Interview Assignment - Solution

Sergey Sinitsa

August 14, 2021

Contents

1 Assignment

For a cluster of points $C = \{\boldsymbol{a}_0, \cdots, \boldsymbol{a}_{N-1}\}$, where $\boldsymbol{a}_i = \{x, y, z, R\}, \boldsymbol{a}_i \in \mathbb{R}^4$ is a point with spatial parameters $\{x, y, z\}$ and reflectively R:

1. Write an algorithm that detects if the cluster C is a traffic sign (traffic sign is composed of retro-reflective sign and a pole, see ".png" - Figure 1).
2. For clusters that was classified as traffic sign, recognize the sign's shape: circle/square/triangle.

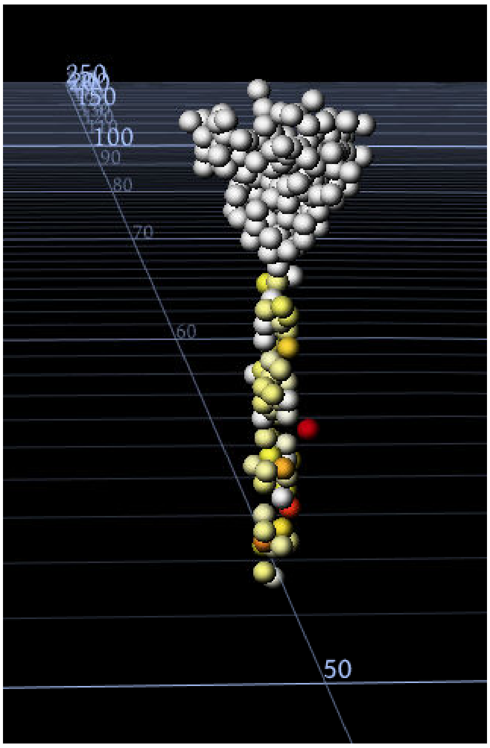


Figure 1: Example image of a point cluster.

1.1 Guidelines

1. An example of traffic sign point cloud is attached as csv file.
2. You should provide clean and clear python code.
3. You can use different packages for algorithmic procedures.
4. You cannot use predefined high-level detectors.
5. The decision is single-frame, meaning, you cannot calibrate or learn based on data.

2 Data analysis

2.1 Provided image.

By looking at the provided image (Figure 1), it is clearly seen that the candidate's assumptions in the interview were wrong. Resolution and SNR simply will not allow us to accurately estimate shapes of the pole and sign, yet the image (Figure 1) justifies the provided hint to look at the reflectively (R) of points - the sign visually differs from the pole.

2.2 Point cloud visualization.

The simplest and naive method is to look at the point cluster as it is. Figure 2 is a visualization of points as 3D scene and 2D projections. Point color corresponds to reflectivity, as R value corresponds to brightness. It is easily observed that the top part has much higher reflectivity (again, as hinted). Furthermore, according to the YZ projection sensor direction is aligned with the X axis.

At the first glance the point cloud is shapeless, but it is an illusion because plots are not scaled properly. Each axis range has to be equal to the others, i.e. 3D scene values range has to be equal for each dimension. In the scaled plots (Figure 3) both, the pole and the sign, are clearly visible. And again,

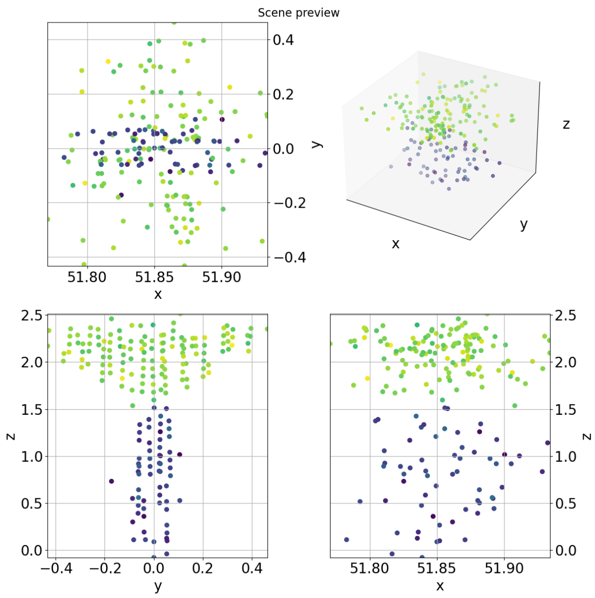


Figure 2: Points cloud visualization as 3D scene and planar projections. Plot are not scaled.

the top part (the plate) is easily differentiated from the pole by it's brightness - R values.

