OPTICAL TACTILE SENSOR FOR SLIP DETECTION OF A GRASPED OBJECT

Kien Ly

Bachelor of Engineering
Mechatronics Engineering

MACQUARIE
University
SYDNEY·AUSTRALIA

Department of Electronic Engineering
Macquarie University

November 7th, 2016

Supervisor: Dr David W. Inglis

ACKNOWLEDGMENTS

I would like to acknowledge Dr. David Inglis for suggesting the mouse sensor as an alternative to the infrared emitter and detecter as well as suggesting the calculation of deflection of a cantilever beam. I would also like to acknowledge Ryan Barnes as he offered insight into the mouse sensor and Brendan Menzies with setting up a robotic gripper.

STATEMENT OF CANDIDATE

I, Kien Ly, declare that this report, submitted as part of the requirement for the award of Bachelor of Engineering in the Department of Electronic Engineering, Macquarie University, is entirely my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualification or assessment an any academic institution.

Student's Name: Kien Ly

Student's Signature: Kien Ly

Date: 7/11/2016

ABSTRACT

Intellgent grasping is required by autonomous robots to lift objects with unknown weight, this requires the gripper robot to have enough grasping force so that the object does not slip out of the gripper but not too much force otherwise the object maybe damaged. The development of a optical tactile sensor can be an approach to intellgent grasping for a robotic gripper. Initial results collected showed that using an infrared sensor in a deforming material will provide slip detection for the gripper robot. Combining this tactile sensor with the gripper robot will create a feedback system that can detect slip and adjust gripping strength for intellgent grasping. The low cost and small size of the tactile sensor offers a cost efficient and adaptable tactile sensor design for a autonomous small gripper robot used to pick and place packages.

The aim of this thesis is to improve an existing design and incorporate elements from other successful tactile sensor designs. Various issues are present with the proposed design which are solved. The infrared sensor will be affected by ambient lighting which can be controlled by placing the sensor in a cavity which no external light source can affect it. There is also the issue of choosing a deformable material which is not too soft otherwise it will always deform or too stiff otherwise it will not deform when the object is slipping. The system must also have a fast response time to detect slip in under one second as well as when slippage occurs at 1 mm or less. This may be solved by using a mouse sensor which has a high response time in microseconds and can detect changes in position under 1 mm.

Contents

Acknowledgments iii

Abstract vii

Table of Contents ix

List of Figures xiii

1 Introduction 1
1.1 Robotic grippers . . . 1
1.2 Previous project contribution . . . 1
1.3 Project objectives . . . 2
1.4 Budget review . . . 2

2 Background Theory 3
2.1 Literature review . . . 3
2.2 Industry grippers . . . 4
2.3 Tactile sensors . . . 5
2.3.1 Strain gauge . . . 6
2.3.2 Ultrasonic Force sensor . . . 6
2.3.3 Optical Fibers . . . 7
2.3.4 Optical decompression . . . 7
2.3.5 Opto-Mechanical . . . 7
2.3.6 Opto tactile sensor for surface texture . . . 8
2.4 Optical sensors . . . 9
2.4.1 Infrared emitter and detector . . . 9
2.4.2 Mouse sensor . . . 10
2.4.3 Dual optical and mechanical tactile sensor . . . 11
2.5 3D printing . . . 11
2.6 Deformation of a cantilever beam . . . 12
2.7 Mechanical properties of materials . . . 13
2.7.1 Shore hardness . . . 14

3 Deformation component design **15**
3.1 Deforming material anaylsis 16
3.1.1 Simulation of deformation 18
3.1.2 EPDM rubber 18
3.1.3 Simulation of curved deformation 19
3.2 Comparison of designs and deductions 20

4 Infrared sensor **22**
4.1 Experiment 1: Infrared detector and emitter 22
4.2 Infrared sensor result 25

5 Mouse sensor **26**
5.1 Sensor overview 26
5.2 Sensor operation 26
5.3 Surface quality 26
5.4 Lifting distance 27
5.4.1 Experiment 2: Lifting distance 27
5.5 Operating temperature 28
5.6 Limitations of sensor 28
5.7 Mouse sensor mount 29
5.8 Mouse sensor results 30

6 SG gripper **31**
6.1 Dimensions and characteristics 31
6.2 Robotic arm software 32
6.3 Modification from proposed plan 33

7 Slip detection system **34**
7.1 System overview 34
7.2 Slip detection sensor 35
7.2.1 Infra red sensor approach 35
7.2.2 ADNS9800 laser mouse sensor approach 35
7.2.3 Old mouse sensor approach 35
7.3 Feedback system 35
7.4 System failure precautions 36
7.4.1 Optical sensor failure detection 36
7.4.2 Gripping arm immediate shut down 36
7.4.3 Drop chute 37

8 Conclusions, improvements and future work **38**
8.1 Conclusions 38
8.2 Improvements and alternatives 39
8.2.1 Dual sensor with damage detection design 39
8.2.2 Internal reflectance 39

8.2.3 Different Gripping robot 39
8.2.4 Different material 39
8.3 Future work and applications 40
8.3.1 Experiment 2: Surface quality 40
8.3.2 Use of old mouse 40

9 Abbreviations **41**

A Tactile sensor design **42**
A.1 Overview 42

B Arduino code for ADNS9800 **45**

C Matlab code **51**

Bibliography **55**

List of Figures

2.1 Tactile sensor model from previous design [10]. . . . 3
2.2 Deformation of sensor with (A) having no slippage occurring. Right image (B) with slippage occurring [10] . . . 4
2.3 Ultrasound tactile sensor sensing how close an object is [12]. . . . 6
2.4 Attenuation due to microbending of a flexible optical fibre [8]. . . . 7
2.5 Optical cavity sensor uncompressed state in (A). The cavity sensor under compression which increase intensity of radiation in (B) [9]. . . . 8
2.6 Opto-mechanical array breaking line of sight when slip occurs from right to left [8]. . . . 8
2.7 IR emitter and detector pair detecting proximity of an object [2]. . . . 9
2.8 Mouse sensor reading the surface with an LED and low resolution camera [13]. 10
2.9 Dual mechanical and optical layout of a tactile sensor [24]. . . . 11
2.10 3D printed filling patterns to create strength in printed hollow objects adding structural strength [17]. . . . 12
2.11 (A) Deflection of a cantilver beam (B) The cross-section of the beam [11] . 13
2.12 Stress - Strain Curve of a ductile material [16] . . . 13
2.13 Shored hardness scale types converted between each type [20]. . . . 14

3.1 Proposed optical tactile sensor design. . . . 15
3.2 Dimensions of the proposed tactile sensor. . . . 16
3.3 Expected Deformation of straight design . . . 17
3.4 Simulation of deflection with 2 cm length (left) and 1 cm length (right) with a 3.82 N force one the edge of both sensors, simulating the sensor holding a can. . . . 18
3.5 EPDM rubber component straight shape cut. . . . 18
3.6 Simulation of curved design deflection . . . 19
3.7 Expected curved design deflection . . . 19

4.1 IR detect and emitter circuit [26] . . . 22
4.2 IR circuit and microcontroller . . . 23
4.3 Expected Arduino voltage output . . . 23
4.4 Caliper experiment setup . . . 24
4.5 Voltage vs Distance of simulating grasping an object . . . 24

4.6 Voltage vs Distance of a slipping object 25

5.1 Expected Surface quality on white paper [25] 27
5.2 Infrared image of mouse sensor 28
5.3 3D printed mouse sensor mount for robotic gripper. 29
5.4 CAD of mouse sensor mount to be attached to the gripper 30

6.1 Measurement of length of gripper and distance between fingers 31
6.2 Flow chart of grasping robot 32

7.1 Overview of intelligent grasping system. 34
7.2 Flow chart of operation 36

A.1 Tactile sensor design 42
A.2 Tactile sensor curved design dimensions 43
A.3 3D printed mount 44

Chapter 1

Introduction

1.1 Robotic grippers

Robotic grippers have been used to lift, hold and place objects without destroying them. This means that it will need to have sufficient gripping force so it does not slip out of the gripper's hand but not too much force that it crushes the object. Intellegent grasping has many applications in the real world such as moving parcels of unknown weight and content for delivery.

In the current automation industry, robotic grippers will move objects with their weights known and an optical sensor is used to detect slip. The gripper would lift the object and if it slipped it would break the line of sight of the sensor to emitter. This thesis will be using an optical sensor with in a tactile sensor to detect slip. The optical sensor will be inside the tactile rubber gripper, when deformed to the point when slip is classified as occuring the gripper will lower the object and apply more gripping force for another attempt to lift the object.

This final report has background theory and literature review of relevant material for this thesis. The literature review is on a previous design using Cameras to simulate optical fibres; there is also a review of an report of industry grippers and problems and finally the operation of a mouse sensor. A review will be conducted on the materials used to make this sensor as well as their properties which make them suitable for this design.

This thesis will also decompose the system components and anaylse the individual components first such as the rubber deforming component, optical sensor (infrared and mouse sensors) and robotic gripping arm. After experiments are conducted to confirm the desired results to achieve the project goals all of the choosen components will be integrated into a feedback system for slip detection in the arm.

1.2 Previous project contribution

This thesis will be improving on another tactile sensor in the literature review, which is the deformating tactile sensor using cameras to simulate optical fibres [10]. This previous project however did not include a feedback system to adjust the grasping force, only to simulate slip detection. This method was published in march 2007 and no improvements can be found on this. Technology has changed in the 9 years which leads to cheaper sensors which have high performance in tactile sensing applications.

