Industrial Placement

Imperial College London

Department of Electronic and Electrical Engineering

Interim Report

Jacky Jiang (CID: 01860908)

Date: June 4, 2023

Contents

1 Introduction

Background The concept of autonomous driving has existed since 1920s. In 1925, Manhattan, inventor Francis Houdina showcased an extraordinary demonstration of a radio-controlled car seen in Figure 1. Without anyone sitting inside, the car started the engine, shifted gears and sounded the horn [2]. In recent decades, the industry has dramatically evolved with emerging companies like Waymo, Baidu Apollo, Tesla etc. In 2014, the SAE (Society of Automotive Engineers) International has created 6 different levels for autonomous driving based on the level of automation [1]. Level 0: No driving automation with only human interactions. Level 1: Driver assistance with some assistance on adaptive cruise control. Level 2: Partial driving automation with consistent assistance on steering, acceleration and breaking. Level 3: Conditional driving automation. Vehicles can operate without human interactions but drivers must be ready to intervene. Level 4: High driving automation and Level 5: Full driving automation are similar that requires no human intervention at all and does not need a safety driver on the vehicle [1]. Tesla for example is the world leading L4 autonomous driving system company. Its Full-Self Driving (FSD) and Autopilot is the world's most advanced driver assistance features and it is currently undergoing beta testing in the North America market. It is highly autonomous that requires little intervention from the driver. From my perspective, Tesla has achieved the ideal autonomous driving that everyone was imagining since 1920s. However, it is still far from perfect. For instance, if accidents happened in front and the police was guiding the traffic, it could not read the body language from the police to conduct certain actions. Also, there are not specific legislation and regulations on autonomous driving vehicles. Whether the car manufacture, the autonomous driving solution provider or the driver should be held accountable is still unclear. Therefore, high level scenarios cannot be put into mass production and thus cannot bring profit to majority companies especially those pure solution providers that do not manufacture cars. The market in China is highly competitive and companies that mainly focus on high level scenarios cannot survive. Therefore, instead of investing into researching level 4 and level 5 autonomous driving, majority solution provider companies are paying more attention on low level scenarios because of their commercial values.

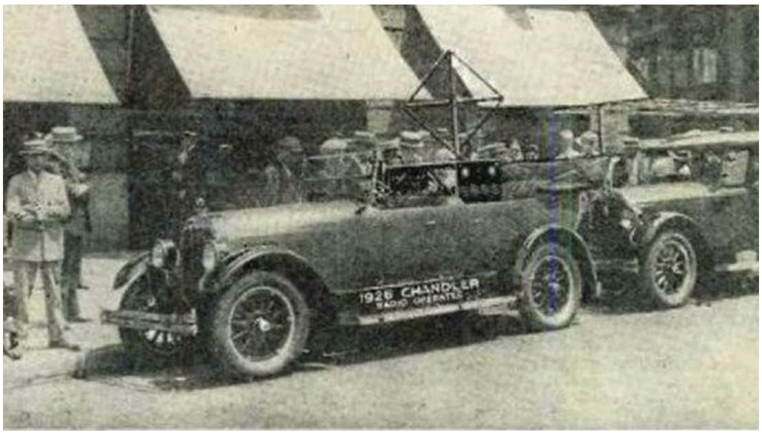

Figure 1: Houdina's Remote Controlled Car. Photo Credit: Discovery Magazine

Company Introduction ZongMu Technology (Shanghai) Co., Ltd., established in 2013, is an ADAS (Advanced Driver Assistance Systems) and autonomous driving technology provider based in Shanghai Zhangjiang International Science and Technology Innovation Center. The company's vision 'From Technology to Production', commits to bring autonomous driving technologies into mass production with R&D centers in multiple major cities in China, in Michigan, United States and in Stuttgart, Germany and manufacture centers in Xiamen, Huzhou. From the beginning, ZongMu has received investment from renowned firms like Qualcomm Ventures, Xiaomi Corporation, Legend Capital etc. and it has been working closely with world-famous partners including BYD, Horizon Robotics, Texas Instruments, Qualcomm etc. The company possess an extensive product layout from algorithm software to system hardware, from intelligent control units to a wide range of intelligent sensors, offering vehicle manufactures with a complete autonomous driving solutions with three branches of products: Smart Driving System, Wireless Charging System for Electric Vehicle and Smart City Solution. The main branch, Smart Driving System, involves AVP (L4 Autonomous Valet Parking), APA (L2 Automatic Parking Assistant), ADAS (Full dimensional Advanced Driving Assistance System), ZATLAS® (Full-Stack High Precision Map) and ZEALOUS (L4 Autonomous Driving Cloud Solution). ZongMu® Tech has reached several milestones over the past 10 years including over 1 billion RMB in Series E financing. It was included in the '2022 China's Top 50 Innovative Companies' list published by Forbes China and ranked the 920^{th} company in the '2023 Hurun Global Unicorn List' with an estimated enterprise valuation of 8 billion RMB.