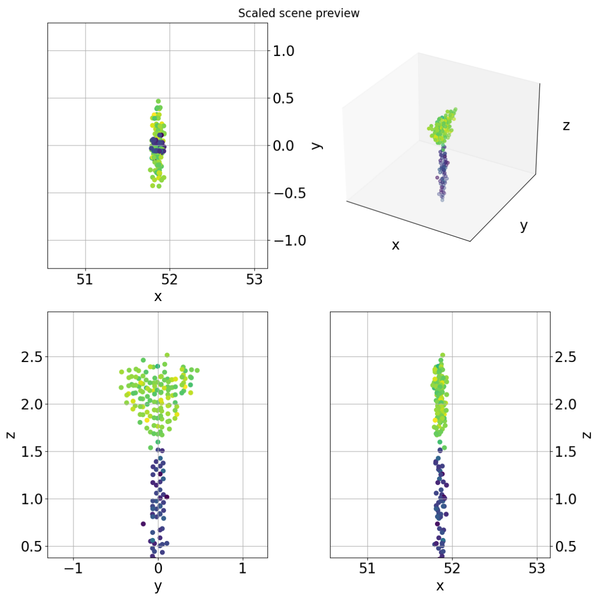


Figure 3: Points cloud visualization as 3D scene and planar projections. Plots are scaled relative to maximum value range.

2.3 Point cloud histogram.

The last procedure is an histogram analysis. Two peaks are expected to be found for R values, and maybe some valuable information about the geometry of the sign. In Figure 4 histograms for each value are shown. As expected, R results in two separate peaks, $\{x, y\}$ are grouped around sign location and z values resemble uniform distribution (or low density area) and Gaussian peak (dense area). Naturally to assume that the uniform section is a pole and the peak is a sign, yet the plate is not always found on top of the pole.

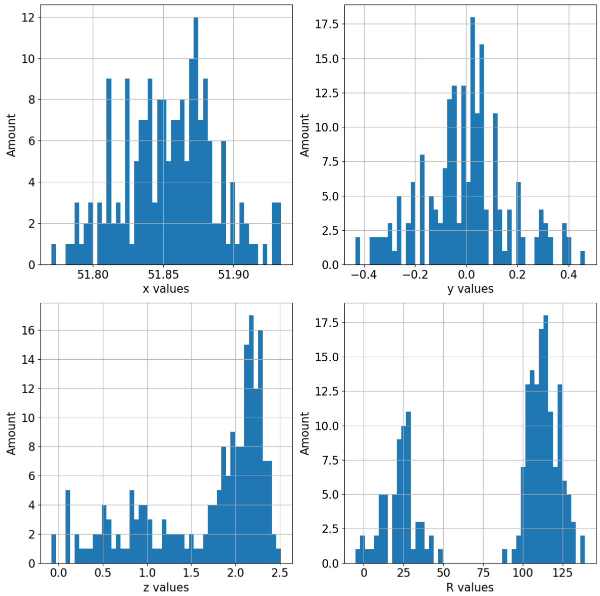


Figure 4: Histograms of point coordinates and reflectivity of point cluster.

3 Problem definition

As in most tasks, the main problem has to be separated into a number of sub problems and solved step by step. Yet, prior to problem definition there are assumptions to be made, as there is no clear definition of a road sign as a point cloud. Moreover, for the real case scenarios one must also consider sensor limitations (e.g., SNR of LiDAR), definition of the object (e.g., standards for road sign geometrical form) and how extreme case examples might look like. All of the above rely on experience with the data, as for every assumption there is always a counter example from the real world (e.g., deformed and old sign's plate is not a highly reflective and may be not a plane), and only experienced person may know which of counter examples may be neglected.

