Use of Imaging Device to Assist Autonomous Vehicles

sdmay18-03@iastate.edu

Introduction

- Souparni Agnihotri
- Ashley Dvorsky
- Eric Himmelblau
- Fahmida Joyti
- John Orefice
- Bowen Zhang
- Joseph Zambreno - Advisor
- SmartAg - Client

Project Needs

- SmartAg currently has an autonomous tractor which is guided by a preset path planning algorithm
- This does not take into account any changes to the environment and must be manually set for each new farm

Project Goal

- Utilize a neural network to detect objects
- Combine with stereo video to find locations
- Provide data in a form usable by the path planner

Design Requirements

Functional Requirements

- The image processing system detects objects including fences and combines in real-time
- Depth determination system calculates relative distance to the object using the stereo cameras
- Information can be used to add object positions to the path planning map

Nonfunctional Requirements

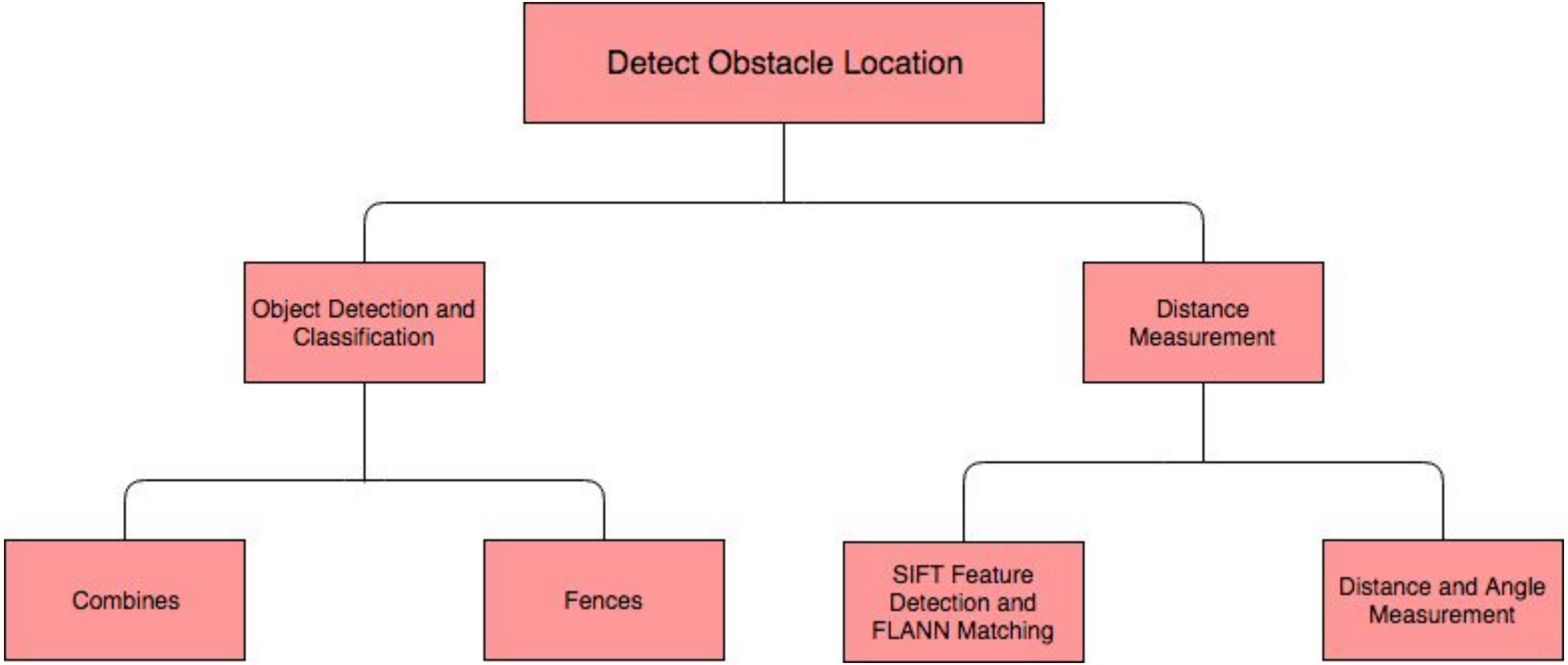
- Speed of real-time object detection system ≥ 15 FPS (NVIDIA Jetson TX2)
- System need to fit into a modern tractor
- System should be able to support both manual and autonomous driving
- Must be powered by the tractor electrical system
- Needs to be easy to use by the target audience (farmers)

