CSIS 0801 Final Year Project

Implementation of An Intelligent Hexapod Robot
Project Plan

FYP Account: fyp14013
FYP Website: http://i.cs.hku.hk/fyp/2014/fyp14013/

Member
- Lai Chak Yui (cylai)
- Lee Siu Man (smlee)
- Ng Siu Lung (slng)
- Yip Sit Wun (swyip)

Supervisor
- Dr. Tam Anthony
- Mr. Lee David

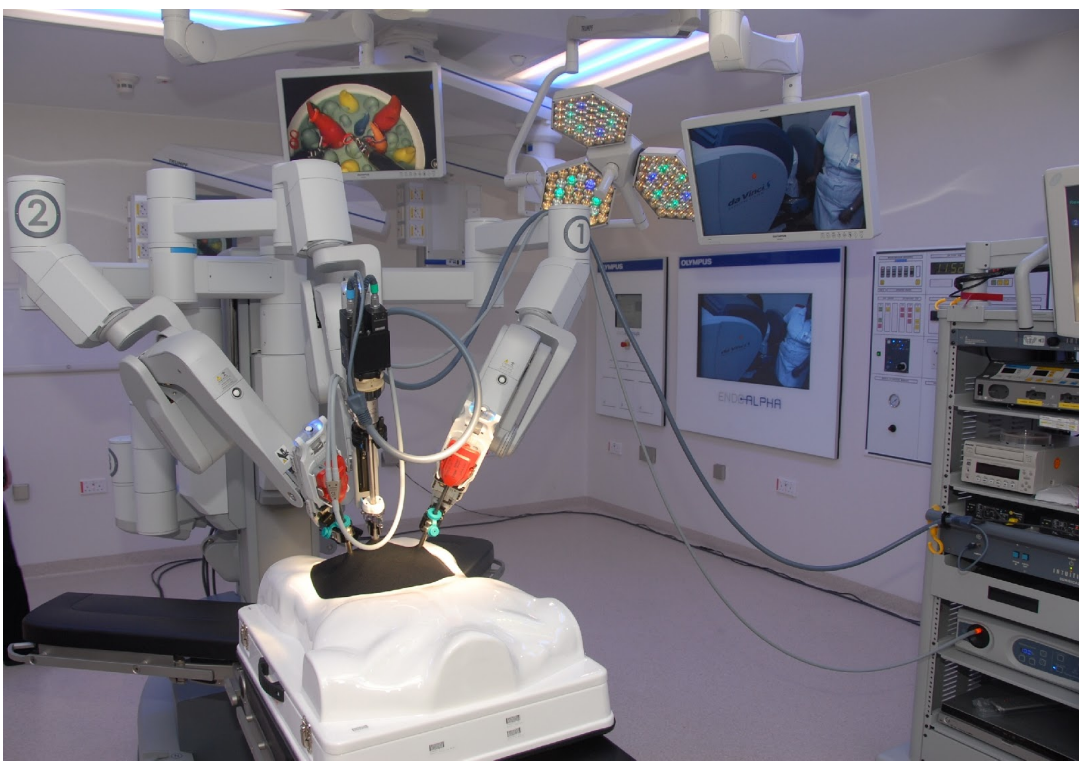
Content

Introduction

Robotics technology has been a hot topic recent years. Due to the development of 3D Printing technology, building an intelligent agent or robot is easier than ever before. Examples like 3D-Printed Cars, robotic prosthetic and even Google Glass are products integrating 3D Printing technology and Artificial Intelligence.

The movie, Terminator, predicted that human will eventually have a war against robots due to its intelligence. Let’s look at it in a less gloomy way. In manufacturing industry, robots are much more productive and accurate than human in some repetitive jobs like manufacturing of computer chips. While in Medical industry, taking The University of Hong Kong as an example, robots have been used to perform minimally invasive surgeries since 2007. [quote] Also, using a 3D modelling software and a 3D Printer allows us to print any customized shape at low cost within hours. Therefore, we believe development of robotics technology is indeed vital to improve quality of life.

With such a high potential in coming decades, 3D Printed Robot is very interesting and meaningful to our team. To understand how it is produced, we would like to develop a smart hexapod robot ourselves.



The da Vinci robotic surgical system used by Faculty of Medicine at HKU

Objective

Our objective of this project is to experience and learn the robotics technology by designing and building a hexapod robot. The technologies involved in this project cover many fields of study, such as robotic mechanics which we need for the design of the robot structure, the study and implementation of different algorithms for controlling the movement of the robot and automated path decision making, and so forth.

The major goals of this hexapod robot are the following:

1. It should be able to walk according to the command given by the user through a mobile application
2. It should be automated in the senses of intelligently avoiding obstacles without human interaction

We based on the difference of functionalities of the robot and divided set the above major goals. For the first goal, it is the demonstration of the core functionality of a Hexapod robot, which is able to walk. For the second goal, it is a goal of higher level because it is the demonstration of automated robot, which means that it is the demonstration of how “Intelligent” our robot is.