3.1 Assumptions

Following above, there are only two assumptions are considered:

1. Road sign has a highly reflective plate, and non to little reflective pole.
2. Plate is formed by a points that aligns with some plane in 3D space.
3. Plate, as 2D object, has some simple geometrical shape as provided in assignment introduction.

3.2 A set of problems to solve

Consequently, there is a set of sub problems to be solved:

1. Check if the given cluster C has bi-modal density of R.
2. If so, estimate a reflectively threshold R_{th} for plate and pole separation.

3. Check if the plate point cluster C_{plate} aligns with some plane in 3D space.
4. If so, estimate a plane coefficients ($\pi : Ax + By + Cz + D = 0$).
5. Perform a shape test to a C_{plate} projected to estimated plane.

4 Method

Overview of the procedure is described as a block diagram at Figure 5, which is similar to a set of problems defined earlier. For a given cluster, the reflectivity density function $\hat{p}(R)$ is estimated, which passes to bi-modality test. If $\hat{p}(R)$ is bi-modal, the algorithm tries to estimate the common plane π for plate points. If coefficients are found, points overcome a set of transformations and are mapped to pixel space where shape of the plate is estimated. Each step is briefly described in this section.

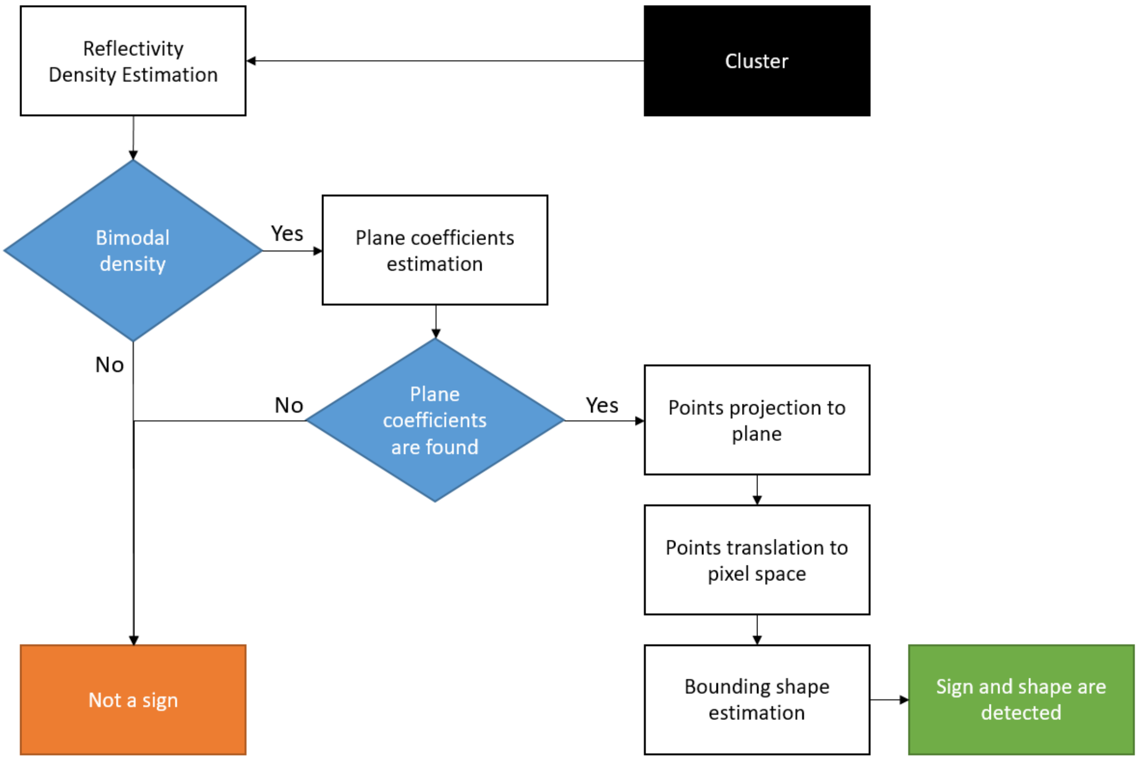


Figure 5: Block diagram of the detection process.

