show.image(result)
```



2. Dynamic lane detection

A 97% quantile threshold is applied for dynamic lane detection.

```
temp.M <- extract.image(result)
temp.MM <- (temp.M > quantile(temp.M, 0.97))*1
show.image(make.image(temp.MM))
```

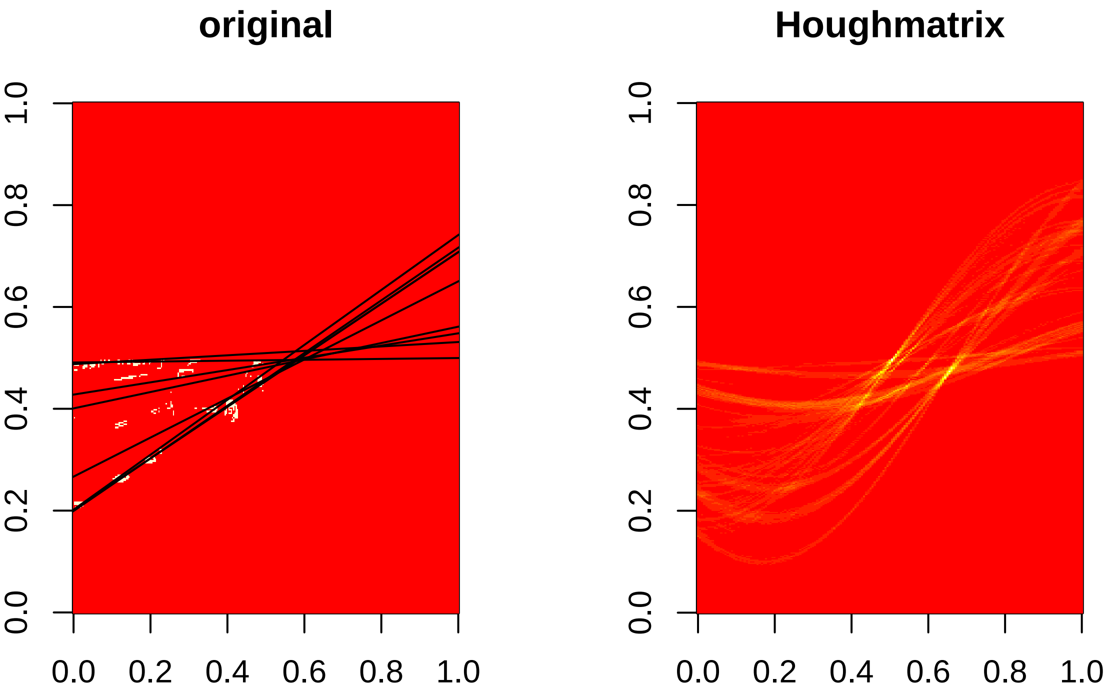


```
temp <- which(temp.MM == 1, arr.ind = T)
```

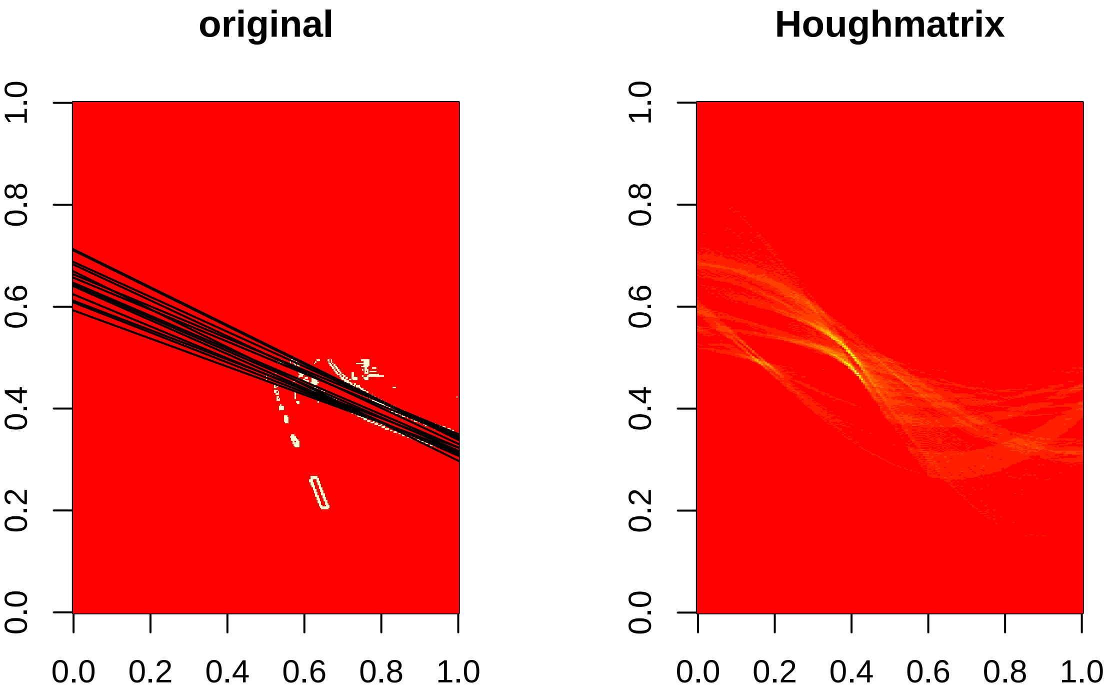
3. Hough transformation

To get robust vanishing point estimation, we first apply hough transformation to left/right part of the preprocessed image, seperately.

