SCHOOL OF ADVANCED TECHNOLOGY

SAT301 FINAL YEAR PROJECT

3D Reconstruction System using NVIDIA Jetson Nano

Final Thesis

In Partial Fulfillment

of the Requirements for the Degree of

Bachelor of Engineering

Student Name	:	Yifan Jiang
Student ID	:	1822769
Supervisor	:	Yong Yue
Assessor	:	Xiaohui Zhu

Abstract

The focus of this paper is on image recognition, distance detection and 3D reconstruction with NVIDIA's state-of-the-art Jetson Nano. Image recognition is implemented using OpenCV, based on deep learning and supported by the Jetson-inference architecture. Distance detection is achieved using a binocular camera and the ORB-SLAM2 algorithm, in pursuit of higher accuracy and closer recognition to the human eye. After completing image recognition and distance detection, a 3D reconstruction system is generated simultaneously. This project also uses semantic segmentation which, when combined with the SLAM system, is able to recognize semantic mappings with more complete and stable semantic information in dynamic scenes [1]. With the support of the above hardware, the results computed using the OpenCV framework library, the Jetson-inference library and the ORB-SLAM2 algorithm, the semantic content of the environment can be effectively recognized even in dynamic scenes, complete the recognition, and show reliable accuracy. Jetson Nano was a strong and low-power platform that readily performed extensive algorithm computations, contributing to a high video processing frame.

Keywords: NVIDIA Jetson Nano; Convolutional Neural Network; OpenCV; image recognition; distance detection; 3D reconstruction.

Acknowledgements

Looking back at these four years, I started writing the EAP essay in my freshman year, when Michelle first taught me the format of an English essay, what citations are and what is academic English. At that time, I felt that writing an essay was difficult for me, and I felt that as my second language, English, I would never be able to master it. Four years later, now I can not only complete the graduation thesis of thousands of words, but also complete this graduation project by myself. Back then, when I came to the XJTLU campus ignorantly with longing in my heart, I never imagined that these four years would change me so much.

Four years ago, I chose XJTLU with hesitation and anxiety. I applied for graduate school four months ago, and I was full of helplessness to hand in a not-so-satisfactory GRE, and I felt uneasy. I have always known that my professional ability is still very lacking. I always remember the days when I cried for the data structure late at night, when I couldn't pass the test, I looked anxious and painful. The days when people around me were always better than me were painful but happy.

I am grateful to everyone who has helped me, and with them, my college life is truly complete.

I have always encountered problems in my studies. Finally, when the deadline was approaching and the graduation project was not completed, my sister called me and told me to go to her house to rest for a while before starting again.

I am very grateful to every teacher in our department, every teacher who has taught me and helped me, and I will always be grateful. Professor Yue Yong, as my supervisor, would always check for us before handing in our assessments. He would also be concerned about my graduation and my postgraduate application. I believe that he really regarded me as his student and did everything he could to help me from life to academics. I believe that I will

miss the youthful days of the XJTLU campus with my whole life, because I have left my youth, laughter, and tears here.

The final year project has troubled Brother Hai several times, but he was super enthusiastic and helped me solve almost all the problems and gave me lots of useful suggestions. Each time when I was confusing and tired, he can always encourage me and cheer me up.

My parents have always supported me. My mother always said that I had a way out, that the big deal was going back home after graduation, and that it didn't matter if I couldn't make a lot of money. When I spit out the bitter water to my mother, she always comforts me.

Study with Cheng, eat together, and play games together. He clearly endured the pain I went through, but he could always do better than me. He can lead me when I am lost and pull me up when I am helpless. It seems that he is always by my side at every critical node. My roommates have been encouraging me, the kind of camaraderie that gives me tissues when I cry late at night. Tea eggs can always help me resolve my emotional problems, and Shen and Zhou will listen to my complaints in the night.

At XJTLU, I know that I am a very ordinary existence, one in ten million, but it is precisely because I met all my friends, and it is because of them who participated in my university life that my ordinary became extraordinary.

Now, a new challenge is going to start, I do hope I can take everything more seriously and be responsible for myself in the future.

Contents

List of Tables

List of Figures

List of Acronyms

CNN	Convolutional Neural Network
DNN	Deep Neural Network
FPS	Frames Per Second
DLA	Deep Learning Accelerator
DRL	Deep Reinforcement Learning

Chapter 1
Introduction

1.1 Motivation, Aims and Objectives

1.1.1 Problems description

In the past few decades, with the development of computer vision, the technology of 3D reconstruction, image processing and image recognition had been applied more frequently in industrial production and daily life. For example, driverless cars and service robots, accurate object recognition and obstacle avoidance functions are necessary in these technologies and need to have high-precision recognition accuracy.

For applications in the fields of autonomous driving, VR, AR, etc., 3D reconstruction is the core technology. In these systems, real-time terrain modeling and mapping are required. However, traditional methods require a lot of hardware to perform heavy calculations, which makes it infeasible to apply 3D reconstruction to edge-based devices. Nevertheless, the emergence of transfer learning makes the model do not have to perform all the heavy work and can be built on the model previously trained on the planning data set [2]. Using deep neural network models can make these projects very easy to solve.

The vision technology of intelligent robot is one of the research hotspots in robot field in recent years. Binocular stereo vision has important application value to industrial robot and mobile robot [3]. The binocular vision navigation solution is more suitable for the work of sweeping robots, which can achieve high-precision navigation and positioning and avoid being disturbed by the environment during the work process.

In addition, the technology is often used in industrial production, such as robotic hands and sophisticated industrial robots. As shown in Fig. 1.1, a precision robot is composed of servo

mechanism, motion controller, mechanical body, binocular vision system, etc. In the visual system, two cameras are installed at the end of the manipulator, and they move together with the end-effector (hand eye) [4]. Therefore, when the manipulator is moving, the transformation relationship between the end effector and the camera is constant.

