CS268: MACHINE PERCEPTION

Homework 4: Build a Lane-Departure Warning System

Fall 2012

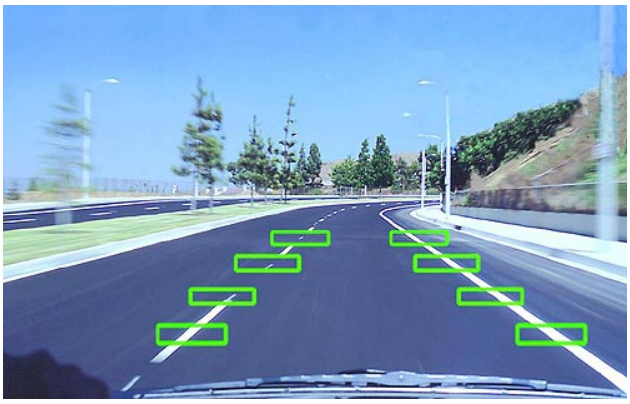

Figure 1: *Lane Departure Warning System Concept*

In Los Angeles, almost nobody *just* drives. Common simultaneous activities include shaving, browsing, texting, makeup, reading, playing solitaire. This makes for a very dangerous transportation environment. While you are a responsible driver and *never, ever* text-and-drive, you are occasionally underslept from all the homework in CS268. So you are determined to make your own lane-departure warning system, to run on your cellphone. You will then place the cellphone on a suction-cup holder on the windshield of your car, outward looking, and have it emit an auditory signal when you are about to depart your lane.

In this homework you will assemble the building blocks of this system.

- Using the camera calibration routine from the OpenCV, calibrate your phone camera. If you do not have a phone with a camera, use the images posted on the courseweb in the directory `LDWS calibrate.zip` to gain practice in camera calibration. **Return the intrinsic calibration parameters of the camera.**

- Choose some sensible *regions of interest* (horizontal strips of pixels) in the image, and – also using OpenCV – run an edge detector to determine the lane boundaries. Test it by taking images of roads or driveways, and try to determine the two lane/road boundaries. If you do not have a way to capture your own images, use those in the courseweb in `LDWS test.zip`. Use RANSAC to handle outlier resposes from the edge detector. **Return an algorithm, best in Python, that detects the edges (calling OpenCV primitives) and fits two lines using RANSAC.**

- Place the camera on the dashboard of your car, and take a picture of a road/driveway. Alternatively, use the data in `LDWS test.zip` in the courseweb. Using knowledge of the fact that lane boundaries are parallel lines on the ground plane, and they are at a distance of $3.66m$ from each other, calibrate the *extrinsic* parameters, that is, determine the position and orientation of the camera relative to a reference frame with origin on the ground plane, with Z axis parallel to the road axis, X axis perpendicular to

it on the road plane, and Y pointing upward from the road plane. Make sure the reference frame has the corrected orientation (that is, $X \times Y = Z$, not $X \times Y = -Z$). Place the origin with $Y = 0$ (i.e. on the road plane), X half-way between the two lane boundaries (i.e. in the center of the road), and Z corresponding to the projection of the camera position onto the road plane. **Return the extrinsic calibration parameters, and the Python script to obtain them** This will be the basic routine that estimates your position relative to the center divider. Generate a warning if the position of the camera along the X axis deviates from the center divider by half the width of the lane (1m).

- (OPTIONAL): Estimate the projective transformation that maps the road plane onto the image plane. Run a Vehicle detector using the OpenCV; use the projective transformation to determine the *distance* to the closest vehicle. Visualize a top view of the road plane, with vehicle markings on it.