Department and Products There are twenty departments and over a hundred divisions in ZongMu. I am currently working as a Computer Vision Algorithm Engineer at the Central Engineering Department (213 colleagues), Computer Vision division (52 colleagues). Our CV division is responsible for pre-research, development, deployment, and final mass production of visual perception algorithms in AVP, APA and ADAS product line, Smart Driving System branch. To be more specific, my team oversees the APA (Advanced Parking Assistant) product line, and the team is developing the 7.0 generation, which will be equipped on Changan Automobile's new electric vehicle platform-C385. The previous generation APA 6.0 has brought remote intelligent parking into reality in which users can control and maneuvering the vehicle through a mobile phone and ZongMu was the first Chinese company to achieve this advancement in industry. APA 7.0 on the other hand aims to provide the first autonomous queuing parking feature in which the users can exit the vehicle outside the parking garage. Queuing to enter the garage, driving through the gate, finding a free space, and parking the vehicle will all be handled by the system. Besides parking, APA 7.0 also features pick-up functionality in which users can schedule in advance pick-up time and location on mobile device. The location can be the evaluator entrance, outside the underground garage, or the entrance of highway service area and the system will then drive the car to exit the parking area and stop at the designated pick-up location. The major breakthrough in technology behind APA 7.0 is a combination of perception hardware system, advanced computer vision and perception algorithm. Perception hardware system has 27 sensors in total including 5 4D imaging millimeter-wave radars designed to solve parking lot perception, 12 next-generation coded ultrasonic radars, 4 high-performance surround-view cameras, and one high-performance front-view camera with resolution up to 8 million pixels. The algorithm part utilizes deep learning and multi-sensor fusion for precise perception, achieves accurate positioning through vision semantics and feature fusion, and finally uses graph search and optimization to handle complex scenarios in the parking lot. Besides these two powerful advancements, through utilization of numerous amount of data and continuous training and optimization of cloud-based algorithms, APA 7.0 is continuously improved in terms of performance, enabling OTA (Over-The-Air) iterations and future upgrades.

2 Projects

Since my team is responsible for APA 7.0, I am involved with the development of computer vision and sensor fusion algorithms for APA 7.0. To be more specific, I am currently working on the Freespace model of APA 7.0 along with my mentor. Thanks to the colleagues in the software department, we could constantly modify and upgrade our model and algorithm through OTA update to the vehicle. Seen in Figure 2 to Figure 4, Freespace is a computer vision model that identifies the available space and path that vehicles can move freely including avoiding other vehicles, pedestrians, pillars, cones, and other obstacles in the parking garage.