4.1 Reflectively density estimation

Reflectively density function $\hat{p}(R)$ is estimated by kernel density estimation (KDE) algorithm with Gaussian kernel. As a concept, KDE is a sliding window that runs over a set of points and returns high values if points are grouped, or lower value if they are separated. By choosing the window width one may control sensitivity of the algorithm to groups. For example, KDE with a small window width will return higher values only for tightly grouped points. In our case, as was observed in Figure 4, R form two clusters - KDE expected to output $\hat{p}(R)$ with two peaks.

4.2 Bi-modality test and plate-pole separation

Given the estimated density function $\hat{p}(R)$ it is tested for a number of modes (local maximums). If there are two modes that are separated at least with some predefined distance (function may be noisy) - the function is bi-modal. The threshold R_{th} for plate-pole separation is found by the detection of local minimum between two found maximas. Separation is straightforward: points with R that are smaller than R_{th} belong to pole cluster C_{pole}, while all the other points to plate cluster C_{plate}.

4.3 Plane coefficients estimation

Even that plate is expected to have plane properties, in practice due to noise in coordinates (x, y, z) of points in C_{plate} they are not aligned perfectly with plate plane. Therefore, the coefficients have to be estimated and calculated. The naive approach is to use least squares estimation (LSE) method, but it is sensitive to outliers, thus will find an optimal plane for all the points. Most likely it will converge for every possible set, but with wrong orientation.

The practical method is to implement LSE with the RANSAC algorithm. For every iteration of RANSAC some set of points is selected, LSE estimates optimal plane coefficients $\{A, B, C, D\}$ in terms of LS error for the selected set of points. Afterward LS error is calculated for whole C_{plate} cluster and compared to previous best result. If lower error is achieved new coefficients are selected. The algorithm will run until at least one of the conditions is satisfied:

- Plane satisfies (in terms of LS error) pre-defined minimum amount of points.
- The iterations are over.

Later is a sign that the algorithm did not converge, therefore $\{A, B, C, D\}$ are not found.

4.4 Point transformation

If plane orientation is successfully estimated the last step is plate shape detection. Yet, points of C_{plate} are in 3D space and shape is 2D. Moreover, points have to be translated to pixel space as well, simply to apply known computer vision algorithms. As a result, points have to overcome set of translations:

1. 3D to plate space ($\mathbb{R}^3 \to \mathbb{R}^3$)
2. plate space to 2D ($\mathbb{R}^3 \to \mathbb{R}^2$)
3. 2D to pixel space (Discretization)

The first step is performed by projection of points to plane π, which requires one point on the plane p_π and normal $\bar{n}_\pi$. Former is an inlier with smallest LS error and normal is $\bar{n}_\pi = (A, B, C)$, thus the problem is reduced to linear algebra calculation.

Second is solved by rotating projected points to the XY plane. In order to perform this operation, the rotation matrix has to be found by solving the equation $\bar{n}_\pi R = \bar{n}_{xy}$. Afterward all projected points in C_{plate} are multiplied by R and are mapped to XY plane. There is no perfect alignment ($z \neq 0$, yet no significant error was found in later stages.

The last step, discretization is performed by setting predefined resolution $H \times W$ and scaling points according to their x, y coordinates w, h by Eq. 1:

$$
\begin{aligned}
h &= H \cdot \tfrac{y - \hat{y}_{min}}{\hat{y}_{max} - \hat{y}_{min}} \\
w &= W \cdot \tfrac{x - \hat{x}_{min}}{\hat{x}_{max} - \hat{x}_{min}}
\end{aligned}
\tag{1}
$$

To margin from the image boundaries to pixels, minimal and maximum values are scaled by some predefined percentage $m_{\%}$.