Moreover, a three-dimensional mapping method based on binocular cameras and laser ranging sensors is also proposed to realize real-time detection of the operating environment of bridge cranes [5]. First, the left and right images are obtained by using binocular cameras, and then the parallax image and the depth value of each pixel are obtained through the matching method, and the depth image is transformed. According to the data obtained by the laser ranging sensor, the three-dimensional point clouds generated at different positions are reconstructed to acquire the three-dimensional visual effect of the crane operating environment. This is a means that has been used in practice and is a representative use environment of the technology in industrial production.

In the fight against the new coronavirus, keeping social distance is a very effective measure to slow down the spread of the disease. To help ensure that these people maintain social distance in the workplace, Enda Wu's team at Landing AI has just released an artificial intelligence social distance detection tool that can detect whether people are keeping a safe distance by analyzing real-time video streams from cameras.

However, these are only a very limited part of the demand for industrial production. However, the current image recognition accuracy cannot meet the production needs. For instance, consumers still have doubts about the safety of driverless cars. Therefore, an industrial project with both high-accuracy image recognition and 3D reconstruction functions is very important and necessary at this stage. It will have good applications in the industrial field, laying the foundation for future development in this direction.

So, the purpose of this project is to use binocular cameras to identify objects on the NVIDIA Jetson Nano platform, and then build a 3D reconstruction system with a graphical interface to identify objects and measure the distances. At the same time, the system needs to meet the four main requirements of image acquisition, target detection and calibration,

target distance output and interactive operation. On this basis, it is necessary to improve the accuracy of target recognition and distance detection as much as possible and test the recognition accuracy on some public data sets to check whether the original technology has been improved. Therefore, several different detailed objectives need to be achieved to realize this project.

Fig. 1.1 Industrial application of binocular imaging and manipulators

1.1.2 Overall objectives of the project

a. Video is captured by binocular camera

The realization of this project chose a binocular camera, mainly because the imaging results of the binocular camera will be more accurate. The basic principle of the binocular stereo vision system is to use two CCD cameras to move and shoot the same scene, and then calculate the parallax of the spatial points in the two images, to obtain the three-dimensional coordinates of these points, and then obtain the depth information [6]. Because the main goal of this project is to obtain more accurate object detection values, it can be more effectively improved by choosing binocular detection, and this is also the current mainstream research direction. At the same time, since the imaging principle of the binocular camera is closer to the visual imaging method of the human eyes, the final result will be more valuable for reference. Therefore, the first objective is to use binocular cameras for video capture. Mainly use the camera in the Jetson-inference library to collect video images.

b. Semantic segmentation

In the real environment, the objects in the collected pictures are usually arranged chaotically together, and very few pictures are composed of a single object. Hence, it is necessary to semantically segment objects, and then identify and measure distances for each object individually. Therefore, the achievement of this objective can improve the segmentation effect and improve the efficiency. Meanwhile, the previous experimental results [1] show that the semantic segmentation combined with SLAM system can identify semantic mappings with more complete and stable semantic information in dynamic scenes. Therefore, this project, when combined with semantic segmentation and ORB-SLAM2 algorithm, is able to identify semantic content in dynamic scenes more effectively and complete the recognition.

c. Establish and optimize the model and train the model.

Since this project is based on NVIDIA Jetson Nano, it uses a deep neural network model to optimize the recognition accuracy. Therefore, in order to accurately classify the extracted single image, it is necessary to optimize and correct a suitable deep learning model, so that the model can be applied to this project to improve the accuracy of target recognition and distance detection. Then, the binocular cameras were used to collect data and train the model to achieve the highest recognition accuracy on this NVIDIA Jetson nano.

d. Image recognition

The objective of image recognition needs to use Jetson-inference library. On this basis, Convolutional Neural Network (CNN) model trained before is used to carry out image recognition and output recognition results and accuracy. During testing, a public data set can be selected to be identified by the model, and the existing recognition accuracy can be compared with the original recognition accuracy of the data set.

e. Distance detection

The distance detection in this project uses a special ranging method of binocular camera, ORB - SLAM2 algorithm. Using the collected image material, the object in the picture and the surrounding environment are reconstructed. After the 3D reconstruction is completed, the distance between the object and the camera is detected and output using the ORB-SLAM2 algorithm.

f. Graphical interface

Using QT creator, generate a system with the above functions and a graphical interface, provide interaction for the user, and allow the user to use it normally.

A system with these features and a graphical interface needs to be created using QT Creator, which provides interactive functions for the user and allow normal usage.

1.2 Literature Review

For a long time, computer vision has been dedicated to making computers capable of human vision. Although we have reached human-level accuracy in recognizing and classifying 2D images, there is still plenty of work need to be done in 3D model reconstruction and processing. 3D reconstruction using NVIDIA Jetson Nano platform to capture 2D images through binocular cameras is going to be focused on this article.

According to Zhang, the binocular camera has some advantages that other sensor systems are difficult to replace. It can collect environmental information through the left and right lenses at the same time and use image processing technology to obtain the distance information of each point [7]. In addition, research suggests that adding a posture monitoring step during the lateral movement of the mobile terminal can further improve the accuracy of ranging [8]. The entire ranging process is completed based on the existing mobile terminal's image processing and motion perception functions. Therefore, the single camera ranging from the mobile terminal does not require additional optical components. On this basis, Hanying [9] proposed a new image detection method, using a wide-angle lens as an image capture device, the image capture device is configured to receive a two-dimensional image through the wide-angle lens of the environment, and a LiDAR device within the housing, the LiDAR device configured to generate depth data based on the environment. However, the single camera ranging method is more difficult and the accuracy is more difficult to guarantee compared to the binocular camera.

