Use of Imaging Devices and Machine Learning Software to Assist in Autonomous Vehicle Path Planning

Project Plan

Team 03

SmartAg: Client

Joseph Zambreno: Advisor

Souparni Agnihotri: Meeting Facilitator

Ashley Dvorsky: Document Manager

Eric Himmelblau: Webmaster

Fahmida Joyti: Test Master

John Orefice: Communications Lead

Bowen Zhang: Hardware Maintainer

Team Website: sdmay18-03@iastate.edu

Version 3: Revised December 2nd, 2017

Table of Contents

List of Figures

List of Tables

List of Definitions

1. Single Shot MultiBox Detector (SSD): A unified framework for object detection with a single network.
2. DarkNet: A state-of-the-art, real-time object detection system.
3. DarkFlow: A superior version of DarkNet that is faster and more accurate.
4. Caffe: A deep learning open-source framework that is used with our SSD model to aid with image classification.
5. *mAP - Mean Average Precision - Mean of AP's in multiclass prediction
6. AP - Average precision - provides a measure of quality across all recall levels for single class classification. Seen as the area under the precision recall curve.

1. Introduction

1.1 Problem Statement

SmartAg, an Iowa based startup, has developed an autonomous tractor system. This tractor is able to use a map of GPS coordinates to autonomously navigate a farm. However, this system works solely off of their precreated map, therefore the autonomous system is not able to react to real time changes in its environmental. If a farmer adds a fence after the map is made, the tractor will simply run it over. To solve this problem, SmartAg is proposing the use of an image recognition system which will work in tandem with the path planning map.

To accomplish this task, we plan to use stereo cameras to capture information about the tractor's environment in real time. The video from these cameras will be fed into an object detection system which will identify if there are any obstacles in the immediate environment. If an obstacle is detected, the system will determine its GPS coordinates from its distance relative to the camera, and add the obstacle to the map so it can be avoided in the future.

1.2 Operating Environment

This will not be a stand alone application and will be dependent on the virtual environment provided by SmartAg. As we are just providing software and no electrical components there will be no weather conditions such as rain or temperature will directly affect our project, but as the tractor will still have to run that would be a consideration of the large scale product. This will be used in fields which can be dusty and we need to take that into consideration when training our algorithms.

1.3 Intended Users and Intended Uses

This product will be used by SmartAg in their autonomous tractor production. It will be integrated into their navigation software and then marketed to large scale production farmers. These farmers will be interfacing with SmartAg's product which will be running our application.

As the tractor is being driving by the path planning software, it will detect obstacles through the stereo cameras, and then identify if there is a border or fence. Upon detecting and identifying an obstacle as such we will first start to navigate away from said obstacle and then pass the GPS coordinates so the algorithm will avoid that obstacle in the future.

1.4 Assumptions and Limitations

i. **Cost -** The camera/sensor and harnessing must be less than $1000.
ii. **Power -** The tractor's electrical system will power our system. We will not be using any external power systems.
iii. **Datasets -** We don't have a sufficient dataset which we can use to train our model. We will be gathering and labeling our own dataset.
iv. **Knowledge of Area -** None of us had much experience with any of the technologies associated with the project. This project requires the team to be able to use image processing libraries such as OpenCV and be familiar with neural networks.We intend to mitigate this issue by setting aside training time for getting familiar with the new technologies we are supposed to be working with.
v. **Testing limitations -** We will be training our model during the harvest season but by the time we will be testing this it will be the winter season during which the weather conditions will be significantly different. This could have a negative impact on the system's ability to reliably identify important objects.
vi. **Hardware Materials -** Currently we do not have production hardware to work with. We will be using our own computers for some work, but we have acquired access to a TitanX box in the lab which has a similar GPU to the one that will be in production.

1.5 Expected End Product and Other Deliverables

At the end of the project, we will have a fully functional object detection system that can identify any object that is within the scope of our dataset or any object that looks similar to another object in the dataset. In addition, we will have a depth calculation algorithm that can calculate an object's position relative to the tractor, primarily off of image data. This information will then be translated into a GPS location which can be used by the path planning algorithm already in place. All of this should be done in real time while the tractor is autonomously moving.