4.5 Shape detection

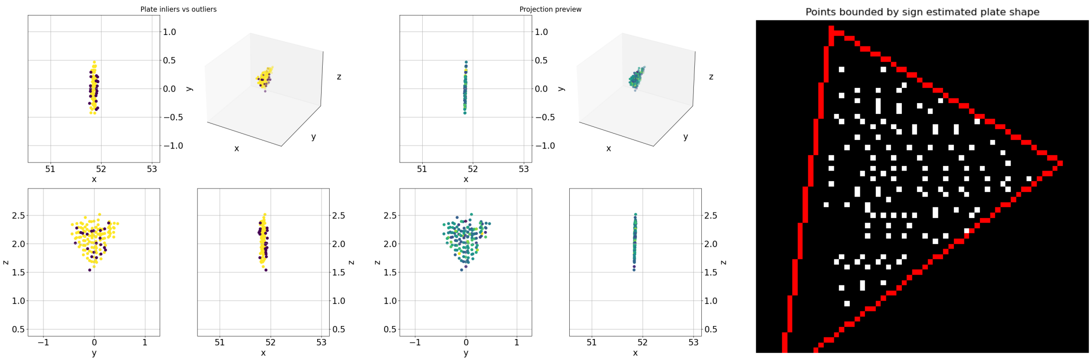
Given a set of points in pixel space it is possible to find the convex hull of a set and later minimum bounding simple shape (triangle, rectangle or circle). Therefore, each minimal area shape is found minimal and area is calculated. The shape with the smallest area is a shape of the sign, as point density is higher (i.e. less empty space is contained in the area).

5 Results

The algorithm was evaluated on provided data and on simulated data.

5.1 Provided cluster

Algorithm successfully detected that cluster is a sign with a triangle shape plate, as it seemed from a preview. The results are shown below:

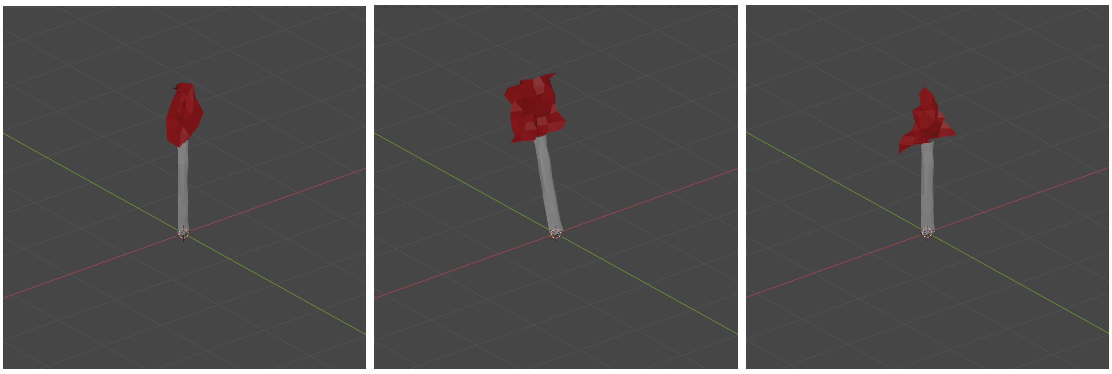


(a) (b) (c)

Figure 6: (a) Inlier and outliers separation by the colors. (b) Projected points to plate plane. (c) Shape detection result.

5.2 Simulated data

Simulated data produced in Blender software and modified by script. There are nine possible scenarios, as there are tree rotation states and three sign plate shapes. Example of scenarios are shown below: Deformation for each produced in the direction of normal, therefore additional noise is added as pre-