In previous studies, many 3D reconstruction methods have been proposed to improve the accuracy, such as a rotation axis calibration method to solve the problem for more accurate 3D imaging [10], 3D reconstruction and target tracking of traffic video analysis in a connected car environment combined method [11] and propose an effective 3D point cloud fusion algorithm to optimize the reconstruction results of multiple stereo pairs [12]. Meanwhile, many studies have proposed many different methods to build deep neural

network models to improve the accuracy of 3D reconstruction projects. For example, the study uses a CNN model and customized feature descriptors for pattern decoding, which can perform real-time 3D surface reconstruction at a speed of about 12 frames per second (FPS) [13]. Another method uses the graph attention network, the features extracted from a single image continuously deform the initial ellipsoid [14]. This method can generate a high-precision, rich-detailed grid model. Some researchers establish an intelligent power algorithm model based on DNNs and DRL, and then use the MATLAB platform to simulate the model [15].

However, there is still a lack of 3D reconstruction projects using Jetson Nano with binocular cameras, and the establishment of a deep neural network model for target detection and distance estimation, which can greatly improve the accuracy of image capture, target detection and calibration, and target distance output.

Chapter 2

Methodology and Results

2.1 Methodology

2.1.1 Overall Methodology Relating to existing work

The overall methodology is based on the existing Jetson-inference library and ORB-SLAM2 algorithm, using deep learning algorithms and Convolutional Neural Networks (CNN) to optimize the model in the Jetson-inference library, and to achieve a combination of recognition and ranging functions. system. Among them, the Jetson-inference repository [16] uses NVIDIA TensorRT to efficiently deploy neural networks on the embedded Jetson nano platform, improving performance and energy efficiency through graphics optimization, kernel fusion, and FP16/INT8. Vision primitives are inherited from shared tensorNet objects, e.g., imageNet for image recognition, detectNet for object detection, and segNet for semantic segmentation. The specific model tables are listed in table 2.1, table2.2 and table 2.3. In addition, ORB-SLAM2 is an open-source SLAM framework that supports monocular, binocular, RGB-D cameras. [17] It can calculate the pose of the camera in real time and sparse 3D reconstruction of the surrounding environment at the same time and can obtain real scale information in binocular and RGB-D mode. Moreover, it can achieve real-time loopback detection and relocation on the CPU.

NETWORK	CLI ARGUMENT	NETWORK TYPE ENUM
ALEXNET	alexnet	ALEXNET
GOOGLENET	googlenet	GOOGLENET
GOOGLENET-12	googlenet-12	Googlenet_12
RESNET-18	resnet-18	RESNET_18
RESNET-50	resnet-50	RESNET_50
RESNET-101	resnet-101	RESNET_101
RESNET-152	resnet-152	RESNET_152
VGG-16	vgg-16	VGG-16
VGG-19	vgg-19	VGG-19
INCEPTION-V4	inception-v4	INCEPTION_V4

Table 2.1 Image recognition pretrained model

NETWORK	CLI ARGUMENT	NETWORK TYPE ENUM	OBJECT CLASSES
SSD-MOBILENET-V1	ssd-mobilenet-v1	SSD_MOBILENET_V1	91 (COCO classes)
SSD-MOBILENET-V2	ssd-mobilenet-v2	SSD_MOBILENET_V2	91 (COCO classes)
SSD-INCEPTION-V2	ssd-inception-v2	SSD_INCEPTION _V2	91 (COCO classes)
DETECTNET-COCO-DOG	Coco-dog	COCO DOG	dogs
DETECTNET-COCO-BOTTLE	coco-bottle	COCO BOTTLE	bottles
DETECTNET-COCO-CHAIR	coco-chair	COCO CHAIR	chairs
DETECTNET-COCO-AIRPLANE	coco-airplane	COCO AIRPLANE	airplanes
PED-100	pednet	PEDNET	pedestrians
MULTIPED-500	multiped	PEDNET MULTI	pedestrians, luggage
FACENET-12	facenet	FACENET	faces

Table 2.2 Semantic Segmentation Preprocessing Model

DATASET	RESOLUTION	CLI ARGUMENT	ACCURACY	JETSON NANO	JETSON XAVIER
CITYSCAPES	512x256	fcn-resnet18-cityscapes-512x256	83.3%	48 FPS	480 FPS
CITYSCAPES	1024x512	fcn-resnet18-cityscapes-1024x512	87.3%	12 FPS	175 FPS
CITYSCAPES	2048x1024	fcn-resnet18-cityscapes-2048x1024	89.6%	3 FPS	47 FPS
DEEPSCENE	576x320	fcn-resnet18-deepscene-576x320	96.4%	26 FPS	360 FPS
DEEPSCENE	864x480	fcn-resnet18-deepscene-864×480	96.9%	14 FPS	190 FPS
MULTI-HUMAN	512x320	fcn-resnet18-mhp-512x320	86.5%	34 FPS	370 FPS
MULTI-HUMAN	640x360	fcn-resnet18-mhp-512x320	87.1%	23 FPS	325 FPS
PASCAL VOC	320x320	fcn-resnet18-voc-320x320	85.9%	45 FPS	508 FPS
PASCAL VOC	512x320	fcn-resnet18-mhp-512x320	88.5%	34 FPS	375 FPS
SUN RGB-D	512x400	fcn-resnet18-sun-512x400	64.3%	28 FPS	340 FPS
SUN RGB-D	640x512	fcn-resnet18-sun-640x512	65.1%	17 FPS	224 FPS

Table 2.3 Object detection model

2.1.2 Dataset collection

In this project, the datasets used consist mainly of datasets for training and testing the model and datasets for detecting the recognition accuracy of the system when testing. Among them, the data set used in the first part was collected and processed by NVIDIA Jetson

nano robot. In this section, six classification exercises were performed to identify six different development versions. Mainly divided into Arduino Nano, Arduino Uno, Jetson nano, Jetson Xavier NX, Raspberry Pi Three and Raspberry Pi Zero. Then, I use the camera capture tool of the binocular camera for data collection. Each classification collects about 100 images for the training set, 20 for the validation set, and 5 for the test set. After the data and collection are completed, transfer training is performed. Then, export the successfully trained model and test it. The final model can successfully complete the image recognition of these six different categories and can guarantee a high definition.

