ILR - 07

Perception System Using Stereo Vision and Radar

Team A - Amit Agarwal

Harry Golash, Yihao Qian, Menghan Zhang, Zihao (Theo) Zhang

Sponsored by: Delphi Automotive

February 16, 2017

Table of Contents

1. Individual Progress

Since the previous progress review, I was retasked with working on the Radar and getting the Radar set up through Ethernet. For this I worked with Harry and did a literature review of the related information on the internet.

1.1 Delphi ESR 2.5

Over the past couple of weeks, Harry and I have been working on getting sensible data from the Delphi ESR 2.5 Radar. After multiple challenges to obtain data from the Radar and communication with Delphi, we were instructed by the team at Delphi to use the Ethernet port to get data instead. Previously, we were using the CAN connector to get data from the Delphi ESR 2.5 radar. The data obtained from the Radar through Ethernet is the raw detection level data in contrast to the processed tracking level data obtained through CAN. Previously, a software called PolySync was used by the team to gather data which is not of any utility anymore.

Figure 1. Ethernet to USB Converter

The initial requirement for gathering the data was to find the IP address of the Radar unit provided to us. The test car used was a 2004 Volvo S60 2.5T AWD acquired by Harry. The Delphi ESR 2.5 was mounted in front of the grill of the test car. This was done using multiple zip-ties and a 3D printed mount that is sturdy and keeps the Radar in place through uneven roads. The radar was inspected for how it responds to the forces experienced due to being at the front of the test car. It was determined to be quite robust to such forces. The voltage generator was connected to an AC output in the Newell Simon Hall basement next to the car. DC power was taken from the voltage generator using alligator cables which provided 24 V as required for the Radar. The wire from the Radar had a female USB port, in which a USB to Ethernet converter was connected, as shown in figure 1. A CAT5e Ethernet cable was attached to the converter. An Ethernet to USB

convertor was connected to the other end, which was further connected to a MacBook Pro using a USB-A to USB-C converter. To complete the setup, the voltage generator was switched on. Once the Radar was drawing current from the Voltage Generator, the process of finding the IP address begun. As instructed by Delphi, the first step was to find the IP Address of the connected laptop/desktop. This was done by checking the system preferences of my computer. Then, the next step was to ping all connections to the computer to locate the correct IP address for the Radar. When this was done, it was expected that apart from the router, another IP Address will be seen, which would be of the radar. During our testing, this did not happen.