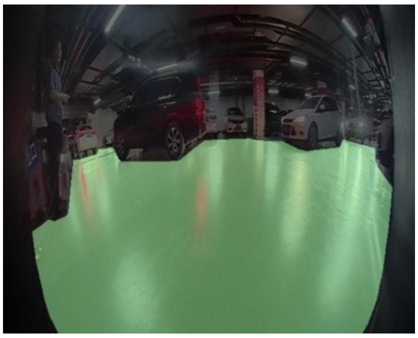

Figure 2: Freespace Visualization 1

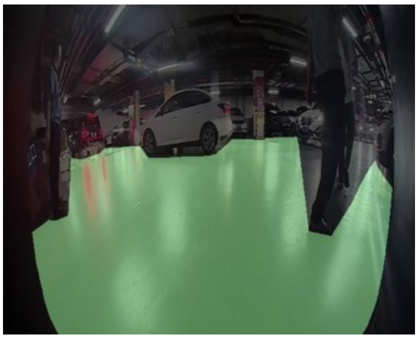

Figure 3: Freespace Visualization 2

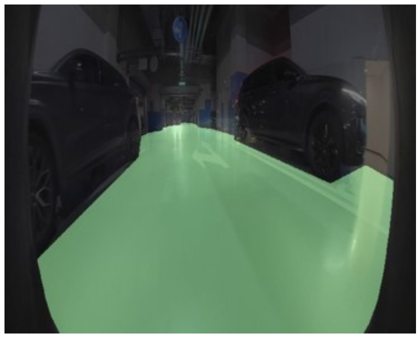

Figure 4: Freespace Visualization 3

Model The model architecture we have developed is a modified fully convolutional network (FCN) seen in Figure 11 with an encoder and a decoder. The model takes input image tensor of size 1 x 3 x 240 x 320 (1 is the batch size and 3 is the number of channels (RGB), 240 is the image height and 320 is the image width), performs binary classification task on the input image, and outputs a score for the available space class and the non-available space class through the softmax activation function at the end of the decoder.

The encoder seen in Figure 5 is composed of a series of downsampling blocks, each comprising multiple 3x3 convolutional layers for feature detection with non-linear ReLU activation function, followed by a 2x2 max pooling layer. Due to its downsampling behavior, the encoder functions as a feature extractor that progressively reduces the spatial dimensions of the feature maps by selecting the maximum value (max pooling), compressing the image representation from high resolution to lower resolution while capturing more global information with increasingly effective receptive fields.

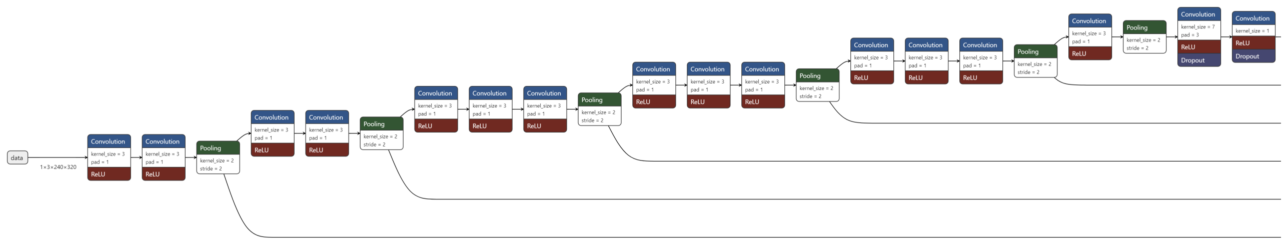

Figure 5: Freespace FCN Encoder

Similarly, the decoder seen in Figure 6 consists of a series of 4x4 deconvolutional layers (also known as transpose convolution) with padding equals 1 and stride equals 2. In reverse to convolutional layer with downsampling effect, deconvolutional layer upsamples the low-resolution feature maps, increasing the spatial dimensions of the input image tensors while capturing more finer-grained information compared to the encoder. Skip connections are also introduced in the decoder to concatenate the outputs of the encoder and the inputs of the decoder. Because spatial dimensions were gradually reduced in the encoder and there might be loss in spatial details, skip connections could help preserve the spatial information from early stage in the encoder and introduce them back in the decoder. Additionally, since skip connections connect multiple layers in the encoder and decoder. This concatenation of feature maps with different resolutions in different layers brings multi-scale information to the model and thus increases accuracy in decision making process with more contextual information, both local and global semantics, being presented. Finally, skip connections help moderate the potential vanishing gradient by allowing gradients to flow from the decoder to the encoder along the short cut connections. However, when