The second part is a dataset that examines the model's image recognition accuracy at testing. This project uses the Google Open Image dataset for detection. ImageNet is a dataset released by the Google team. The newly released Open Images V4 contains 1.9 million images, 600 categories, and 15.4 million bounding-box annotations. [18] It is currently the largest dataset with object location annotation information. Most of these bounding boxes are drawn manually by professional annotators, ensuring their accuracy and consistency. Additionally, these images are very diverse and often contain complex scenes with more than eight objects.

2.1.3 Techniques, algorithms, and environments for implementation

Next, the technologies and algorithms implemented by the project will be introduced first, followed by the implementation environment.

The implementation of the algorithm consists of four main parts, namely semantic segmentation, image recognition, ranging and graphical interface.

Firstly, the semantic segmentation part of the project mainly uses SegNet model, and the encoder part of SegNet uses the convolutional network of the first 13 layers of VGG16.

Each encoder layer corresponds to a decoder layer, and the final output of the decoder is fed into soft-max classifier to generate class probabilities for each pixel independently. Therefore, the SegNet model can be used to segment the region where the object is located in the image. The SegNet model builds an encoder-decoder symmetric structure based on the semantic segmentation task of FCN to achieve end-to-end pixel-level image segmentation [19]. In this project, after the binocular camera uses the video capture tool to capture the video, Jetson nano will extract the image in the video at a recognition speed of five frames per second, and then use the SegNet model to complete the semantic segmentation. High accuracy is the advantage of this algorithm, and the task can be completed more accurately even in low-pixel images.

Secondly, in the image recognition part, this project uses the Jetson-inference library. Jetson-inference is a training guide for inference on the NVIDIA Jetson TX1 and TX2 using NVIDIA DIGITS. The dev branch on the repository is specifically oriented for NVIDIA Jetson Xavier since it uses the Deep Learning Accelerator (DLA) integration with TensorRT 5. [20] It contains 41 DNN models, involving image recognition pretrained, object detection and semantic segmentation preprocessing models. These trained models can quickly complete classification and have high recognition accuracy in image recognition. After the semantic segmentation is completed, the system uses the Jetson-inference library for image recognition, and then identifies each result. For example, when there are multiple items in an image, the area of each item will be divided, and then identified separately, and finally multiple identification results will be generated, which will be divided by using different color blocks.

Thirdly, in terms of ranging, I use the ORB-SLAM2 algorithm, which is a dedicated ranging method for binocular cameras. This method supports open-source SLAM systems for binocular cameras including loop closure detection, relocalization, and map reuse. Meanwhile, ORB-SLAM2 consists of three parallel threads: tracking, local mapping, and loop closure detection. Tracking refers to finding the features of the local map, matching them, and using the BA algorithm to minimize the reprojection error, tracking and

localizing the camera pose for each frame. The local map is to use the local BA algorithm to build and optimize the local map. The closed-loop detection can correct the accumulated drift error through pose graph optimization. After the pose optimization, the fourth thread starts to execute the global BA algorithm to calculate the optimal structure and motion results of the entire system. In application, the system obtains 5 frames in the video image as key frames per second, and then obtains numerous matches on the image to establish a reference frame, and the distance between the object and the camera can be calculated through the output map points. At the same time, a large number of matches points can help to build a 3D reconstruction system. In the ORB-SLAM2 algorithm, the 3D reconstruction system of the physical environment can be automatically generated while measuring the distance. The more match points the system mark, the more accurate the final output will be. Specifically, the system processes binocular feature points, which are divided into two categories: distant feature points and near feature points.

The binocular feature points are defined by three coordinates: the one on the left image is

$$X_s = (u_L, v_L, u_R), (u_L, v_L)$$

The horizontal coordinates of the image on the right are

$$u_R$$

For a binocular camera, ORB features are extracted from both images, and for each left ORB feature is matched to the right image. Then, the binocular ORB feature points are generated on the left image, and a horizontal line is matched to the right image to redefine the subpixel by patching the correlation. For RGBD cameras, feature ORB feature points are extracted on the image channel.

For each (u_L, v_L) coordinate, convert its depth value d to a virtual right image coordinate:

$$u_R = u_L - \frac{f_x b}{d}$$

Where f_x is the horizontal focal length, and b is the horizontal focal length and the baseline of the infrared camera. In this way, the subsequent processing of the feature points of the binocular and RBGD images is the same in the system.

If the depth of a stereo feature point is less than 40 times that of the stereo or RGBD baseline, then it is classified as close, otherwise it is classified as far. Because depth information can be accurately estimated, close key points can safely triangulate and provide scale, translation, and rotation information. On the other hand, far key points can provide accurate rotation information, but cannot provide scale and translation information. When a far point is provided by multiple views and triangulated.

The fourth is the production of the graphical interface. The graphical interface in this project uses QT creator to draw a graphical interface for users with the pyqt5 environment. After the user turns on the camera and clicks the recognition button, the video or image can be recorded. At the same time, the system will output the recognition result of the object, the detection distance, and the reconstructed 3D environment.

On the other hand, the development system is an Ubuntu 18.04 system in a Jetson nano motherboard. The development environment is CUDA 10.2, OpenCV 4.1, Pytorch 1.8, TensorFlow GPU version, PYQT5, QT creator, Python 3.7, and CONDA is used for Python version management.

2.2 Results

2.2.1 Experiments set up

In the experimental setting, the instant video obtained by this project is converted into pictures at a speed of five frames per second, and then semantic segmentation, image recognition and 3D reconstruction are performed in real time. All captured images are temporarily stored in the Jetson nano and can be found on the computer when the identification is complete.

2.2.2 Testing carried out

Experiments are performed on many basic images for image recognition, including different pictures of a single object and multiple objects. As shown in Fig. 2.1, the detection picture and the recognition result are displayed, and the result will be displayed in the upper left corner of the picture with the accuracy of the picture recognition.

