Computer Vision

wznd85

December 1, 2017

1 Design choices

1.1 Pre-filtering

I masked the top and bottom of the image to ignore the car bonnet and the sky. I had then performed Gamma correction on the images before realising that this decreased the quality of the illuminant variant images. Finally, I converted the left image into both the HSV and illuminant invariant colour spaces for use in finding the region of interest.

1.2 Region of interest

I created 4 separate methods to obtain a region of interest:

- find a mean of a rectangle of pixels just in front of the car and use thresholds to find all the similar pixels in the image;
- use histograms to find the most frequently occurring pixels and choose the pixels that are close to those values;
- use seeded region growing (SRG) to grow a region in the shape of the road;
- similar to the first method, but instead of using a rectangle use 2 different pixels as the seeds.

Although SRG sometimes produced the best results, this was dependent on the seed chosen. Furthermore, it was the slow and hence, impractical in a real-time scenario. For the most part, the first method works very well, as shown by 3. However, sometimes significant shadows -present even in the illuminant invariant image - skew the mean of the pixels in the rectangle, resulting in the pavement being masked instead of the road, as shown by 6.

After performing region-of-interest masking on both the HSV and illu- minant invariant images, I then used bitwise AND to generate a combined mask, which I dilated and contoured to remove smaller regions. However, after significant testing it became obvious that using the illuminant invariant mask alone would be better. I used morphology functions, as well as finding the biggest contour to produce a cleaned version of the mask. Finally, I checked if the cleaned mask was big enough to use in the next section; if not, I used the original illuminant invariant mask.

Figure 1: Combined mask on image 1506942643.476350_L

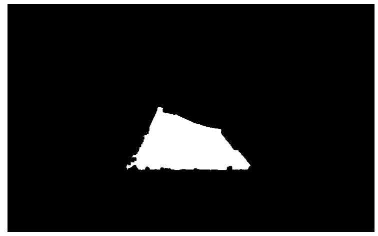

Figure 2: Improved mask on image 1506942643.476350_L

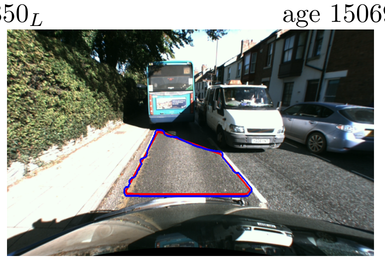

Figure 3: Final plane result on image 1506942643.476350_L

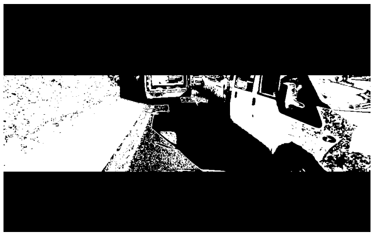

Figure 4: Combined mask on image 1506942644.476266_L

Figure 5: Improved mask on image 1506942644.476266_L

Figure 6: Final plane result on image 1506942644.476266_L

1.3 RANSAC

Before RANSAC, I performed histogram equalisation on the grayscale images before inputting them into the disparity code in order to improve the quality of the map. I then used the mask generated previously on the disparity map, and finally created a 3D points image to be used in RANSAC.

I implemented RANSAC using a plane model on the 3D points image, choosing 40 iterations, a threshold of 0.5, and a value of at least 150 data points needed. I added a check to make sure the Y component was the biggest component of the normal coeffcients and also attempted to choose the best plane based on average distance of points to the plane, instead of the number of inliers, however this did not improve results.

1.4 Drawing

I took two approaches to drawing: the first involved drawing a convex hull around the inliers; the second involved drawing polylines between all the inliers, dilating the lines, and finally drawing a contour around the new shape. This shape is more complex but has the benefit of avoiding certain objects.

To draw the normal, I simply chose a random point from the inliers on the plane, and found another point using the normal coeffcients, and then converted both the points into 2D space to draw an arrow between them.

1.5 Feature Detection

I used an existing HOG detector to find pedestrians in the image. This works well most of the time (7), however occasionally results in a false positive (8).

