Lidar And Radar Systems

Assignment-3

Master Mechatronics

submitted by:

Nishanth Nandakumar

Mtr.No:32971

February 11, 2022

Prof. Dr.rer.nat. Stefan Elser

1 Assignment 3

1.1 Introduction

The main objective of this project is to use the 3D Bounding Boxes ground truth provided to predict the position of the car. This is achieved by using the ground truth values for two frames, finding the velocity of the car between the two frames, and using this to predict the bounding box in the third frame as shown in figure1. The predicted bounding box is then used along with the ground truth bounding box for the same frame to calculate the intersection over Union(IOU).

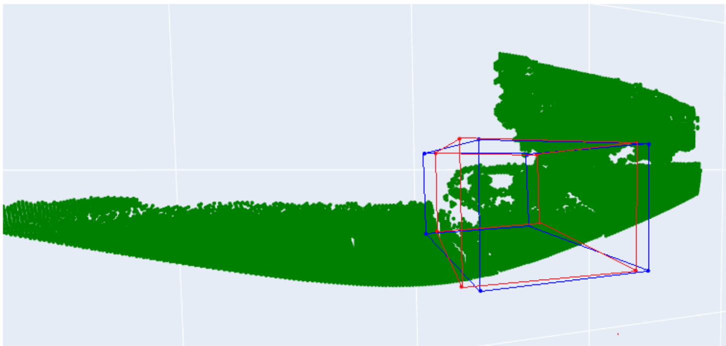

Figure 1: Blickfeld point cloud with predicted(blue) and ground truth(red) bounding boxes for frame 19

The dataset provided consists of two recorded data from Blickfeld and Velodyne sensors with the ground truth bounding boxes. Here we will be using Blickfeld data as the ground truth remains the same between the two sensor data and it doesn't effect the final result.

1.2 Algorithm and Calculations

The ground truth bounding box consists of six values:

- x - X coordinate of center of the car
- y - Y coordinate of center of the car
- z - Z coordinate of center of the car
- w - Width of the car
- l - Length of the car
- h - Height of the car
- yaw - Rotation around the z-axis

When the car is in motion, the change in position is depicted by the x,y,z, and yaw readings. As the dimensions of the car remain the same there will be no change in w,l, and h readings. To predict the bounding box for the car we need to find the change in position of the car between the frames. The time interval between the frames is considered to be constant, we can calculate the change in distance using d_0 and d_1 values from two ground truth bounding boxes as in equations1 and 2. We calculate the change in position as shown in equation3. The change in position calculated is then added to the ground truth of the second frame to predict the car's position in the third frame as shown in equation5.

$$d_0 = (x_0, y_0, z_0, w_0, l_0, h_0, yaw_0) \tag{1}$$

$$d_1 = (x_1, y_1, z_1, w_1, l_1, h_1, yaw_1) \tag{2}$$

$$d_{err} = d_1 - d_0 \tag{3}$$

$$d_{err} = (x_{err}, y_{err}, z_{err}, 0, 0, 0, yaw_{err}) \tag{4}$$

$$d_{pred} = (x_1 + x_{err}, y_1 + y_{err}, z_1 + z_{err}, w_1, l_1, h_1, yaw_1 + yaw_{err}) \tag{5}$$

The bb_pred function takes the ground truth of the two bounding boxes and returns the predicted bounding box values for the third frame.

```
def bb_pred(boundingbox1, boundingbox2):
    deltad = boundingbox2 - boundingbox1
    bbox_pred = boundingbox2 + deltad
    return bbox_pred
```

Now that we have the predicted bounding box, we use the make_boundingbox function from task 1 to get the eight corners of the predicted bounding box.Once we get the predicted bounding box from the above algorithm, we need to compare it with the actual ground truth and calculate an error between the two. Here, we use the Intersection over Union(IOU) metric which is used to measure the accuracy of an object detector as shown in figure 2. This depicts IOU calculation for a 2D bounding box, since we are predicting a 3D bounding box we have to consider volume instead of the area as shown in equation6.