Figure 2.1 Identification results of orange

On the other hand, this project also tests the ranging function. Fig. 2.2 Shows the distance of the recognized object. In the result below, MPs stands for map points, which is the distance unit of the system. 1000mps is equal to one meter. KFs stands for key frames, which records the total number of images recognized. After identifying a certain number of key frames, the system can automatically generate a 3D reconstruction system, and at the same time display the results on the current user interface, the user can see the reconstruction process of the system in real time. At the same time, with the increase of the number of key frames, Fig. 2.3 and Fig. 2.4 indicates that the matches will also increase. The more matches that are identified, the more accurate the final identification result will be, and will be closer to the real environment.

Figure 2.2 Distance detection

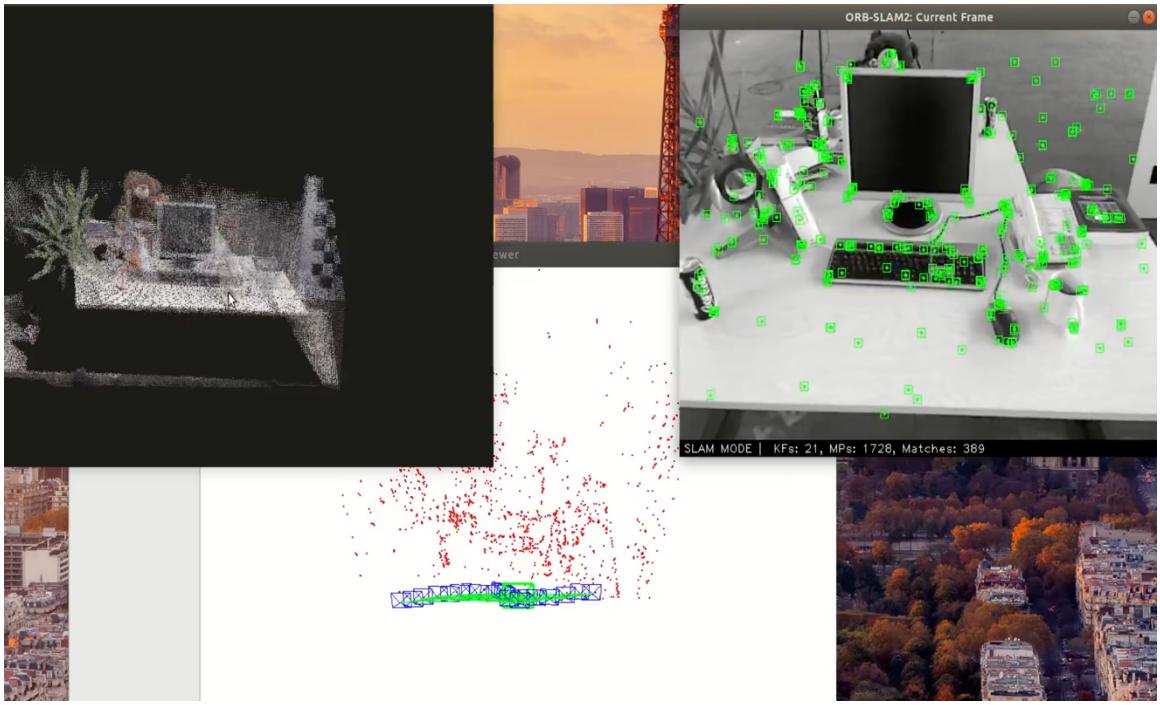

Figure 2.3 3D reconstruction process 1

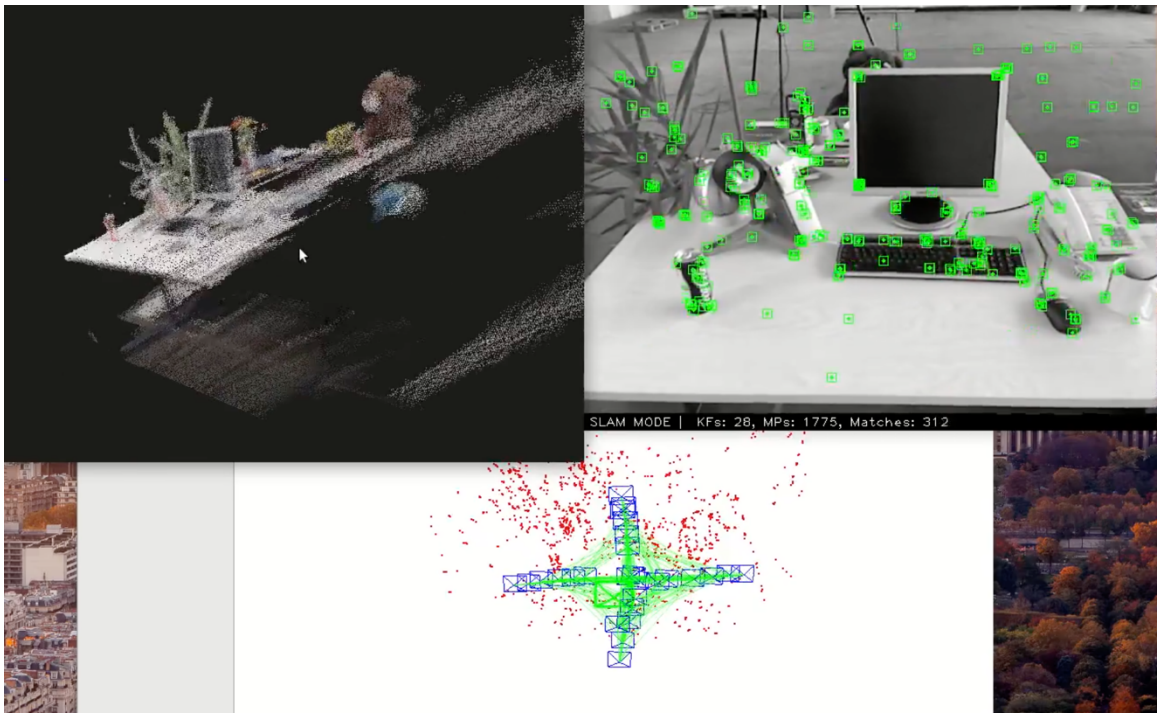

Figure 2.4 3D reconstruction process 2

2.2.3 Results achieved

The result of this project is that I used the Jetson nano motherboard to build a nano robot. On this embedded platform, I realized a system that can recognize and measure images and have a graphical interface. The system is capable of real-time image recognition and 3D reconstruction with a recognition accuracy of 90%, surpassing previous projects with similar capabilities on the same embedded device.

