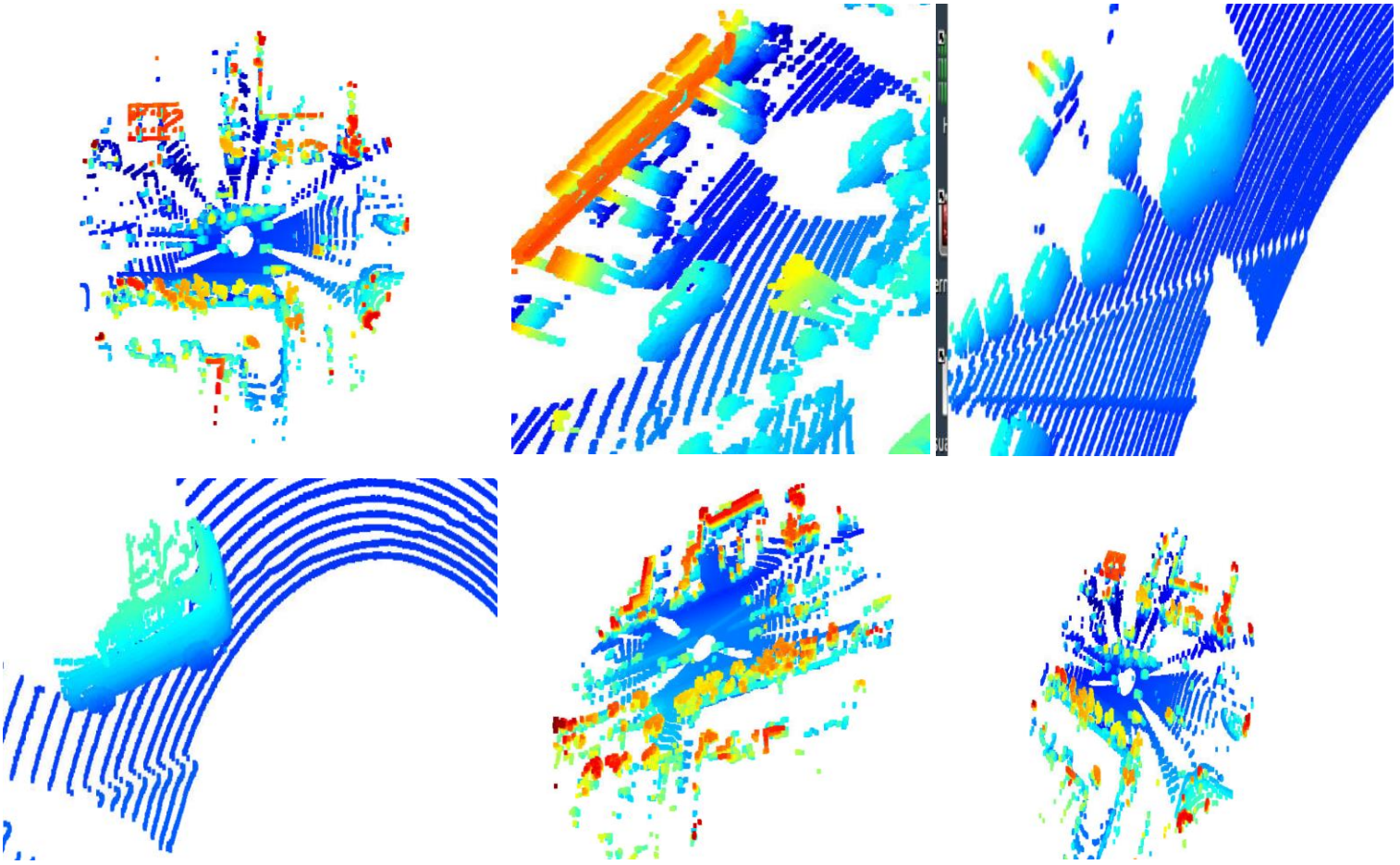
Section 1: Compute Lidar Point-Cloud from Range Image