Operation Environment

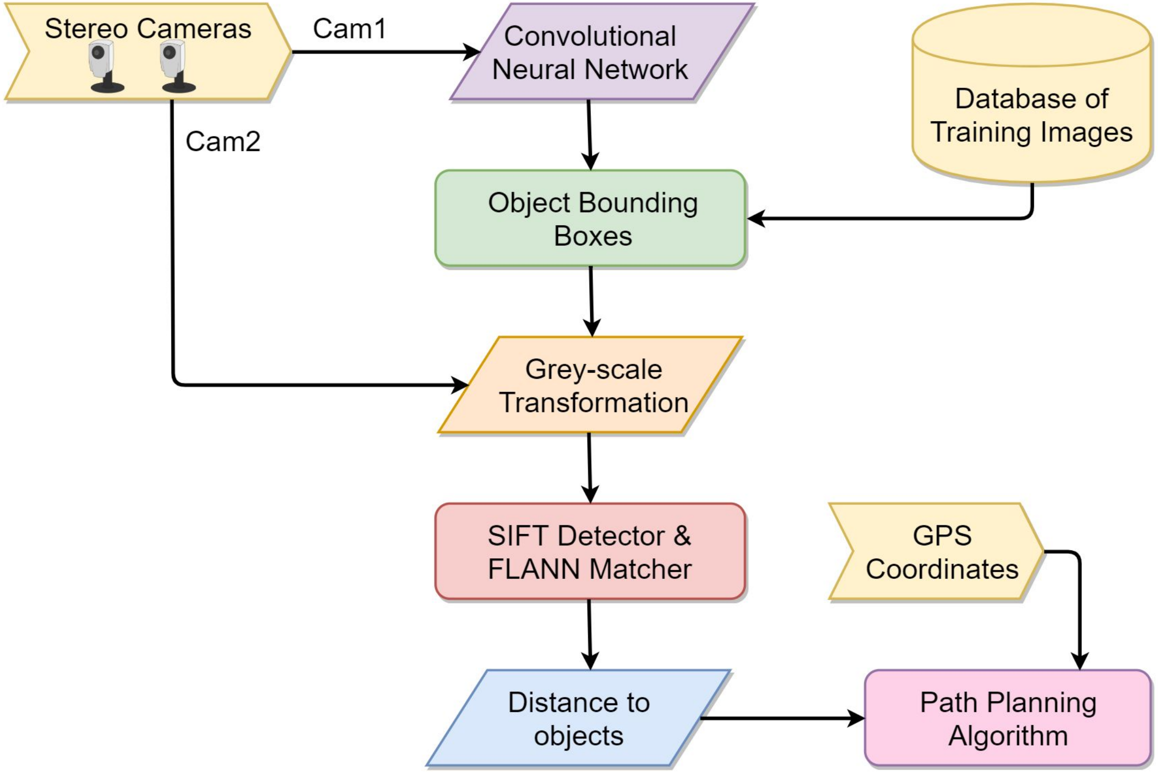
- Our software will be integrated in the SmartAg Virtual Environment
- Assuming fair weather conditions for normal tractor use
  - If a farmer would not take the tractor out, our product is also not safe to be used