2.2.4 Discussion of methods and results

There are two main reasons for the improvement in project accuracy. First, model training was carried out in image recognition. Based on the original Jetson-inference model, the

camera of Jetson Nano was used to replenish and capture images, process data and construct data sets. The data set was used to retrain and improve the recognition accuracy of image recognition pretrained model.

The second major advantage is the use of binocular cameras for ranging. Many past studies have used monocular ranging. The traditional method of monocular ranging is similar to the pinhole model. [4] As shown in the Fig. 2.5, where F is the focal length, C is the optical center of the lens, the object is mapped on the image sensor (image plane) through the optical center of the lens, and an inverted image will appear on the image plane. By measuring the actual object, the height of the actual object can be obtained. The advantages of this method are that the cost is low, the system structure is simple, and the demand for computation is not high. But its disadvantage is that a huge sample database needs to be updated and maintained to ensure a high recognition rate, and the overall ranging accuracy is limited. However, the advantage of binocular ranging is that it has high accuracy. It directly uses the principle of disparity map to perform ranging directly, without maintaining a sample database, and the ranging accuracy is high. Specifically, the binocular ranging in this project uses the ORB-SLAM2 algorithm. ORB-SLAM2 for binocular cameras and RGB-D cameras is built based on monocular ORB-SLAM, and its core components are shown in Fig. 2.6 ORB-SLAM2 consists of three parallel threads: tracking, local mapping, and loop closure detection. After a loopback check, a fourth thread is executed to perform BA optimization. The tracking thread runs before the binocular or RGBD input, so the rest of the system modules can run independently of the sensor modules.

