Arduino based obstacle avoidance robot

Report Name	Progress Report
Author (User Id)	Daniel Atkinson (daa9)
Supervisor (User Id)	Dave Barnes (dpb)
Module	CS39440
Date	November 16, 2012
Revision	0.6
Status	Draft

Contents

1 Project Summary

1.1 Aim

The aim of the project is to build and program an autonomous robot that can move around freely within an environment and avoid coliding with obstacles it may find.

1.2 Background

I tend to experiment with electronic components on a regular basis. Due to this I try and make small systems and get them working, originaly nothing more than a simple timer or an audio amplifier. Naturaly the progression from this would be to move onto microcontrollers.
I like to see things move, it keeps my interest. The satisfaction I get when having started with nothing and going through to get something built and moving is what makes me think up new ideas, how can this be modified?, how can this be made better?. A robotic project the perfect thing for me to do.

2 Current Progress

2.1 Related Works

There are already lots of examples of work in this area. Cameras can be used to find a path between objects using different image proccessing techniques. These require a fairly high level of processing due to the amount of data in each image. This can also sometimes lead to mistaking visible patterns in the environment as obstacles or empty space when there is none, such as a mirror or a patterned carpet. [3]

2.2 Technologies

2.2.1 Sensors

- LDR (light dependant resistor)
  Small resistor that changes its resistance depending on how much light it is exposed to. This could be used to detect if the robot is very close to bumping into an object and avoid it as the object would shadow some of the light from getting to the resistor, like a bump skirt.

- Camera
  A camera could be used to detect objects in front of it using various image processing techniques. This method is good because it can potentialy map a relatively large area in a single image. On the other hand it requires more processing to do which can be slow and result in having colided into an object or being stuck in a tight space before it has finished processing. A camera and the proccessing linked to using it for obstacle avoidance also increases power consumption, decreasing total runtime.

- Infra Red
  Used to detect distance from an object. An emiter and a reciever linked to work like the light dependant resistor but using infra red instead of visible light. Depending on intensity it can be used to detect distance. These could be aranged all around the robot like a halo to detect objects from all sides.

- Sonar
  Again emiter, reciever style approach. Emit an ultrasonic wave to bounce off of whatever

is in front of the emitter and then time how long it takes to be picked up by the reciever to determine how far away the object is. This method comes with its drawbacks. Due to how sound waves behave when they interact with the environment by bouncing off of it. If the surface is angled or curved the sound can bounce away from the reciever, either not returning to it at all and giving a false reading of there beign nothing in front of it, or be bouncing off of multiple objects back to the reciever giving a false longer reading than there actually is.

2.2.2 Actuators

2.2.3 Materials

I considered several materials for the robot chassis to be built of.

- Wood
  This would be the easiest material to make the chassis from as it is cheap, easy to cut into the intended shape and easy to screw components to. Also it being non-condictive would help when mounting circuit boards to it.

- Plastic
  The lighter option. Good due to its low weight but not as strong as wood and could bend or snap under load of many of the heavier components such as motors or a power source. Also can be more expensive than wood to aquire and cut into the desired shape. It is also non-conductive, again useful to mount conponents to.

- Steel
  A stonger material that can take much heavier loads but is itself rather heavy compared to wood and plastic. This extra weight will put more stress on the motors used to drive to robot and may even need the motors to be more powerfull. It is conductive which means more materials will be needed to make a non-conductive mounting platform for electronic components.

- Aluminium
  A much lighter metal than the Steel, but still heavier than wood or plastic of the same thickness. It can take heavier loads than the plastic. It is conductive, making it useless to mount components onto without a mounting platform of some kind.

Aluminium seems to be the best all round. It is stong but not as heavy as steel. It can act as a heat sink for the motos if they are mounted diectly. Also it is not too expensive to buy in small amounts.
In addition to the aluminium base I have decided to use plastic for mounting components to the base. To make this easier I have built at 3D printer for fabicating custom components to specification.

