# Project of Digital Guide Dog


Dongxun Sun
Umass Amherst
Amherst, MA
dsun@umass.edu

He Zhang
Umass Amherst
Amherst, MA
hezhang@umass.educreate

Hui Sun
Umass Amherst
Amherst, MA
huisun@umass.edu



***Abstract*****—*****This paper is about the design of an aid for visually impaired people. D435 camera sensor is used to collect and obtain picture and video information, which is transmitted to Raspberry PI through USB-C communication for image and video processing to obtain depth map. The depth image can be used to analyze the obstacles and distance in front of the user to indicate the distance between the obstacles in front of the user. The IMU sensor is used to detect the Angle of the camera, and the vibration sensor provides the user with an intuitive indication of the obstacles ahead.***



***Keyword—image process, depth image, auxiliary equipment***


## I. Introduction

There are about 10 million visually impaired people in the US. It always hard for them to go outside to fit in the world with all the inconveniences. Some of them usually go out with guide dogs [1], but it's not easy to train a dog to be a qualified dog[2]. In this project, we would like to offer them an easier way to avoid hitting any obstacles in a path. The idea of the project is to develop a device to assist visually impaired person. According to this idea, the group decide to start using the technique we can access at the university to come up with a product that can actually alert visually impaired person to avoid the obstacle in their way. Currently, the product is a simple connection between the camera and the Raspberry PI, there is a monitor connected to the Raspberry PI for testing purpose. The design of our products is based on wearable devices, which are designed to be used on belts, and its requirements and specifications are the following:

- Required work during day time.
- Long enough to finish a trip. Working time: 2 hours
- Work for person with normal walking form.
- Horizontal FOV: 90deg
- Detect range: 0.3~30 feet
- Light enough for people to easily carry. Weight <= 1.5 lb
- Detect obstacle that are large enough to stop people from walking. Detected object size >= 50 sq inches

## II. Design

### *A. Software Design*

In our design architecture, there are four logical layers which are Hardware layer, Hardware drive interface layer, Core Algorithm layer and User Interface layer respect. The hardware layer, in yellow, Fig.1, is the collection of physical sensors, like cameras, IMU, beepers and so on. The Hardware interface layer, in orange, Fig.1, is the hardware driver layer, it's responsible for interacting with the physical sensors used in our project, including get the raw data from. Sensors and driving the sensors react to the user as feedback. The blue one in Fig.1 is our core algorithm layer, it is in charge of processing the raw data to turn to the depth information we need and to make the final decision for users depending on the processed depth data. The upper green part of Fig.1 is the software interactive layer to communicate with users.

