Individual Lab Report #8

Mar. 2, 2017

Menghan Zhang

TeamA

Amit Agarwal

Harry Golash

Yihao Qian

Zihao Zhang

Individual progress

Last time, I worked on integrating the stereo vision and object detection algorithm so that we can get the depth information of the objects. At that time I added the depth information and bounding boxes on the RGB images. And I said that I wanted to learn more about the PCL visualization method. After that we discussed about the visualization method of the final results. There were three methods we thought about.

1. Using PCL to do the 3D visualization.

   The Point Cloud Library (PCL) is a large scale, open project for point cloud processing. The PCL framework contains numerous state-of-the art algorithms including filtering, feature estimation, surface reconstruction, registration, model fitting and segmentation. These algorithms can be used to stitch 3D point clouds together, extract key points and compute descriptors to recognize objects in the world based on their geometric appearance, and create surfaces from point clouds and visualize them.

   A point cloud is a data structure used to represent a collection of multi-dimensional points and is commonly used to represent three-dimensional data. In a 3D point cloud, the points usually represent the X, Y, and Z geometric coordinates of an underlying sampled surface. When color information is present, the point cloud becomes 4D. The architecture

   Using PCL can show the 3D information of all the points in the environment, it can actual visualize the entire 3D environment. But it need times to learn more about that and it cost lots of time to configure and maintain on the system. After thinking twice about that, we thought for our project, we're only interested in visualizing the specific objects we want to know, we do not care about other points in the environment, so we decided to give up PCL.

2. Visualizing from top view.

   Since the information we cared is more about what will the objects look like in our own vehicle, that means the relative position. In that case, a top view image can directly show the position we cared about and also is not hard to implement.

   After deciding the visualization method, we discussed about the inputs of this model. After combined all the information from two sensors and algorithms we used for detection and tracking, the inputs should contained objects class(from detection), object number(from tracking), the x position and depth information(from stereo vision and radar) and velocity information(from radar and tracking). Since we are implementing in C++, for each frame, inputs should be a 2D vector, each rows refer to one object, and contains all the information I mentioned before.

   I draw a grid to be the background picture and each pixel refered to 10cm in real word. The original point is in the middle of the bottom line where our car is located. Our system supposes to perceive the distance up to 150m and x range up to 100m. For x, the right is the positive orientation and for depth, up is the positive. I used blue rectangle to represent the cars and green circles are pedestrians and added red text information contains the details about that. The Figure 1 shows the results.

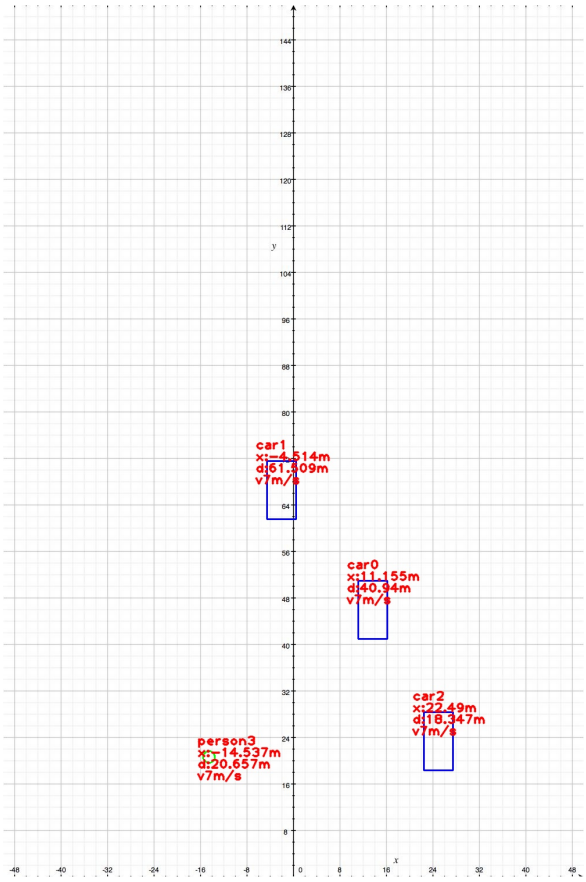

Figure 1. Visualization Results

3. Adding the text information on the RGB image.
   Just like the last PR showed(Figure 2), it is little bit shy to represent the depth information which is really important in our project. So we decided to choose the second method.

Figure 2 Last PR results

Challenges

I used OpenCV to do the visualization, kept the name window and draw new patterns on that when another frame results input. I'm worried about whether it can achieve real-time and can keep working for a really long time. I'll search more about the visualization method in OpenCV and try to improve that next week.

Teamwork

This time, Yihao searched more about the tracking algorithm and improve the stereo vision algorithm. Zihao was working on the GPS, Amit and Harry was keep trying to get meaningful information from Radar. There is indeed one communication problem between me and Yihao, I guess sometimes he was really mad at my language mistakes and thought I didn't treat the work seriously, I'm really sorry about the mistakes and I was trying to improved that, but it takes time. But he was not willing to talk to me, even if I want to discuss about how to break the work. If I don't know what's he's doing, I will have a great chance works on the same things with him and left some work behind. Also, that's not possible for me to integrate the system if he's not willing to give me his code.

Future work

1. Try to intergrade the stereo vision and detection network directly. Last time we used txt file to save the results from both systems and calculated the depth for objects.
2. Do more research tracking and try to modify that into working on multiple objects.