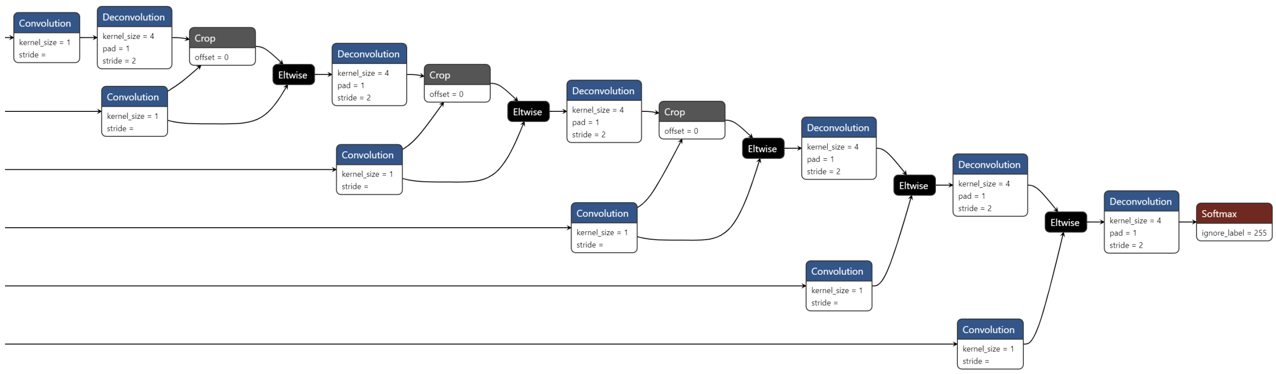

Figure 6: Freespace FCN Decoder with Additional Convolutional Layers

testing the model, we have noticed a significant increase in the number of parameters and thus a noticeably longer training time. After investigation and debugging, we have concluded that the increase was a result of directly concatenating the encoder feature maps to the decoder feature maps, which has essentially doubled the number of parameters. To tackle this problem, we have introduced additional convolutional layers with reduced number of output channels shown in Figure 6. Therefore, instead of connecting the output of the convolutional layers in the encoder directly to the output of the deconvolutional layers in the decoder, skip connections will first go through a convolutional layer before concatenation. This has substantially moderated the model's complexity while still maintaining the effect of skip connections. Furthermore, Crop operations are added in the decoder which take decoder feature map and skip connection feature map as input and align them in terms of spatial dimensions so they can be concatenated correctly. Finally, element-wise sum operations perform summing over elements from the cropped feature maps and the feature maps from the skip connections. Along with deconvolutional layers, convolutional layers, and crop operations, together they form the feature fusion block, and the decoder is composed of a series of feature fusion block. The Freespace model has been installed on the collection vehicles and is currently undergoing real-scenario testing.

Data Processing In the Deep Learning course, I have learnt that pre-processing of data is extremely vital and it have been proven true in my placement project. The data I used for continuous training comes the four surround-view cameras installed on the data collection vehicles running in Shanghai and Chongqing. When the collection vehicles are sent to carry out real-scenario testing in designated parking garage, our test engineers will be on site and constantly monitor the performance by conducting integrated system testings. When they have observed deviation or malfunctions, they upload the recorded scenario in the format of bag file with documented time sequence onto the shared problem-solving platform on cloud. Bag file is commonly used in Robot Operating System (ROS) framework. It is extremely powerful and convenient as it supports offline playback at exact time sequence occurred during the session. Because there are multiple sensors installed on the vehicles, I need to examine whether the problem was caused by the Freespace model. Therefore, I will login in to the platform and download the recorded files, replay, make predictions with the Freespace model and visualize the output to see if there are any prediction error and locate the frames where error happens. If there is error, I will relabel the frames which will naturally become the new training data, and retrain the model to adjust performance. If there is no error, I will pass this problem to my colleagues who are responsible for hardware sensor algorithms to further examine the cause for the problem. As explained in the model section, Freespace model takes input of image input of size 320 x 240 but not video. Therefore, the bag files are reproduced in video format, and dissected into one frame per second. Since the original resolution of the cameras are 1920 x 1280, frames are further resized into 320 x 240 to match the input image size. After pre-processing the data, the model can now take the frames as input, and make Freespace predictions.

Visualization As mentioned in the previous section, after model predictions, I need to examine whether the cause for the problem is relevant to the Freespace model or it is due to other hardware algorithms. However, it is almost impossible to inspect the classification results by looking at pure scores. Therefore, it is extremely important to visualize the model predictions. My mentor has written structure of the visualization scripts in python and has walked me through the steps and required knowledge. I have finished and optimized the rest of the script. The first step of visualization is to convert continuous probability into binary masks. Demonstrated