2. Proposed Approach and Statement of Work

2.1 Functional Requirements

1. The image processing system shall be able to detect objects such as fence, ditches and terraces in real time using an appropriate Neural Network.

2. Use depth determination techniques to find how far away an object is and its dimensions so that the tractor can successfully circumvent them.
3. Add object positions to the path planning map so that they can be avoided in the future.

2.2 Constraints Considerations

We have limited experience in terms of image recognition, machine learning and artificial intelligence so there will be a lot of learning and experience that will have to be built during this project. This project will only last through the end of the spring semester of 2018 and all deliverables will have to be submitted at that point. We will also have to work closely with our client to ensure that our scope is not too large, but can still meet all of the functional requirements they have set.

2.3 Technology Considerations

We will be utilizing OpenCV[1], Python and MobileNet-SSD [2] Neural Network to aid with image processing. We had to decide between using three different Neural Networks to achieve our tasks: DarkNet[3], DarkFlow[4], and SSD[5]. The primary advantages and disadvantages of each neural network are listed in *Table 1* below.

Neural Network	Advantages	Disadvantages
DarkNet	- Runs at 45 FPS using an Nvidia Titan X GPU - Understands generalized representation of objects.	- Difficulty to predict objects in groups. - Loss function treats the error the same for small and large bounding boxes. -Has trouble when it sees unfamiliar aspect ratios or configurations
DarkFlow	-Runs at 67 FPS using an Nvidia Titan X GPU -Darknet predicted only 98 boxes per image but Darkflow predicts more than a thousand boxes per image	- Difficulty determining two similar, but differently sized objects. - Still has difficulty predicting objects in group. - Still has trouble when it sees

[1] Reference to OpenCv
[2] Reference to MobileNet-SSD
[3] Reference to DarkNet
[4] Reference to DarkFlow
[5] Reference to SSD

	- mAP of 78.6 on a 544 x 544 image - Recall of the convolutional neural network increases from 81% to 88% compared to DarkNet	unfamiliar aspect ratios or configurations
SSD	- Significant improvement in speed for high accuracy detection compared to the others. - Runs at 59 FPS with 74.3% mAP when tested on the VOC2007 using an Nvidia Titan X GPU - Naturally handles objects of various sizes (due to the combination of different feature maps with different resolutions) -Can detect objects in multiple aspect ratios	- Relies on Caffe which is difficult to set up on a Windows computer. - Difficulty distinguishing between similar objects (ex horse vs zebra) - Lower accuracy when identifying small objects compared to large objects

Table 1: Comparison of Available Neural Nets

Our initial research shows that Darkflow is a clear choice over Darknet since it offers clear improvements for our uses, however, it still has to be compared to SSD. After additional research and testing, our final decision is to use SSD over darkflow. SSD provides both more accurate object detection and faster performance when compared to Darkflow making it a clear choice for our project. Additionally, it's primary disadvantages will not have a large impact on our project. The difficulty regarding installation will be a only be present early on in the project and won't be a concern later on. Further, we are not concerned with identifying small objects, and in our use case if it mistakenly categorizes one object as a similar looking one, it is likely that we will still want to avoid it, so this should not be an issue either.

Once we got the SSD set up on the TitanX box, we trained and tested the model and got the expected mAP value and loss. We are in the process of figuring out how to integrate it with a webcam so we can do real-time object detection.

2.4 Safety Considerations

As we are working with a process that typically involves human interactions we need to thoroughly test this application to ensure that it will be safe to run autonomously when there could be people around. Additionally, we need to guarantee that it causes minimal damage to property on the farm.

2.5 Previous work / literature review

The core of the image classification, object detection and depth perception has already been experimented with and implemented by several credible sources. For image classification we had the opportunity to choose which neural network to use (DarkFlow, DarkNet, SSD). Just running these Neural Networks helped us do image classification with high confidence.