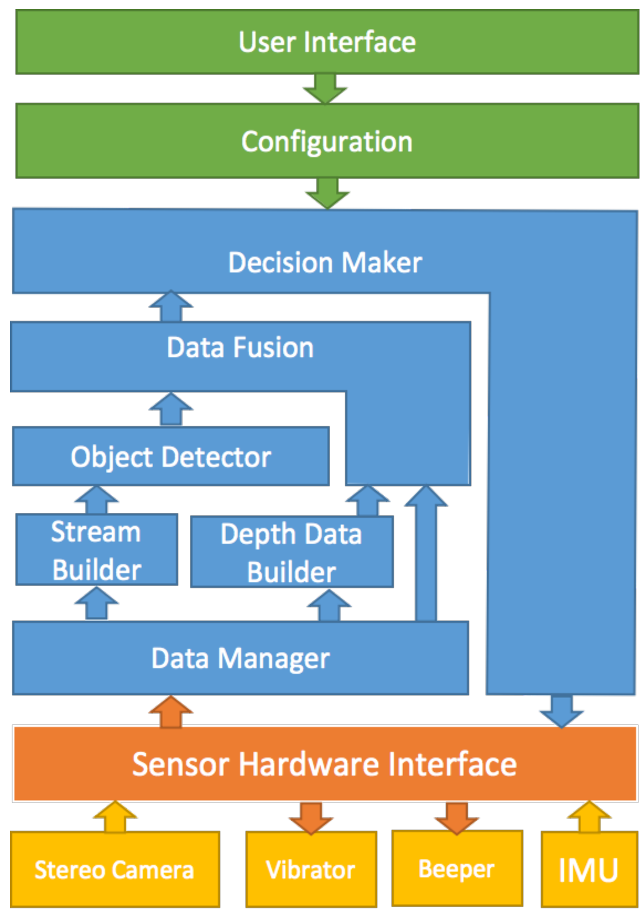


Fig.1. Software architecture

The data flow is as Fig.2, raw data are included 32bit RGB data from sensors cameras in addition to the angular velocity, linear velocity, and orientation come from the sensor of IMU are also included in the raw data package. all of these data passed through the sensor driver, the data manager responsible for allocated the buffer and stored the data streaming, the stream builder and depth data builder are in change of build the RGB frame and Depth frame, after object recognition analyzed, all of the information like the object size, type, and location with depth data will be delivered to the data fusion algorithm module to fuse, and then to the final decision module to generate the feedback which will go back to the user by driving sensors. The application user interface we designed for our product is shown in appendix I.

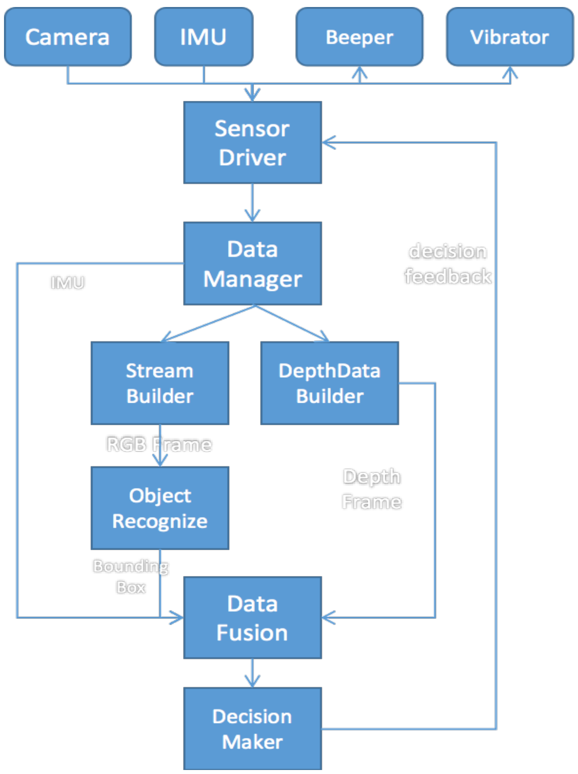


Fig.2. Data flow

*B. Hardware Design*

In terms of hardware design, digital guide dog, as a wearable device, detects obstacles ahead through camera sensor, IMU sensor corrects the tilt state of the device in real time, and uses vibration sensor and sound module to prompt obstacles ahead. Our block diagram like Fig.3.

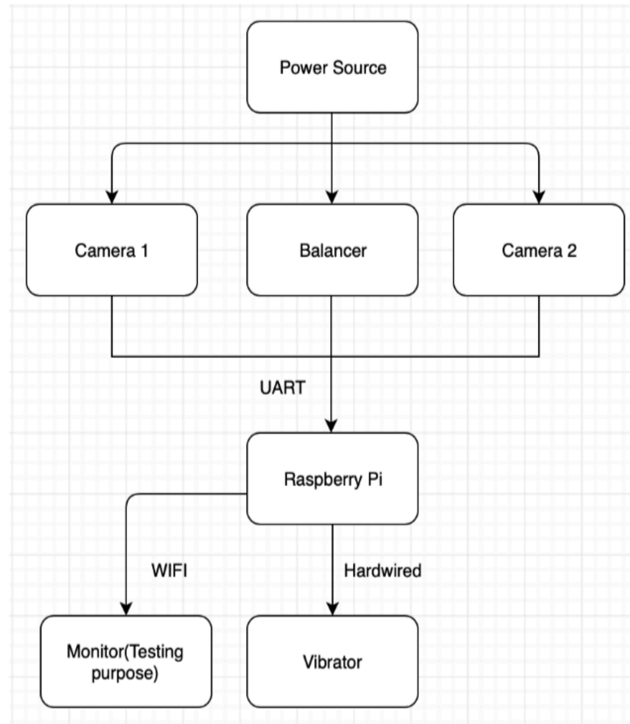


Fig.3. Block diagram

In the selection of main control chip, we used Raspberry Pi 4B as the Microcontroller development board for the whole project. The main advantages of using this development board are[3]:

- Based on the Cortex A-72 processor, 1.5-ghz CPU and 4GB RAM, it can perform a high speed of real-time video acquisition and processing;
- With plenty GPIO, it can respond in real time, switch the working state of the sensor and read the data;
- It is small in size and can be used for power supply during work, which is in line with the lightweight characteristics of wearable devices.