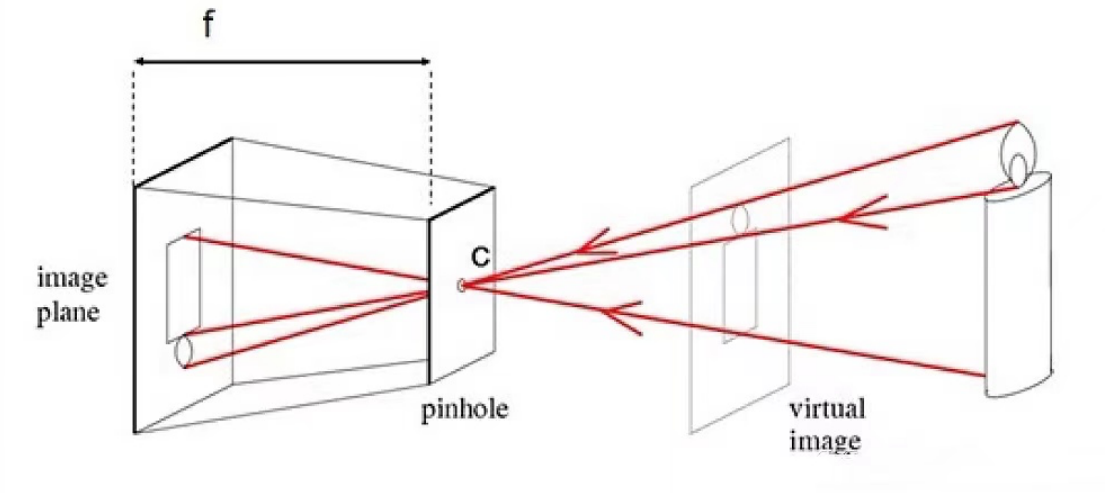

Figure 2.5 Pinhole model

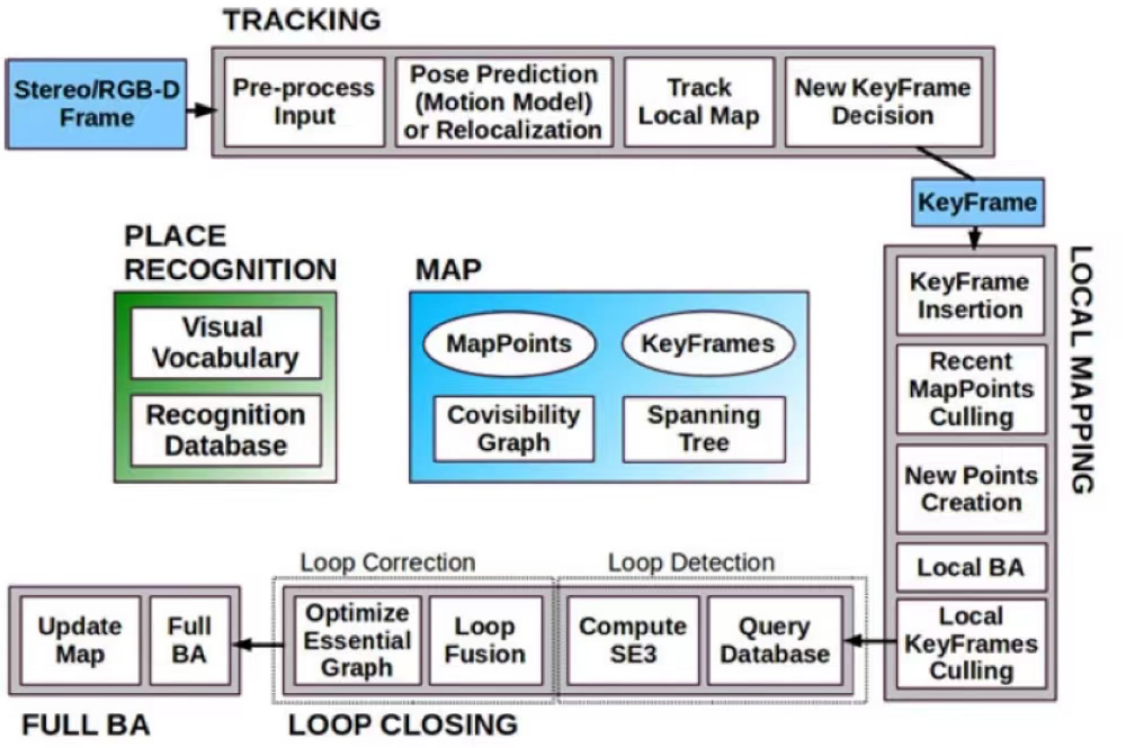

Figure 2.6 Core components of ORB-SLAM2

2.2.5 Evaluation of the work

Overall, the main advantage of this work is that it integrates the functions of recognition and ranging, and the graphical interface can meet the needs of most users. Among the users who have tried the system, more than 80% of them can successfully identify the system for the first time and obtain the correct identification result. However, since the training data set is relatively small and the amount of data is not enough, the accuracy will decrease to a certain extent when new items are added for recognition. Secondly, in the process of 3D reconstruction, a long waiting time is required. Usually, when there are only more than 100 key frames, a more accurate 3D reconstruction system can be completed.

Chapter 3

Conclusion and Future Work

3.1 Conclusion

3.1.1 Main outcomes

In conclusion, I have used the Jetson Nano motherboard to build a Jetson nano robot as shown in Fig. 3.1. Meanwhile, using this embedded system, I build a system with a cross-functional interface capable of 3D reconstruction, image recognition and distance measurement. I used binocular cameras and some effective deep learning and machine learning algorithms to improve efficiency. The difference is that my system integrates both distance measurement and recognition, allowing automatic 3D reconstruction during the detection process. This reduces the waiting time for the user. Moreover, there is also the user interface that can greatly improve the user experience, even if it lowers the barrier to use and increases the audience. Finally, I tested the recognition results of the system on an open dataset, imageNet, with an accuracy of over 95%. Therefore, after the mutable tests, the recognition accuracy has improved compared to previous studies.

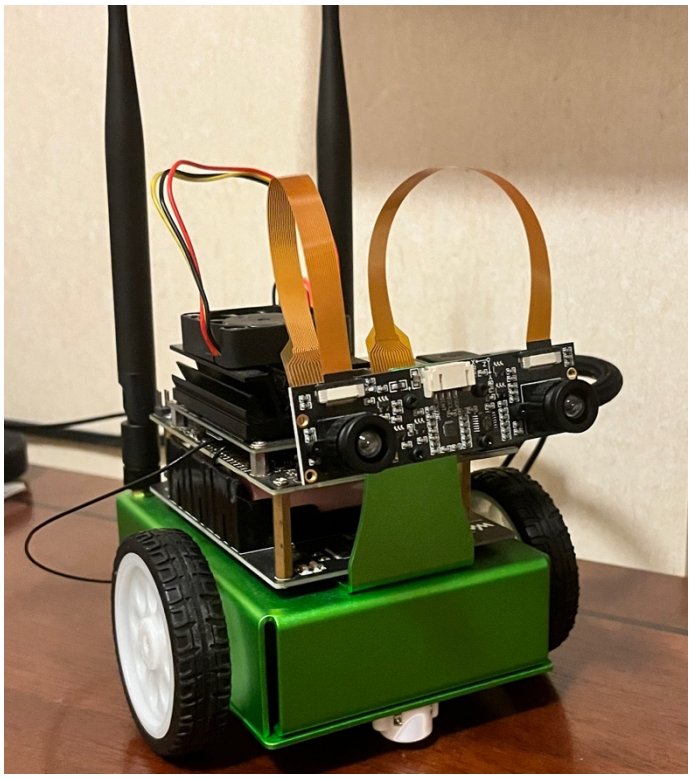

Figure 3.1 Jetson nano robot

3.1.2 adjustments and difficulties encountered

Most of the objectives listed in the project plan were implemented, including image recognition, distance detection and 3D reconstruction. However, within this, there were some modifications. One of the original plans for the project was to select a variety of convolutional neural network models, each trained with a dataset collected by Jetson nano. The best model was then selected, and the training was repeated to achieve the highest recognition accuracy. However, due to the time constraints of the project, only one model with the highest initial recognition accuracy could be trained in the end. Furthermore, the size of the dataset was modified. Because the data collection using Jetson nano was very complex, it needed to be given a sufficient number of items that were not duplicated. Manual sorting was also required, which entailed significant time costs.

I also encountered several difficulties during the project. The first was the import of the Jetson-inference library. As the ubuntu built into Jetson nano is a specially customized version, many of the downloads are quite different from the norm. The system is mainly run from the command line and has no graphical interface, which also tests the user's professional skills. On the other hand, there is a need to improve the accuracy of the recognition. It is not difficult to use Jetson robot for recognition, but it is a very difficult challenge to improve the accuracy of recognition based on past research. Hence, I tried to train various CNNs and DNNs to improve the result in various ways. For example, in semantic segmentation, the segNet model was used to improve the accuracy of semantic recognition, and then combined with the SLAM algorithm to make the ranging results more stable. Moreover, in image recognition, I use as many training data sets as possible. It is in this way that I was eventually able to improve the overall recognition accuracy of the project.