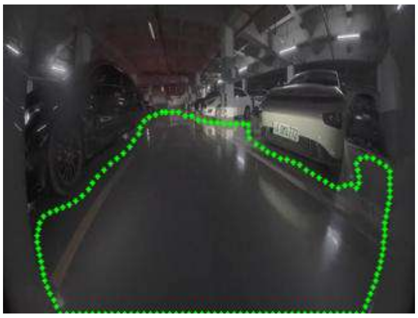

Figure 7: Freespace Contour 1

Figure 8: Freespace Contour 2

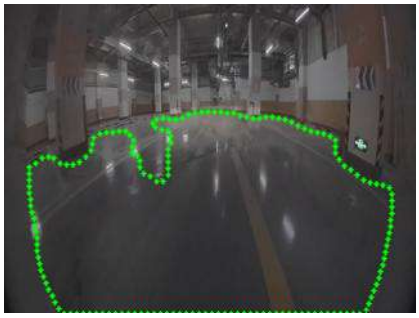

Figure 9: Freespace Contour 3

in Figure 12, the predicted scores for two classes are stored in net.blobs['upscore32_2cls'].data[0] array and a threshold of value 0 is applied to both classes. Values larger than the threshold values are stored as 'True' back in a_0 and a_1 for two classes and vice versa. The next line essentially selects the class 0 or 1 with maximum response at each pixel location and the results would be binary segmentation masks saved as PNG images for later usage. The next step of visualization is to draw the contour lines representing the Freespace on the original image. Demonstrated in Figure 13, the binary masks created in the previous steps will now undergo morphological operations with pre-defined kernels using the opencv library. Erosion and dilation essentially denoise the binary masks, fills the gap within the objects and strengthen the edge of objects, enhancing the quality of the binary masks. Next, Freespace contours are extracted by cv2.findContours from the pre-processed binary masks and drawn on the original image to demonstrate the effect. Figure 7, Figure 8, Figure 9 are visualized contour visualization made from the model predictions. As shown, the model has clearly identified the available path and space while avoiding all the vehicles and pillars in Figure 7. However, one can easily observe that the contour was incorrectly extended on the wall in Figure 8 due to the reflection of headlights from the vehicle, and it failed to identify all the available space in Figure 9 because of ceiling lights' reflection on the ground. Therefore, after analyzing the contour graphs, I will document the cause of error and create a new log on the problem-solving platform to provide feedback to the test engineer and inform the team that the error was caused by the inefficiency of Freespace model on light intensity changes and I will start re-training the model and provide evaluation within a given time period. Also, I will select the original frames where missclassification has happened and send them to the data engineers in the team to relabel the frames that will later be used as training data.

Re-training The re-training process is relatively easy. My mentor has written the re-training script for other projects and sent me a copy to study. I have then recreated one script for re-training the Freespace model. Seen in Figure 15, the re-training script basically sets up the caffe environment and GPU, load the pre-trained weights, adjust the interpolation layer and runs for 40 iterations. After receiving relabeled images from the data engineers, I can now put the original image along with the ground truth label and start re-training the

Freespace model.

Model Validation To evaluate the performance of the model after training with new data, my mentor and I have processed and designed three sets of validation data with specific scenarios: crossing vehicle, crossing pedestrian, and light. Also, We have implemented several metrics to numerically evaluate the model performance: Precision (see Equation 1), Recall (see Equation 2), F1 measure (see Equation 3), IOU (Intersection Over Union, see Equation 4) and mIOU (see Equation 5, where N is the total number of image pairs and IOU_i is the IOU value for i^{th} image pair) which is mean IOU calculated across four surround-view cameras for one scenario. For this part, my mentor has written the framework for the script and I have implemented all the algorithms for the metrics. After running the validation script which compares the predicted images generated from the validation set and the ground truth label, it will output an excel file demonstrating all the metrics shown in figure Table 1. Also, seen in Table 2, a mIOU threshold is applied to set a minimum standard and a value higher than the threshold for the designated scenario is considered pass and can be updated to the overall system. Finally, I will make a presentation on the weekly team meeting to demonstrate this week's improvements with a PowerPoint including tables with figures and visualized contours.

Conclusion After spending two month on the Freespace project, I have completed all the python scripts needed in all process. I have also created a word document that explains the scripts and procedure in details. Until this stage, the Freespace project can be carried out under a standard procedure: receive data, process data, visualization, re-training and validation and the mIOU values has significantly increased because of continuous training at a frequency of twice a week. It is absolutely exciting to utilize the knowledge I have learnt in Deep Learning and Computer Vision course in processing data, modifying the model, writing algorithms and scripts. Besides coding on my own, we have weekly team meetings on every Thursday and Friday afternoon to discuss the problems and progress. I will use this time especially to communicate with our test engineers and hardware engineers about testing strategies and hardware performance. For instance, if I have received increasing number of problems indicating the model's limitation under changes in light intensity, I will specifically negotiate with the test engineer to run system testing under changing light intensity. Also, if the reported problem was not due to Freespace limitation, I will discuss the problem with our hardware engineers to see if there is anything we could compensate the hardware system in modifying the current model. Furthermore, the Freespace model cannot function when there are too many water stains on the floor, causing abnormal light reflection. Under this circumstance, the hardware system will intervene but this requires further training and testing. Apart from the Freespace project, I have also been working on another project involving 3D perception. Because of confidentiality issues, I cannot explain the project in details but my job in this project is to write and test pythons script for processing fish-eye high way data that includes extracting frames from videos and projecting point cloud data onto the extracted frames to overlay the objects. Until this stage, I have finished writing and testing the scripts and start processing the point cloud data while optimizing the scripts to increase execution time.