Design Approach

Functional Decomposition



Block Diagram



Technical Details

Training MobileNet SSD

Training set: 2200 images

Testing set: 80 images

Classification measure:

IoU = Area of intersection / Area of union

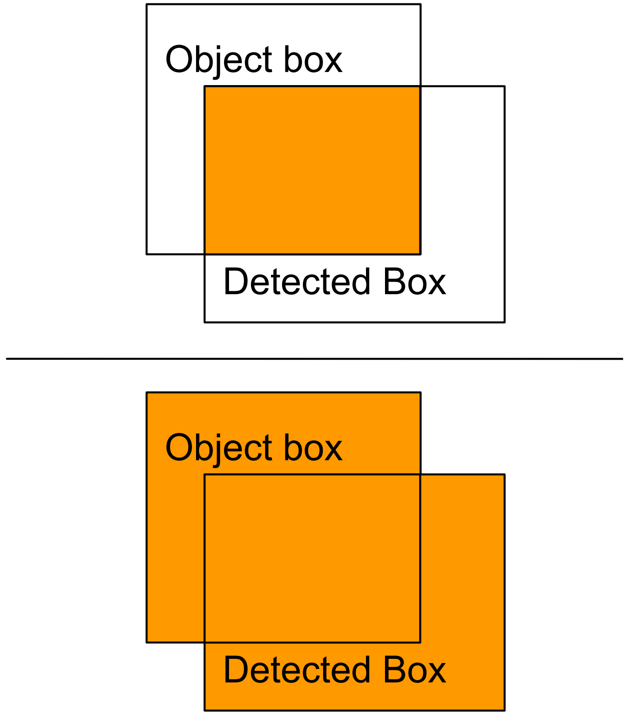
True positive (TP): IoU > 0.5

False positive (FP): IoU <= 0.5

Precision = TP / (TP + FP)

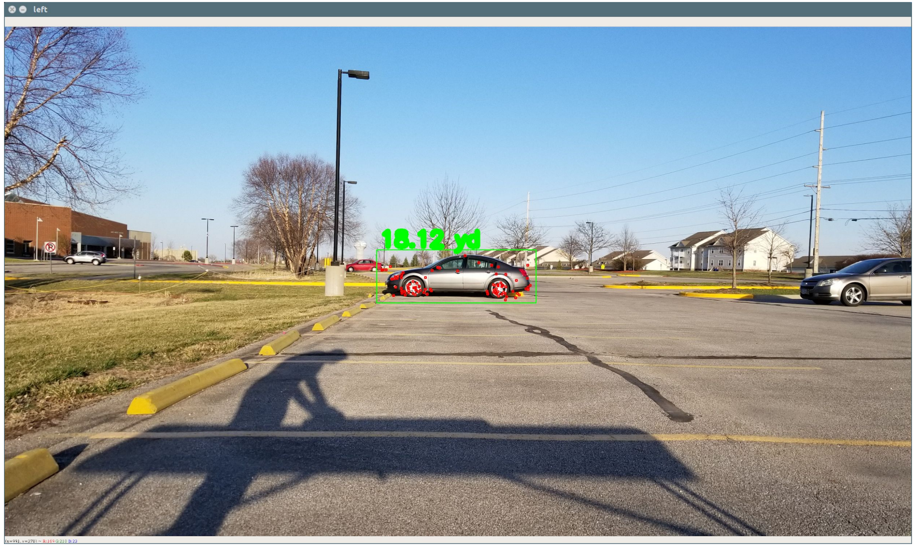
$$\text{Mean Average Precision (mAP)} = \frac{1}{|classes|} \sum_{c \in classes} \frac{TP(c)}{TP(c) + FP(c)}$$

