SLAM on Smartphones

ECE 445

Project Proposal

YIXIAO LIN

FENGYUANSHAN XU

TA: Mustafa Mukadam

September 18, 2013

Table of Contents

1. INTRODUCTION

1.1 Statement of purpose

When it comes to distributed robotic system, great limitation exists for higher level programming. A positioning system would allow coordinate based behavior. Thereafter, it reduces the complexity of higher level robotic programming. We want to build a positioning system that detects the environment around itself and creates a map. The map information can be easily used by other robotic developers in any indoor environment.

1.2 Objectives

Goals:

- Reduce the complexity of the indoor robotic positioning problem
- Compatible map data transfer to android smart phones as well as computers

Functions:

- Sensor platform with accurate angle control
- Continuously collects the distance data in an automatic way
- Noise filtering of the sensor raw data and process data
- Transfer processed map data to an Android phone
- Generate map of surrounding environment

Benefits:

- Simple devices fit in any kind of indoor environment without additional set up requirements
- Easy to apply to any other robotic platforms

Features:

- Extremely portable device
- Simple to set up and use for any robotic platform
- Generate map information of the indoor environment

2. DESIGN

2.1 Block Diagram

As shown on figure 1 below, this is our top level system diagram.

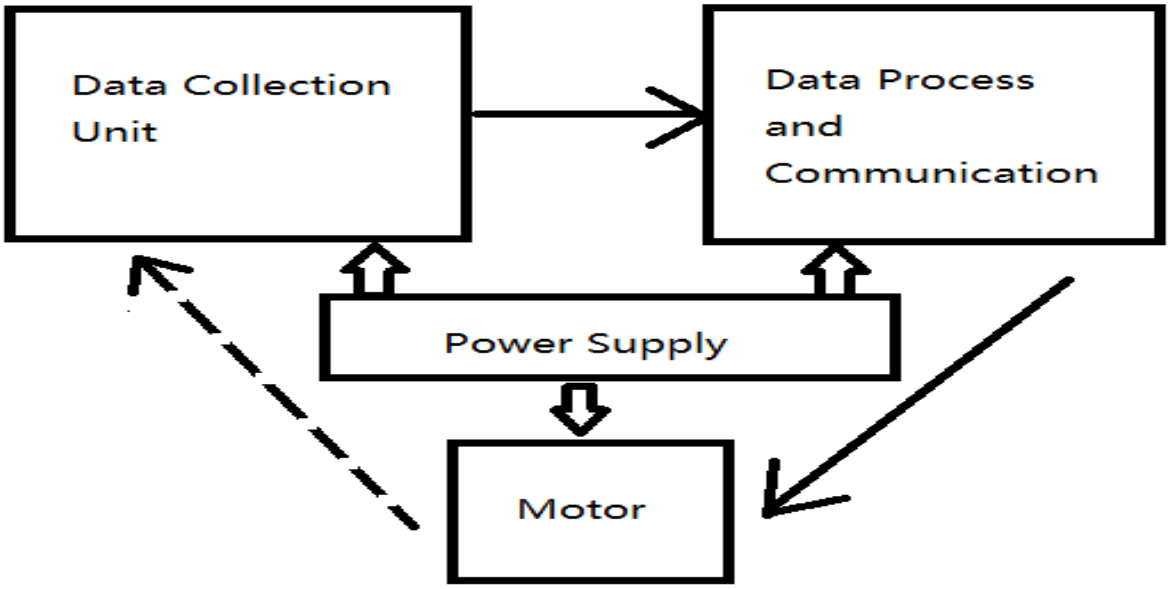

Figure 1. Top Level System Layout

The figure 2 is the block diagram for each top level system block.