Figure 7: Example of pedestrian detection on image 1506942486.479530_L

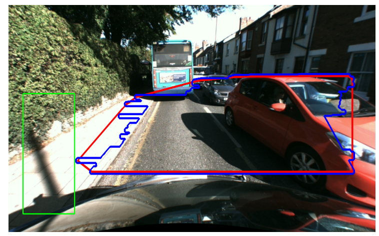

Figure 8: Example of false positive on image 1506942638.476344_L

1.6 Extras

I chose to use Z instead of Zmax in the project_disparity_to_3d and project_3D_points_to_2D_image_points functions to improve visual appearance of results.

2 Results on first image

Figure 9: Original image

Figure 10: Result of applying first image mask

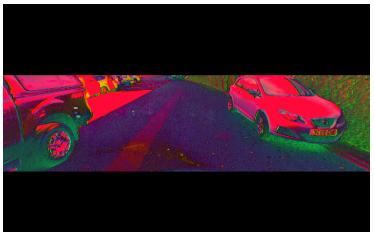

Figure 11: Image in HSV colour space

Figure 12: Image in illuminant invariant colour space

Figure 13: Mask from illuminant invariant image

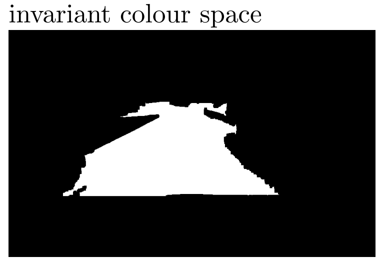

Figure 14: Result of filling biggest contour in mask

Figure 15: Grayscale image from left image

Figure 16: Grayscale image optimised for disparity

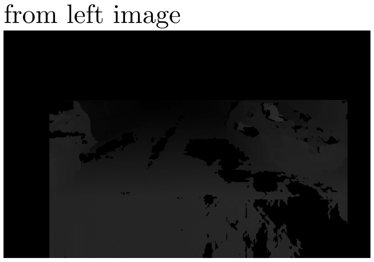

Figure 17: Disparity map obtained using original grayscale images

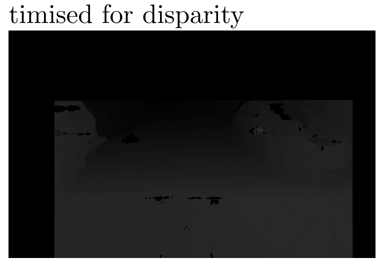

Figure 18: Disparity map obtained using optimised grayscale images

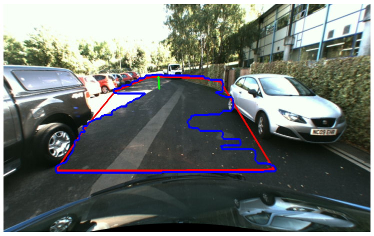

Figure 19: Final plane result using normal disparity map

Figure 20: Final plane result using optimised disparity map

3 Evaluation

I randomly chose 6 images and created ground truth versions of the planes; the ground truth planes were drawn by hand.

I took two measurements:

- Percentage of pixels in ground truth plane that appear in my resulting plane;
- Percentage of pixels in my resulting plane that do not appear in the ground truth plane.

The first measurement is a measure of how well my solution masks the road as a low value would indicate that my plane was not covering much of the road. However, the second measurement is also needed to examine how many outliers there are as otherwise you could simply draw a plane around the whole image and always mask the pixels in the ground truth image, resulting in a distorted view of how well the solution worked.

The pictures used are shown below:

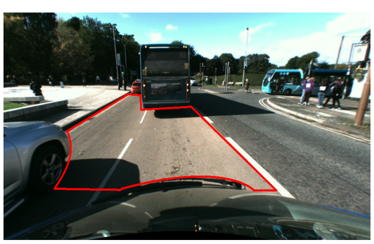

Figure 21: Ground truth for 1506942594.475307_L

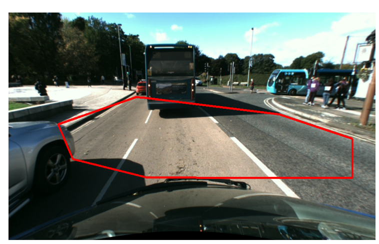

Figure 22: My result for 1506942594.475307_L

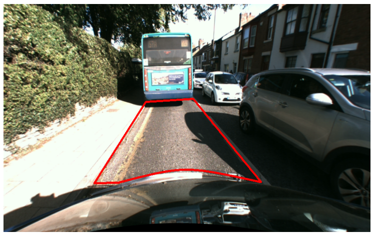

Figure 23: Ground truth for 1506942716.476855_L

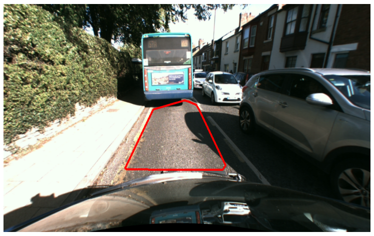

Figure 24: My result for 1506942716.476855_L

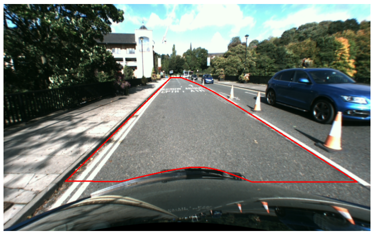

Figure 25: Ground truth for 1506942913.476390_L

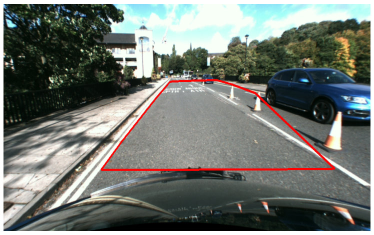

Figure 26: My result for 1506942913.476390_L

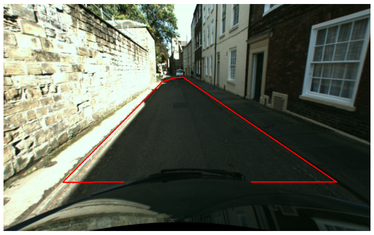

Figure 27: Ground truth for 1506943282.479159_L

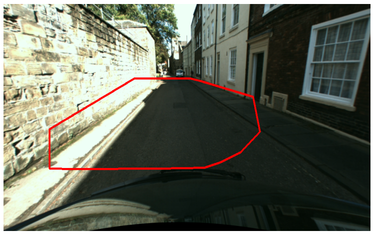

Figure 28: My result for 1506943282.479159_L

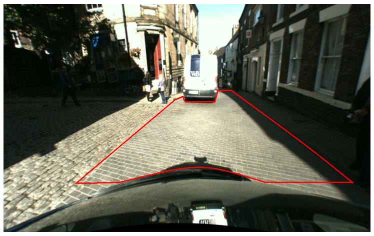

Figure 29: Ground truth for 1506943342.478196_L

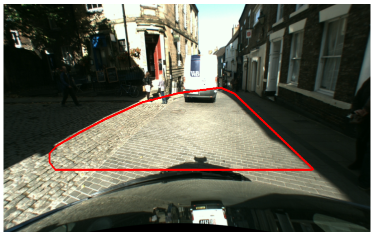

Figure 30: My result for 1506943342.478196_L

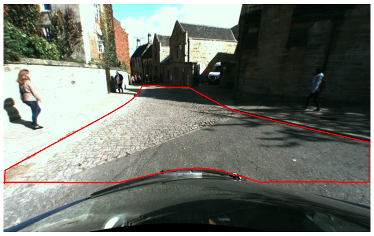

Figure 31: Ground truth for 1506943456.478291_L

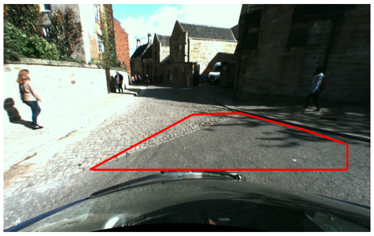

Figure 32: My result for 1506943456.478291_L

Figure 33: Ground truth for 1506943836.379759_L

Figure 34: My result for 1506943836.379759_L

The result for the first measurement was 68%.

The result for the second measurement was 32%.