We also aim to detect distance to objects via a combination of cameras and radar sensors. This has been approached in a multitude of ways in the past. One approach is to use a suite of laser distance sensors (Lidar/Radar) but these may fail in fuzzy environments (such as a row of wheat stalks). Another approach is to use multiple cameras to calculate distance similar to human eyes. Another approach is to calculate a depth map from a single still image using a trained neural network. This approach may be time consuming, so a more lightweight NN that analyses a stream of still images (a video feed) from the camera can be used.

We will continue to use the many available online resources to better understand the inner workings of these neural nets. One excellent source that we plan to continue using is Stanford's openly available course, CS231n: Convolutional Neural Networks for Visual Recognition[6]. Additionally the online textbook, Neural Networks and Deep Learning[7] is an accessible yet informative source that we plan to reference throughout this project

In addition to testing objects, we must measure their relative distance. This can be done by using a single image to construct a depth map as documented by Saxena et al[8]. Stereo images can also be used to determine depth information, a method which has been proven through millions of years of evolution. Many sources, including the University of Washington's CSE 455[9] and Induchoodan et al[10] provide extensive documentation regarding the methodologies and mathematics behind doing so.

[6] Reference to CS231n
[7] Reference to Neural Networks and Deep Learning
[8] 3-D Depth Reconstruction from a Single Still Image
[9] University of Washington, CSE 455
[10] Induchoodan et al

Companies like General Motors[11], Tesla [12], Google cars[13], and NASA and John Deere[14] are also working on developing autonomous vehicles. For our project, we will primarily focus on detecting objects and sending the GPS coordinates of the objects to SmartAg's path planner instead of developing a complete autonomous vehicle. For this, there are several sources for depth determination available online.

2.6 Possible risks and risk management

One of the first risks that need to be addressed was whether or not the project scope is too large for a senior design project. To mitigate the risk we will have a well defined project plan with realistic goals and timelines. This will ensure that we have enough time to properly research the technologies needed, implement all parts of the system, and thoroughly test them.

There are also additional risks concerning the generation of a training set of images. One concern is simply whether or not we will be able to feasibly collect a large and varied enough dataset to guarantee reliable object detection and classification. Additionally, any images that we manually collect will likely be taken during a different time of year than the tractor will be operating. This could potentially lower the accuracy of the image recognition system when it is being used in production.

2.7 Proposed Diagram

The performance and accuracy of our image detection and distance determination system will rely heavily on the types of Neural Networks we use. Below is the basic block diagram for our system. The external hardware we are using are a camera and a GPS/Radar which will work hand-in-hand with the Neural network to do classify images and determine distance.

[11] Reference to General Motors
[12] Reference to Tesla
[13] Reference to Google
[14] Reference to NASA and John Deere