1.3 Project objectives

The project objectives is to improve an existing concept, which is a deforming tactile sensor using an optical sensor. This existing concept will be modified to be included into a feedback system with a robotic arm to correct grasping force. The following improvements are:

- Improvements can be in sensor size, it must be small and compact. The previous project has used a 5 cm diameter circle with a length of 6 cm [10].
- The sensor must be cheaper as well the sensor should cost less than $50 to produce. The previous project used 3 cameras as sensors.
- The sensor must be adaptable to fit any robotic gripper.
- The feedback system should be fast enough to detect slip in milliseconds.
- The sensor should be able to detect slip that is less than 1 mm.
- The slip detection and correction should not damage the object being lifted.

1.4 Budget review

The budget for this thesis project is $300, other resources can be provided by Macquaire University for free such as 3D printing, hand tools and measurement devices. The majority of the budget was spent on the mouse sensor which costs $90 AUD ($35 for one sensor) including shipping. Around $15 was spent on the rubber block which can be cut to various sizes. The Robotic gripper costs $700 with all the motors, gripper arm and hardware which was provided by macquarie university. The infrared sensor emitter and detector was also provided.

Chapter 2

Background Theory

2.1 Literature review

This thesis will be creating a gripping robot with a photo sensor and prove that it works or provide improvements so that it does work. This article is about the design of a tactile sensor with an optic fibre and a sensor to detect slippage. Various issues mentioned in this article are optical sensors used, types of lighting and their results, camera positions, colour of object and materials. The method of gripping is that it uses a flexible material with small ridges or holes in it. These holes will contain the optical fibre and photo sensor too as seen in figure 2.1 [10].

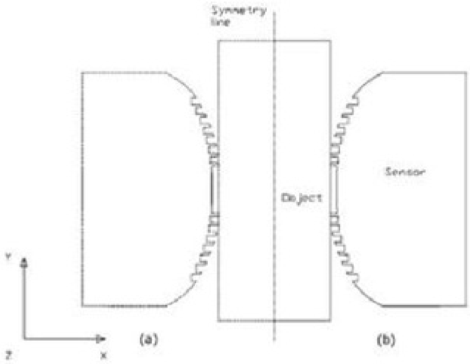

Figure 2.1: Tactile sensor model from previous design [10].

When the object is gripped there will be no deformation (a) but when there is not enough force it will then start to slip. When slipping, the material will deform due to friction between the object and the gripper. This will then cause a change in light intensity of the sensor because of the deformed holes becoming a different shape as seen in figure 2.2 below in (b) [10].

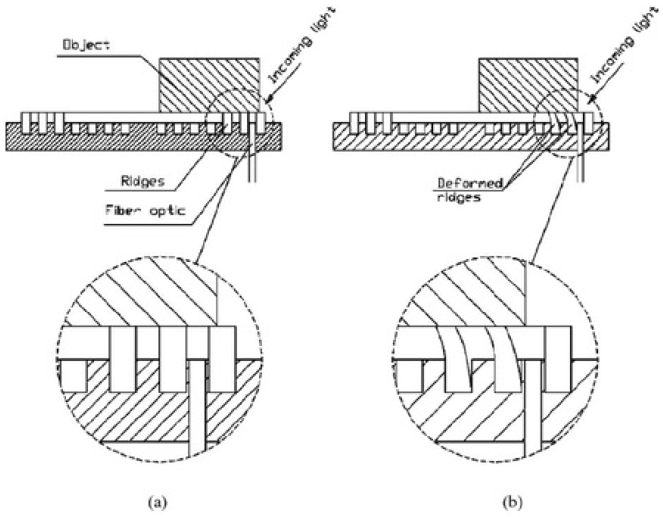

Figure 2.2: Deformation of sensor with (A) having no slippage occurring. Right image (B) with slippage occurring [10]

From this article, we can consider all the issues already presented. We can use the method for this except with an infrared emitter and detector. We should also consider materials since a material that is too flexible (soft) will always bend even when just holding the object without lifting or moving yet. On the other hand, a material that is too rigid will not bend at all and will not detect the slipping object until it has completely fallen out. So experimenting with different materials to find a suitable one is essential, then having a rubber layer on the outside to increase friction between the gripper and object as mention in the above literature.
Therefore the desired information from this section is that the tactile sensor to be designed should have a cavity which the optical sensor is placed inside. The main concept should be that the material deforms and attenuation of the reflected light occurs when an object is slipping due to insufficient gripping force. The proposed design will modify this above sensor to having only one cavity instead of multiple to reduce amount of sensors used. The reduced amount of sensors will decrease the money spent as well as compacting the sensor down to a small size which is the goal of this thesis.

2.2 Industry grippers

This will be the first literature to review because it provides an insight into designing a robotic grip system. It talks about how complex a robotic gripping system can be according to the task to be performed. This article explores the types of gripping arms whether they are 2 claws/prong or multi claw. The angle in which the gripper approaches

the object is also to be considered as approaching at an angle is very complex as problems can arise. There is also a lot of emphasis on using the right amount of force as it will cause multiple problems [5].
The following arm types are mentioned in the literature with pneumatic a more preferred type used because it can apply a small pressure and keep on applying it until it grips the object. The material used in the gripping is also considered as rubber material is most preferred since it has high co-efficient of friction. There is also a section on how large the gripping machine is since it can affect how long the production line is. Finally, there is a section on increasing reliability of the gripping robot with the following essential points [1]:

- Gripping the parts securely will decrease the likelihood of dropping the object while in motion.
- Encompassing the whole object will help to hold it securely and align it in the gripping jaw in the presence of uncertainties in the surrounding environment.
- Minimising the finger length will also help to securely hold the object since longer than necessary fingers will increase chances of misalignment.
- There is enough approach clearance; this is mentioned above on how large the machine is. This is whether the machine might hit another machine, wall, equipment, etc.
- Designed for proper gripper type interactions, the shape of the gripper should match the object which is being held.

Therefore if this was used in an industrial environment, reliability of the tactile sensor and gripper system can be improved by having sufficient gripping force which is mentioned in the above point. Creating a specific gripper shape to encompass the object will also increase reliability such as a curved shape gripper to pick and pack pipes or round objects, a flat gripper to pick up flat surfaces such as boxes, etc. Having a smaller tactile sensor will minimise the finger length which in turn minimises the chances of misalignment. Using a small gripping arm will minimise the chance of collision with unexpected objects.

2.3 Tactile sensors

There are many types of tactile sensors that have been developed. Optical tactile sensors have an advantage over many of the sensors that have electronic components in them (such as the strain gauage) since they do not have current in the contacting skin which can affect some applications. This can be pick up sensitive instruments and objects affected by elecromagnetism. Combining two methods of tactile sensor also increases the reliability of the system in case of failure.

2.3.1 Strain gauge

A mteallic strain gauge is place in a flexible material. When slip occurs, the strain gauge will start to stretch and it electrical resistance will start to increase. These can be small enough to be placed on a human sized robotic hand. [8]. These types of sensors have current running through it therefore it would not be an ideal tactile sensor to be used for a universal gripping system that lifts objects with unknown weight and material properties (whether it is affected by electromagnetism). This sensor was considered for a dual tactile sensors which uses optical and mechanical sensor for a secondary slip detection method incase the optical sensor fails.

2.3.2 Ultrasonic Force sensor

This sensor is used to measure the thickness of objects and detect slip, there are multiple ultrasound sensors within a elastic skin. When the object is graspped it will deform the elastic skin. The ultrasound on the other side of the skin will then read that there is an object in the a certain area due to the sound waves bouncing back earlier than expected. Slip occurs when the sensors start to see that shorter waves are being produced downwards [12]. This type of tactile sensor can also be considered to be used in a dual tactile sensor design however it also has current running through it which is the same problem that the strain gauge tactile sensor has. This is also used as a secondary slip detection method incase the optical sensor fails as well as showing the position of the object in the gripper. This will allow the gripping system to ensure that the object is in the correct position and minimise misalignment when grasping.

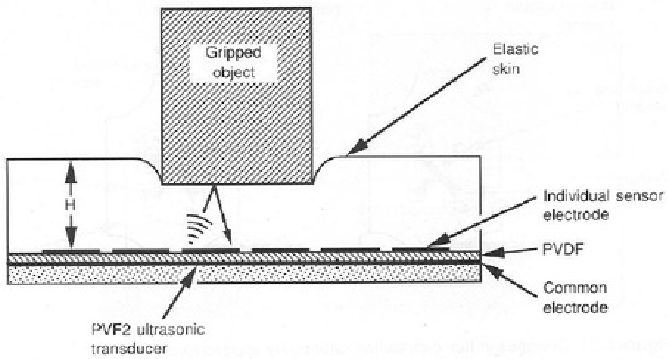

Figure 2.3: Ultrasound tactile sensor sensing how close an object is [12].

2.3.3 Optical Fibers

Optical fibers operate on internal reflection of its light. When used in a gripper robot, when an object is slipping the optical fiber will bend towards the direction it is slipping and cause attenuation. [8]. This is the optical sensor method used in the previous work contribution, however we will replace this with an infrared sensor to decrease the cost and size of the sensor.

