Yihao Qian
Team A: Aware
Teammates: Amit Agarwal Harry Golash
Menghan Zhang Zihao (Theo) Zhang
ILR10
Aprial.06, 2017

Individual Progress

During the last week, I was in charge of setting up the system, ROS node and researching on how to improve the system.

System set up:

The requirements for object tracking are as follow:

1. Ubuntu 14.04.5
2. CUDA (Parallel computing)
3. OPENCV3.2+ contribution (Object tracking)
4. Ros-indigo

Since I had a lot of experience in setting up the system, I set up Ubuntu14.04.5 and OPENCV3.2+contribution by myself. I also gave some suggestions and helped Menghan and Zihao to set up the system.

Stereo Vision:

The requirements for stereo vision system:

1. Stereo vision system works in real time
2. Stereo vision system achieves more than 80% accuracy

At this stage the system can achieve requirement 2. However, the time that is spent on processing the image is related to several parameters below.

1. Image resolution
2. CPU or GPU implementation

The algorithm is now being implemented in CPU on single thread, and there are 2M pixels in the image. The more pixels the algorithm is going to deal with, the more time the algorithm it will take. Right now, the algorithm would take 1s to compute the disparity for one image. It seems that using 1s per frame is not a big deal, however it may lead to huge problem. The reasons are as follow:

1. Dropping bags: When the input bags speed is larger than output bags speed, some of the bags may drop randomly. However, what we need is to draw the bounding box on the depth image which is accord with the image that is used for object detection. Our object detection system works 80fps. The stereo vision system works 1 fps. That is the reason why the node that is used to receive the object detection message and stereo vision depth merely cannot get the bags that are sent at the same time.
2. Object tracking: The usage for object tracking is to compute the velocity of the object. As you can find our pipeline in figure1. The object tracking node receives the package from stereo vision and object detection nodes. If stereo vision node outputs the bag at the speed of 1s/image. Even object tracking works perfectly during this period. We cannot get the

velocity of the object, since the depth information is the same during the 1s time.

The requirements of this project are very strict. Not only we need to satisfy the accuracy requirement, but also we need to build the system that could run in real-time. In order to satisfy both requirements, I researched on different algorithms to satisfy the requirement. As for stereo vision, I tried SGBM algorithm and ELAS-CPU version. However, at this stage we need to change to another algorithm so that our system could work in real time. The ELAS-GPU version could use GPU to compute the depth information. ELAS-GPU version could work 5 fps. You can find a screen shot in figure 2.

Challenges:

System integration:

As I mentioned before, we need to satisfy very strict requirements. Even each of the sub-system works well individually, it does not mean that the system could run perfectly as a whole. For example, both object detection and stereo vision require a lot of compute resources. Integrating those sub-systems together would reduce the frame rates of both system and may lead to lose of message.

Teamwork:

During the past weeks, I set up the system and changed the stereo vision system from CPU to GPU. Menghan and I integrated our object detection system in ROS system. Zihao and I set up the stereo vision node. Harry and Amit worked on getting data from radar. They could get and parse data from radar.

Name	Teamwork
Yihao Qian	System set up, changed from ELAS-CPU
Zihao Zhang	Stereo vision node set up
Menghan	Object detection node set up
Amit	Set up radar and visualize radar
Harry	Set up radar and visualize radar

In the next week. We will integrate object detection system with object tracking system. We will first build a node for ELAS-GPU. I am planning to work on that. Zihao is going to build GPS node. Menghan is going to combine object detection system with object tracking system.

Figures:

