UDACITY

PROJECT SPECIFICATION

Advanced Lane Finding

Camera Calibration

CRITERIA	MEETS SPECIFICATIONS	REVIEWER TIPS
Have the camera matrix and distortion coefficients been computed correctly and checked on the calibration test image?	OpenCV functions or other methods were used to calculate the correct camera matrix and distortion coefficients using the calibration chessboard images provided in the repository. The distortion matrix should be used to un-distort the test calibration image provided as a demonstration that the calibration is correct	Most students will use the OpenCV functions we demonstrate in the lesson to do this procedure, but it is fine if they use other methods, as long as they arrive at the correct numbers (they should all arrive at essentially the same result), and the un-distorted test image looks ok.

Pipeline (single images)

CRITERIA	MEETS SPECIFICATIONS	REVIEWER TIPS
Has the distortion correction been correctly applied to each image?	Distortion correction that was calculated via camera calibration has been correctly applied to each image.	Again, most students will use the OpenCV functions to perform this step, but other methods are fine, as long as they arrive at the correct result (an un-distorted image) in the end. This is essentially the same as applying the distortion correction to the test calibration image, just that now that it's been tested it is applied to real-world images.
Has a binary image been created using color transforms, gradients or other methods?	At least two methods (i.e., color transforms, gradients) have been combined to create a binary image containing likely lane pixels. There is no "ground truth" here, just visual verification that the pixels identified as part of the lane lines are, in fact, part of the lines.	Here we are expecting students to combine color and gradients to get a satisfactory result, but they are free to bring in other methods if they wish.

CRITERIA	MEETS SPECIFICATIONS	REVIEWER TIPS
Has a perspective transform been applied to rectify the image?	OpenCV function or other method has been used to correctly rectify each image to a "birds-eye view"	Here again, most students will use OpenCV functions, but they are free to take other approaches.
Have lane line pixels been identified in the rectified image and fit with a polynomial?	Methods have been used to identify lane line pixels in the rectified binary image. The left and right line have been identified and fit with a curved functional form (e.g., spine or polynomial).	The methodology here is pretty much wide open, but students should implement some series of steps to identify the pixels that are part of the lane lines, separate the left line from the right and fit each with a curve.
Having identified the lane lines, has the radius of curvature of the road been estimated? And the position of the vehicle with respect to center in the lane?	Here the idea is to take the measurements of where the lane lines are and estimate how much the road is curving and where the vehicle is located with respect to the center of the lane. The radius of curvature may be given in meters assuming the curve of the road follows a circle and the position of the vehicle within the lane may be given as meters off of center.	Here the main idea is to get the student thinking about how to transform their measurements into control outputs like how much to turn and where they are located in the lane. As long as the estimates are somewhat reasonable the specifications have been met.
Has the result from lane line detection been warped back to the original image space and displayed?	The fit from the rectified image has been warped back onto the original image and plotted to identify the lane boundaries. This should demonstrate that the lane boundaries were correctly identified.	The result in this case should be an image with the lane markings drawn on it. The lane markings drawn should generally do a good job of identifying the actual lane on the road.

Pipeline (video)

CRITERIA	MEETS SPECIFICATIONS	REVIEWER TIPS

Does the pipeline established with the test images work to process the video?	The image processing pipeline that was established to find the lane lines in images successfully processes the video. The output here should be a new video where the lanes are identified in every frame, and outputs are generated regarding the radius of curvature of the lane and vehicle position within the lane. The identification and estimation don't need to be perfect, but they should not be wildly off in any case. The pipeline should correctly map out curved lines and not fail when shadows or pavement color changes are present.	It's difficult to get this to work perfectly, so a rough approximation of the lane position is fine in this case. The main thing is that students are able to detect curving lines and that their algorithm doesn't fail when shadows or color changes are present.
Has some kind of search method been implemented to discover the position of the lines in the first images in the video stream?	In the first few frames of video, the algorithm should perform a search without prior assumptions about where the lines are (i.e., no hard coded values to start with). Once a high-confidence detection is achieved, that positional knowledge may be used in future iterations as a starting point to find the lines.	There is quite a bit of room for creativity here, but basically we would like students to perform some kind of search for the lines initially until they determine they have found a high-confidence detection of the lines.
Has some form of tracking of the position of the lane lines been implemented?	As soon as a high confidence detection of the lane lines has been achieved, that information should be propagated to the detection step for the next frame of the video, both as a means of saving time on detection and in order to reject outliers (anomalous detections).	There is room for creativity here, some students may get serious and try Kalman filters or other methods but basically the idea is just to take the information from the last detection (or several detections to inform the search for lane lines in the next frame.

Readme

CRITERIA	MEETS SPECIFICATIONS	REVIEWER TIPS
Has a Readme file been included that describes in detail the steps taken to construct the pipeline, techniques used, areas where improvements could be made?	The Readme file submitted with this project includes a detailed description of what steps were taken to achieve the result, what techniques were used to arrive at a successful result, what could be improved about their algorithm/pipeline, and what hypothetical cases would cause their pipeline to fail.	As long as a student puts some thought and effort in to meet specifications this item should pass. Mostly we're looking for a student to be able to articulate what they have done to achieve their result, where their algorithm might fail on other datasets, and what could be done to improve it.

Reviewer FAQ Student FAQ