2.2.4 Power Source

As this robot is intended to move around freely the powersource cannot be supplied by an ordinaly power outlet, it has to be self contained. This means it will have to be a battery.
The battery will have to be several cells or a single high output cell due to the size of motors I will be using.
I will need as many Amp hours as possible for longer runtimes. This could be achieved with several cels linked together in series (end to end) to increase voltage and link more together in parrallel (side by side) to increase amp hours (runtime).
The lighter of the batteries would be to use lithium polymer (LiPo), these come in up to 11.2

volt packages which is not quite high enough to run the motors I have chosen.
A lead acid battery is the other choice with high enough output. A single battery can output 12 volts and can be supplied with high amp hour ratings compared to the lithium ion alternatives. The chosen battery is rated at 12 volts and 4 amp hours. It weighs more than the LiPo but is a more convenient package as it takes up less space on the robot.

2.2.5 3D Printer

As mentioned previously I have built a 3D printer. This was not built exclusively to aid in this project but it was a factor in the puchase decision.
The printer was not designed by me, it was bought as a kit. The construction of the device aided as a good learning proccess and deepened my understanding of how 3d printing works. The build area of the printer is 200x200x260mm (length x width x depth) so it can produce fairly large components.

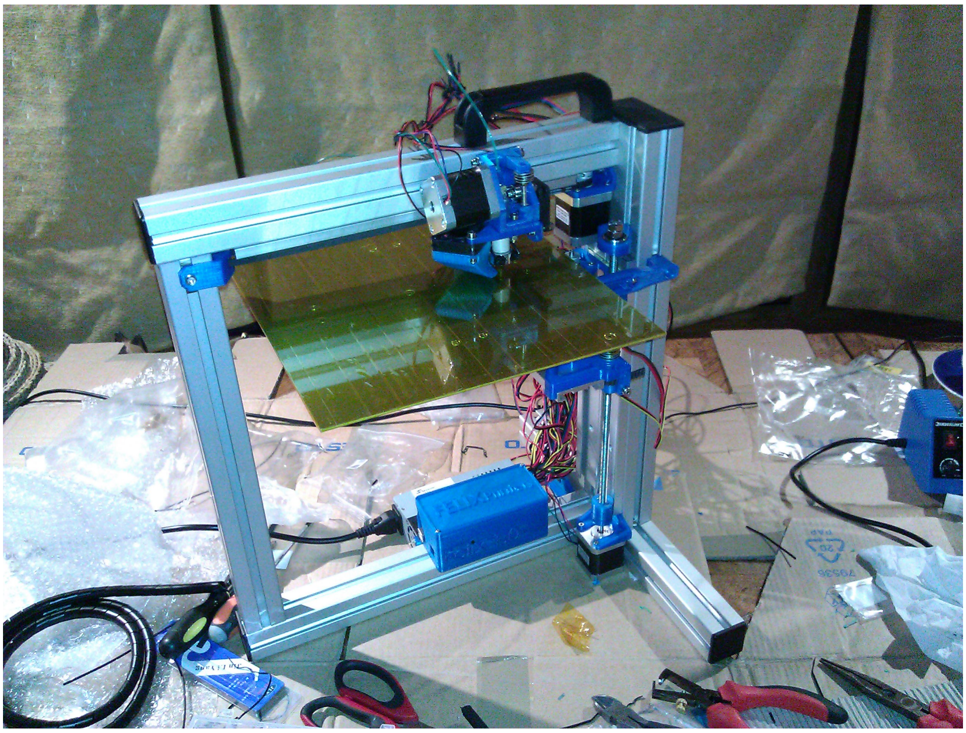

Figure 1: 3D Printer

2.3 Initial Learning

After choosing to use the arduino due to it having such a vast amount of support and examples I bought one and began to learn how to use it, using the arduino website itself as a guide (arduino.cc). On here there are many tutorials with circuit diagrams and example code on how to get the basics working, which I found very helpful.
The first things to learn were how to interact with the development board from my computer, then move onto getting it to output something. Making lights blink was easy, and communicating over a serial connection was also relatively easy. It got a bit harder when trying to recieve useful input from a sensor.
The first input I recieved was from an Infra Red sensor, using this to determine how far an object was from the sensor itself.
Something I discovered very quickly was that even if the sensor was not moved and the object

it is pointing at is not mvoed the valus recieved from the sensor vary. This 'noise' could cause a problem. The obvious solution was to take a range of samples and average them to get a less eratic reading.
To further add to that I may add a threshold and if the value from each reading it too different from the current average of a range of the most recent readings, then discard that value as to not push the average.