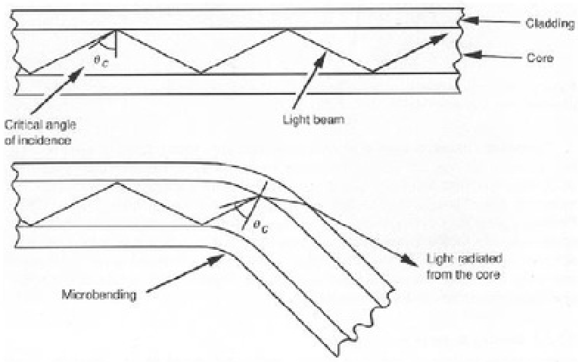

Figure 2.4: Attenuation due to microbending of a flexible optical fibre [8].

2.3.4 Optical decompression

This tactile sensor uses a cavity in which it emits light into, and is reflected back at the detector in the same position. As the cavity is compressed, the intensity of the radiation can be described by Reimer and Baldwins equation which describes the scattered energy will increase as the thickness of the decreases (decompressing cavity). The below image shows how it works when compressed and uncompressed [9]. This optical tactile design has an advantage that it is not affected by ambient light as it is all incased in its own rubber cavity. This controls some of the external environment factors which allow it to be used outdoors under the sun.

2.3.5 Opto-Mechanical

This sensing method will have a elastic material as the contact skin connected to a board with various holes. The skin will have a small spike connected to it and when and object is gripped it will push the skin down and break the line of sight for a photo emitter and diode. When the object is slipping, it will move down the sections of theres holes and slip can be detected [8]. This is a good example of a optical tactile sensor due to its simplicity and low cost of components. However the array of photo emitters and detectors will increase the size of the gripper which is undesirable from the project objectives in the introduction.

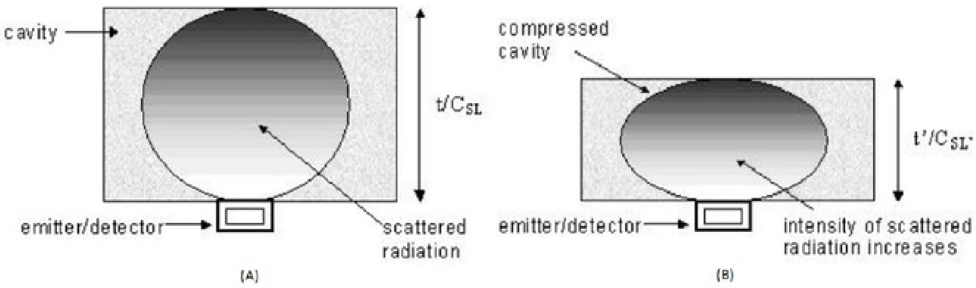

Figure 2.5: Optical cavity sensor uncompressed state in (A). The cavity sensor under compression which increase intensity of radiation in (B) [9].

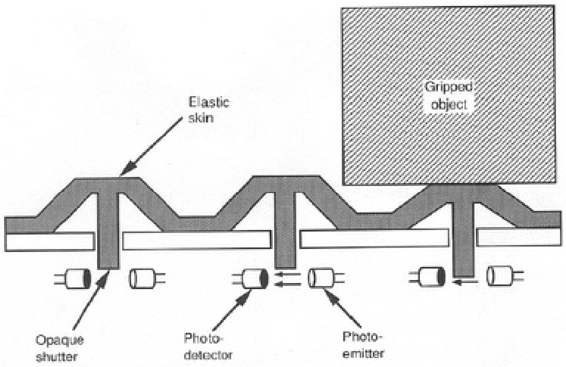

Figure 2.6: Opto-mechanical array breaking line of sight when slip occurs from right to left [8].

2.3.6 Opto tactile sensor for surface texture

A small bump will travel across the surface, since the surface is rough; it will have ridges. These ridges will be in contact with a mirror which will move it reflecting light onto the photo transistor. These can be translated into voltages and show if an object is rough due to how it behaves. A rough surface will have lots of spikes in voltages which as smooth surface will have little to no voltage spikes [15].

2.4 Optical sensors

Infrared emitter and detector sensors are desired for this tactile sensor because it is small, cheap, has a low response time in microseconds [23] and can have wires soldered to the leads which allow it to fit in any mount. Therefore most of the project objectives are achieved with this optical sensor. A mouse sensor operates similar to an infrared sensor, it uses an LED too however instead of a phototransistor it uses a small low resolution camera which an image is taken thousands of times a second. Using a mouse sensor achieves all of the project objectives that the infrared sensor does as well as detecting changes in position when the objects slips less than 1 mm. Using a mouse sensor will increase the cost of producing a mouse sensor however it will still be under $50 to produce a tactile sensor with higher performance in achieving the project objectives.

2.4.1 Infrared emitter and detector

This type of sensor is used due to its compact size, it consists of an emitter and detector pair as seen below in fig. . The emitter can be an LED which can have visible light as well as Infrared or even UV (ultra violet) light. The detector which is usually a photoresistor will then vary its resistance according to how much IR light it is exposed to. There are various factors that can affect the result of this such as the surface of the object it is reflecting from [2]. If the surface of the object is not very reflective e.g a black surface, it will not lower it will not have a desired response, therefore a white or metallic surface is desirable. Ambient light is also an issue for an IR emitter and detector, this can be solved in many ways such as modulating the signal e.g. 38k modulation so the detector can only look for the IR light at that frequency [23]. Using the IR detector and emitter in the dark can also prevent ambient light affecting the sensor.

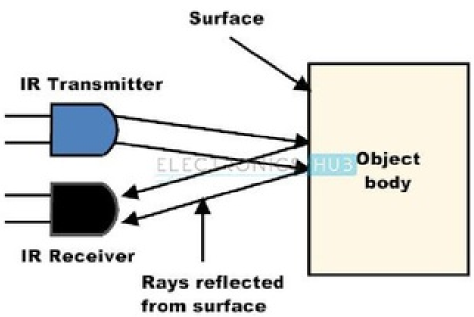

Figure 2.7: IR emitter and detector pair detecting proximity of an object [2].

2.4.2 Mouse sensor

There are two basic mouse types, optical and laser mouse. Both mouses have similar components, both have a IC with a CMOS sensor, a light pipe (lens) and a LED [13]. These moues generally use a red light source because red LEDs were cheaper and readily available however any colour could be used. There are blue LEDs mouses with special sensors to detect UV light it emits.

Optical mouses use regular LED as a light source while the laser mouse uses a laser. They both will illuminate the surface below the CMOS sensor which consist of an array of photodetectors that will capture the image. These mouses will capture from 1000 to 6000 images per second, these images are then sent to the digital signal processor (DSP). The DSP will then analyse the images and determine the position of the mouse and direction if it has been moved [27]. When using the optical mouse on a very smooth surface or a less reflective surface they might not function effectively and they also have less DPI then a laser mouse.

Laser mouses work by using the laser Doppler velocimetry principle which the laser produced will reflect off the surface and be detected at a different frequency when emitted due to movement of the mouse [6] .

Figure 2.8: Mouse sensor reading the surface with an LED and low resolution camera [13].

Some issues with mouse sensors are the acceleration and lifting off distance. Optical mouses do not a lot of acceleration, which can also be called resolution error vs speed. This is caused by noise captured with the camera, the mouse may not keep up and can overshoot. This means that if a mouse was to move very quickly 5 cm across and the same mouse is also moved 5 cm acrossed slowly. They will be in different positions as the quick mouse will overshoot and be further. Lift off distance is the distance that the mouse still operated can when elevated from the surface [6] . Optical mouses have a short lift off distance which means it needs to be close to the reflective surface in order to operate. This issue can be solved by placing the mouse very close to the tactile contacting area.

Mouses also have a parameter called DPI (Dots per inch) which determines how many pixels the CMOS sensor has. In modern gaming mouses, this can be adjusted however older mouses may not be. The higher the DPI the more sensitve the mouse would be mean you can move 2 cm across with the mouse but the mouse will translate that into half monitor screen. This may be usefully for some applications such as confined work spaces. This parameter can affect the tactile sensor providing a false positive result to slippage as there will be some movement during gripping and not lifting.

2.4.3 Dual optical and mechanical tactile sensor

Recent trends in tactile sensor designs have a mechanical and optical sensor layout. In figure 2.9 the reflective ball will move when it is gripping and the springs will move backwards towards the light sensor which is the component labelled 9. When slip is detected, the ball will move in the direction of the slip so the springs will compress in this direction too showing move voltage towards one side. The optical fibre will detect the movement and change in position of the ball,

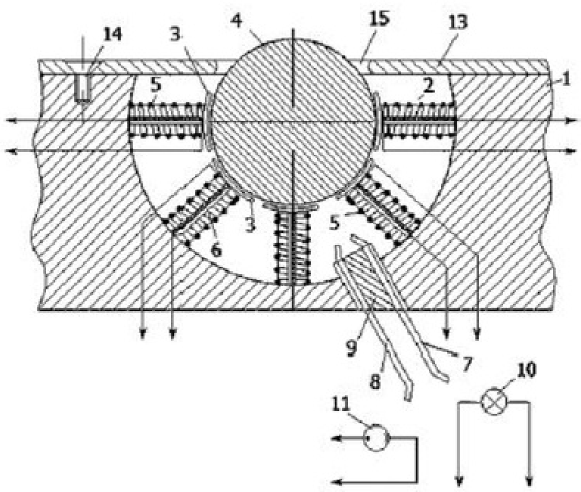

Figure 2.9: Dual mechanical and optical layout of a tactile sensor [24].