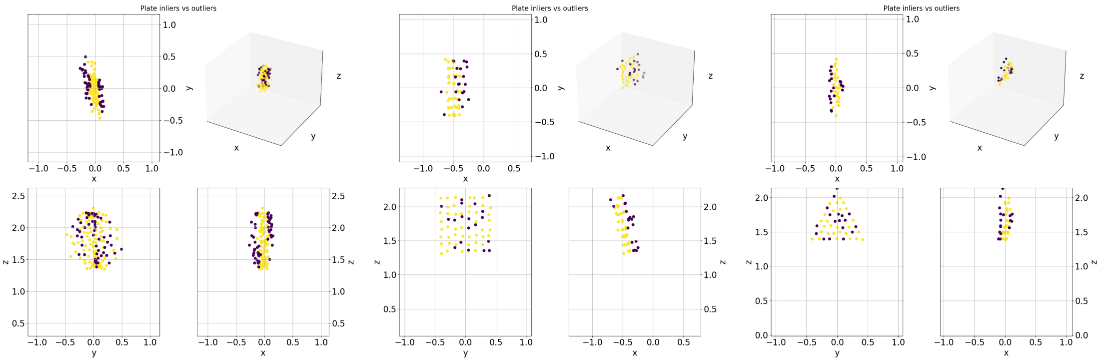


(a) (b) (c)

Figure 7: (a) Circle shaped sign. (b) Rectangle sign. (c) Triangle shaped sign.

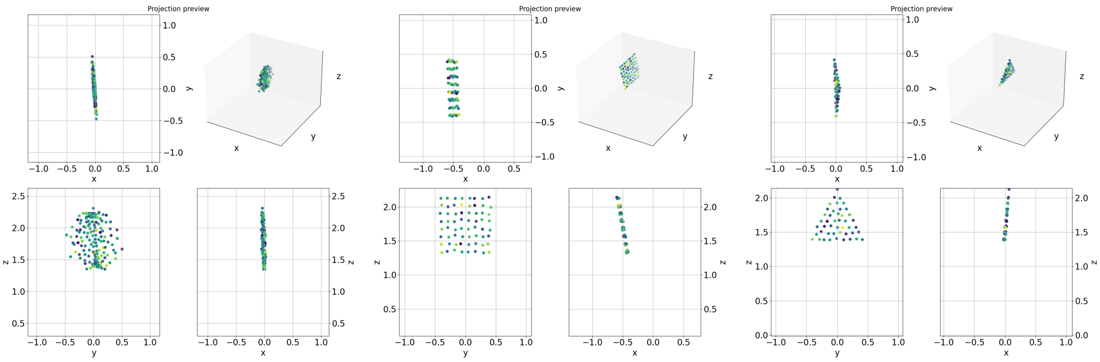
processing. The reflectivity values are sampled from two normal distributions to match bi-modal density that is found in the provided cluster.

The results are provided below for each case in the same order.



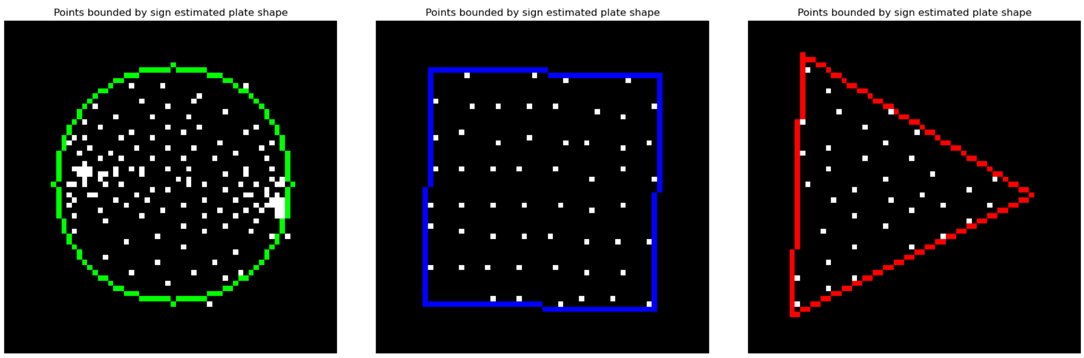
(a) (b) (c)

Figure 8: Inliers vs. outliers after plane estimation. (a) Circle shaped sign. (b) Rectangle sign. (c) Triangle shaped sign.



(a) (b) (c)

Figure 9: Plane projection. (a) Circle shaped sign. (b) Rectangle sign. (c) Triangle shaped sign.



(a) (b) (c)

Figure 10: Shape detection. (a) Circle shaped sign. (b) Rectangle sign. (c) Triangle shaped sign.