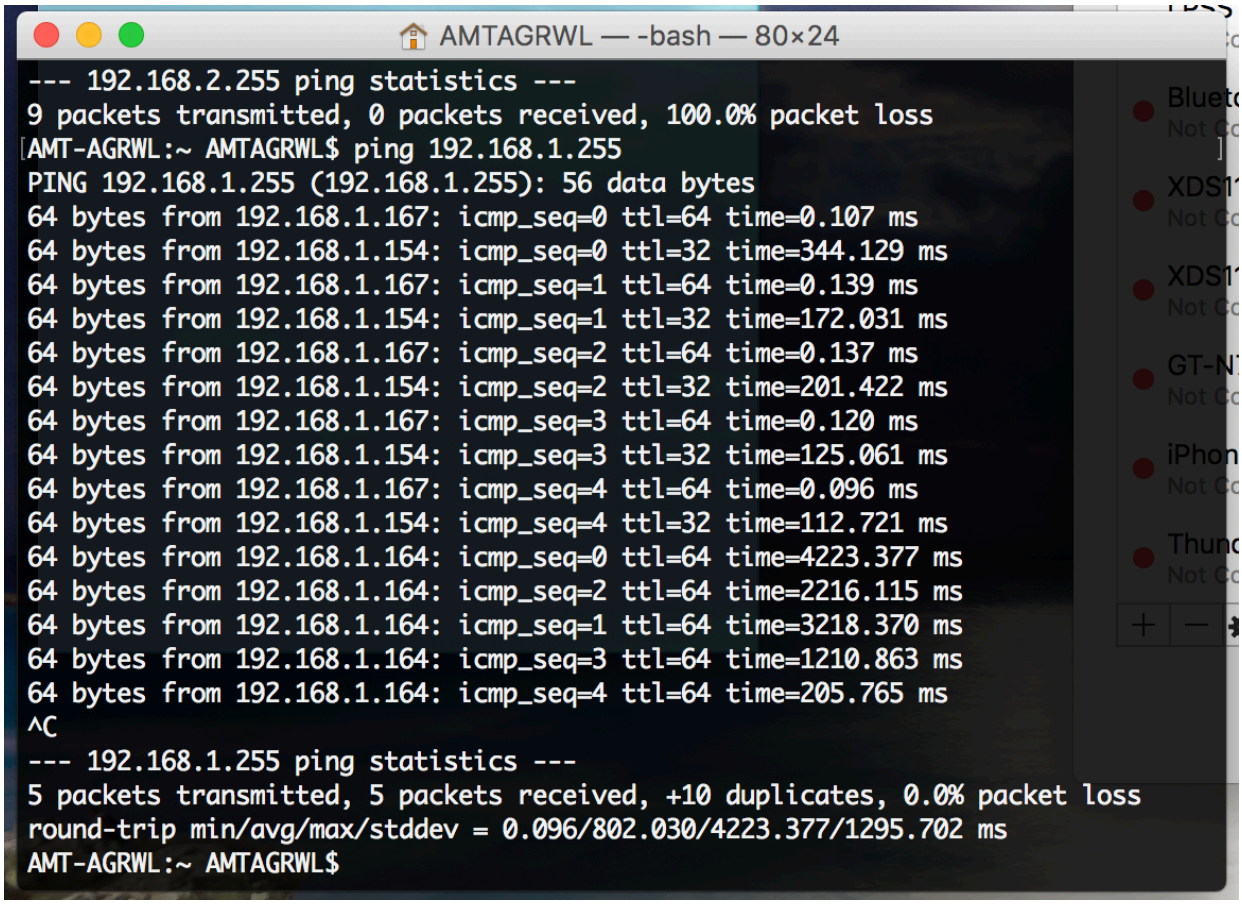

Figure 2. Terminal Window – Successful Ping

Multiple tries were made to solve this issue. The first one was trying an Ubuntu machine. The same procedure of setting up the Radar was performed. Using the terminal application on the system, the IP Address of the machine was procured. Once this was done, all systems on the network were pinged using the ping command in Terminal. Doing this showed the IP Address if the router but, again, not of the Radar. This was repeated multiple times, but no results were achieved. After this, the wiring was checked extensively including all the converters and adapters for any faults or issues. No issues were found with the wiring. Next, the working status of the Radar was verified. This was done by using the setup from last semester over CAN. This involved

changing a lot of cables. Data was obtained using PolySync Studio. The data was still noisy but the visualization was vivid enough to understand that the Radar was working. I notified Delphi about the challenges that we were facing.

Delphi's representative, JongHo Lee, came to campus this Monday, February 13th. He verified our setup and found nothing wrong with it. He used another Radar that he had brought from Delphi in place of the Radar that the team had been using since the start of the project. The same procedure of finding the IP Address was followed on the MacBook. The process yielded the IP Address of the replacement Radar as shown in figure 2. Then the netcat command was used to receive data from the Radar. Port 5555 was used to receive the data in the form of messages through the Ethernet connection. The messages received still need to be parsed to understand the data in them.

2. Challenges

The major challenges faced were due to the faulty Radar provided by the Delphi. In addition, the cable connections were a little hard to understand.

2.1 Faulty Radar

The issue with the provided Radar not working through Ethernet was eventually resolved when Delphi gave a working Radar in place of the faulty one. This challenge wasted a lot of time spent on the Radar by Harry and me. I do not think this was a big issue in technical terms. Although, due to the amount of research done on Ethernet connections, I have learnt some things that may be helpful in the future.

3. Teamwork

The work this week was performed well by all 5 team members. Harry and I worked on the Radar and getting data from the Radar using the Ethernet port. Yihao worked on porting the stereo vision code from Matlab to C++. Menghan did a literature review of the current state of the art algorithm for Object Tracking. In addition, she worked on correlating objects with depth estimates instead of just pixels as received from the depth map. Zihao familiarized himself with Computer Vision and what part he will be working on, in addition to working on object tracking. All members worked well and in a timely manner.

4. Future Plans

In the future, we plan to successfully obtain and parse data from the Radar. On the Computer Vision side, the team will make progress in Object Tracking algorithm and potentially start implmenting the alogrithms. I will be working on a literature review for the Radar documentation to parse the messages obtained.

5. References

[1] "Delphi ESR 2.5". PolySync Support Center. N.p., 2016. Web. 12 Nov. 2016.