2.5 3D printing

A 3D printed adapter has been suggeseted to hold the tactile sensor onto the robotic arm. This is an advantage because there are many types of robotic gripper arms so a 3D printed mount can reduce the cost of converting the design in this thesis. The Fused Deposition

Modelling printed is used because it is a cheap and common small 3D printer [19] which was available from Macquaire University.

The 3D printer used is the UP Plus 2 3D printer which uses Acrylonitrile butadiene styrene (ABS), the resolution can be adjusted so it prints a very high resolution but at a lower speed. The setting choosen will be a low resolution so it can print faster however it will fill in solid blocks in the CAD design with a rectilinear filling as seen below in figure 2.12. Therefore using 3D printed casing which mounts the tactile sensor to gripper arm will increase the adaptability of this design which is a project objective.

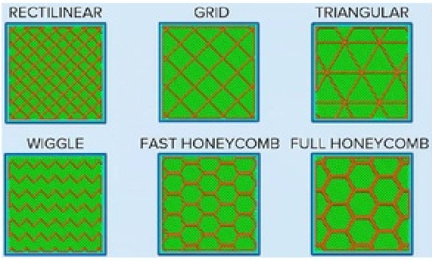

Figure 2.10: 3D printed filling patterns to create strength in printed hollow objects adding structural strength [17].

2.6 Deformation of a cantilever beam

The main principle behind on how this tactile sensor operates is the deflection of a cantilever beam. This is because the tactile sensor is fixed on one end to the robotic gripper and a load will then be placed on the other end of the beam, which is the weight of the object being lifted. There are various deflection formulas to be used depending on how weight is distributed on the beam, an example can be a hollow rectangular beam as seen below in figure with an end load only2.11. To calculate the end loaded deflection of a this beam the variables required are area moment of inertia of the beams cross section (mm^4), force applied at the unfixed end of th beam (N), Length of beam and modulus of elasticity in the material of the cantilever beam [11].

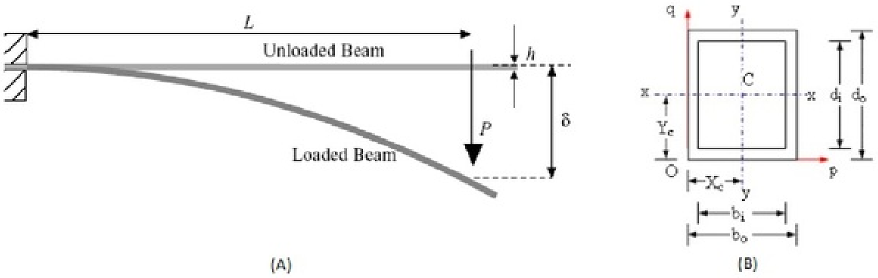

Figure 2.11: (A) Deflection of a cantilver beam (B) The cross-section of the beam [11]

2.7 Mechanical properties of materials

The material used for the deforming part of this sensor will be made of rubber. Rubber is choosen because it has a high co-efficient of friction which will make it an ideal material to hold the object and prevent further slipping. Compare this to a smoove surface such as plastic or metal, they will slip off the object unless a large amount of force is applied which can damage the object. The rubber will also prevent scratching on the surface of the objected lifted and when slip occurs compared to metal or plastic as the contact material. Some mechanical properties determines the limits of deformation in the rubber is Yield Strength which is the point where the material stops all elastic behaviour which is the elastic region shown below in figure 2.12 [16]. Within this region, the material will return to its original shape however above it is when its starts permanent deformation that is undesirable.

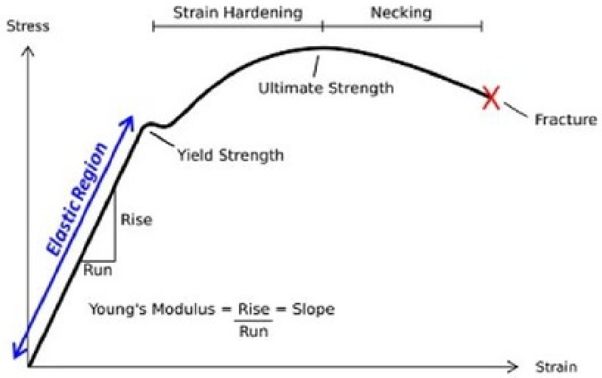

Figure 2.12: Stress - Strain Curve of a ductile material [16]

2.7.1 Shore hardness

The shore hardness test is used to determine the toughness of a material, usually in polymers and how they resist permanent indentation [20]. The shore hardness can provide an estimate on how difficult a material is to cut as well as how stiff a polymer is. In figure 2.13 the desired stiffness of the rubber deforming component will be in the soft region which is a 50 to 70 in shore 00 scale, 10 to 30 in the shore A scale however not available in shore D scale.

Figure 2.13: Shored hardness scale types converted between each type [20].

Chapter 3

Deformation component design

This chapter will anaylse the rubber component of the tactile sensor as seen in below in figure 3.1. It will calculate the deflection of a cantilever beam which was mentioned in 2.6 (not to scale, full dimensions are in appendix A). The deflection formula will be an end loaded cantilever beam and a hollow rectangular beam. This is because this shape is easier to cut as well as being able to lift objects with flat surfaces e.g boxes. This tactile sensor will initailly be lifting cans only because they have a standard dimension as well as a reflective surface for the optical sensor.

Ethylene Propylene Diene Monomer (EPDM) is choosen as the rubber component due to it's availablility as well as having a shore hardness of 70. It also has a high coefficient of friction due to being a rubber and addition properties such as water resistance, UV resistance, cyclic fatigue resistance [7]. These additional properties allows this sensor to be used in the external environment and exposure to the sun and wet weather.

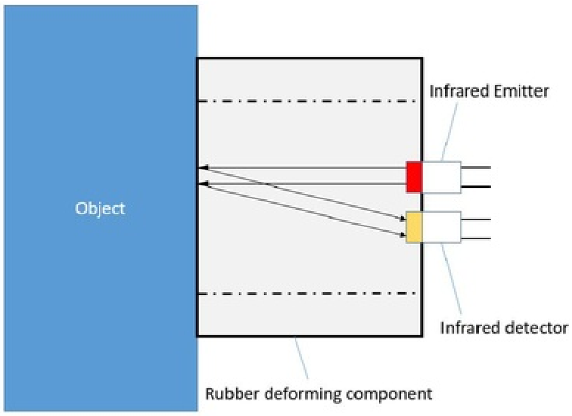

Figure 3.1: Proposed optical tactile sensor design.

Properties	Expected values (MPa)
Tensile strength	25
Yield Strength	14
Young's modulus	16.79
Shear modulus	5.5

Table 3.1: Mechanical properties of EDPM [7]

3.1 Deforming material anaylsis

The proposed design of the tactile sensor will look like the figure 3.2. It will have the following mechanical properties. EPDM is used because it was readily available for purchase and the shore hardness test score is 70. The sensor will be 2 cm by 3 cm and will have a thickness of 2 cm. The expected deflection for the 2 cm length is 0.155 mm. The deflection can be reduced by reducing the length of the beam. The load applied will be similar to the weight of a soda can which is 0.390 kg.

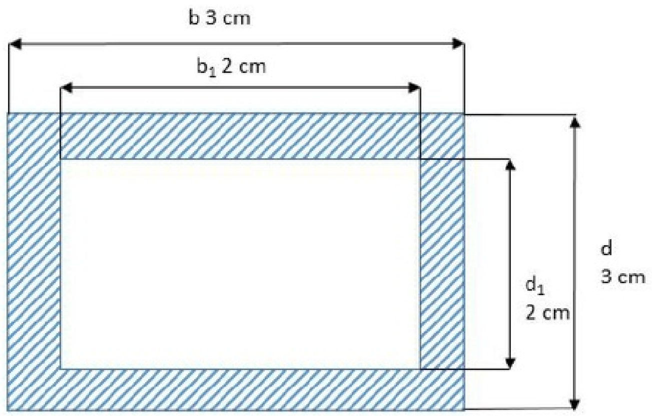

Figure 3.2: Dimensions of the proposed tactile sensor.

To calculate the area moment of inertia we will minus the outer with the inner area using the following equation 3.1 and from figure 3.2, b is 3cm, b_1 is 2cm, d is 3cm and d_1 is 2cm. The area moment of inertia will equal to 5.416 x 10^4mm^4 using the equation below.

$$I_{xx} = \frac{bd^3}{12} - \frac{b_1 d_1^3}{12} \quad (3.1)$$

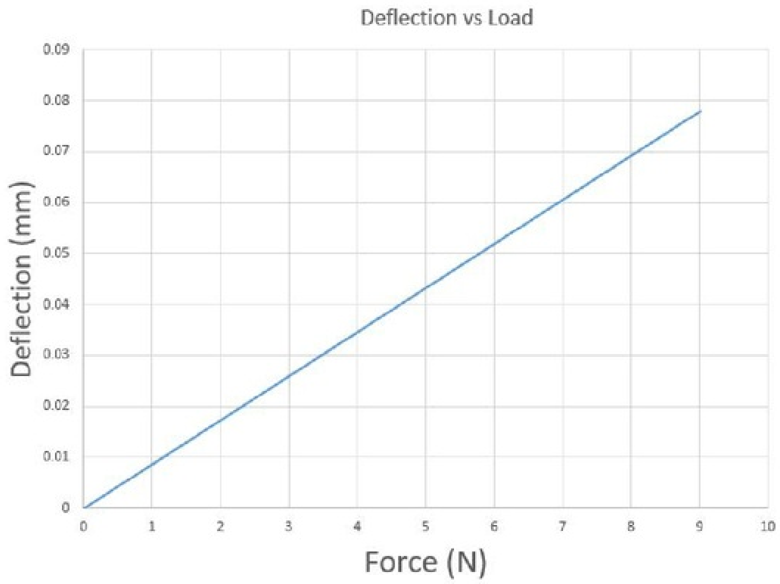

Figure 3.3: Expected Deformation of straight design

The deflection of an end loaded cantilever beam is the equation 3.3 below. It wil also use area moment of inertia of a hollow rectangular beam which is seen in equation 3.1

$$\delta = \frac{FL^3}{3EI} \tag{3.2}$$

Where:

δ = Deflection of beam (m)

F = Force or load (N)

L = Length of beam (mm)

E = Modulus of Elasticity (N/mm^2)

I = Area moment of Inertia (mm^4)

3.1.1 Simulation of deformation

The deflection of just the weight of can is shown in the below using Autodesk inventorto simulate it. All mechanic properties of the rubber is in the above Table 3.1. In figure 3.4 we simulate how the deflection changes as the length is change. The longer 2 cm thickness beam (left image) has deformed more than the shorter 1 cm thickness beam (right image). This simulation matches the hand calculated value however shows where the sensor will deform. The load for each simulation is 3.82 N load of the edge of the contacting surface.