Next I moved onto motors. I had a 9 volt DC motor in the kit that came with my Arduino starter pack. Getting the motor to turn was very easy, just apply current to one side and have the other linked to ground. Simple.
Getting the motor to turn both forwards and backwards seemed liek it would be harder. But due to this motor runnign well off of 5 volts this was not an issue as an Arduino can output around 5 volts on its output pins and also link these pins to ground as well. As such I can just switch which pin is ground and which is supplying voltage to change the direction of the motor. Of course this will not be so easy for larger motors as the Arduino will not be able to supply enough voltage in this manner. I will require a motor driver.

2.4 Prototype

To build the prototype I bought a small plastic chassis kit which inclided the chassis, two wheels, a caster for the rear 'wheel', two motors and two sets of gears for the motors. All I needed to add was the microcontroler itself, a motor driver and a power source to get this running.

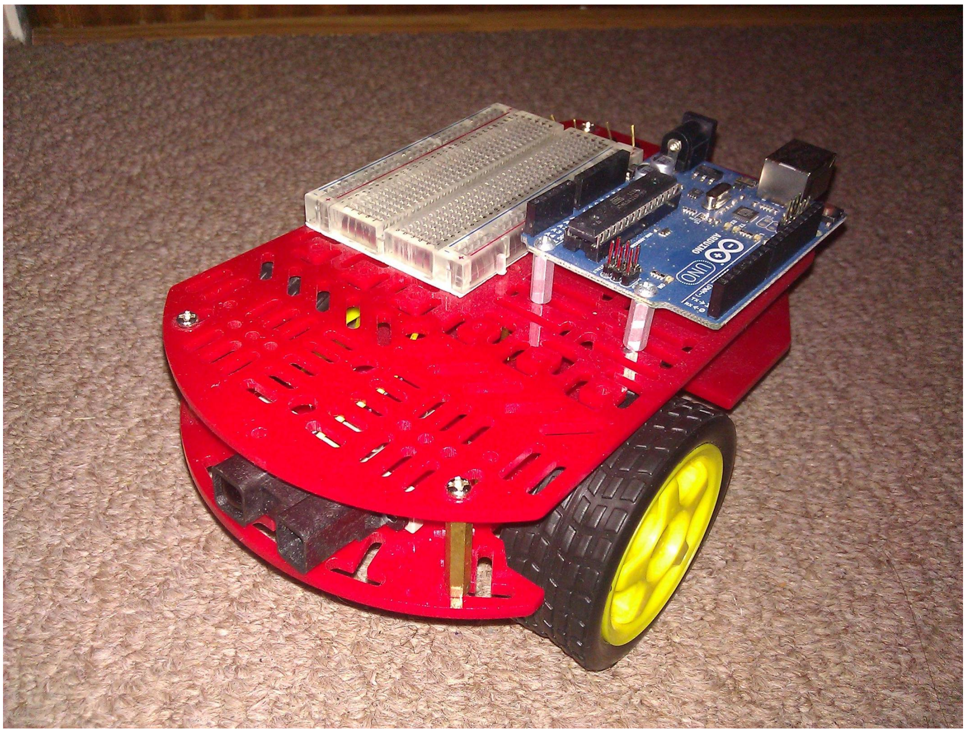

Figure 2: Plastic prototype

2.5 Current Build

So far a basic chassis has been built out of aluminium and both the motor driver and the arduino have been mounted onto it.

