Advanced Lane Finding Project

Rahul Radhakrishnan
2020 /August/05

Here's my resubmission. I refined the YUV & HLS thresholds to get a more refined lane detection.

However, at one point (0.24s) of the video, there is a flicker, which I am unable to get rid off.

```
def Pipeline(original_image):

    test_image = original_image

    grad_x = abs_xsobel_thresh(test_image, orient='x', Thresh_min=30, Thresh_max=150)
    grad_y = abs_ysobel_thresh(test_image, orient='y', Thresh_min=20, Thresh_max=100)
    mag_binary = mag_thresh(test_image, sobel_kernel=9, mag_thresh=(100, 105))
    dir_binary = dir_threshold(test_image, sobel_kernel=15, thresh=(0.1, 1.0))
    s_binary = s_threshold(test_image, sobel_kernel=17, thresh=(100,255))
    l_binary = l_threshold(test_image, sobel_kernel=17, thresh=(80,255))
    h_binary = h_threshold(test_image, sobel_kernel=17, thresh=(150,255))
    y_binary = y_threshold(test_image, sobel_kernel=17, thresh=(20,255))
    color_binary = np.dstack(( np.zeros_like(grad_x), grad_x, s_binary)) * 255

    combined = np.zeros_like(mag_binary)
    combined[((grad_x == 1) | (y_binary == 1) & (s_binary == 1) & (l_binary == 1))] = 1

    warped = warp_image(combined)

    Window_Image = detect_lane(warped)

    Unwarped_Image = unwarp_image(Window_Image)

    Final_Image = cv2.addWeighted(test_image, 1, Unwarped_Image, 0.3, 0)

    Radius = rad_of_curvature(Final_Image)

    Car_Position = car_distance(Final_Image)

    cv2.putText(Final_Image,"Radius of Curvature is " + str(Radius)+ "m", (100,100), 2, 1, (255,255,0),2)

    cv2.putText(Final_Image,"Distance from center is {:2f}".format(Car_Position)+ "m", (100,150), 2, 1, (255,255,0),2)

    return Final_Image
```

I also coded the car position with respect to the lane size as shown below.

```
def car_distance(image):

    test_image = image

    grad_x = abs_xsobel_thresh(test_image, orient='x', Thresh_min=30, Thresh_max=150)

    mag_binary = mag_thresh(test_image, sobel_kernel=9, mag_thresh=(100, 105))

    s_binary = s_threshold(test_image, sobel_kernel=17, thresh=(170,255))

    l_binary = l_threshold(test_image, sobel_kernel=17, thresh=(170,255))

    y_binary = y_threshold(test_image, sobel_kernel=17, thresh=(170,255))

    combined = np.zeros_like(mag_binary)
    combined[((grad_x == 1) | (s_binary == 1) & (l_binary == 1) & (y_binary == 1))] = 1

    distance = car_position(combined)

    return distance
```

The radius of curvature can also be seen in the video, though I am not very sure about the numbers.