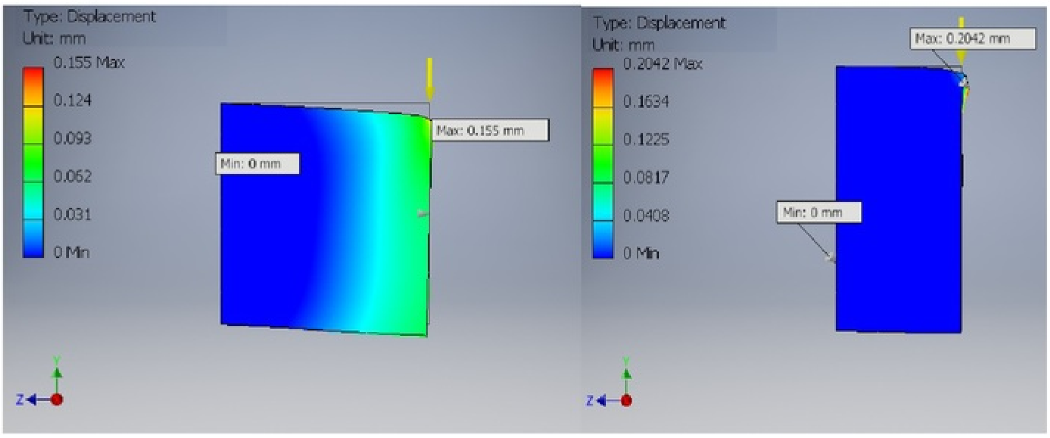

Figure 3.4: Simulation of deflection with 2 cm length (left) and 1 cm length (right) with a 3.82 N force one the edge of both sensors, simulating the sensor holding a can.

3.1.2 EPDM rubber

Cutting the EPDM was difficult, the standard method of cutting was a sharp knife. However this was much too difficult so a hack saw was used instead to speed up the process. The rubber was much more stiff than expected which is shown in the results of calculation and simulation.

Figure 3.5: EPDM rubber component straight shape cut.

3.1.3 Simulation of curved deformation

An alternate tactile rubber design was made to fit around a can. This is expected to better accomodate the shape of a can, with a measured can being diameter of 6.5 cm. As mention in section 2.2 encompassing the object that is going to be lifted will increase stability from external environment and help to hold it securely. This curved design is created to specifically lift cans, other various shapes can be created to lift certain objects. A full detailed CAD of the dimensions of this sensor is in appendix A.3.

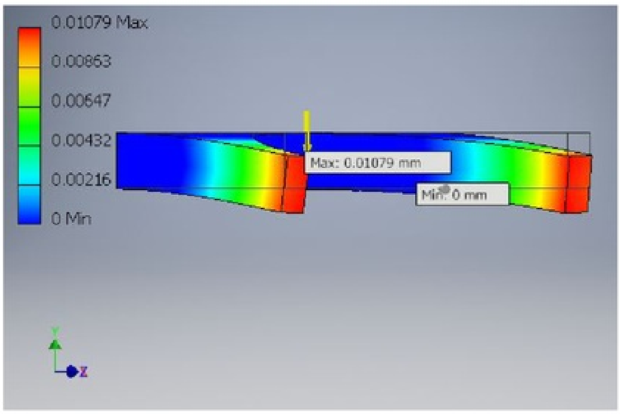

Figure 3.6: Simulation of curved design deflection

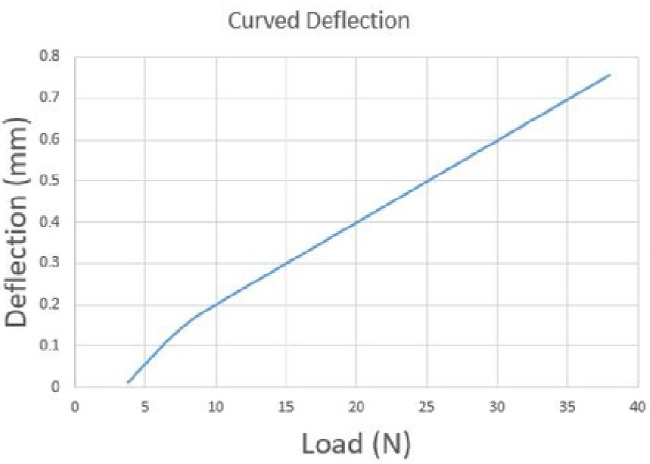

Figure 3.7: Expected curved design deflection

Shape	Force applied (N)	Length of rubber (mm)	Displacement (mm)
Straight	3.8	10	0.155
Straight	3.8	20	0.2085
Curved	3.8	10	0.01079

Table 3.2: Comparison of curved and straight deflection

3.2 Comparison of designs and deductions

The results shown are very small e.g 0.155 mm of deflection which can be difficult to be viewed by the visible eye. A critical error has been made when purchasing the rubber block as it displayed the shore hardness however there are many variations of the shore hardness tests as seen in Section 2.7 and figure 2.13. The displayed shore hardness of 70 on the rubber block was interpreted as shore hardness 00 which is a soft rubber however the advertised had the shore hardness A scale which is a medium hard rubber which is as stiff as a car tyre.

Simulating as well as calculating the deflection of cantilever beam gives details on how the sensor will behave when it is loaded. As shown in figure 3.4, the rubber will have a deflection of 0.155 mm and a third of the beam will start to bend which is not shown in the calculations as it will only reveal that it has a 0.155 mm deflection. When the same load is used except a shorter beam length less deflection is observed and the most deformed section will be on the top edge of the beam. When the curved beam is loaded, we can see that most of the deforming section will be focused on the prongs of the beam. This will require a different sensor placement because the whole curved surface which should be in contact with the object is not being deformed. Comparing hand calculations and running a simulation has determined that the curved shape is not suitable for this tactile sensor

Graphing the expected deflection which the weight of three cans a straight line is observed. This is because as mentioned in Section 2.7 during the elastic region it will have a straight line characteristic however if it passes the yield strength of 14 MPa in Table 3.1 it will start to deform permanently. In this case, the limit of three cans has not reached the yield strength of the EPDM rubber.

We can conclude that we will require a shore hardness of 70 on the shore 00 scale or 30 on the shore A scale to be able to cut the soft rubber to shape. The tactile sensor dimensions will stay the same however a different material be used to replace EPDM. A desirable deflection would be 5 mm or 0.005 m. rearrangeing the deflection formula we get the following equation where force is still 3.82 N, Length is 10 mm, deflection of the beam is the desired 0.005 m and the area moment of inertia in the x axis is still 5.416 x $10^4 \mathrm{mm}^4$. The Young's modulus will be 4.677 N mm^2 which is equivalent to low density polymers such as silicon, polyethylene, nylon, etc [3].

$$E = \frac{FL^3}{\delta 3I} \tag{3.3}$$

Chapter 4

Infrared sensor

This chapter will have an overview of the optical sensor used which is an IR emitter and detector pair as well as the microcontroller used. The microcontoller used is an Arduino Uno which is readily available, cheap and small in size. The Arduino is also choosen because of past experience with the software coding. An IR emitter and detector is used because it is low powered, cheap, readily available and small in size. The IR detector is also used instead of a regular photoresistor because of ambient lighting which may affect the sensor as mentioned in Section 2.4.1.

4.1 Experiment 1: Infrared detector and emitter

The infrared detector and emitter will look be based off figure 4.1 and will be connected to the micro controller (Arduino) in figure 4.2.

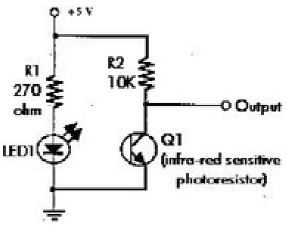

Figure 4.1: IR detect and emitter circuit [26]

We expect to see the result below in figure 4.3 as voltage becomes high when an object is closer. The voltage will then decrease as it is moved away. This experiment is decsigned to show that when the gripper is applying force there will be a change in voltage. After it has slowly decreased it will stant in a constant range as it is lifting the object. When slip occurs we will see a change in voltages, spikes may occur as the object is deforming the tactile sensor.