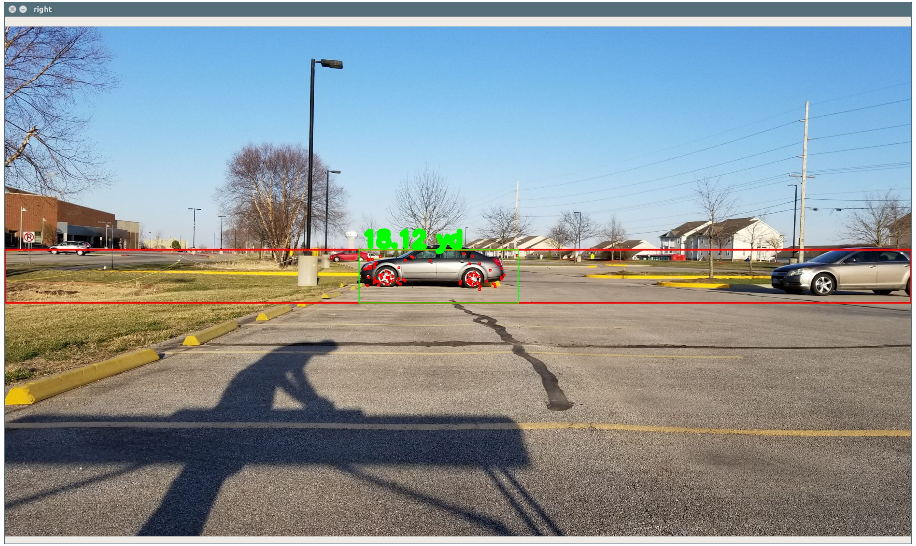
IoU =



Distance Measurement

- Object matching with SIFT and FLANN
- Distance measurement using camera intrinsics and image disparity