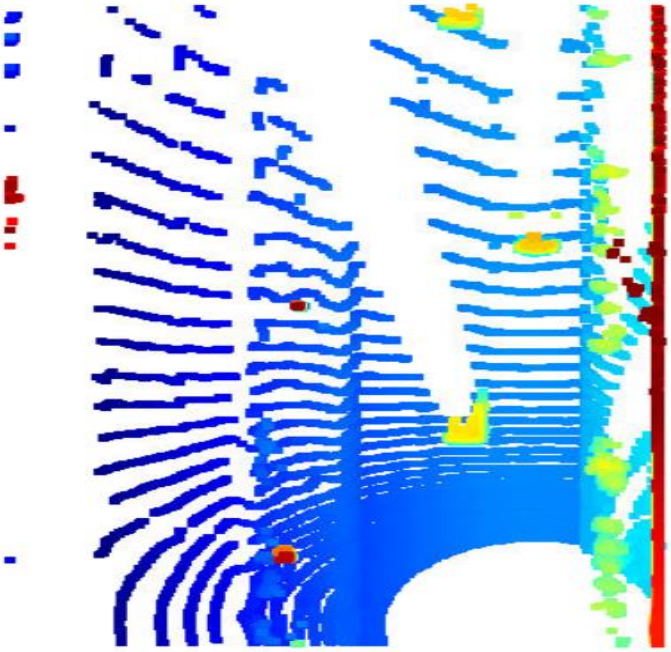
Visualize range image channels (ID_S1_EX1)

Visualize LIDAR point-cloud (ID_S1_EX2)



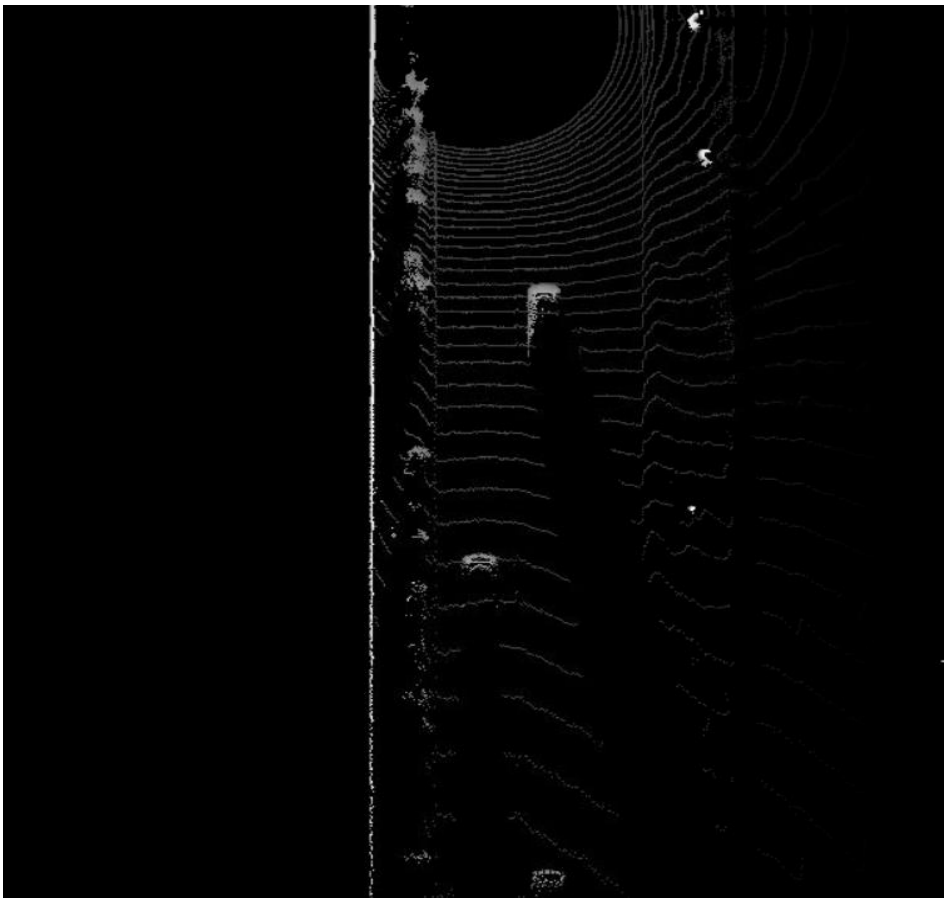
Section 2: Create Birds-Eye View from Lidar PCL

Convert sensor coordinates to BEV-map coordinates (ID_S2_EX1)