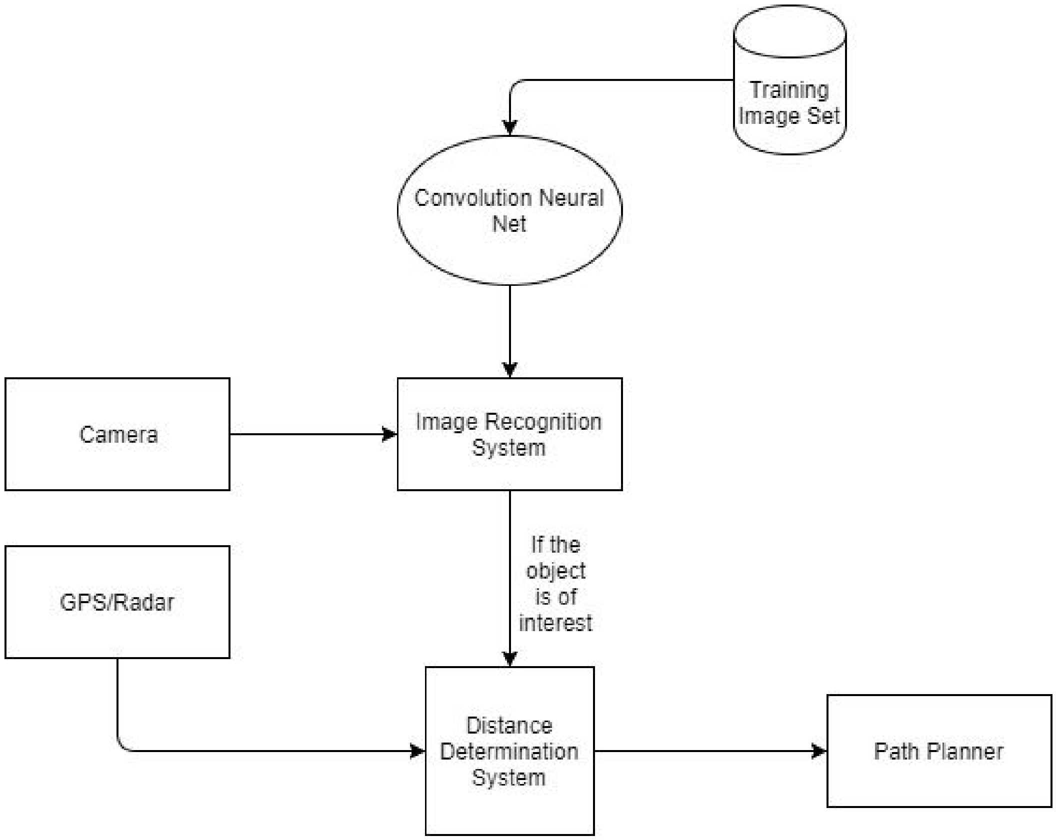

Figure 1: System Diagram for Object Detection and Distance Determination System

2.8 Project proposed milestones and evaluation criteria

Milestones	**Description**	**Planned Date *(tentative)**
Research/Learning	1. Neural Networks 2. OpenCV 3. Darknet	2017-9-30
Experiment with Different Methodologies	1. Test various preexisting image classification networks. 2. Determine which one to use.	2017-11-15
Image Recognition System	1. Identify and label objects of interest (fences, roads etc.)	2018-01-30
Depth Perception	1. Identifying how far away an object is. 2. Determines GPS position of object to be added to path planning map	2018-02-28
Testing	1. Test algorithm and train more as	2018-04-20

	necessary	

Table 2: Major Project Milestones

2.9 Project tracking procedures

We will meet weekly and separate tasks into two week sprints with tasks based off of previous performance and indication of the future schedule. We should ideally have tasks split so that everyone can do all of their work in the two week sprint. Within each sprint, we will us a kanban board to organize the tasks assigned to each individual and ensure that they are completed according to schedule.

We are also having a gantt chart to keep track our current schedule.

2.10 Statement of Work (Task objective, approach & expected results)

2.10.1. Research and Learning

a. **Objective of Task:** Become familiar with the tools required for this task, including Python, openCV, DarkFlow, DarkNet and SSD.
b. **Task Approach:** Split the group into teams of 2. Each team would research one of the neural networks (DarkFlow, DarkNet, SSD). They will also do side experiments on Python and OpenCV. In the meetings, each group will share their ideas and the team will come to a unanimous decision on which neural network to use.
c. **Expected Results:** Extensive research will be done on the Neural Networks so that the team will be able to make the right decision on which network to move forward with. All the team members will be comfortable with Python and OpenCV.

2.10.2. Set up Image Recognition System

a. **Objective of Task:** Use the Neural Network selected to do image recognition in real time. Extend that to recognize specific images of interest (fences, roads etc).
b. **Task Approach:** We started by gaining remote access to the Titan X box in which we set up the Neural Network (SSD). We trained given data on the network and got the desired loss and mAP values. We have begun shortlisting a feasible set of objects we wish to recognize and generating a training set of images. We will use this dataset to train on the SSD neural network and then test it to see how well our neural network is classifying those images.

c. **Expected Results:** Training and testing the MobileNet-SSD with the dataset that we acquire should give us the following results:
   i. A mAP score of at least 0.70 on the training set and loss function of less than 0.6 on the training set.
   ii. Achieve a minimum accuracy of 80% on the test set and have it run at a minimum FPS rate of 15 in real-time when tested with a webcam

2.10.3. Determine Object Position (depth) from Image

a. **Objective of Task:** Determine the relative position of an object to the camera based on an image of it and translate this to a GPS coordinate.

b. **Task Approach:**
   i. Identify the object using our image recognition system
   ii. Calculate the distance of the object from the tractor
   iii. Use the GPS coordinates of the tractor to convert relative position to GPS coordinates
   iv. Send the GPS coordinates to the path planner used by SmartAg to assist in path planning

c. Expected Results: Create a system which can take an image with and object identified as an input and produces GPS coordinates representing the images object's location as an output.

