Individual Lab Report #10

Apr. 6, 2017

Menghan Zhang

TeamA

Amit Agarwal

Harry Golash

Yihao Qian

Zihao Zhang

Individual progress

This time, our team mainly focused on building nodes in ROS to get the pipeline of whole system. We have finished to build the image capture node, Elas stereo vision node, SSD object detection nodes. The depth measurement node is almost done, but I still need to modify the coordinate transformation according to the message format of stereo vison depth map. Current system can be shown in figure 1 which is saved from rqt_graph. I'm in charge of SSD and depth measurement node.
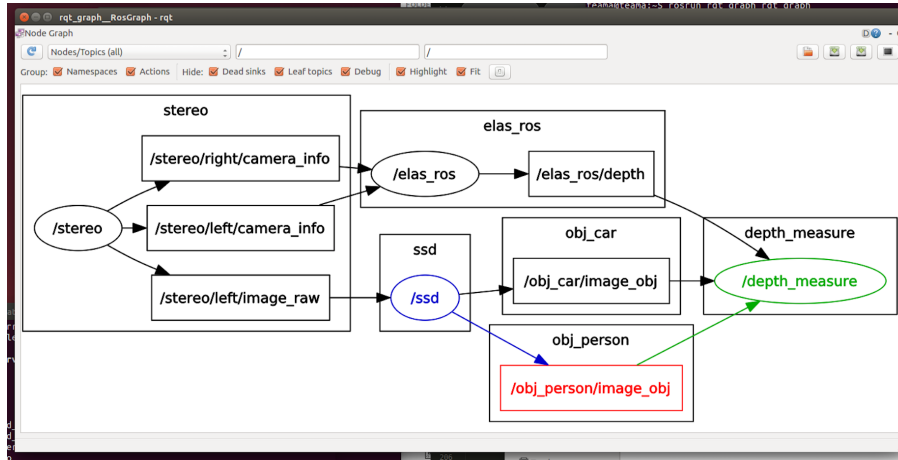

Figure 1. Current Status

1. SSD detection ROS node

    Last time, I modified Caffe and SSD into ROS node but failed to launch the node because of some environment and configuration problems. This week, after searching online I figured out the reason that is because the author has updated the network and added several new layers and gcc 4.8 is not compatible with CUDA 8.0 we need to update gcc to 5.3 or latest version.

    This node subscribes to the image_raw data captured by left camera ("/stereo/left/image_raw") and published two topics, one topic is about the detection results of car ("/obj_car/image_obj") another is about pedestrians("/obj_person/image_obj"). The format of the output topics is

    - Header header (copied from image_raw header so that the timestamp is saved which can be use to synchronize with other nodes and sensors)
    - String type (car or person)
    - Image_rect[] obj (save the detection results format is [x, y, width, height])

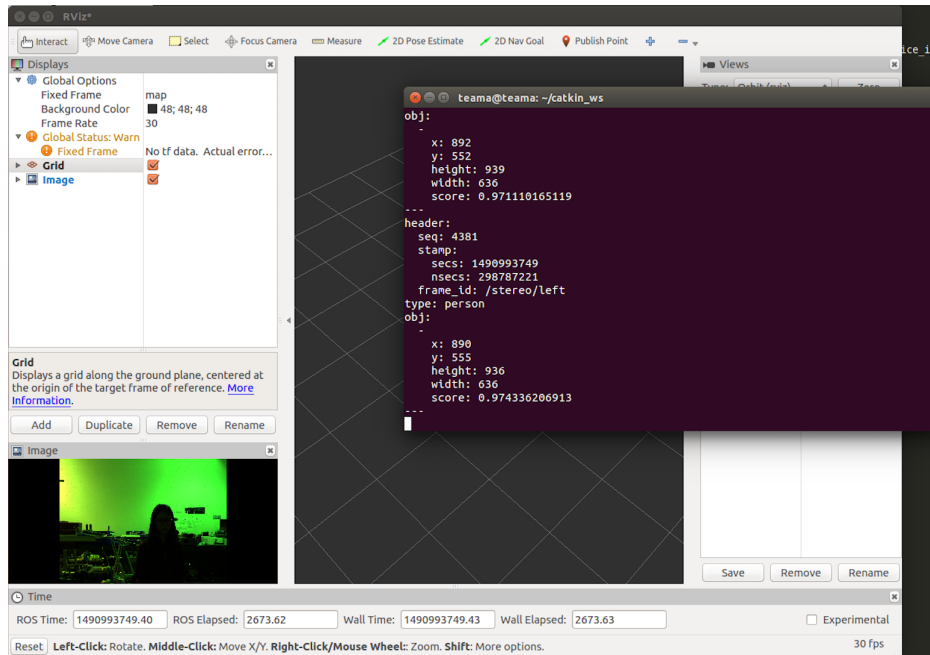   The results can be shown in Figure 2.

Figure 2. SSD nodes results

2. Depth measurement node

As we can see from Figure 1, This node needs to subscribe 3 topics, depth map published by elas_ros node, detection results published by SSD node. Then publish the topic named "/depth" which have 3D position of objects. The main function of this nodes can be divided into 3 part.

- Synchronize between depth map and detection results.

  I use the approximate synchronize policy provide message_filters. Message_filters is a utility library for use with roscpp and rospy. It collects commonly used message "filtering" algorithms into a common space. A message filter is defined as something which a message arrives into and may or may not be spit back out of at a later point in time. The Synchronizer filter synchronizes incoming channels by the timestamps contained in their headers, and outputs them in the form of a single callback that takes the same number of channels. The C++ implementation can synchronize up to 9 channels. The message_filters::sync_policies::ApproximateTime policy uses an adaptive algorithm to match messages based on their timestamp.

- Transfer the detection results into depth map coordinate.

  Because of the different message types and formats between the SSD and Elas, I need a function to unify this two message type.

- Calculate the depth of the objects

  Like the method I mentioned in ILR 07, I filter the points which depth beyond 1~100m (the maximum margin that we can detect and calculate depth) in bounding box then get the median depth value in these points.

Challenges

1. Synchronization

   Because of the different frequency and process speed between SSD and Elas, we need to use approximate policy. But the stereo vision node processes too slow, it need 1s to

generate one depth map, the message queue waits to synchronize need to be above 100 to successfully synchronize two topics. We need to make the stereo node work in GPU to speed it up.

2. Message type
I'm not quite sure what's the format of depth map. When I echo the topic it seems like all the points in the depth map is 0~255 which need to be decoded to the depth information. After I use cv_bridge to transfer that into cv::mat with mono8 format, it seems like some depth information with mm unit. But I'm still not clear whether the x and y in this map same as the standard cv::mat which means columns and rows or not.

Teamwork

This time, Yihao Zihao and I settle down the time to work together every day so that we can help each other to get familiar with ROS as soon as possible and supervise each other to spend more time and efforts on this project. I'm in charge of SSD and depth measurement node Zihao is in charge of image capture and stereo vision nodes, Yihao is working on GPU version of stereo vision node and help us to solve some problems. Amit and Harry visualized data from Radar and build ROS node.

Future work

1. Make depth measurement node work.
2. Build object tracking node.