Through the camera sensor to collect photos and video information, we use the integrated Intel D435 industrial camera sensor, like Fig.4. The structure is shown below, with the IR projector and the RGB module and one camera on each side. The combination of a wide field of view and global shutter sensor on the D435 make it the preferred solution for applications such as robotic navigation and object recognition[4]. We will also use this sensor to test the category of obstacles ahead and quantify the probability of object types in the future development.

Fig.4. Intel D435 sensor

Its specification parameters are shown in table 1, and it is well used both indoors and outdoors, with an identifiable distance of nearly 10 meters, and a FOV of 90 degrees. Image and video data transmission can be realized through USB-C, which is the most suitable sensor for our project. Obtain the distance of obstacles within 10 meters in front of the user to ensure that the user has enough time to avoid obstacles.

TABLE 1 THE SPECIFICATION OF INTEL D435

| | | |
|---|---|---|
| **Features** | **Use Environment**<br>Indoor/Outdoor | **Maximum Range**<br>Approx. 10 meters. Accuracy varies de<br>on calibration, scene, and lighting condi |
| | **Image Sensor Technology**<br>3μm x 3μm pixel size | |
| **Depth** | **Depth Field of View (FOV):**<br>Approx. 90° x 60° x 95° | |
| **Components** | **Camera Module:**<br>Intel RealSense D430 + RGB Camera | |
| **Physical** | **Length x Depth x Height**<br>90 mm x 25mm x 25mm | **Connectors:**<br>USB-C |

Through the communication between Intel D435 and raspberry PI development board, the processing and output of picture and video data can be realized. We can now get the D435 sensor to capture the video stream in real time and process it as a depth map video stream. In the later design, we will add IMU sensor to obtain the location of the camera module in real time. When the straight Angle of the camera deviates from the horizontal direction, the product will prompt to correct the position to prevent wrong identification. It will also debug the vibration sensor and the raspberry PI's own sound module for obstacle detection, and the system will give users a hit.

*C. Depth Data Quality Evaluation*

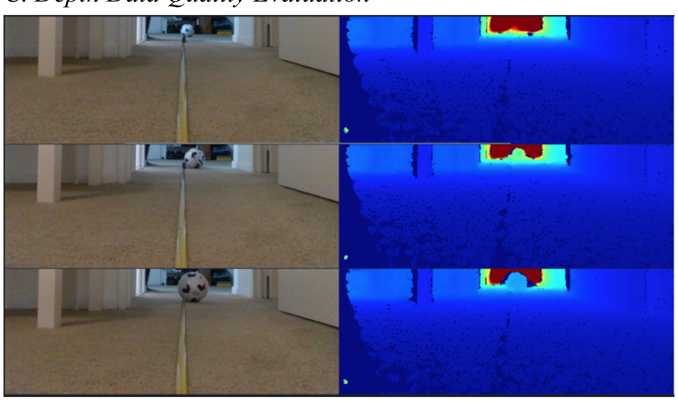

Fig.5 Quality evaluation

We evaluated the quality of depth data we got from our system. The evaluation method is that we used the tape measure to measure the distance between a standard soccer ball to our camera sensor with different sampling distance within 5 meters, and those distance would be our data ground truth, at the same time, we get the distance extract from the depth image captured from our system. compare the data with the ground truth, we can have an error table shown below. You can see from the table, within 3 meters, we got very high accuracy, even the maximum error is around 0.35 in 5 meters, it's satisfying enough for our requirements.