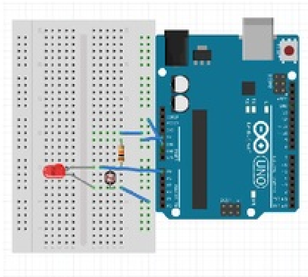

Figure 4.2: IR circuit and microcontroller

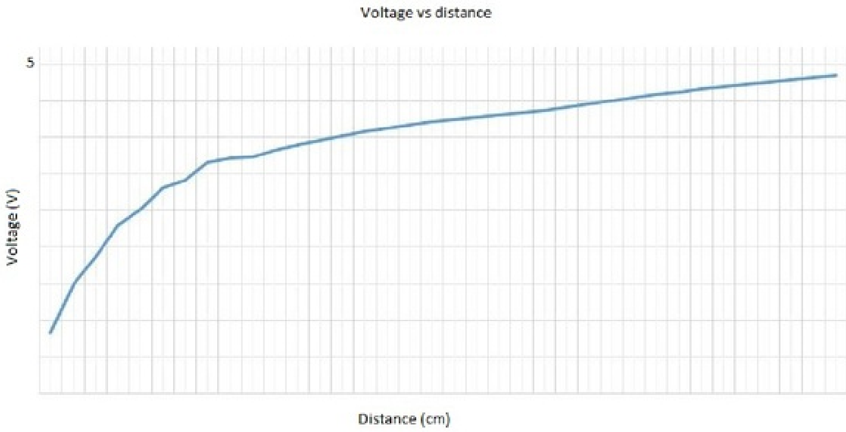

Figure 4.3: Expected Arduino voltage output

The first experiment is to test if the sensor design works. In a dark room, a white card is attached to a caliper and is placed 1 cm above the IR detector and emitter and moved 5 cm upwards and back down three times. The white card has a thickness of 3 mm we account for this by moving the caliper from 53 mm to 3mm. Results in figure 4.5 show there are a spikesr whenthe card is close. The second experiment is moving a piece of paper 1 cm from left to right 3 times. In figure 4.6 we see that there are 3 spikes to show movement of the paper. The second experiment confirms that we can use this sensor design as the tactile sensor will move downwards if slip is detected which is similar to the sideways movement as the tactile sensor starts to vibrate as it is dropped.

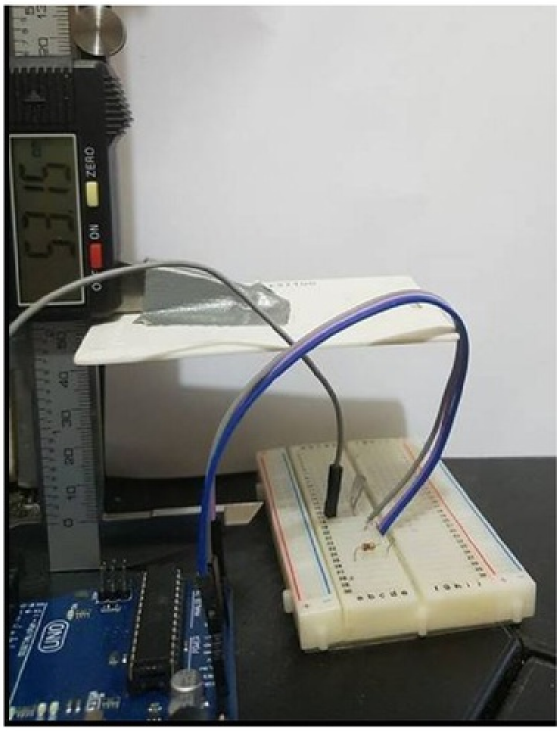

Figure 4.4: Caliper experiment setup

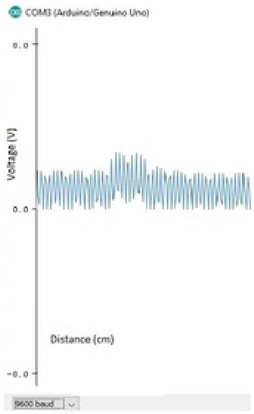

Figure 4.5: Voltage vs Distance of simulating grasping an object

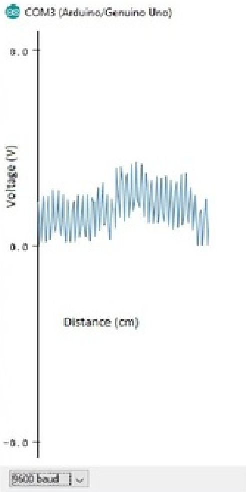

Figure 4.6: Voltage vs Distance of a slipping object

4.2 Infrared sensor result

The results of the IR sensor test have shown the expected and desired voltage outputs to be matching. As the object is is closer, the voltage will increase as expected and observed. This is shown in figure 4.5 as voltage versus time in the Arduino output monitor but we expect that a certain distance will have a unique voltage. The maximum voltage output is observed at 0 cm or touching the infrared sensor will be 5 V output. The response time of the sensor can be as small as 15 microseconds [18]. Therefore this sensor has the desired results as well as fitting the project object of being small in size and cheap. Since the circuit is very compact, this sensor will help to contribute to a smaller design since the deforming rubber can be reduced in size. The overall cost for the circuit and microcontroller is under $20.

Chapter 5

Mouse sensor

5.1 Sensor overview

This sensor is used because mouse sensors have a small response time, giving a instantaneous response which is required. As mentioned in section 2.4.2 it captures at least four thousand pictures every second. Using this sensor can be an alternative to the deforming material tactile sensor, this sensor does not require a deforming material. It can read movement in x and y axis and give distance travelled compared to just voltage by the infrared sensor. This means that it can detect slippage in less than 1 mm and at a much faster rate then the infrared sensor. This sensor is also relatively cheap at $35 AUD at the time of purchase.

5.2 Sensor operation

This sensor operates by detecting the image below it, if it is not moving then the movement bit is low. If movement has not occured in a while, standby mode is active which reduces power consumer by the sensor. When movement is detected, the motion pin will become high and address 0x02 [25] will update. This will then tell the two x axis registers Delta_X_L and Delta_X_H (address 0x03 and 0x04) and two y axis registers Delta_Y_L and Delta_Y_H (address 0x05 and 0x06) to read and update its bits.

5.3 Surface quality

This is the amount of features that the sensor can identified in the current frame. As mentioned in section 2.4.2 the mouse sensor will take thousands of pictures and identify features of the surface it is on. When the mouse sensor is moved the features identified have their positions tracked. Surface quality in the ADNS9800 sensor is the SQUAL register (0x07 address) will identify the amount of features on the surface [25].

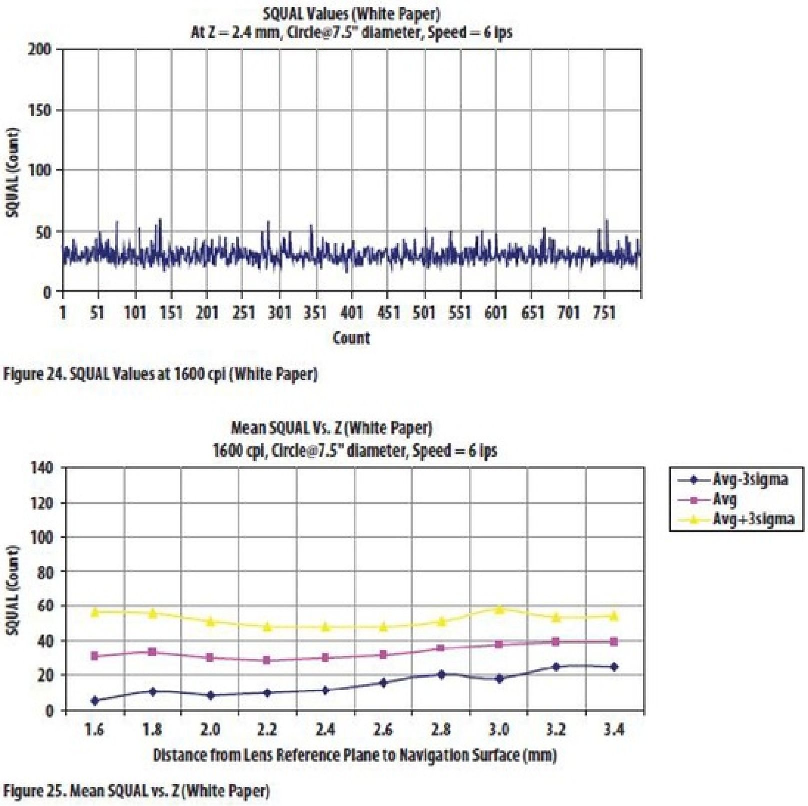

Figure 5.1: Expected Surface quality on white paper [25]

5.4 Lifting distance

The lift detection of is important as mention in section 2.4 will determine which surfaces it can read on and the dimensions of gripper cover. The nominal lifting distance is 2.4 mm [25], the 3D printed cover for the sensor is 2 mm from the sensing surface. The lifting distance will change for each surface as well as lens used. The lens used is a ADNS-6190-002 gaming laser lens.

5.4.1 Experiment 2: Lifting distance

In Table 5.1 a paper with text is placed under the sensor and moved 5 cm horizontally. Any response from the Delta_X_L or Delta_X_H will indicate that it was able to identify a feature process it to a change in x-axis from the arduino code. We will do this on a white paper table 4.1 as well as a soda can with it's text table 4.2.