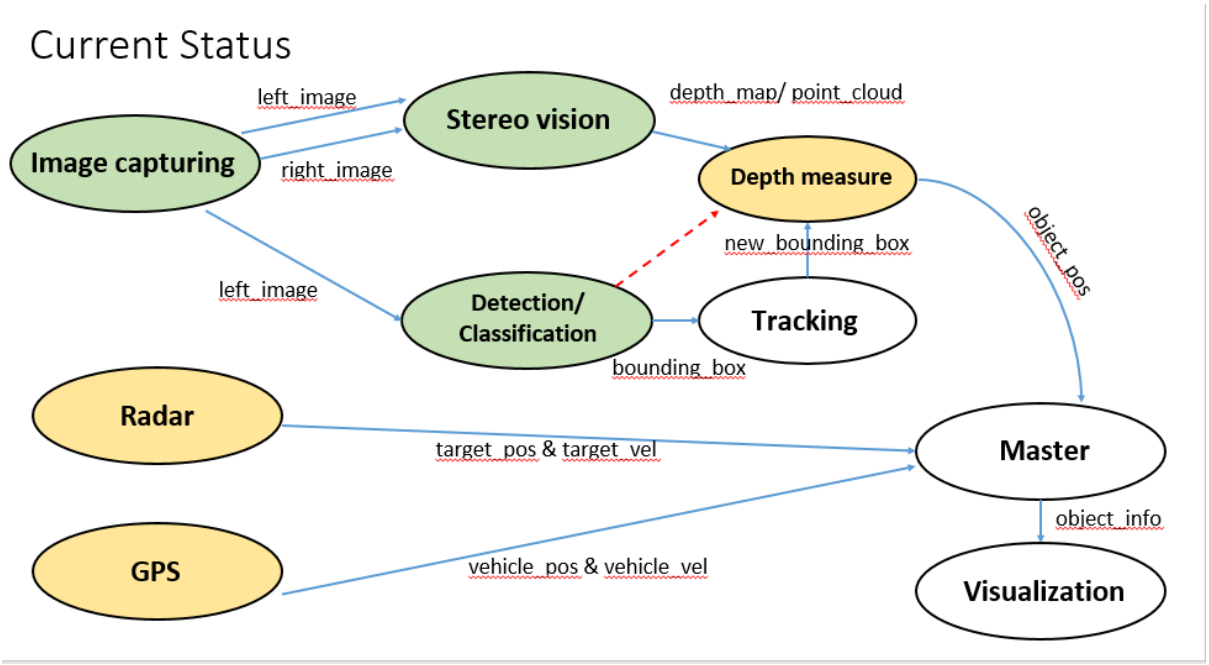

Figure 1. System structure

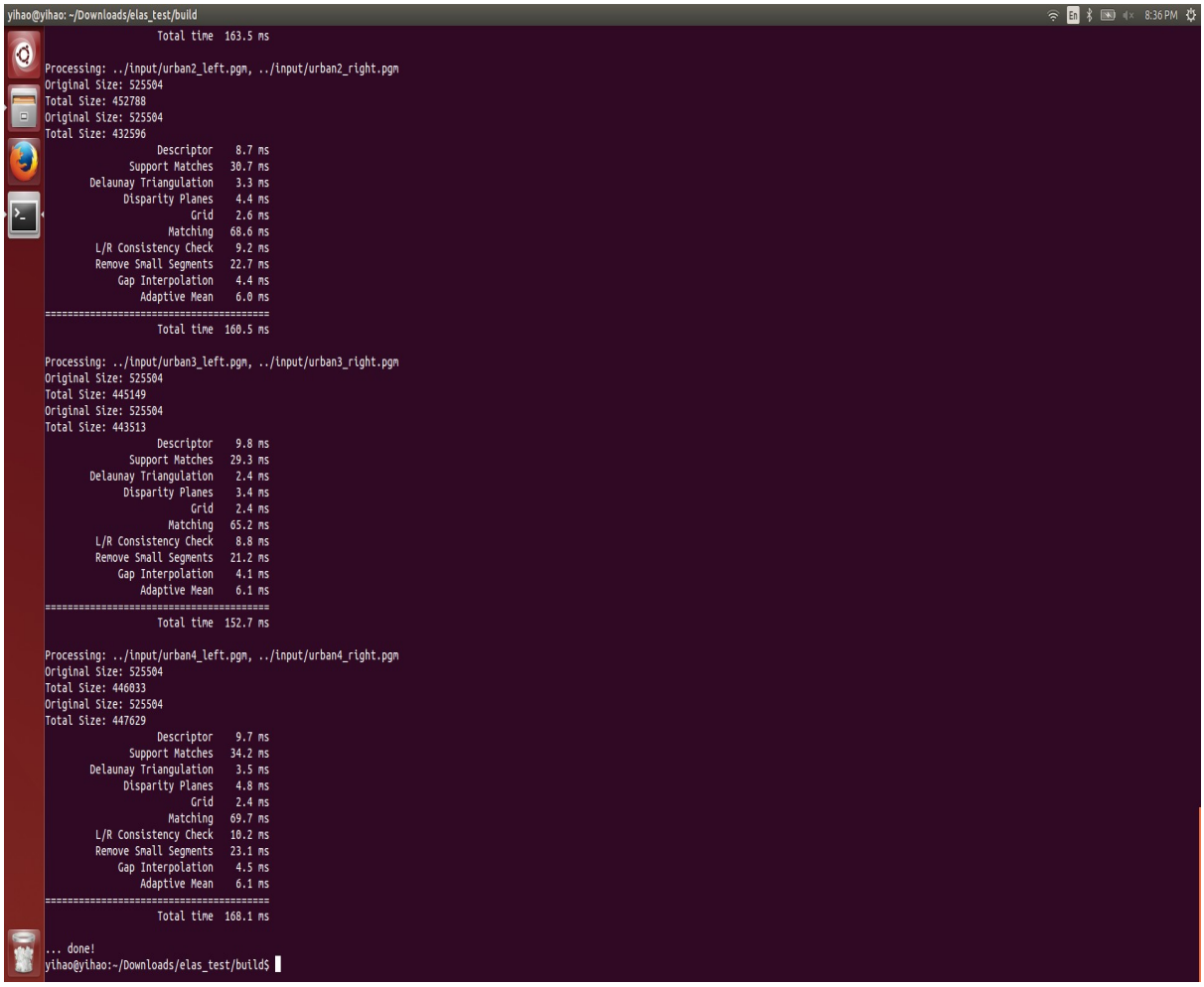

Figure 2. ELAS-GPU version