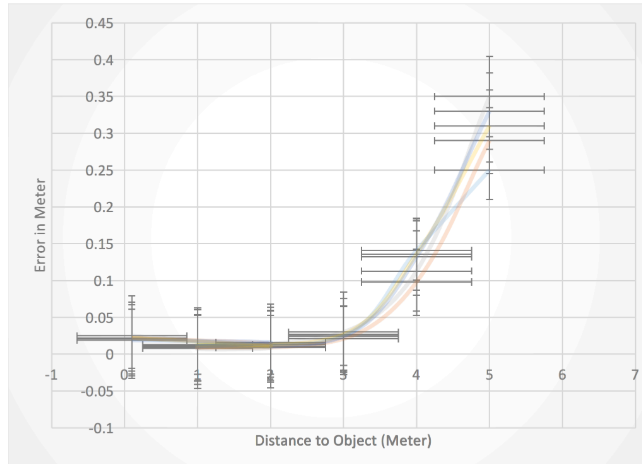


Fig.6. Depth data error

*D. Our wearable design:*

For the combination of components, a wearable device is the best choice for practical use. Our current thinking is similar to placing the camera in front of the user and the power module and microcontroller on one side of the body. Similar to a belt, these components are stuck on the top for fixation. Our design like Fig.7.

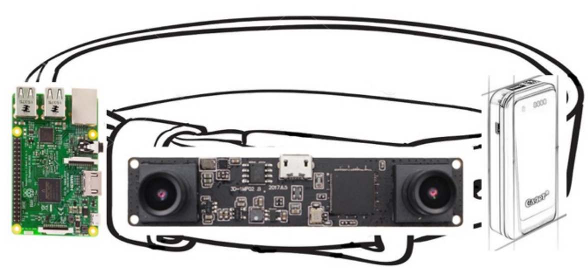

Fig.7. Wearable design

## III. ALGORITHM

The basic 3D reconstruction algorithm is like this below, first we have to calibrate the stereo cameras to find the intrinsic and extrinsic parameters described the location and orientation of two cameras. Calibration is carried out acquiring and processing more than 10 stereo pairs of checkerboard pattern[5]. After calibration, the next step is the rectification which is to remove lens distortions and turn the stereo pair in a regular standard form, then the step of stereo correspondence aims at finding homologous pixels in the stereo image pair. The step of triangulation computing to shifted and matched to generate the depth of each pixel, and get the disparity map [6].

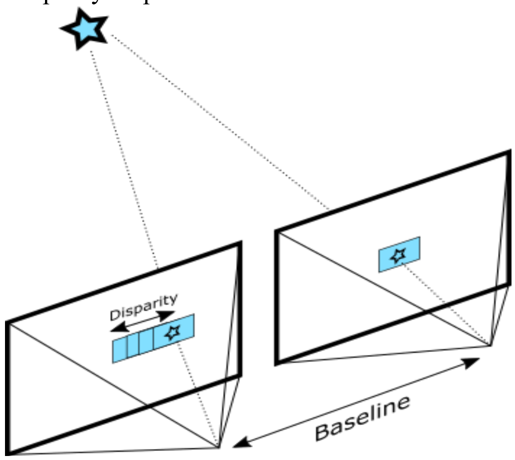


Fig.8. Disparity map

Generate the disparity map which measures the different offset in pixels of two images, pseudo-code is as below[7]:

Create a "minSSD" array equal to the size of the image, with large initial values.

Create a "disparity" array equal to the size of the image.

```
for k = 0 to MAX_SHIFT do
  Step 1: Shift right image to the right by k pixels
  Step 2: Perform Sum of Squared Differences (SSD) between left image and shifted right image
  Step 3: Update the minSSD and disparity array.
      for each pixel coordinate (i,j) do
        if ssd(i,j) < minSSD(i,j) do
          minSSD(i,j) <= ssd(i,j)
          disparity(i,j) <= k
        end
      end
end
```

## IV. Prototype

We made the project website which collected the project idea, detail design block diagram, power point file for presentation and demo video. The website screen shot is shown below, visit the project website by the link:

http://www.ecs.umass.edu/ece688/f19-s20/digital-guide-dog/

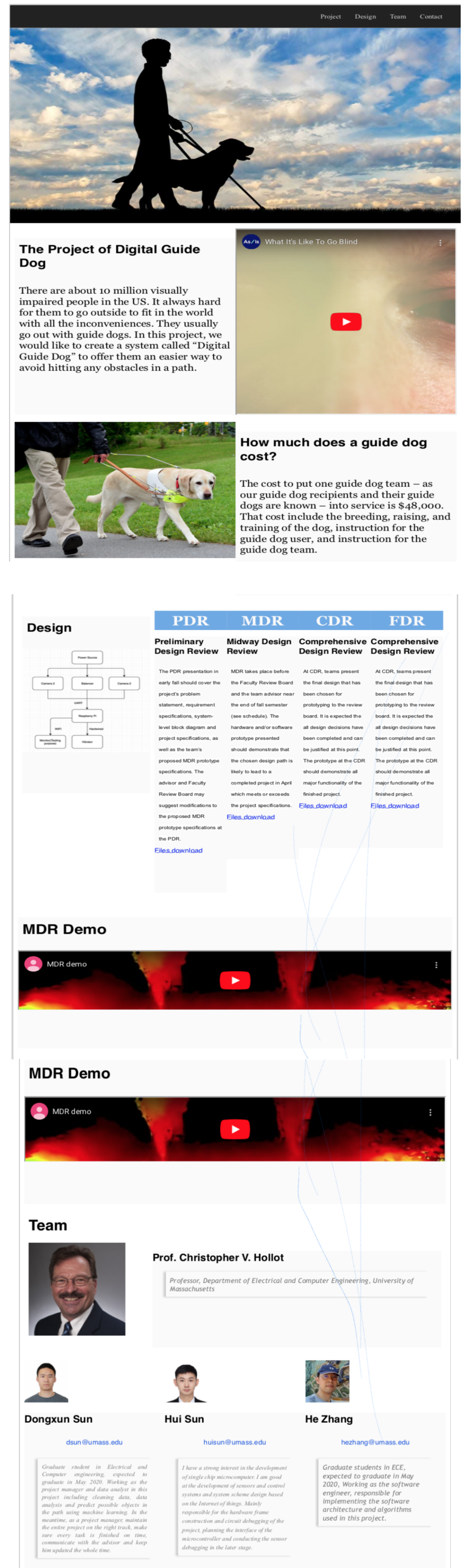


Fig.9. Website page

## V. RESULTS

As for result for this project of this semester, we reached our goal of generating the real time depth video. By using the binocular vision system, we accomplished this goal with the two cameras we hardwired together. All the data came through the two cameras flows into the Raspberry PI, which we used to do all the implementation of the algorithm to, in order for us to get the depth video. Currently, we have the video tested and it looks it met our requirements. The result shows enough fluidity for this stage and the outcome of the data is clear.

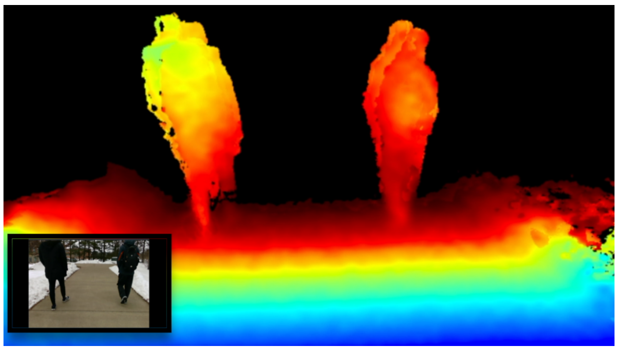

Fig.10. Depth map

## VI. CONCLUSION

In our project, we created the depth video with the live time video at this time. By using Intel camera and the technique from the image processing area, we can get the general distance between the user and the obstacle in order to have the user alerted so they can avoid contact with whatever stands in their way. What we have done this semester is everything we planned by the beginning of the semester, and we already have a rough plan for next semester. We all have the faith to finish the project nice and clean.

## VII. PROJECT MANAGEMENT

In terms of project management, basically, we have the project on track from every spec. The team has a meeting every Thursday, give the weekly update to our advisor, Dr. Christopher V. Hollot. The update usually include what we have done both in software and hardware, and the plan for the next week as well. Prior the PDR and MDR, we both did a review presentation with Dr. Hollot and he always provide very helpful advice.

TABLE 2 DIVISION OF LABOR

| | | |
|---|---|---|
| Dongxun Sun | Project Manager | Contact with the advisor and keep the project on the right track, gathered data from the system for future work, currently have about 1 GB of clean data. |
| He Zhang | Software engineer | Design and implement all the core algorithms |
| Hui Sun | Hardware engineer | Development board and sensor debugging and implementation |

In the future, in order for this project to fully functional, we plan to have machine learning tech implemented to help the system simply identify the object stand in the users' way. We plan to extract all the pixels colored as a human, clean all the data extracted from the data set and use machine learning model to train those pixels to see if we can reach the goal of letting the Raspberry PI to recognize a few objects. We will have our best effort to reach the goal, but according to the feedback we get after MDR, we also understand this can be touch. For the hardware part, we need to have the vibrator working for field test, the vibrator is the key feature for the final test, that is the main alarm to let the user know when there are obstacle stands in the users' way.