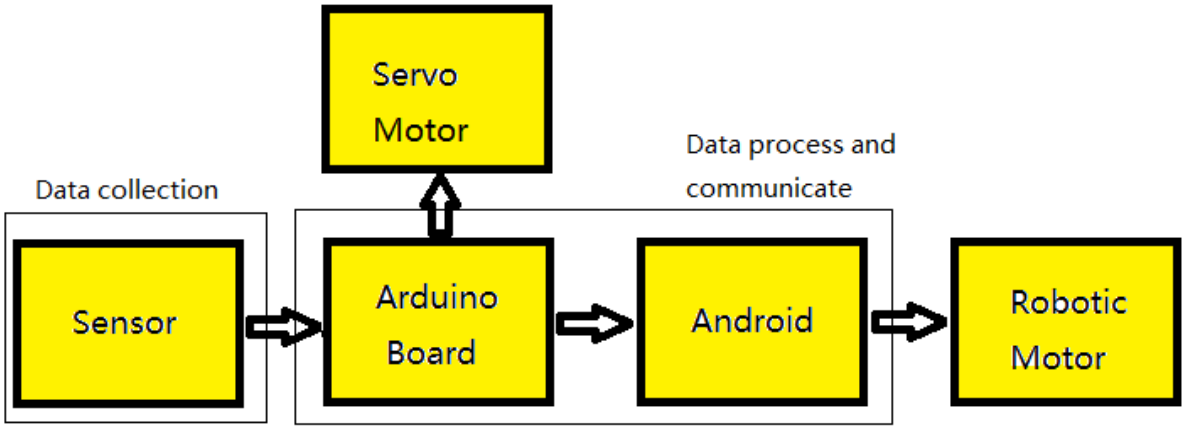

Figure 2. Block diagram

2.2 Block Description

Summary:

As shown in the top level system layout, our design is consisted of three parts. Data collection unit will collect the data which is the distance between the robot and the surrounding obstacles. The data will flow into the data process and communication unit. A map of surrounding environment will be generated. A path for exploring the room will be generated for the robot as well. Then the newly planned path will drive the motor for robot to move and drive the motor of the sensor to collect distance data in another direction.

Sensor:

There are two devices act as sensor for our design:

1. Ultrasonic sensor

We are using an Ultrasonic Range Detection Sensor DYP-ME007. This is used for detecting the range between the robot and the obstacles nearby. This sensor can send an ultrasound pulse at a frequency of 40KHz. It then waits for receive the echo back pulse and calculates the time taken in microseconds. With 5V DC working voltage and 15mA working current, this sensor has sentry distance from 2cm - 5m and accuracy up to 0.3cm. The output of this sensor is an analog signal. We can determine the distance between current position and the obstacle according to the voltage level. The output voltage will be passed to Arduino Board for processing. The sensor is driven by a servo motor to collect the data from all directions.

2. Inertial Measurement Unit

This unit will be used to detect the position of robot. A linear accelerometer measures non-gravitational acceleration of the robot. For each of the six degrees of freedom (x,y,z and θx,θy,θz), IMU integrates over time the sensed acceleration, together with gravity, to calculate the current velocity. With 3.3V input and I2C interface, the output signal from this unit can be passed to our Arduino board for processing. We can determine how far we are from the previous measurement point, which makes our map generation more practical.

Arduino Board:

We choose Arduino RoMeo v2 to be our data processor. It will control the servo motor and generate a map of surrounding obstacles. We are going to write a program for this board to process the output voltage from the sensor. Therefore, Arduino board can determine how far an obstacle is away from the robot at the direction. Then based on current map information, Arduino board will control the servo motor to rotate by an angle to get the distance data in another direction to complete the map. The control signal will be sent to the servo motor controller. There is an USB interface on this Arduino board, which enables the communication between the Arduino and Android platform. The map created by the Arduino board will be passed to the Android phone through the USB port.

Servo Motor:

We choose servo motor over stepper motor for the reason that servo motor allows for precise control of angular position and velocity. We can determine at which angle our sensor is currently at. This motor will be assembled on top of our robot. The digital servo motor controller directs operation of the servo motor by sending velocity command signals to the amplifier, which drives the servo motor. Operating

voltage is about 4.8V to 6V.

Android:

Currently, we are using Samsung Nexus S Android smartphones to control iRobot Create platforms. The phone is placed on top of iRobot Create. It sends commands to the robots using Bluetooth communication. We will develop program in java to control the robot exploring behavior.

iRobot Create:

This platform will be controlled by Android phone. It will move according to the phone's command. The platform has two wheels; it can go straight, turn left/right.