$$IOU = \frac{VolumeofIntersection}{VolumeofUnion} \tag{6}$$

To calculate the IOU we need to get the minimum and the maximum of the x,y,z values from the eight corners of the predicted bounding box and the ground truth bounding box using the function minmaxbb provided. This function returns an array with the minimum and maximum values.

```
def minmaxbb(bounding_box):
  #Here we get max and min for bounding box
  #Initialize the variables with some maximum and
  #minimum values
  x_max = -10000; x_min = 10000;
  y_max = -10000; y_min = 10000;
  z_max = -10000; z_min = 10000
  for corner in bounding_box:
    if corner[0] >= x_max:
      x_max = corner[0]
    if corner[0] <= x_min:
      x_min = corner[0]
    if corner[1] >= y_max:
      y_max = corner[1]
    if corner[1] <= y_min:
      y_min = corner[1]
    if corner[2] >= z_max:
      z_max = corner[2]
    if corner[2] <= z_min:
      z_min = corner[2]
 val = [x_max,x_min,y_max,y_min,z_max,z_min]
  return val
```

Now, we use the bb_iou function provided to calculate the IOU. We provide the output of the minmaxbb function for the predicted bounding box and the ground truth bounding box. To calculate the volume of intersection we take the minimum of the two maximum values and maximum of the two minimum values for x,y,z as shown. To calculate the volume of union we add the volumes of the predicted bounding box and ground truth bounding box and negate the volume of intersection to account for overlapping. IOU is calculated using equation6.

```
def bb_iou(boxA, boxB):
  # determine the (x, y)-coordinates of the
  #intersection cuboid
  xA = min(boxA[0], boxB[0])
  xB = max(boxA[1], boxB[1])
  yA = min(boxA[2], boxB[2])
  yB = max(boxA[3], boxB[3])
  zA = min(boxA[4], boxB[4])
  zB = max(boxA[5], boxB[5])
  # compute the volume of intersection cuboid
  interVolume = (xA - xB) * (yA - yB) * (zA - zB)
  # compute the Volume of both the prediction and
```

```
#ground-truth cuboids
boxAVolume = (boxA[0] - boxA[1] ) * (boxA[2] -
             boxA[3] ) * (boxA[4] - boxA[5] )
boxBVolume = (boxB[0] - boxB[1] ) * (boxB[2] -
             boxB[3] ) * (boxB[4] - boxB[5] )

iou = interVolume / float(boxAVolume+boxBVolume
      - interVolume)
return iou
```

Table1 provides the IOU values for every frame in the Blickfeld Record1 dataset. After finding the IOU we need to calculate the precision, recall, and F1 score for the algorithm. Precision provides us information on the ability of our algorithm to provide only relevant instances. Recall provides information on finding the relevant instances in the dataset. We take the harmonic mean between precision and recall to calculate the F1 score. This provides us with the best balance between the two for the algorithm [Koe18]. The recall, precision, and F1 scores are calculated using the equations7 8 9 respectively.

Record1							
Frame	IOU	Frame	IOU	Frame	IOU	Frame	IOU
2	0.779398	12	0.821221	22	0.918283	32	0.640934
3	0.738349	13	0.684312	23	0.899573	33	0.796748
4	0.740382	14	0.590068	24	0.755651	34	0.756825
5	0.711057	15	0.844045	25	0.916351	35	0.749610
6	0.533504	16	0.560903	26	0.847218	36	0.836819
7	0.708702	17	0.679676	27	0.637178	37	0.701248
8	0.451933	18	0.752304	28	0.648468	38	0.711288
9	0.509574	19	0.794411	29	0.749983	39	0.755419
10	0.840206	20	0.712026	30	0.748517	40	0.753663
11	0.821145	21	0.820260	31	0.740715	41	0.621377