2.10.4 Testing

a. **Objective of Task:** Thoroughly test the image recognition and object positioning systems.

b. **Task Approach:**
   i. Create a test plan and test cases for individual components.
   ii. Perform tests and address any bugs/unexpected results.
   iii. Retest until all test requirements are met.
   iv. Create an integration test plan.
   v. Perform tests and address any bugs/unexpected results.
   vi. Retest until all test requirements are met.
   vii. Inform client of any bugs or performance issues which were not resolved.

c. **Expected Results:** Prove that our final product meets SmartAg's criteria by successfully passing all tests.

2.11 Testing requirements considerations

To ensure that our end product meets all requirements set by the client and does so without any unexpected bugs, each part of our system will be tested independently, and then again once everything has been integrated together. Regarding independent testing, the primary components that will need to be tested are the object detection system and the distance determination system.

The object detection system will need to be tested for both accuracy and speed. Our system will be considered satisfactory if the neural net achieves a mAP score of at least 0.70 on the training set. That is, for every 10 images received, at least 7 would be classified correctly. Additionally, in order to minimize expected risk, we will try to minimise the loss function and attempt to keep it below 0.6. This will allow us to be confident that it will not miss any major obstacles in the tractor's path. Additionally, since this system will need to perform in real time, we would like it to run at a rate of at least 15 frames per second on production hardware. While all neural nets we've considered claim at much higher speeds than this, they are often tested on cutting edge desktop level hardware, which is far more powerful than the embedded hardware we will need it to run on. This means that although training can be done on desktop computers, testing should be reserved for production hardware.

Accuracy in the distance determination system is another critical aspect of this project. While it is important that we can identify potential obstacles, this is of little value if we do not have a good idea of where they are. Fortunately, this can easily be tested in a simulated environment. Determining the distance to a person or a chair in a room should be relatively similar to finding the distance to a fence in a field. This means that preliminary testing can be in an environment which is easily replicated so that we can compare results and see improvements over time. Once we are confident that our algorithm is effective in a simulated environment, it can be further tested in a farming environment.

After being thoroughly tested individually, we must guarantee that these two systems can work in tandem, and further show that they can work with the pre-existing path planning algorithm. This will be done in a similar manner to the testing of the distance determination system. We can begin by testing in a simulated environment, using data from the object detection system as an input to the distance determination system. After we establish that we can correctly identify an object of interest and accurately find its relative distance, we can replicate the test in a farm environment on a running tractor. Of course, at this point all testing should be done on production hardware, specifically a John Deere tractor with stereo cameras and an Nvidia Jetson TX2 GPU. Outdoor testing should be performed in various weather and farm conditions so that we have a comprehensive understanding of our system's true capabilities.

2.12 Assessment of Proposed Solution

We believe that the above proposed solution will be satisfactory in meeting all of the needs of our client SmartAg. We have clearly defined all functional requirements of this project and have identified the constraints imposed upon us. Further, we have taken into consideration technological requirements and hindrances as well as safety concerns when proposing this solution. We have used of this information, combined with review of related literature, to suggest a solution which minimizes any risks and meets all necessary requirements. With this solution in mind, we have broken the project up into several primary milestones which are accompanied by industry proven progress tracking procedures to guarantee that they are completed on schedule. Finally, we have carefully come up with testing methodology to ensure that all parts of our system perform reliably and as expected.