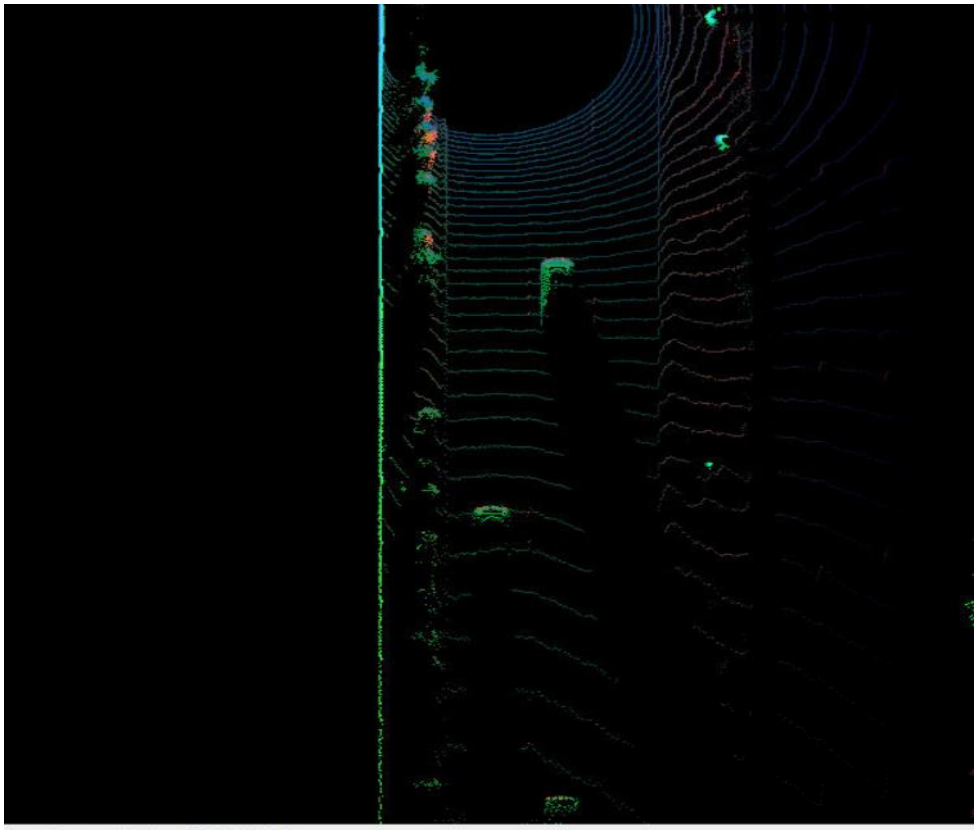


Compute intensity layer of the BEV map (ID_S2_EX2)

Compute height layer of the BEV map (ID_S2_EX3)



BEV-map (Density, Height, Intensity)