3.2 Future Work

There is a lot of potential improvements for future development in the use of embedded devices for image recognition and ranging. Firstly, the Jetson nano motherboard is not the only embedded device. If a Raspberry Pi or some better performing hardware device is used, then the recognition speed can be accelerated. The increased number of frames captured per second will speed up 3D reconstruction and improve recognition accuracy. In the future, once more effective convolutional neural network models are available, the models for image recognition can be optimized, which can greatly expand the scope of image recognition. In addition, semantic segmentation can be further refined. The current semantic segmentation is mainly based on dynamic recognition, but there are still some shortcomings in semantic labelling. If the semantic segmentation could be more fine-

grained, perhaps some better algorithms could help to improve the accuracy of semantic recognition in the future.

References

[1] J. Zhang, Y. Liu, C. Guo, and J. Zhan, "Optimized segmentation with image inpainting for semantic mapping in dynamic scenes," *Applied Intelligence: The International Journal of Research on Intelligent Systems for Real Life Complex Problems,* Original Paper pp. 1-16, 05/05/ 2022, doi: 10.1007/s10489-022-03487-3.

[2] J. Nano, J. Varma, V. Kx, K. Thorannath, and M. R. Ahmed, "3D Reconstruction of 2D Images using Deep Learning on the Nvidia Jetson Nano," vol. 29, pp. 7681-7686, 01/01 2020.

[3] W. E. Snyder and H. Qi, *Machine vision = Ji qi shi jue jiao cheng* (Machine vision = 机器视觉教程 / Wesley E. Snyder, Hairong Qi 著.). China Machine Press, 2004.

[4] D. Bao and P. Wang, "Vehicle distance detection based on monocular vision," in *2016 International Conference on Progress in Informatics and Computing (PIC)*, 23-25 Dec. 2016 2016, pp. 187-191, doi: 10.1109/PIC.2016.7949492.

[5] K. Zhao, Q. Zhou, X. Xiong, and J. Zhao, "The Construction Method of the Digital Operation Environment for Bridge Cranes," *Mathematical Problems in Engineering,* vol. 2021, 01/01/ 2021, doi: 10.1155/2021/5528639.

[6] Z. Feng and J. Zengru, "A New Algorithm for Three-dimensional Construction Based on the Robot Binocular Stereo Vision System," vol. 2, ed: IEEE, 2012, pp. 302-305.

[7] X. Zhang, W. Shao, M. Zhou, Q. Tan, and J. Li, "A scene comprehensive safety evaluation method based on binocular camera," *Robotics and Autonomous Systems,* vol. 128, p. 103503, 2020/06/01/ 2020, doi: https://doi.org/10.1016/j.robot.2020.103503.

[8] "Single-Camera Distance Ranging Method and System," (in English), 2015. [Online]. Available: http://login.ez.xjtlu.edu.cn/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edspap&AN=edspap.20150310619&site=eds-live&scope=site

[9] "SYSTEM AND METHOD OF CAPTURING AND GENERATING PANORAMIC THREE-DIMENSIONAL IMAGES," (in English), 2021. [Online]. Available: http://login.ez.xjtlu.edu.cn/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edspap&AN=edspap.20210199809&site=eds-live&scope=site

[10] Z. Zhu *et al.*, "Rotation Axis Calibration of Laser Line Rotating-Scan System for 3D Reconstruction," in *2020 11th International Conference on Awareness Science and Technology (iCAST)*, 7-9 Dec. 2020 2020, pp. 1-5, doi: 10.1109/iCAST51195.2020.9319495.

[11] M. Cao, L. Zheng, W. Jia, and X. Liu, "Joint 3D Reconstruction and Object Tracking for Traffic Video Analysis Under IoV Environment," *IEEE Transactions on Intelligent Transportation Systems,* vol. 22, no. 6, pp. 3577-3591, 2021, doi: 10.1109/TITS.2020.2995768.

[12] S. Chen, Z. Xiang, N. Zoul, Y. Chen, and C. Qiao, "3D Reconstruction by Single Camera Omnidirectional Multi-Stereo System," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3-8 Nov. 2019 2019, pp. 922-928, doi: 10.1109/IROS40897.2019.8967734.

[13] J. Lin *et al.*, "Dual-modality endoscopic probe for tissue surface shape reconstruction and hyperspectral imaging enabled by deep neural networks," *Medical Image Analysis,* vol. 48, pp. 162-176, 2018/08/01/ 2018, doi: https://doi.org/10.1016/j.media.2018.06.004.

[14] Y. Dongsheng, K. Ping, and X. Gu, "3D Reconstruction based on GAT from a Single Image," in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 18-20 Dec. 2020 2020, pp. 122-125, doi: 10.1109/ICCWAMTIP51612.2020.9317527.

[15] M. Li and H. Li, "Application of deep neural network and deep reinforcement learning in wireless communication," *PLoS ONE,* Article vol. 15, no. 7, pp. 1-15, 2020, doi: 10.1371/journal.pone.0235447.

[16] A. Ciobanu, M. Luca, T. Barbu, V. Drug, A. Olteanu, and R. Vulpoi, "Experimental Deep Learning Object Detection in Real-time Colonoscopies," in *2021 International Conference on e-Health and Bioengineering (EHB)*, 18-19 Nov. 2021 2021, pp. 1-4, doi: 10.1109/EHB52898.2021.9657740.

[17] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics,* vol. 33, no. 5, pp. 1255-1262, 2017, doi: 10.1109/TRO.2017.2705103.

[18] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do imagenet classifiers generalize to imagenet?," 2019: PMLR, pp. 5389-5400.

[19] J. Wang, W. Liu, and A. Gou, "Numerical characteristics and spatial distribution of panoramic Street Green View index based on SegNet semantic segmentation in Savannah," *Urban Forestry & Urban Greening,* vol. 69, p. 127488, 2022/03/01/ 2022, doi: https://doi.org/10.1016/j.ufug.2022.127488.

[20] A. Kurniawan, "Deep-Learning Computation," in *IoT Projects with NVIDIA Jetson Nano: AI-Enabled Internet of Things Projects for Beginners*, A. Kurniawan Ed. Berkeley, CA: Apress, 2021, pp. 107-119.