Table 1: IOU readings for each frame in Record1 dataset

$$recall = \frac{truepositives}{truepositives + falsenegatives} \tag{7}$$

$$precision = \frac{truepositives}{truepositives + falsepositives} \tag{8}$$

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \tag{9}$$

To calculate recall, precision we need true positives, false positives, and false negatives. We consider a threshold from 0 to 1 in steps of 0.1. True positive is the total number of frames in Record1 with IOU greater than the threshold value in Table1. False-positive is the total number of frames in Record1 with IOU less than the threshold. False negatives are the points that are labeled as negatives by the algorithm but are actually positive, in our case this is 0. Table2 provides us with the readings for different threshold values.

Record1						
Threshold	TP	FP	FN	recall	precision	F1
0	40	0	0	1	1	1
0.1	40	0	0	1	1	1
0.2	40	0	0	1	1	1
0.3	40	0	0	1	1	1
0.4	40	0	0	1	1	1
0.5	39	1	0	1	0.975	0.987341772
0.6	35	5	0	1	0.875	0.933333333
0.7	29	11	0	1	0.725	0.84057971
0.8	10	30	0	1	0.25	0.4
0.9	2	38	0	1	0.05	0.095238095
1	0	40	0	#DIV/0!	0	#DIV/0!

Table 2: Recall, Precision and F1 score for different threshold

The above procedure was repeated for the Record2 dataset but due to the availability of only five ground truth bounding boxes we could get IOU for three frames 8,9, and 10 as 0.69409, 0.55329, and 0.69777 respectively. The recall, precision, and F1 score for this with IOU greater than 0.5 are 100% each.

1.3 Conclusion

The Record1 and Record2 data for Blickfeld sensor is used to predict the bounding boxes based on the ground truth bounding boxes. The algorithm to predict the bounding boxes is explained in the above section. We use Intersection over Union to obtain a score for the predicted bounding box when compared to the ground truth. The recall, precision, and the F1 score for the algorithm are calculated for different threshold values. If we consider 0.5 to be an average threshold for the algorithm, it has recall, precision, and F1 scores of 100%, 97.5%, and 98.7% respectively on the Record1 dataset. The algorithm has a 100% recall, precision and F1 score for the Record2 dataset at a 0.5 threshold.

2 Appendix

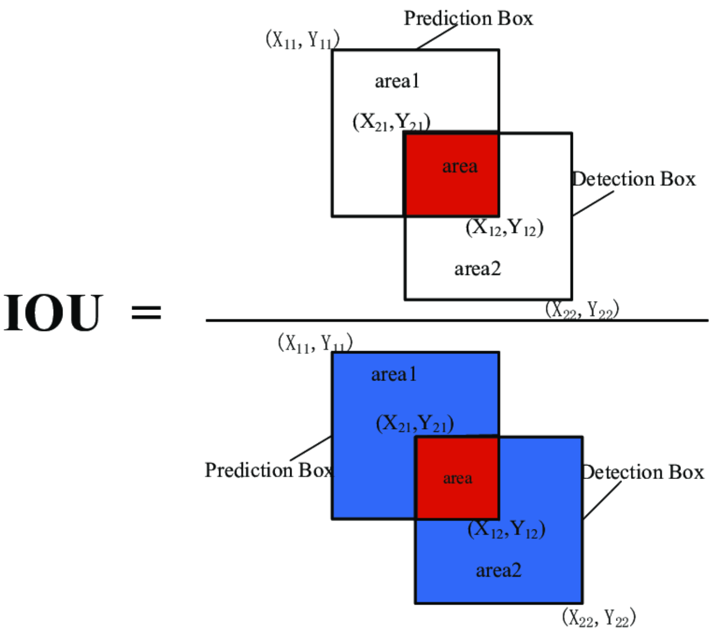

Figure 2: Calculation of IOU Source:[CZZ19]