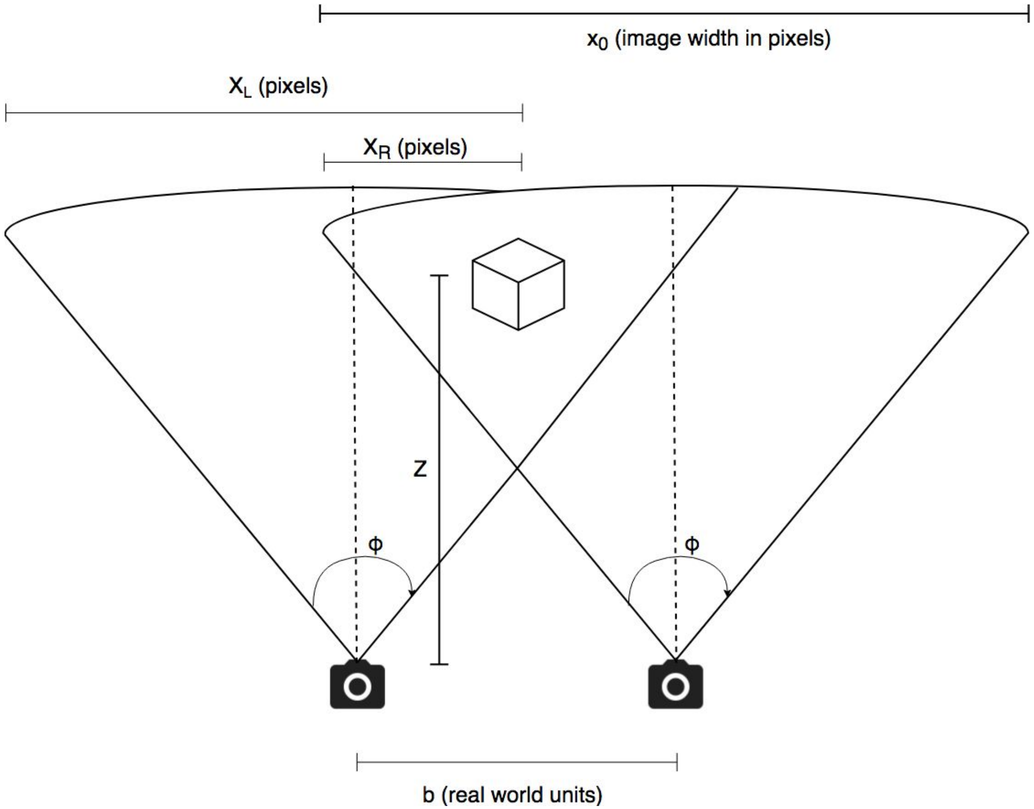


Distance Measurement

$$z = \frac{b * x_0}{2tan(\frac{\varphi}{2})(x_L - x_R)}$$

$$f = \frac{x_0}{2tan(\frac{\varphi}{2})}$$

$$z = \frac{fb}{x_L - x_R}$$



Software

- OpenCV
  - Open source computer vision library
  - Manipulation of video feed.
  - Implementing SIFT & FLANN algorithm
- Tensorflow Object Detection API
  - Open source framework built on top of tensorflow.
  - Used widely for satisfying computer vision needs.
- MobileNet SSD
  - Fastest and accurate results based on previous research.

Hardware

- Identical USB Cameras
  - Provided by our client
- Amazon EC2 Instance
  - Contains GPU which speeded up our training process
  - Supports Tensorflow and all dependencies needed to setup the training process

Challenges and Mitigations

Object Detection	Distance Measurement
• Setting up environment for training the neural network	• Finding matching features in both video feeds
• Acquiring sufficient training and testing data	• Issues with cameras
• Misclassification of objects with low quality videos	• Issues with camera calibration