2.5.1 Base

This consisted of cutting a sheet of aluminium to the desired shape, cutting two more pieces, bending them at a 90 degree angle and cutting a hole in them to mount the robots motors too. These two pieces were then bolted onto the underside of the larger piece. A further piece was bolted onto the base plate extruding out of the back to support a rear wheel.

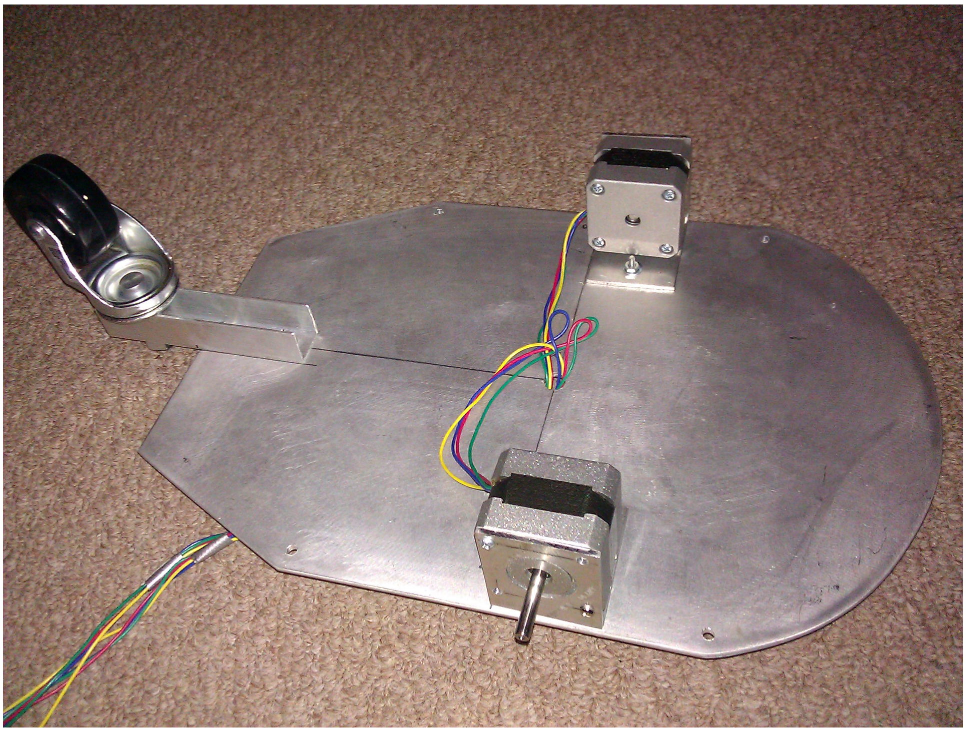

Figure 3: Aluminium chassis

2.5.2 Components

Using the 3d printer I made some mounting pieces to put electronic components onto the emtal chassis. First something small and easy. These are a sharpe infra red sensor mount and an ultrasonic sensor mount, the models for which I downloaded off of thingiverse.

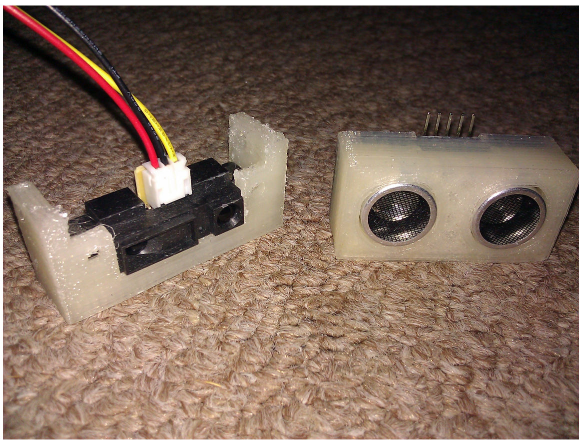

Figure 4: Printed sensor mounts

2.5.3 Chassis

In addition to the mounted components and the base chassis, some large wheels have been added so that it can actually drive. These wheels are intended for off-road use and should handle and terrain the robot may end up trying to traverse.