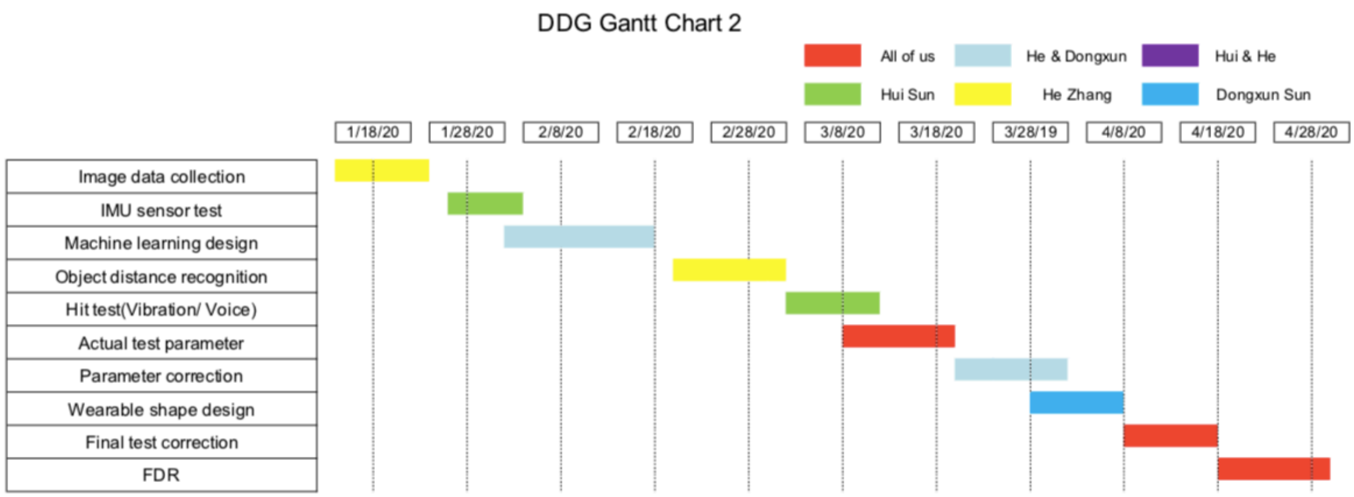


Fig.11. Gantt chart for next semester

# VIII. APPENDIX

Appendix I


Fig.12. DGD application user interface

## IX.References


[1] Morita, C. (2019). *How Much Does A Guide Dog Cost?*. [online] Puppy In Training. Available at: https://puppyintraining.com/how-much-does-a-guide- dog-cost/ [Accessed 16 Dec. 2019].

[2] Servicedogcentral.org. (2019). *How long does it take to train a guide dog? | Service Dog Central*. [online] Available at: https://servicedogcentral.org/content/node/348 [Accessed 16 Dec. 2019].

[3] Guo, T. (2019). [online] Blog.csdn.net. Available at: https://blog.csdn.net/zhufeng88/article/details/75452271 [Accessed 16 Dec. 2019].

[4] Intel.(2019). [online] Available at: http://www.imtelrealsense.com/depth-camera-d435/ [Accessed 16 Dec. 2019].

[5] Mattoccia, S. (2019). [online] Vision.deis.unibo.it. Available at: http://vision.deis.unibo.it/~smatt/Seminars/StereoVision.pdf [Accessed 16 Dec. 2019].

[6] Intel® RealSenseTM Depth and Tracking Cameras. (2019). *The basics of stereo depth vision – Intel® RealSenseTM Depth and Tracking Cameras*. [online] Available at: https://www.intelrealsense.com/stereo-depth-vision-basics/?cid=em-elq-38282&elq_cid=3569199 [Accessed 16 Dec. 2019].

[7] Li, T. (2019). [online] Blog.csdn.net. Available at: https://blog.csdn.net/real_myth/article/details/51206579 [Accessed 16 Dec. 2019].