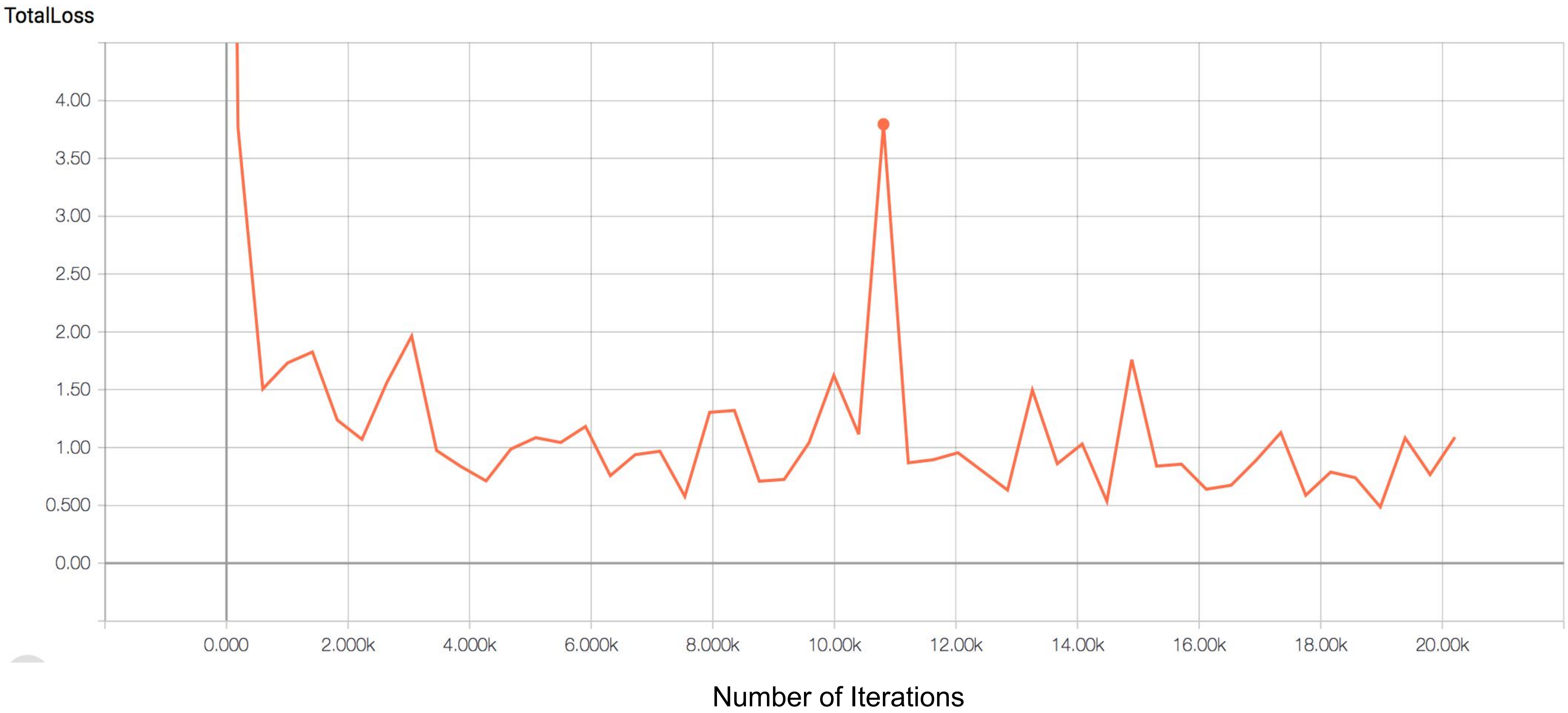
Testing Environment

- Iowa State University Campus
- Testing image set (separate from training data)
- SmartAg's local test field with a modifiable tractor and test obstacles.

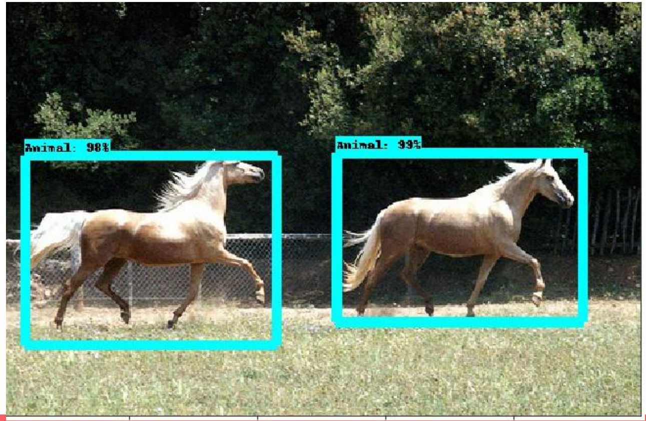
Object Detection Testing Strategy

- Static images
- Video feed
- Stereo camera, real-time video feed.

Object Detection Training Results



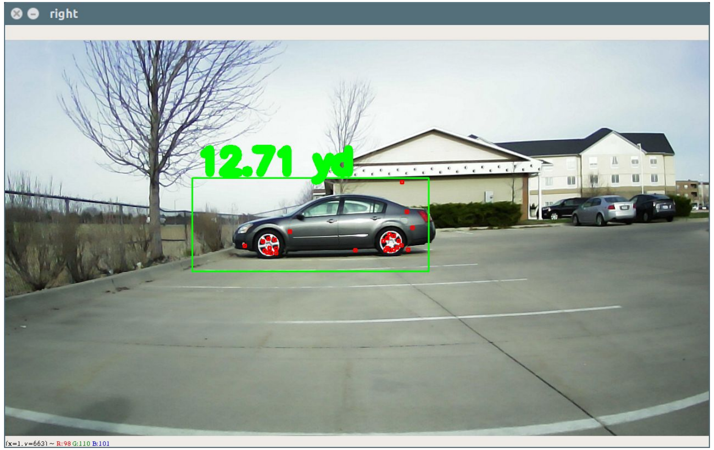
Testing results output images



Distance Testing Strategy

- Manually measure distance to an object
- Compare with results from distance system
- Repeat at different baseline and object distances

Distance Results



15 yards

20 yards

30 yards

Demo

Demo Video

Left Camera

Right Camera

Thank you!

Questions?