```
## Using Hough1 --> complete
```



```
## Using Hough1 --> complete
```

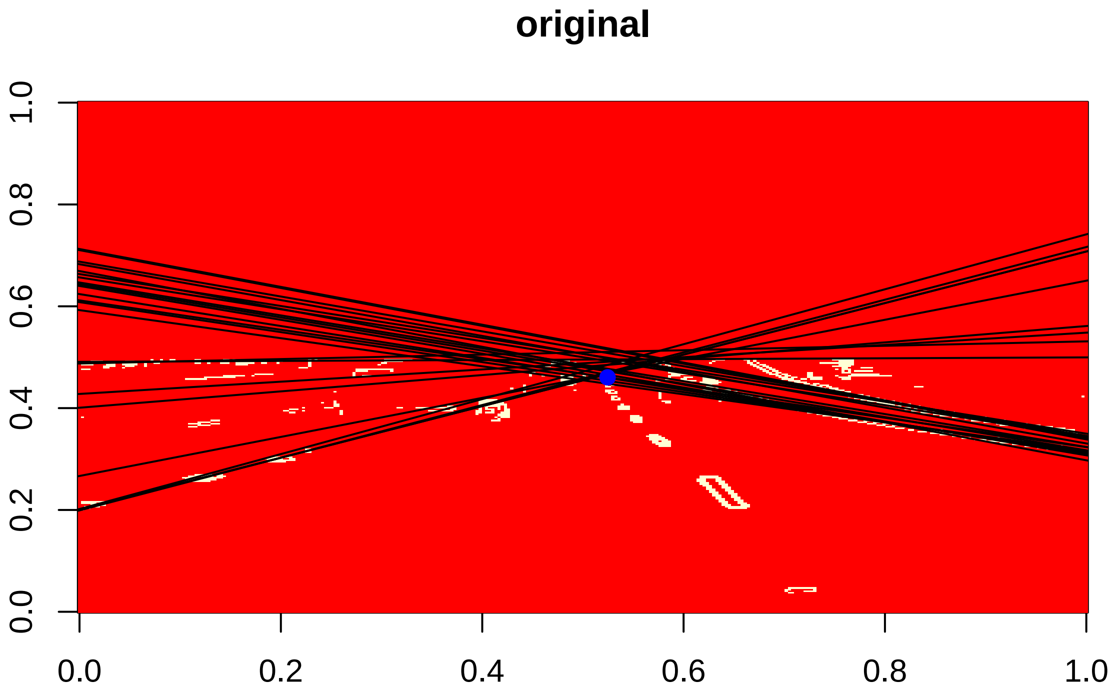


Combine together we have:

```
## Using Hough1 --> complete
```

```
## Using Hough1 --> complete
## Using Hough1 --> complete
```

```
## Using Hough1 --> complete
## Using Hough1 --> complete
```



Visualization

test train