Evaluation In this placement project, I have applied and learnt many technical skills mentioned above. However, there are non-technical skills that played a vital role in productivity. For instance, during school terms, seldom did I work on multiple projects or tasks simultaneously or in the same day. However, during the placement project, I need to constantly switch between scripts and file directories because there may be multiple problems occurring at the same time and our test engineers and hardware engineers are waiting for results. Therefore, I need to solve multiple instances simultaneously. At the beginning, it was easy to make mistakes and sometimes I have executed the wrong script which accidentally deleted images that are ready to be processed, so I had to download the files from the cloud platform again and restart the whole process. After a few days of multi-tasking, I could handle multiple problems easily. Besides this, I have also learnt from my mentor to create an excel file to document all the data on my hand and the progress. It is extremely useful as it helps keep a record of the status of all files especially which ones are waiting to be returned from the data engineers and which ones have not yet been processed. With the help of this excel record, my efficiency has been further improved in terms of organization.

3 Further Planning and Management

Further thoughts While executing the re-training procedure, I have noticed that a few improvements can be made to enhance the efficiency. First, when examining the cause of testing problem after visualization, it is simple to identity the error with limited number of frames. However, it could be time-consuming when the number of frames are large and if there are multiple problems needed to be solved at the same time. Therefore, I am planing to design and train a small classification model to automatically selects frames with errors from the visualized contour graphs. Moreover, instead of training the model on visualized contour graphs, the model could be directly trained with processed binary masks because the process does not require manual intervention if a vision model is in place and thus does not require visualization step. If this can be accomplished, there will be a huge improvement in efficiency. Speaking of the feasibility of training this classification model, the amount of training data is sufficient as we have accumulated numerous amount of data from the collection vehicles from Shanghai and Chongqing. However, the time cost and resource cost of training a new model should also be taken into account. Also, there are multiple scripts in the process that needed to be executed separately and I am planning to merge all scripts into two scripts: one for data processing and one for re-training to further improve the efficiency.

Deliverable and Planning Referring to Figure 10, I have finished writing the data processing scripts and re-training scripts, marked in light pink. Because our team will hand over the entire system to Changan Automobile in early August and they will conduct internal testings, the main focus at the moment is still solving the small classification errors that occurs and retraining the model, marked in bright pink. The deliverable for this project is the well-trained Freespace model with a standardized re-training procedure. Therefore, I have put this task as my first priority from now to early August. In the mean time, I have already started merging all the scripts and will get this done as soon as possible. Also, I will begin designing and training the small classification model starting in June. Hopefully, I could finish and test this model to be used in speeding up the current procedure. However, even if this cannot be finished before early august, this classification model can still be used in other projects with small modifications. Therefore, I have put this task as my third priority for the Freespace project in the third line. After successfully transferring the system to Changan Automobile, my team and I will focus on continuing monitoring the system performance and make necessary updates and modifications if there is any request. Apart from Freespace, I have the 3D Cloud Point Data project on my hand as well and as I mentioned in the previous section, I have already finished all the scripts, marked in light orange and will be starting processing the data in the following week and this project will be carried out as long as my placement lasts. The deliverable for this project are the script I have finished and a data processing procedure to guide the data engineers. Finally, I will begin another project involved with Pseudo Point Cloud in June and I will begin by studying the code with my mentor. The deliverable for this project is a briefing document which will be used to apply for international patent. If anything above did not go as planned, I will first inform my mentor and supervisor, and then we will hold team meetings to discuss the project progress in details and see if we could solve this internally with more resources or we could asks for help from other teams.