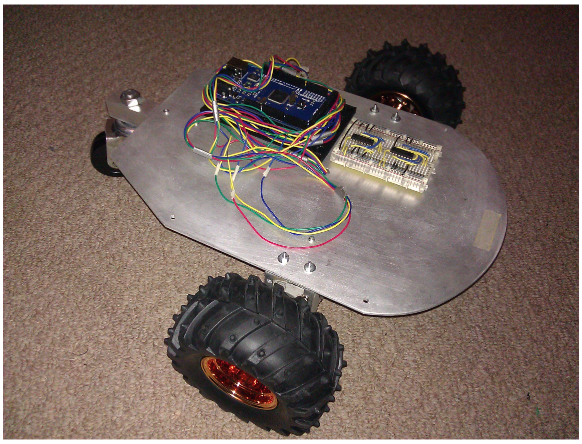

Figure 5: Chassis with wheels

2.6 Utility

I put a small amount of time into building a portable device to interact with the main robot with main the purpose of reading live debug output. This device was thought of because following aroudn the prototype with a laptop plugged into it became very anoying and quite awkward because if the robot made a tight turn it could get tangled up in the cable.
This resulted in using some components I had lay around to build a temporary solution. I used an Arduino fio, this has a built in LiPo battery socket, charging circuit and has an xbee module socket. An xbee is a wireless radio popular with elecronics hobbyists for project such as this. It provides a simple serial interface over wireless, exactly what I needed, Also a 2x16 character LCD display.
It's not pretty but it works for a temporary/initial solution. I had the intention to have this

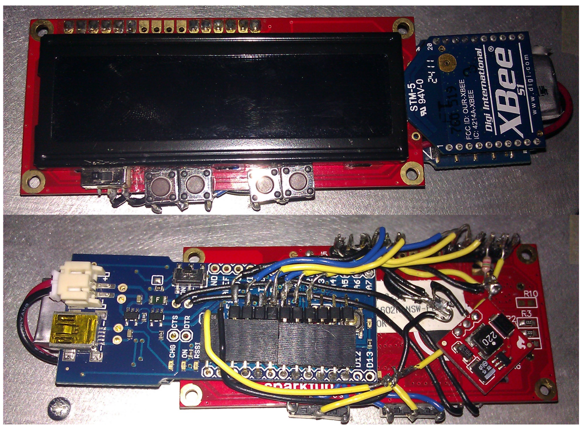

Figure 6: Small wireless monitor

mounted on my wrist so I would not have to constantly have to pick it up and put it down to use. A glove was used to hold it to my wrist.

Figure 7: Wrist mounted device

3 Planning

3.1 Subsystems

The robots functions will be divided into subsystems/modules.

- Controller
  The controller unit will be a microcontroller that can recieve input from all the other devices and process what to do with the data and delegate tasks to the other systems to perform more complex tasks.
- Motors
  Due to the motors being steppers, they require constant control to get them to move. Manually controllign the magnets inside the motors to get them to turn step by step or using an arduino library to it both require constant use of the microcontroller to do so. This could be threaded or off-handed to another controller. The off-handing approach is what I am planning. This will take input from the control system.
- Sensors
  This system will constantly take sensor reading or when prompted to do so. It will send this data to the controller to decide what to do with.

Annotated Bibliography

[1] Various, "Getting started with raspberry pi gpio and python," http://lwk.mjhosting.co.uk/?p=343, Aug. 2012, accessed 19 October 2012.

How to use Python on a Raspberry Pi to control the general purpose input output pins on the board

[2] ——, "I2c communication between two arduino boards," http://absences.sofianaudry.com/node/10, Aug. 2012, accessed 20 October 2012.

Details on how to get two arduino development boards to communicate with each other, I intend to use this to be able to offhand tasks to another board, such as driving motors independantly from the sensor processing

[3] ——, "Robo realm," http://www.roborealm.com/tutorial/Obstacle_Avoidance/, Aug. 2012, accessed October 2012.

Tutorial on obstacle avoidance using a camera