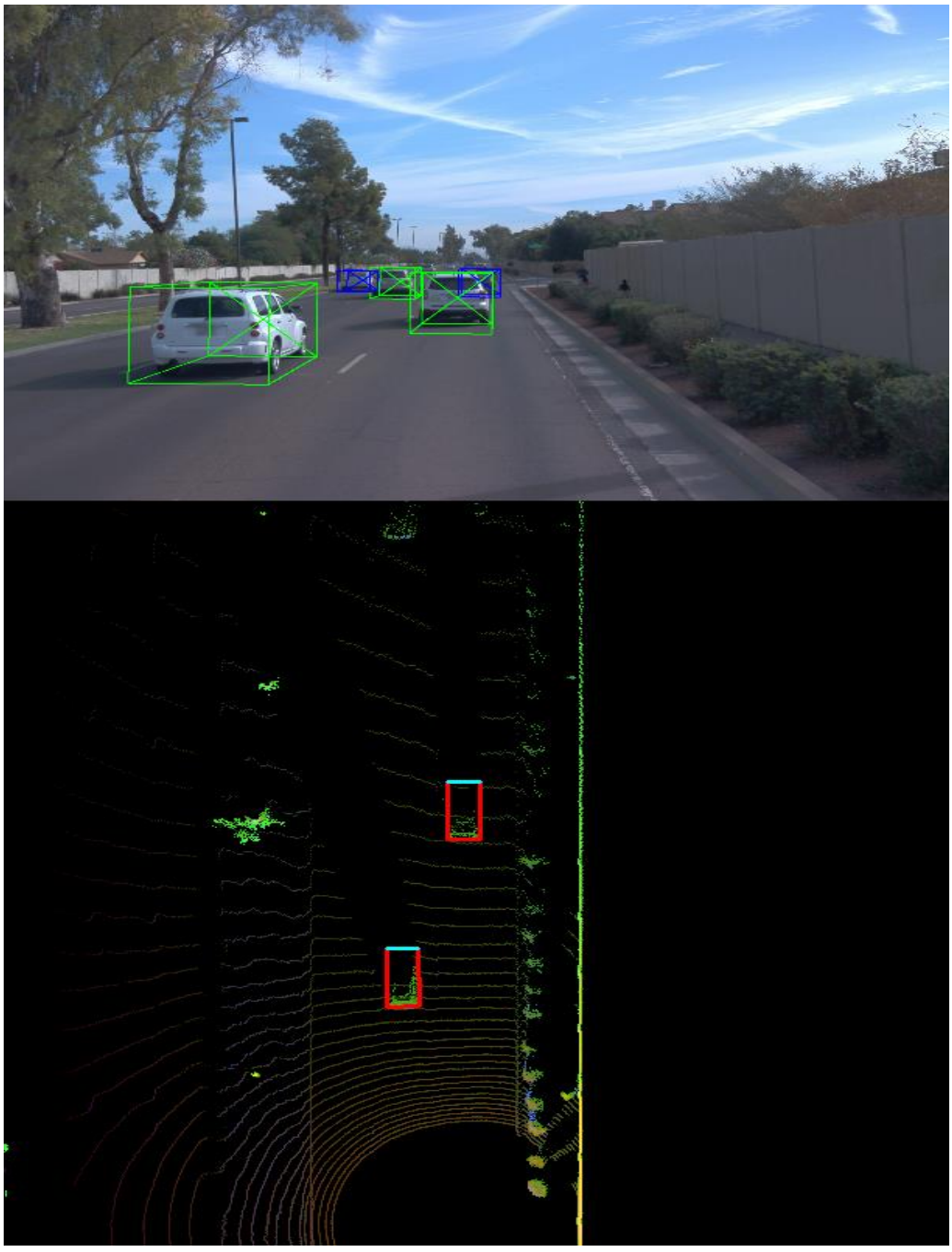


Section 3: Model-based Object Detection in BEV Image

Add a second model from a GitHub repo (ID_S3_EX1)

```
[[9.7324532e-01 3.5126199e+02 2.1873851e+02 1.0573186e+00 1.626
0259e+00
```

Extract 3D bounding boxes from model response (ID_S3_EX2)



Section 4: Performance Evaluation for Object Detection

Compute intersection-over-union between labels and detections (ID_S4_EX1)

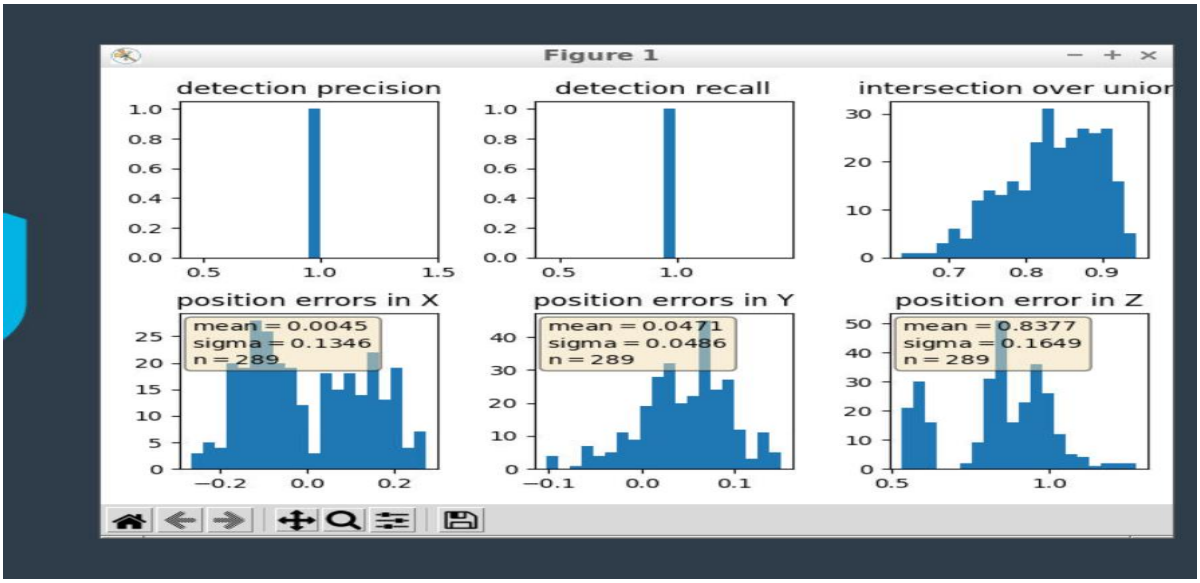
```
ious :
 [0.8234346907261101, 0.8883709520530157, 0.875289820918529]
 center_dev :
 [[0.14017868041992188, -0.019716262817382812, 1.0292643213596193],
 [-0.08345603942871094, 0.06975579261779785, 0.8291298942401681], [
0.08367729187011719, 0.025097429752349854, 0.8929607095304846]]
```

Compute false-negatives and false-positives (ID_S4_EX2)

```
TP :  6
FN :  0
FP :  0
precision = 1.0, recall = 1.0
```

Compute precision an d recall (ID_S4_EX3)

```
TP :  289
FN :  17
FP :  15
precision = 0.9506578947368421, recall = 0.9444444444444444
```



Confirmation Test

```
TP :  306
FN :  0
FP :  0
precision = 1.0, recall = 1.0
```