Power supply:

The power supply will be used for the controller, the sensor and the servo motor. The power supply will be consisting of a 9V battery and an adapter.

2.3 Performance Requirements

1. Ultra-sonic sensor feedback accurate, with false tolerance up to 0.01 meter if within 5 meter radius
2. Servo motor able to drive the sensor accurately from 0 to 360 degree; False tolerance up to 1 degree.
3. IMU position estimation is accurate, with false tolerance up to 5% of the total displacement
4. Arduino board map generation efficient, generate a complete map in 30 seconds after receiving the sensor data.
5. Real time map data transfer from Arduino to android phone safely
6. iRobot create explores and maps any room in 5 minutes.

3.0 Verification

3.1 Testing Procedures

1. Compare the data obtained and the real surrounding environment with distance from 0.05 m to 5 m
2. Initialize the servo motor position to 0, test 20 random positions. Then we compare the servo motor position and the ideal position.
3. Generate position data by moving IMU from point A to point B. Compare the position data generated by IMU and real position.
4. After generating a new map, a test LED lights up. After servo motor finish moving from 0 to 360 degree, the time LED lights up should be less than 30 seconds.
5. Shortly after the Arduino has a new or updated map, the Android phone receives the same information; the delay should now be preserved by users, signal passed to Android phone should stay in the range of 5.00 ± 0.25 V
6. Generate a relative complete map of a room (no bigger than a typical living room) in 5 minutes

3.2 Tolerance Analysis

Since the mapping heavily depends on distance data, the tolerance analysis will be based on the accuracy of our ultra-sonic sensor. To ensure the accuracy of sensor measurement, we will compare the sensor estimated distance with real distance. The room diameter should not be larger than 10 meter. Any point that is further away than 5 meters from the sensor would not be guaranteed to meet the 0.01 meter accuracy specification.

4. Cost Analysis

4.1 Labor cost

Name	Hourly Rate	Total Hours	Total
FENGYUANSHAN XU	$20.00	180	$9,000
YIXIAO LIN	$20.00	180	$9,000
Total	$40.00	360	$18,000

4.2 Unit Cost

Item Name	Quantity	Cost($)
Ultrasonic Sensor	1	7.00
Arduino RoMeo	1	40.00
Servo Motor	1	15.00
IMU	1	50.00
Base	1	30.00
Total		142.00

Section	Total
Labor	$18000
Parts	$142
Total	$18142

4.3 Schedule

Week	Task	Responsibility
9/16	Finalize the proposal	Yixiao Lin
	Read papers on servo motor and in door slam using ultrasonic sensor	Fengyuanshan Xu
9/23	Research Arduino to Android communication, order ultrasonic sensor and required arduino board.	Yixiao Lin
	Design Review, order servo motor, study Arduino code for controlling servo motor.	Fengyuanshan Xu
9/30	Test servo motor performance.	Yixiao Lin
	Handle servo motor and inertial measurement unit. Assemble servo motor with ultrasonic sensor	Fengyuanshan Xu
10/7	Work on program for data processing	Yixiao Lin
	Test ultrasonic sensor on servo motor	Fengyuanshan Xu
10/14	Finish code for map generation	Yixiao Lin
	Test ultrasonic sensor with IMU	Fengyuanshan Xu
10/21	Simulate the program on Arduino	Yixiao Lin
	Connect sensor, motor and IMU to Arduino and test the data processing program	Fengyuanshan Xu
10/28	Grab papers on communication between Android and Arduino	Yixiao Lin
	Work on code pass data from Arduino to Android	Fengyuanshan Xu
11/4	Test path planning program on Android phone	Yixiao Lin
	Work on program for map generation	Fengyuanshan Xu
11/11	Combine all parts together	Yixiao Lin
	Finish program for map generation	Fengyuanshan Xu
11/18	Complete the project by fixing remaining issues	Yixiao Lin
	Final test to devices and functionalities	Fengyuanshan Xu
11/25	Tolerance analysis	Yixiao Lin
	Verification of specifications	Fengyuanshan Xu
12/2	Demo and presentation	Yixiao Lin
	Demo and presentation	Fengyuanshan Xu
12/9	Final paper	Yixiao Lin
	Final paper	Fengyuanshan Xu