3. Estimated Resources and Project Timeline

3.1 Personnel Effort Requirements

Task	Estimated Hours Required	Description
Research and Learning	120	Since we will be in specialized subteams, each person will be able to focus on a single research area. This assumes that we will have a sufficient understanding of the technologies needed if each person spends 20 hours total researching his/her respective topic.
Compare Methodologies	320	This will be the largest aspect of our project since we will need to brainstorm different methodologies, learn any method specific skills needed to implement them, create POCs for each, and compare them to determine which is the most promising.
Image Recognition System	120	This task will involve expanding on both our training and testing data sets. We will then need to train the system and ensure that it works in real time.
Object Positioning System	150	Depending on whether we decide to do this solely using a single image as data, use stereo images, or use peripheral sensors to assist in this task, it will have various complexities. Since there is currently a lot of uncertainty

		regarding this task, we want to allot time for any unexpected roadblocks.
Testing	200	Testing will be spread throughout this project and is definitely important, so we intend to spend a significant portion of time on it. Once the bulk of development is complete we will need to come up with narrow tests and tests for edge cases to guarantee that we have a reliable system.

Table 3: Major Project Task Descriptions and Work Estimation

3.2 Other Resource Requirements

- Camera
    - Stereo wide angle
- GPS
- Embedded Nvidia GPU (Jetson TX2)
- NeuralNet to recognize objects (SSD)
- Image Datasets
    - Large training set
    - Intermediate testing set for comparisons
    - Smaller testing set
- Peripheral Sensors
    - Radar

3.3 Financial Requirements

Our overall budget per until cannot exceed $1,000 per unit including software and any additional sensors or materials that we add.

3.4 Project Timeline

The Gantt chart (*Figure 2*) is the timeline for the whole project.

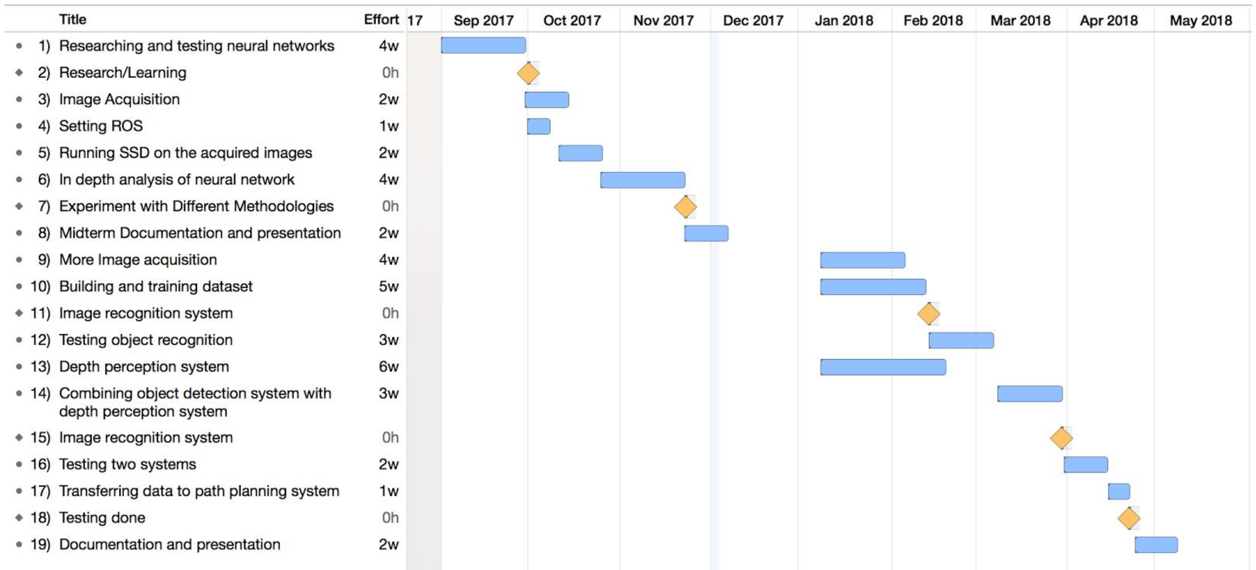

Figure 2. Gantt chart for the project

4. Closure Materials

4.1 Standards

Our team will be following the below standards:

- Version Control:
  - We will be using github for doing version control for all our project code.
  - We will be using Google software suite such as Google docs, sheets etc for maintaining our documentation such as the Design Document,Project plan etc since it allows to keep track of the revisions we made to our document and can be accessed easily from the menu by selecting File -> Version history
- Code Review:
  - Each of our team member will be working on separate branches in git
  - Once they have finished working on a feature they will submit a pull request
  - Two of the team members will review the code to determine whether the code satisfies the requirements for the feature and integrates well with the existing code.
  - The new code will then be merged with the master branch
- IEEE Standard - Ethically aligned design: A vision for prioritizing human well being with artificial intelligence and autonomous systems.
  - The intention of this standard is to cover, the ethical concerns on AI/AS through a rigorous regard to the problem from different perspectives. The ultimate objective of this initiative is to provide guidelines/procedures/standards to prioritize human well being in the forthcoming evolutions on artificial intelligence and autonomous systems.
  - The main purpose of this standard is - “to ensure every technologist is educated, trained, and empowered to prioritize ethical considerations in the design and development of autonomous and intelligent systems”

4.2 Conclusion

We will be working as a team to improve upon Smart Ag’s current path planning software by adding the feature of object detection and image recognition. We will research our neural network options, experiment with different methodologies, implement the image recognition system, implement the distance calculation system, and thoroughly test all parts to guarantee they meet the project requirements. We have a strict deadline with multiple sprints that will be accomplished by sub squads to ensure that we will be working towards our goal with enough time and accountability.

This is a project that our team has prioritized as a high priority and are willing to work hard to complete to full satisfaction. Technology is the future of the farming industry and being surrounded by an agricultural community this is something that is very close to all team members.

4.3 References

1. “OpenCV Library.” OpenCV Library, opencv.org/.
2. Chuanqi. “MobileNet-SSD, GitHub, 5 Aug. 2017. https://github.com/chuanqi305/MobileNet-SSD
3. Pjreddie. “Darknet”. GitHub. 1 Oct. 2017. https://github.com/pjreddie/darknet
4. Thtrieu. “Darkflow.” GitHub, 24 Sept. 2017, https://github.com/thtrieu/darkflow
5. Balancap. “Balancap/SSD-Tensorflow.” GitHub, 10 Apr. 2017, https://github.com/balancap/SSD-Tensorflow.
6. Johnson. “Convolutional Neural Networks”. Stanford University. 10 May. 2017
7. Michael A. Nielsen, "Neural Networks and Deep Learning", Determination Press, 2015
8. Saxena. “3-D Depth Reconstruction”. Stanford University. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.78.7759&rep=rep1&type=pdf
9. University of Washington. “Stereo and 3D Vision”. University of Washington. https://courses.cs.washington.edu/courses/cse455/09wi/Lects/lect16.pdf
10. Induchoodan et al. “Depth Recovery from Stereo Images”. IEEE. Nov 27. 2014 http://ieeexplore.ieee.org/document/7019764/
11. General Motors. “GM Advances Self-Driving Vehicle”. GM. 9 Oct. 2017. http://www.gm.com/mol/m-2017-oct-1009-lidar1.html
12. Telsa. “Full Self-Driving Hardware onA ll Cars”. Tesla https://www.tesla.com/autopilot
13. Google. “Self Driving Car Project”, Waymo https://waymo.com/
14. NASA and John Deere. “How NASA and John Deere Helped Tractors Drive Themselves”. NASA and John Deere. 16 Nov. 2016. https://www.nasa.gov/feature/directorates/spacetech/spinoff/john_deere
15. Weiliu. “Caffe”. GitHub. 10 Oct. 2017. https://github.com/weiliu89/caffe/tree/ssd
16. “Feed the World Smarter.” Smart Ag, https://www.smart-ag.com/
17. Allanzelener. “Allanzelener/YAD2K.” GitHub, 2 July 2017, https://github.com/allanzelener/YAD2K.
18. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2016.91
19. Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2017.6
20. The PASCAL Visual Object Classes Challenge 2007. (n.d.). Retrieved from http://host.robots.ox.ac.uk/pascal/VOC/voc2007/21. The PASCAL Visual Object Classes Challenge 2012. (n.d.). Retrieved from http://host.robots.ox.ac.uk/pascal/VOC/voc2012/