Table 5.1: Lifting distance on Coke can and White paper

Lifting distance (mm)	Mouse sensor active(Coke Can)	Mouse sensor active (White Paper)
0	Yes	Yes
1	Yes	Yes
2	Yes	Yes
3	Yes	Yes
4	No	No

5.5 Operating temperature

During normal operation, the sensor when touched was hot. Further tests had to be conducted to determine the temperature during normal operating conditions. This is because the sensor has no heat sink or any other thermal management and may require one. The picture in Fig. 4.2 had a reading of 41.6 degrees celsius during 30 minutes of continous back and forth motion over a book. The absolute maximum it can operate at is 85 °C therefore this sensor does not require any heat management and will not soften the abs plastic used for the 3D printed housing. The ABS mount will start to soften and reach its glass transition phase at 105°C and become almost liquid at 230°C [22].

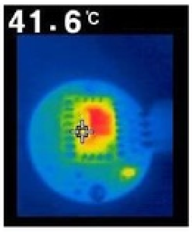

Figure 5.2: Infrared image of mouse sensor

5.6 Limitations of sensor

Voltage supply: The default voltage supply will require 5 volts from a DC power source on the sensor. This requirement stems appears from the circuit board purchased with the sensor which it is mounted on. This sensor has two operating modes, a 5 volt mode and 3 volt mode. For the 5 volt mode, the minimum required voltage is 4 V and the maximum is 5.25 V. These two voltages can be supplied by the arduino 5 V and 3.3 V pins. To activate 3.3 V mode, the some tracks behind the sensor will be required to be cut.

Lifting Distance The lifting distance of the sensor can be between 1mm to 5mm depending on the surface plane it is reading. This can further be increase by changing the lens from the ADNS-6130-002 gaming lens to a different lens ADNS-6130-001 or created however this will be extremely difficult as a scratch on the lens will create several reading errors. The nominal reading distance is 2.4 mm

Reading speed This sensor has a maximum reading speed of 150 inches per second (381 cm/sec) which means that it can only detect a length of 150 inches a second. If speed of the sensor is higher, it will result in more distance travelled which the sensor will not be able to keep up and process the incoming information.

Operating temperature The adns9800 will have a maximum operating temperature of 85°C and a minimum of 0°C to operate. However in the datasheet there is no mean temperature it operates at.

Frame rate This sensor has a default of 4,000 frames per second which can then be increased to a maximum of 12,000 frames per second. This will control the response time of the sensor as it can detect movement in millimeters in less than a second which is required to create an adequate tactile sensor.

Reset delay This is the time when it is idle and no new images are being processed and then the sensor is moved again. This movement will activate the motion pin which will result in it being high so the sensor can start processing data again. The minimum time it can start generating data again is 30 milliseconds. This is 0.03 seconds and is predicted to be when the gripper has stopped applying the gripping force and starting to lift the object.

5.7 Mouse sensor mount

In figure 5.4 we see that the 3D printed mouse sensor mount is working with a 2 mm distance between the lens and the sensing surface (left image).

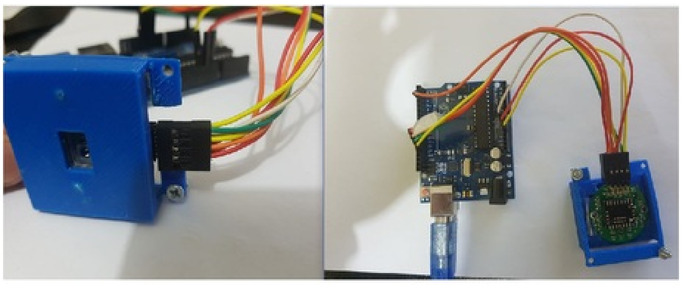

Figure 5.3: 3D printed mouse sensor mount for robotic gripper.

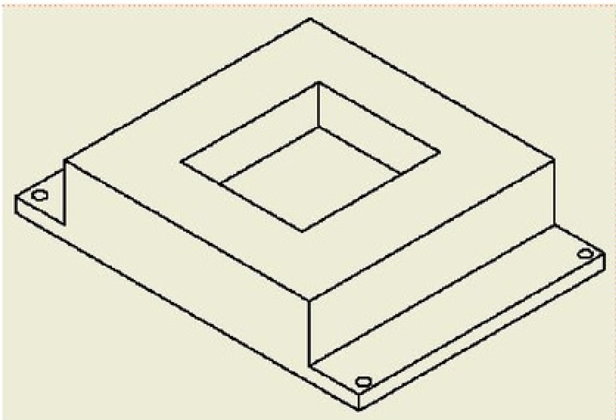

Figure 5.4: CAD of mouse sensor mount to be attached to the gripper .

5.8 Mouse sensor results

The results from the experiments above show that the lifting distance in Table 5.1 can detect motion at 3 mm above the surface a soda can and white paper however greater distance than this will result in no motion detect. This will mean that the maximum thickness we can make the lens to the surface is 3 mm. The thermal image of the sensor shows that the sensor has a temperature of 41.6 ° when used continously in the 3D printed mount. This shows that it is under the maximum operating temperature in section 5.6 and there is sufficient air flow therefore now further heat management is required as it is also below the glass transition phase of 105°C. There are limitations to this mouse sensor however they do not restrict the design of the 3D printed mount too much other than having a limit of 3 mm from the surface. This sensor however will restrict the size of the tactile sensor used since it has a depth of 9 mm and a diameter of 34 mm.

Chapter 6

SG gripper

The gripper robot used is a AX-12A smart robotic arm with an SG gripper seen in figure 6.1. The arm is weaker than anticipated with the ability to only lift 0.499kg which is just one standard soda can. The joints rotate can at 180 ° so there is some good mobility. The budget for a robotic gripper was $200 or less however this gripper was readily available and already assembled for use.

6.1 Dimensions and characteristics

The dimensions of the gripper also limit the 3D printed mount designs. Having only 6 cm of space when opened, the mouse sensor will have be 1 cm in length. Therefore it limits the objects that can be grasp to less than 5 cm. From the datasheet [21] and measurements by using a spring scale and loading the SG gripper, it can only lift 500 g. The reach of the gripper is 44 cm from the center of base.

Figure 6.1: Measurement of length of gripper and distance between fingers

6.2 Robotic arm software

This arm uses open source software to control it, such as matlab, labview, python, C, C++, eclipse and visual basic. Matlab was choosen because it is a familar program. There is code in the appendix C to control the arm, a gui is created with sliders to control the various servo motors. There was various other codes which were provide from the crust crawler website however they were jsut templates, which was filled in to control this gripper. Limits are placed to stop the arm from rotating too much which may twist the wires and damage the servo motors. This arm has been successfully controlled through matlab however the mouse sensor stopped working before it could be integrated into the software.

A sample gripping program is used to simulate how the corrective grasping robot should operate. In figure 6.2 is how the robot will operate. The tactile sensor will be implemented into the gripping program which the arduino com port will be read by the mat lab.

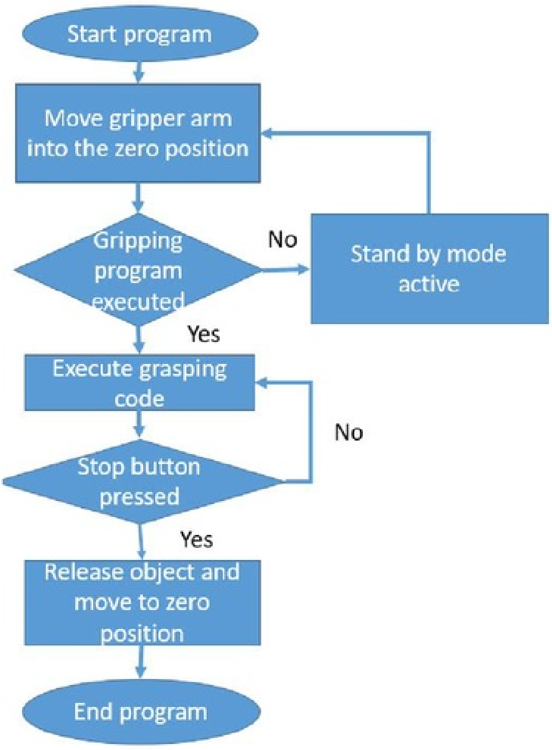

Figure 6.2: Flow chart of grasping robot

6.3 Modification from proposed plan

The gripper arm has the limitations of only lift 0.5 kg and an open space of 6 cm for the sensor and object therefore instead of lifting cans, a new flat panel with a string attached to weights. The weights will then be varied however dynamic weight will have to be accounted for. When the object is lifted, the object will start to swing which could show slip occuring when there is not. To account for this, the speed of lifting will be reduced in the mat lab code with a delay in the lifting code between each iteration.

Chapter 7

Slip detection system

7.1 System overview

The following layout is how the system will work in figure 7.1:

1. The sensor will gather and process images and changes in the x direction.
2. Data from the sensor will sent to the arduino.
3. The Arduino code will process this data into changes in distance. Alternatively, Matlab can read the values from the Arduino without the Arduino software open.
4. Matlab determine whether the sensor has reached a limit so it will have to lower the arm and apply more gripping force.

