STEREO CAMERA AND SOLID STATE LIDAR OBJECT DETECTION ON SMALL DELIVERY ROBOT

Jiarun Wei[1,†], Ding Zhao[2,†]

[1]Carnegie Mellon University, Pittsburgh, PA
[2]Carnegie Mellon University, Pittsburgh, PA

ABSTRACT

Stereo camera is popular in object detection due to its dense data. It works by matching the key points of the left and right images, using the geometric constraints to calculate the depth map, and then feed these into a neural network. Due to the geometric constraints, it only works for a certain range of objects and is unstable for some objects. Here we present a demo to stabilize the detection using solid state lidars, which is a very cost efficient solution especially for small and slow robots.

Keywords: Object detection, Computer Vision, Solid-state Lidar, Autonomous Delivery

1. INTRODUCTION

The main goal of this work is to stabilize the detection of objects using a stereo camera and lidar. The

2. METHODS

The stereo camera object detection can be decomposed into several parts: feature extraction, reigion proposal, reigion of intersection alignment, keypoint prediction and finally 3D bounding box estimation, which involves deep neural networks such as ResNet and binocular geometric constraints. In the following experiment, the lidar point clouds are used to verify the detection.

Feature extraction The ResNet is one of the most commonly used feature extraction methods. Its architecture is shown in figure 1.

$$
\begin{aligned}
v_t &= \left(y - \frac{h}{2}\right) / \left(z - \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta\right) \\
u_l &= \left(x - \frac{w}{2}\cos\theta - \frac{l}{2}\sin\theta\right) / \left(z + \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta\right) \\
u_p &= \left(x + \frac{w}{2}\cos\theta - \frac{l}{2}\sin\theta\right) / \left(z - \frac{w}{2}\sin\theta - \frac{l}{2}\cos\theta\right) \\
&\cdots \\
u'_r &= \left(x - b + \frac{w}{2}\cos\theta + \frac{l}{2}\sin\theta\right) / \left(z - \frac{w}{2}\sin\theta + \frac{l}{2}\cos\theta\right)
\end{aligned}
\tag{1}
$$

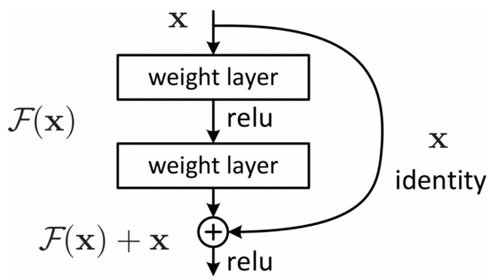

FIGURE 1: RESNET BLOCK

Reigion proposal

3. SYSTEM DESIGN

As is shown in figure 2, the delivery robot can be divided into 6 parts: stereo camera, router, battery, solid-state lidar and the processor.

Stereo camera The stereo camera is used for the detection of dynamic obstacles such as human and vehicles. Though the depth obtained by the stereo camera is not as accurate as the lidar, its object detection ability is a lot more better due to the RGB triple input channels.

Solid-state lidar The lidar is used in multiple tasks: mapping, localization and object detection.

4. EXPERIMENT

Figure 1 and 2 shows 2 views of the stereo camera object detection result of a person. The results are shown in blue bounding box with green vertices. The strips in the figures are the lidar point cloud. It is shown that the person in the bounding box is closely surrounded by the strips, which shows that the lidar point cloud overlaps with the camera object detection. Figure 3 shows the path planning result of the vehicle. The green line is the global planning path, the red line is the local planning path and the blue line is the localization.