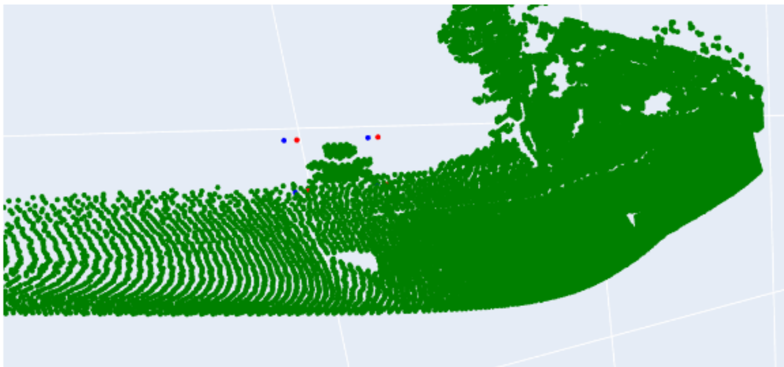

Figure 3: Blickfeld point cloud for Record1 with predicted(blue) and ground truth(red) bounding boxes for frame 04

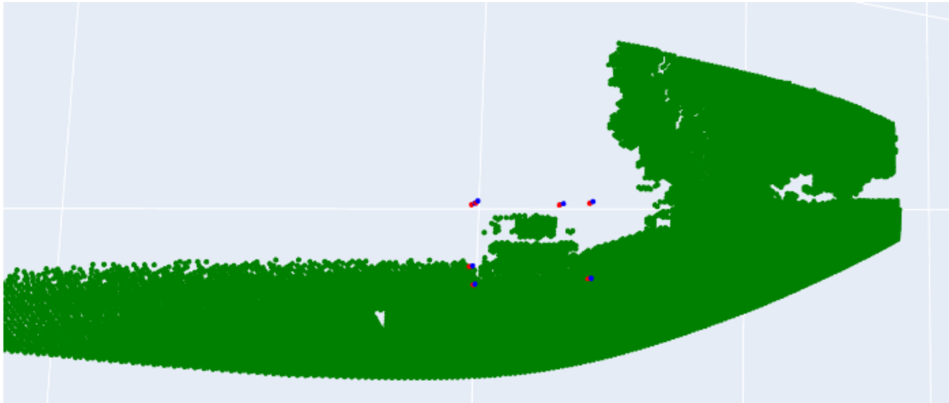

Figure 4: Blickfeld point cloud for Record1 with predicted(blue) and ground truth(red) bounding boxes for frame 10

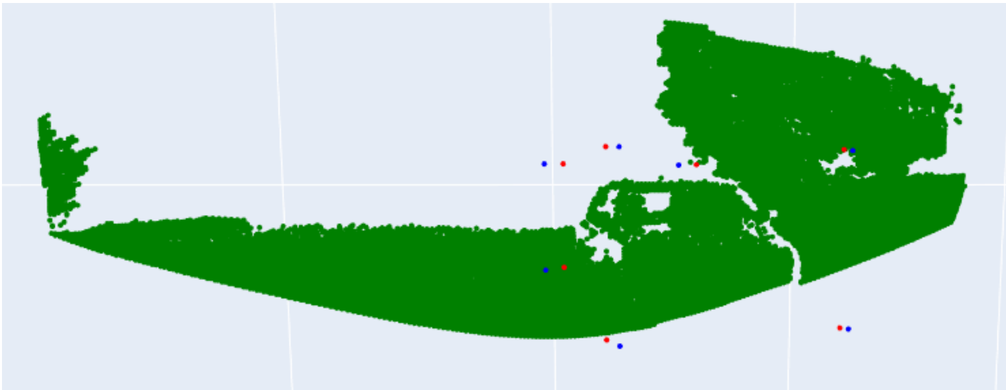

Figure 5: Blickfeld point cloud for Record1 with predicted(blue) and ground truth(red) bounding boxes for frame 21

