Hello everyone, the topic of my pre today is Bionic Mechanical Goose Based on Water Surface SLAM Mapping. We named this project as GOOBOT. Because the solution we finally chose is a bionic goose-shaped robot, so we named the robot GOOBOT, Also it means good robot.

**Firstly, I want to introduce some initial** inspiration **and goals of the project.**

One day, Professor Zhu was walking by the Weiming Lake and thought: Can we make a type of water robot to help connect garbage and monitor water quality？So our project was born. We want to complete a water robot that can precisely control its actions and be intelligent enough to accomplish some tasks.

**Now, let’s learn some basic features of the robot.**

First of all, the physical structure, it must be able to float on the water, as it is a water robot, also it needs to have robustness in some scenes.

In terms of movement, it should be able to precisely control its own posture, speed, and position to achieve autonomous cruises according to the planned route.

And then information gathering, Before autonomously planning a path, it is also necessary to have the ability of learning the surrounding environment, such as:

- using its Visual or laser to gather information of the water surface and bank, processing the data and building models, and then autonomously plan a definite cruise route to complete the task expected

We also want the robot to be bionic, which means it can perfectly blend into the natural surroundings, without the feeling that it stands out or is disharmonious

-This requires our robots to be smaller, emit less noise when running, and have an appearance similar to some animals.

- At the same time, in order to be enjoyable, it can also interact with people and complete some funny actions.

Refering to Professor Zhu’s requirements for amphibious, bionic fins or other mechanical structures may be added in the future.

**Currently we divide the project into six parts,**

**First of all,** **we conducted a detailed investigation,** however, there are few successful examples of small surface robots. Most of them are for large surface robots or underwater robots.

**So we independently design the shape and physical structure of the goose**

We place the center of buoyancy directly above the center of gravity, to ensure that its slight sideways will not cause the instability of the overall system, otherwise it may lead to overturning.

-The belly of the goose is hollowed out, There are battery storage unit and development boards. Two propeller motors are symmetrically equipped under the abdomen of the goose, used for the power of the movement.

-The neck of the goose is equipped with a steering gear, which can complete the rotation of the gooseneck.

-And the head of the goose is equipped with a camera, responsible for collecting visual information of the environment.

Of course, the current bionic robot goose still has large space left to install more sensors and other equipment

**This picture shows the electrical system design of goobot.**

At present, there are three types of sensors, transmitting the external information into the system.

The remote control can control the goose in case of some special events, such as if the robot failed to conduct the right movement, we can set the manual control mode.

The JY61 gyroscope is responsible for detecting the posture, and the detecting results are passed to the Raspberry as the input of the PID.

The camera is responsible for taking real-time images for Slam.

We choose robomaster as the motion control board, to convert motion information into PWM wave output, so as to control the speed of the motor and complete the control of behavior.

The Raspberry serves as the upper board, receiving gyroscope information, doing PID control, and outputs it to the robomaster board for execution.

The real-time picture is transmitted to the server through the socket, and the server builds the map in real time.When the analyzing and map-building is done, the server can send instructions to guide the goose's direction or cruise route.

**The next stage is to realize the communication between various modules, so that they can be linked as a whole.**

We can also look at this picture. Except that the communication between the Raspberry and the server is still being debugged, the other communication functions have been successfully developed.

The figure also marks the next stage of work focusing on PID debugging and SLAM development.

**Finally, let’s summarize the work done on the hardware.**

**读PPT**

Chen Yang will talk about the detailed situation here

The biggest difficulty is that, we are kind of unfamiliar with the IDE and set up the environment. Here I will talk about the problems encountered in detail.

We tried to calculate the quaternion through the angular velocity and angular acceleration returned by the six-position sensor, and then integrate it into Euler angles.

Then Eliminate drift errors through the acceleration of gravity.

But according to tutor's theoretical analysis, we found that the acceleration of gravity cannot eliminate the drift of the yaw angle anyway, it eliminates the pitch angle and roll angle.

In the end we decided to replace the nine-position sensor to solve the problem.

**The next topic is the slam part. Its full name is Simultaneous Localization and Mapping**

Based on our original design, Can I use the camera to complete the slam?

The difficulty is that, the single camera mapping can only infer the size of the object, it cannot judging the distance between the object and the camera itself. So it perform the task by judging the difference in the size of the object between the two frames. What’s more , it has no idea of its own state such as its velocity.

And there are very few feature points recognized by the camera on the water surface, which adds to the difficulty.

Based on the certain unknown waters that we want to map, such as Weiming Lake, we can sum up the information on the shore, to outline the movable range of the goobot, and then deal with the garbage on the water surface Perform modeling and calibration.

I conducted some experiments with the two existing algorithms. ORB corner detection algorithm and canny edge detection algorithm.

读PPT

**In our next stage,**

we first want to conduct a launch test first, and collect some data as a basis for subsequent debugging.