†Joint first authors

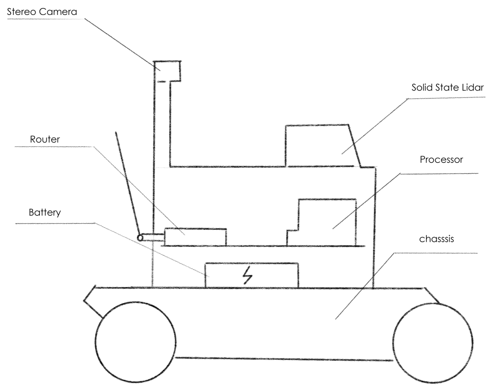

FIGURE 2: SYSTEM DESIGN

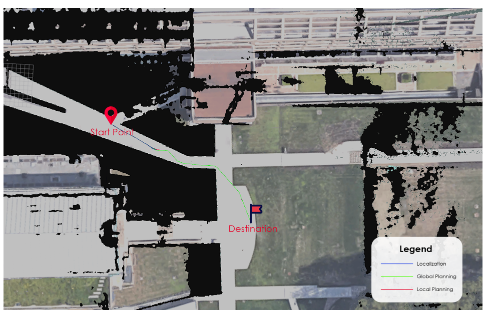

FIGURE 3: COST MAP

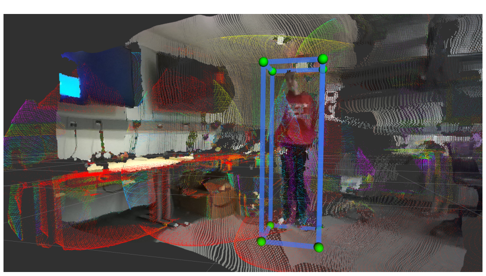

FIGURE 4: OBJECT DETECTION VERIFIED BY LIDAR

5. RESULTS

According to the experiment, the stereo camera's object detection is relatively accurate even under indoor envirnonment. The lidar's point cloud is also shown to have a very good alignment with the depth map given by the stereo camera.

ACKNOWLEDGMENTS

Place any acknowledgments here.

REFERENCES

[1] Li, Peiliang, Chen, Xiaozhi and Shen, Shaojie. "Stereo R-CNN based 3D Object Detection for Autonomous Driving." *CoRR* Vol. abs/1902.09738. URL 1902.09738, URL http://arxiv.org/abs/1902.09738.

[2] He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing and Sun, Jian. "Deep Residual Learning for Image Recognition." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: pp. 770–778. 2016. DOI 10.1109/CVPR.2016.90.

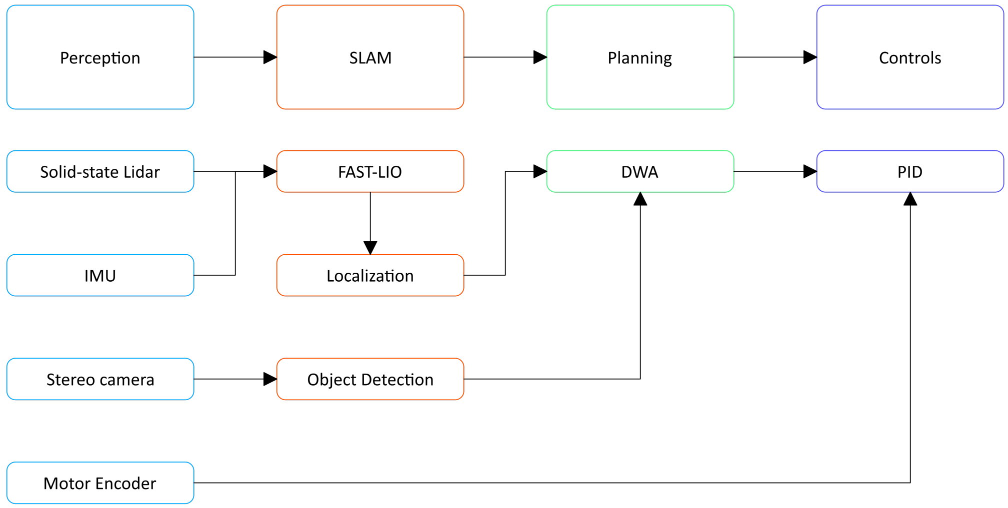

FIGURE 5: FLOW CHART

APPENDIX A. CODES FOR CAMERA LIDAR CALIBRATION

```python
import pyzed.sl as sl
import cv2
init_params = sl.InitParameters()
# runtime = sl.RuntimeParameters()
init_params.camera_resolution = sl.RESOLUTION.HD2K
# init_params.camera_fps = 30
zed = sl.Camera()
zed.open(init_params)
if not zed.is_opened():
  print("Opening ZED Camera...")
zed.grab(init_params)
image = sl.Mat()

zed.retrieve_image(image) # Retrieve the left image
cv2.imwrite("test.png", image.get_data())
```