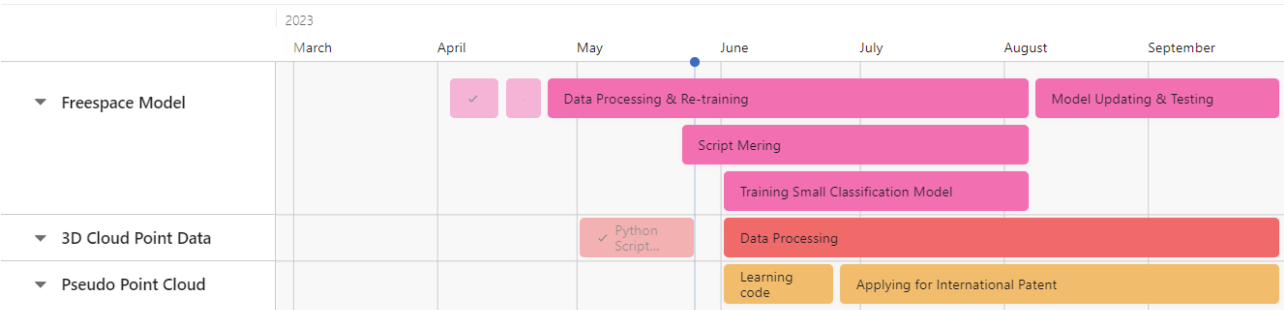

Figure 10: Projects Timeline

4 Reference

References

[1] R. Press. Sae levels of automation in cars simply explained (+image). Available at https://www.rambus.com/blogs/driving-automation-levels/ (2020/06/10).

[2] A. Reiser. History of autonomous cars. Available at https://www.tomorrowsworldtoday.com/2021/08/09/history-of-autonomous-cars/ (2023/04/23).

5 Appendix

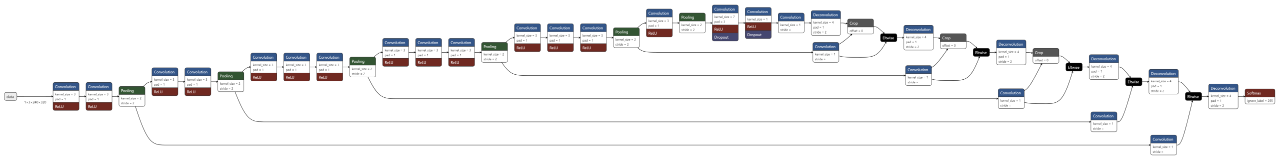

Figure 11: Freespace FCN Model

```python
a_0 = net.blobs['upscore32_2cls'].data[0][0, ...] > threshold
a_1 = net.blobs['upscore32_2cls'].data[0][1, ...] > threshold
a_0 = np.array(a_0, dtype=np.uint8)
a_1 = np.array(a_1, dtype=np.uint8)
out = net.blobs['upscore32_2cls'].data[0].argmax(axis=0)
```

Figure 12: Visualization code snippet 1

```python
erosion = cv2.erode(input, kernel, iterations=1)
dilation = cv2.dilate(erosion, kernel2, iterations=1)
contours, hierarchy = cv2.findContours(dilation, cv2.RETR_TREE, cv2.CHAIN_APPROX_NONE)
cnt = contours[np.argmax([cv2.contourArea(cnt) for cnt in contours])]
cv2.drawContours(img_ori, cnt[0:-1:5], -1, (0, 255, 0), 3, lineType=cv2.LINE_AA)
```

Figure 13: Visualization code snippet 2

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1_{measure} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

$$IOU = \frac{TP}{TP + FP + FN} \tag{4}$$

$$mIOU = \frac{\sum_{i=1}^{N} IOU_i}{N} \tag{5}$$

Metrics	Front	Right	Left	Rear
Precision	0.97401	0.97513	0.97405	0.97306
Recall	0.85065	0.75072	0.72948	0.71076
F1 measure	0.45404	0.42412	0.41708	0.41065
IOU	0.84627	0.74783	0.72635	0.70718

Table 1: Validation Metrics

Date	Target	Method	Channel	mIOU (Crossing vehicle)	Flag
2023/05/12	Freespace	Python	front,left,rear,right	0.756907	Pass

Table 2: Validation Metrics

Figure 14: Model Validation Presentation

```python
setproctitle.setproctitle("200w_small_freespace")

weights = "./model/624836_0_1lr10_iter_8000000.caffemodel"

caffe.set_device(2)
caffe.set_mode_gpu()

solver = caffe.SGDSolver('solver_half.prototxt')
solver.net.copy_from(weights)
interp_layers = [k for k in solver.net.params.keys() if 'up' in k]
surgery.interp(solver.net, interp_layers)

for _ in range(40):
    solver.step(200000)
```

Figure 15: Re-training code snippet 1