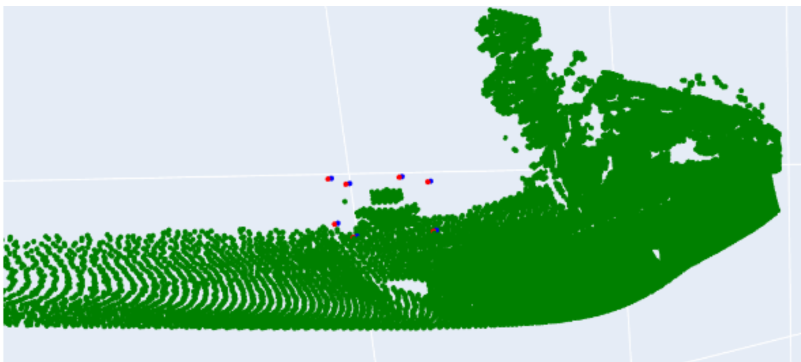

Figure 6: Blickfeld point cloud for Record1 with predicted(blue) and ground truth(red) bounding boxes for frame 36

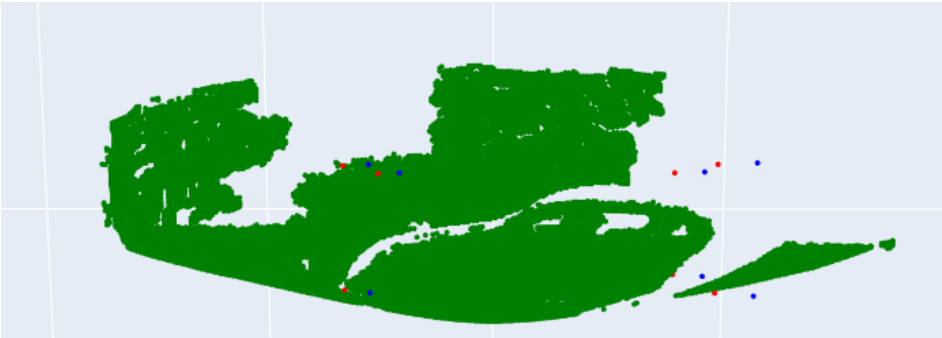

Figure 7: Blickfeld point cloud for Record2 with predicted(blue) and ground truth(red) bounding boxes for frame 08

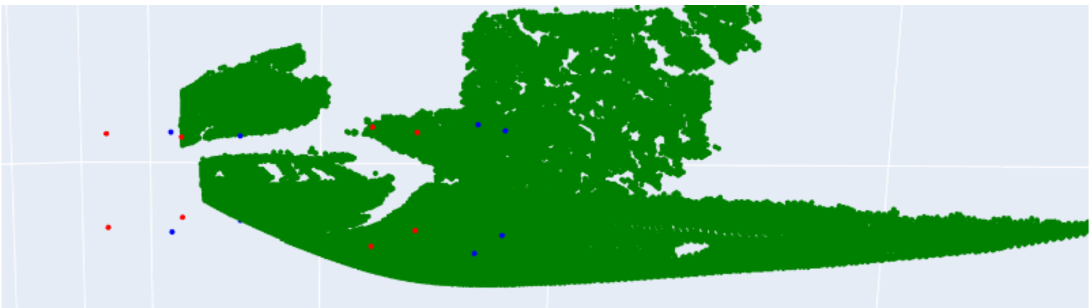

Figure 8: Blickfeld point cloud for Record2 with predicted(blue) and ground truth(red) bounding boxes for frame 09

References

[CZZ19] Cheng, Shuhong ; Zhao, Kaopeng ; Zhang, Dianfan: Abnormal Water Quality Monitoring Based on Visual Sensing of Three-Dimensional Motion Behavior of Fish. In: *Symmetry* 11 (2019), Nr. 9, S. 1179

[Koe18] Koehrsen, Will: *Beyond accuracy: Precision and recall.* https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c. Version: Mar 2018