Challenge 4 - Lane Detection System

Friday, April 1, 2016 8:17 AM

Relevance: Autonomous driving is becoming a reality. And one of the key component is a camera-based lane detection.

	
Input	Sample results. *First four numbers are distances at each end of line and the second two numbers are new point and total points.*

Goal: Write an algorithm that will automatically find two lanes that are closest to the car in an image as quickly as possible.

Work on:
Update detect_lanes.m
- Main function which returns the two lanes
- [rho1, theta1, rho2, theta2] = detect_lanes (input_im)
  - [rho1, theta1] - first lane
  - [rho2, theta2] - second lane

Useful concepts
- Various image enhancements and filtering
- Edge detection
- Hough transform

Dataset:
- All images are color. Same size within each video. The size of images between videos may vary.

Get started
- Download train.zip [LINK TO ZIP] that contains
  - one folder called "train"
  - inside "train", you will find
    - detect_lanes.m: an awful attempt at solving this challenge. *Simply returns two random lines*
    - grade_detect_lanes.m: computes the performance based on the images included in this folder
    - *.png: images
    - *.csv: excel file that contains locations of two lanes in (rho1, theta1) and (rho2, theta2)
- Note that grade_detect_lanes has code that visualizes the results and displays. Feel free to delete if you don't need it. The testing code will not have such visualization code.

How to test before submitting
- Place your detect_lanes.m inside "train" folder
- Run grade_detect_lanes.m

File requirements. *Note that not following will mean that you get 10%*

- one zip file with
  - detect_lanes.m
  - other matlab files that you have written.
  - lastname_firstname_detect_lanes.ppt (if you are presenting)
- ZIP file. ***Nor rar, 7zip, … plane old zip.***

How TA/instructor will grade

1. Download your assignment
2. Unzip and move your file to "test" folder
3. Run grade_detect_lanes.m for 60 sec per image on each image. We will test on total of 30 images.
   a. **1-4 point** if the returned lanes are both "accurate".
      i. Accurate if all x-positions at y=1 and y=480 are both within ground truth
      ii. 1 point if every point (not sum of distance) is within 100 pixels of ground truth.
      iii. 2 points if within 75 pixels
      iv. 3 points if within 50 pixels
      v. 4 points if within 25 pixels
   b. No penalty for wrong answer.
   c. Only one attempt for each image
   d. Order of images tested is same for everyone.

What can I use?

- General principle:
  - Try to focus on using the concepts that you learned in the class.
  - You must thoroughly understand what you are using. *You will be asked to clearly explain how the method that you used works either during presentation or on-demand Q&A session with the instructor.*
  - Thus, simply downloading some other "state-of-the-art" method and using it will earn
    - will be considered "not meaningful" submission
    - 0 for the presentation
- Specific to this challenge:
  - Allowed to use
    - built-in function of
      - all concepts that we covered up to challenge 3 (including auto thresholding and convolution)
  - Cannot use
    - built-in function of
      - hough transform
      - edge detection
  - Other info
    - The heart of the method must be based on "the concepts that we have learned".