The importance of automation is that tele-operation of robot is always inadequate in real world situation. The increase in autonomy of a robot can result in the speed up of response made by the robot and the improve in accuracy. For instance, in a nature disaster scenario that human cannot approach the area and can only rely on robot to execute the rescue task. If every operation of the robot needed to be controlled through tele-operation, it is obvious that the efficiency of the rescue task would not be high due to the latency and other factors.

Besides the above major goals, we also aim to enhance the functionalities and intelligence of the robot in future development. This can achieved by extending the system that we will develop for the Hexapod robot to adapt different sensors or even other robotics accessories.

Design and implementation

1. The Design and implementation of 3D Model

Body Structure of hexabot robot

A robot can be statically stable once there are three legs in contact with the ground, and the centre of mass of the robot is with the triangle formed connected the three legs. Therefore, comparing to a quadruple robot, it takes us no work to make a statically stable hexapod robot, once we moves 3 legs at maximum each time. Moreover, as more legs are available in a hexapod robot, even one or two legs are temporarily broken, the robot can still move. To enhance the degree of movement and perform more complex motion, we decided to build a hexapod robot with 3 Servos per leg.

3D Modelling software: SketchUp

We will use SketchUp, a 3D modelling software, to draw our robot based on the kinematics and the design we prefer. After installing a open source plugin called SketchUp STL, we can use SketchUp to export 3D Model to a .STL format, a universal 3D file format supported by most 3D printers. Then we will print different parts of the robot with the 3D printer we selected, XYZ Printing da Vinci 1.0.

Printing components: da Vinci 1.0

We will use XYZ Printing da Vinci 1.0 to print different parts of the robot in this project. It is a easy-to-use 3D printer which is factory assembled and calibrated. We first need to imported different .STL files from SketchUp into XYZware, an software bundled with the printer. The software features internal checking to validate whether the model is 3D printer or not.

We first disassemble the robot and print the robot in part level, the main body, joints, feet, to reduce the risk of printing failure. After printing the parts of the robot with acceptable quality, we will assemble back the robot with the micro-computer, sensors attached.

2. Kinematic Algorithm

As mentioned above, we would have the mirco-computer, either Raspberry Pi or WRTnode, as the brain of the Hexapod robot. The main responsibility of it is to control the movement of the robot by implementing specific kinematic algorithm. One of the major challenges of this project is to study and find out the suitable kinematic algorithm which suits the design of our Hexapod, and to design and implement such algorithm into a feasible solution for the Hexapod to use.

According to the findings of our early studies in robot kinematics, there are two general equations for solving the robot kinematics problem.

1. Forward Kinematics
2. Inverse Kinematics

However, since both equations are just general ideas on how to solve the spatial problem (i.e. calculating the position of a robot arm by the given arm length and degree of each joint), we need a more specific solution which is more efficient and best fit the hardware design of our robot.

For instance, the variations of the robot design such as whether each arm should have two joints or three joints, such difference in the degree of movement would require a different kinematic algorithm in order to make the robot arms move as our desired way. In short, we need to control the servos which act as the joints on robot arms by applying the proper kinematic algorithms.

3. Intelligence of the robot (Obstacle avoidance)

Our desired robot is an “intelligent” one that it will be able to decide its own walking path and avoiding obstacle without any human interaction. To achieve this, we will design and implement different algorithms for the robot to work with different sensors.

First, we will implement a solution working with the ultrasonic sensor. With the ultrasonic sensor attached to the Hexapod robot, our basic approach is that we first implement a function in the micro-computer which constantly triggers the sensor to emit inaudible sound. When the echo is detected by the sensor, the sensor will return a value to the micro-computer. By tracking the time lapse between the emission and reflection, the distance from the robot to obstacle can be calculated with the equation of propagation of wave.

With the obtained data, we need a second algorithm for analysing it and doing the path decision making. The most basic decision will be either turn left or turn right if there is an obstacle presented in front of the robot. The algorithms will repeat analysing the data received by the ultra-sonic sensor and deciding the feasible path to walk.

4. Server on mini computer

We will use C++ as our programming language. The server program will keep running on the mini computer once the hexapod robot is booted. There are four main functions we are going to implement in the server program.

Communication between smart phone applications and the server itself

We are going to us “XML-RPC” as the communication channel. PRC is procedure remote call. It means the smart phone application will be the remote device and it will send requests(in XML format) to the server to call procedures it wants. For example, a user press the forward button on the smart phone application. Then the smart phone will send a XML request to the server which asking for forward procedure. The advantage of using XML-RPC is that it is independent of specific protocols.

Controlling sensors

The server will instruct how sensors behave. For example, the server will define when to trigger the ultrasound sensors and when to get back the echo signals. And then it will analyse the data collected to calculate the distance of the obstacle.

Controlling servos

The server will instruct how servos behave. For example, when the server receive a forward procedure call. It will instruct different servos that how many of degree should the servos rotate so that the hexapod robot can move forward.

Storing the kinematic algorithm

Our kinematic algorithm will be stored in the server. Whenever the server receive a movement procedure. The server will make good use of the algorithm to calculate how should the servos should behave.

5. Design and implementation of Smartphone App

We will build an mobile application to patrol the hexapod robot. With the user-friendly graphical user interface, users can control the motion of robot with ease by clicking buttons. Users can also voice out the commands, or rotating the smartphone to navigate the robot.

The user first connect the smartphone to the WiFi hotspot initiated by the micro-computer of the robot, or connect the smartphone to the network which the robot has connected. When the application is initiated, it will detect the presence of the hexapod robot. Once the application establish connection with the robot over the WiFi, the application will communicate with the robot by sending commands using XML-RPC protocol. The application will also features voice-to-text library to encode users' voice commands into string.

We plan to develop this mobile app on Android platform first due to the available resources. For future development, mobile app on other platform such as iOS platform is also possible.

6. Basic flow of using hexapod robot

Turn-on the robot

- the robot itself boot the Linux OS, pre-loading the server and corresponding “intelligent” functions
- some “intelligent” functions run automatically once startup, e.g. the obstacle avoidance function if implemented

Connect the smartphone to the same network as the robot

- the smartphone establish connection to the robot and enable data-transfer between the smartphone and the robot server

Using on-screen button/voice to patrol the robot and see how it behaves

- The smartphone receives users’ command and send the commands to the robot using XML-RPC
- the servers receivers command, the micro-computer decodes and executes the command with the built-in and implemented functions
- the servers send the results back to the smartphone if necessary (e.g. photos or videos captured by the webcam)

7. Micro-computer of the Hexapod (Hardware-level)

To build an “intelligent” hexapod robot, we will attach a micro-computer to the robot as its core. The micro-computer is Linux-based which features great expandability. Different hardwares and sensors, including servos, ultrasound sensor, camera, will be connected to the micro-computer. By installing proper drivers, the micro-computer can co-work with the hardware to perform different actions. The credit-card size computer has built-in WiFi module (or connect it with a WiFi USB adapter), it can communicate with smartphones, allowing users to patrol the robot with the app.

We will first evaluate two linux-based little computers, Raspberry Pi and WRTnode. Though both of them run embedded-linux, OpenWRT, they features different hardwares and expandabilities. We will decide which one to be used in our final version after performing thorough evaluation and testing.

8. Shape and colour recognition

If resources are available, we would like to implement colour and shape recognition into our robot. Therefore, the robot will be able to recognise basic shapes like square, oval, triangle and base colours like red, blue, yellow and green. To achieve this goal, we will attach a camera to our robot. Using the camera, the robot constantly record the surroundings. The micro-computer will use some of the frames and analyse that particular frame. To recognise the basic shape, the computer will run edge detection to find the border of the shape. Then it will count the number of continuous edges to identifies what shape is presented in that frame.

Timeline

We will use iterative method to build our project. We will set all targets on the first day of a week. And then accomplish them and tested them at the end of a week. As a result, we have weekly versions of our project which will be incremental.

Mid-Sep - Oct

- Study of the project
  - All materials bought, linux system, the 3D printing software, kinematic algorithm.
- Project Plan Proposal and Website
  - Finish the project plan proposal and built a website to show our project in an effective way

By First week of October

- Gather and all requirements and design the architecture
  - Server of the robot, 3D printing model design, android application

At the Second week of October

- Starting implementation (using iterative method)

End of October

- Android App 1.0
  - Can send and receive message with server
- Test Print and Combine the 3D Model
  - Print 3D models from online resources.
  - Combine it with the materials bought.
- the server can make use of Ultrasound sensor to get the distance of obstacle
- Naive movements of Robot
  - Server control the servo to rotate

End of November

- Tailor 3D printing model
  - is similar to our design(make a 3d model ourself)
- Walkable Robot
  - Server have basic logic for forward,backward

End of December

- Android 3.0
  - Better UI Design

- Finalize 3D Printing Model
- 3D printing model printed is exactly the same as our design
- Server have the basic logic forward,backward,turn left, turn right.
- Natural Movement of Robot
    - Adopt the first version of Kinematic Algorithm to the movement of robot
- Add Sensor

End of January

- Android 4.0
    - Alter the UI and functionalities of the it depends on the server/sensors progress.
- Enhance the Kinematic Algorithm
- The hexapod can move around on a plain surface without any control. It will turn around when it detects an obstacle.

End of February

- Finalize the Kinematic Algorithm and check the correctness
- Add a webcam on the robot. The smart phone application can see what is in front of the hexapod continuously.

End of March

- The robot can move along a stair
- User can enter a text on the smart phone application, the robot will speak it out.

End of April

- Testing all features and optimization

End of May

- Ready to deliver the FYP project.