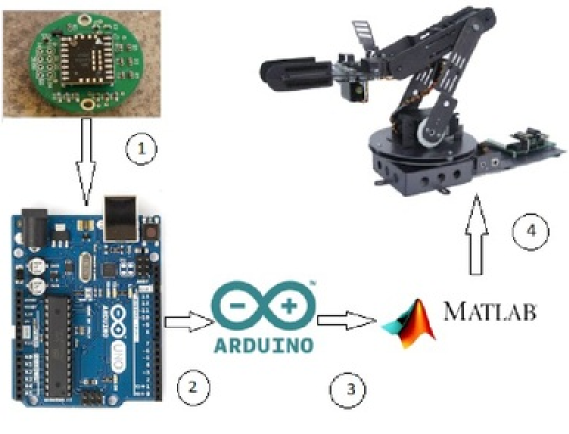

Figure 7.1: Overview of intelligent grasping system.

7.2 Slip detection sensor

Various alternative optical sensors can be used for slip detection besides the use of cameras which this thesis is based of improving. These are also different to the literature review tactile sensors. The following approaches will use different optical sensors however they will all use an Arduino microcontroller.

7.2.1 Infra red sensor approach

This sensor is small as well as cheap which costs $1.95 for an emitter and detector pair. This is the simplest tactile sensor design and approach which is also the cheapest and can be made into the smallest tactile sensor (only required to cover the IR emitter and detector pair which is 1 cm by 1cm and 1 cm length). However due to a mistake in material selection, this sensor may not be used so an alternative method is required. This sensor consist of the deforming rubber component,

7.2.2 ADNS9800 laser mouse sensor approach

This is the choosen as there is code already available however some editing will be required. The code will require 2's complement of the bits to convert them into values we can read. Since this sensor is from America, the distance will be initially in inches which will require the x values to be multiplied by 2.54 (1 inch is 2.54 cm). The CPI will need to be divided by the xydat[], this converts movement into inches where we can convert to centimeters. This sensor is the most expensive at $35 for one sensor. Various other features are mentioned in Chapter 5.

7.2.3 Old mouse sensor approach

This method will take any generic or spare optical mouse available, deconstruct it and use it. A new 3D printed housing will be required for this sensor and jumpers will have to be soldered on to the MISO, MOSI, GND, Voltage input or SDIO, SCK, GND and Vcc depending on which mouse sensor is used. Then coding is required to read the x and y axis registers and process the bits into x and y distances. The cost of this can vary from online stores, a cheap mouse at $5 to a office mouse at $15. Due to both ADNS9800 sensors being unable to read values anymore from unknown reasons and shipping from America to Australia will take two weeks, this approach will be used.

7.3 Feedback system

The feedback system will attempt to use a mouse which has a USB connect to the computer and coordinates of the mouse are read off matlab. This mouse sensor will then require a custom 3D printed gripper mount. Matlab will read the mouse cooridnates and then execute its program to attempt to lift an object, it will attempt to lift it in 3 increasing steps before dropping it and increasing the lifting force by one and a half of the steps.

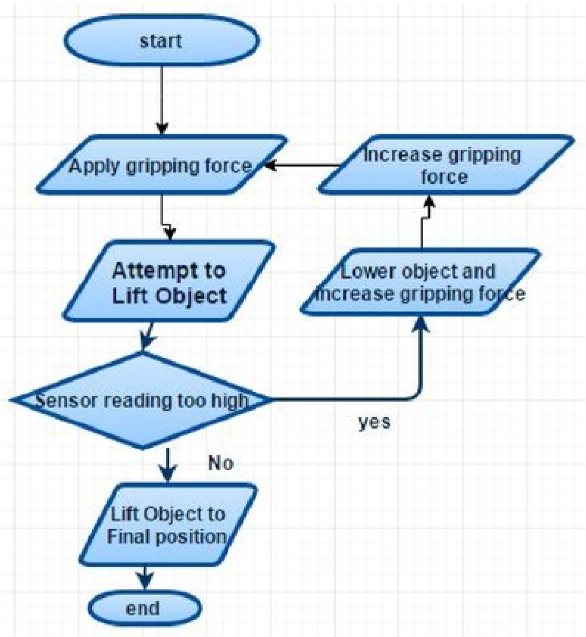

Figure 7.2: Flow chart of operation

7.4 System failure precautions

7.4.1 Optical sensor failure detection

To detect where the optical sensor is working, an arduino boolean code can be used to check where there is voltage in digital pin 12 and 11 which are the input and output pins to the mouse sensor. If there is no voltage in one of these the boolean will return a 0 from the & function and the arduino will make an unused pin high followed by a failure message.

7.4.2 Gripping arm immediate shut down

Using the following Matlab code "function stop_button_Callback(hObject, eventdata, handles)" it can generate a stop button and when pressed it will stop and shut down the gripper. In case of an emergency or accident, a button is used however further conditions can be used to trip this stop function. If the optical sensor is not working, the arduino can activate an unused pin as high and mat lab will constantly check this pin. If it is high, the gripper robot will release its grip and move to the zero position and print a optical sensor failure message. If the optical sensor pin is high and the program executed, the program will not be execute and display a optical sensor failure message.

7.4.3 Drop chute

A bag stretched under the robotic gripper can be used in the case that there is insufficient grasping force while the object is already in the air and slip occurs. This case can happen when an object is long.

Chapter 8

Conclusions, improvements and future work

8.1 Conclusions

In conclusion we can see that from the expected voltage got an increase as it is moving closer and a decrease in voltage as the reflective surface moves away. This confirms that the design will increase in voltage as it is gripping and will stay in a constant range until slip occurs. We can also see that moving the reflective surface sideways, we get flucuations in voltage which means that when the object is slipping it behave in a similar way.

If the tactile sensor deforms too much, the length of the beam can be halved to decrease the deflection. This can be seen in figure 3.4 where the 1 cm thickness sensor has deflected less. If the lifting distance limit is encountered, placing the sensor as close to the end of the tactile sensor as possible can help resolve this. A tactile sensor with a curved deforming shape will not work fir this design since the tips will deform instead of the middle of the sensor which is where it is desired. The hand calculation shows how much deformation is occuring however it does not show how it will look, which the simulation does therefore a conclusion can be made that it is not suitable. A material with a lowered Young's modulus is required since the higher the modulus of elasticity the more force it will require to be deformed.

The results from the mouse sensor however were useful in the application of this tactile sensor. They had a lower response time compared to the infrared sensor. The infrared sensor was much cheaper than the mouse sensor however the mouse sensor achieved all the project objectives where as the the infrared sensor was lacking in detecting slip smaller than 1mm due to the noise it had. The feedback grasping system was not created however the mouse sensor worked with arduino providing changes in distance and the robotic arm was able to move using matlab.

8.2 Improvements and alternatives

8.2.1 Dual sensor with damage detection design

The new trend in tactile sensor design is having two types of sensors for extra protection against slip. There can be many other sensors to choose from however a force sensor is suggested for this. The force sensor will be placed on the contacting face of the tactile sensor. For the slip detection method, when the object is slipping there will be a voltage increases as it moves along the strip.

To determine whether the object is being crushed, the size of the tactile sensor face will have to be increased. When the object is gripped, and there is a large difference in where voltage is on the sensing face e.g only in the middle. This will indicate that the tactile sensor is crushing the object for example when the tactile sensor is trying to grip an box with an object and some empty space between, the gripper will crush the box or packaging.

8.2.2 Internal reflectance

The issue of ambient lighting and low reflecting material can both be solved by modifying the design to have a cavity which is painted completely with a metallic silver or gold leaf is stuck on the inside wall. Gold is a good reflector of infrared light as well as aluminium [14]. This modification will completely remove any external light sources as well as not relying on the reflectance of the object being lifted. This is similar to the opaque cavity tactile sensor mentioned in Section 2.3.4.

8.2.3 Different Gripping robot

The use of a different gripper robot with more lifting strength can be an improvement to this grasping system. Increased lifting strength will vastly increase the ability of the grasping robot to lift heavier objects. The goal is to lift up to 4 kg, which is around the weight of mail packages.

8.2.4 Different material

The next choice of deformable material would be silicon, silicon mould mixes can he readily purcahsed online and have the Young's modulus required to deform in the desired length. It wil be a low Young's modulus under EPDM. Materials that also have a long elongation will be considered too (steep stress to strain slopes are stiffer).

8.3 Future work and applications

With the improvements this tactile sensor can be used in most environments including under direct sun light since ambient light does not affect it. Another application can be in a parcel and letter delivery role. This grasping robot will carefully deliver parcels with unknown weight, shape and surface reflectivity to their destinations. This is required because there is a current problem with Auspost having labour costs causing debt. This tactile sensor since it is cheap and compact can be used on a mail delivery robot to help reduce labour costs.

The gripper robot when improvements implemented, the cavity improvement can allow for under water use. The underwater robot can be used to clean the sea such as autonomously clean the bottom of the sea. Alternatively if can also be used to pick up small seafood without damaging the shell or animal autonomously.

8.3.1 Experiment 2: Surface quality

This experiement could not be conducted and recorded due to the sensors not working anymore. The surface quality is useful to determine the amount of features that can be read on the soda can as well as various other surfaces and how it can vary as lifting distance is increased.

8.3.2 Use of old mouse

Due to both sensors not working close to the deadline of this thesis, the use of an old mouse has only made partial progress in feedback system.

Chapter 9

Abbreviations

LED	Light Emitting Diode
IR	Infrared
CMOS	Complementary metal oxide semiconductor
DPS	Digital Signal Processor
DPI	Dots per inch
EPDM	Ethylene Propylene Diene Monomer
ABS	Acrylonitrile butadiene styrene

Appendix A

Tactile sensor design

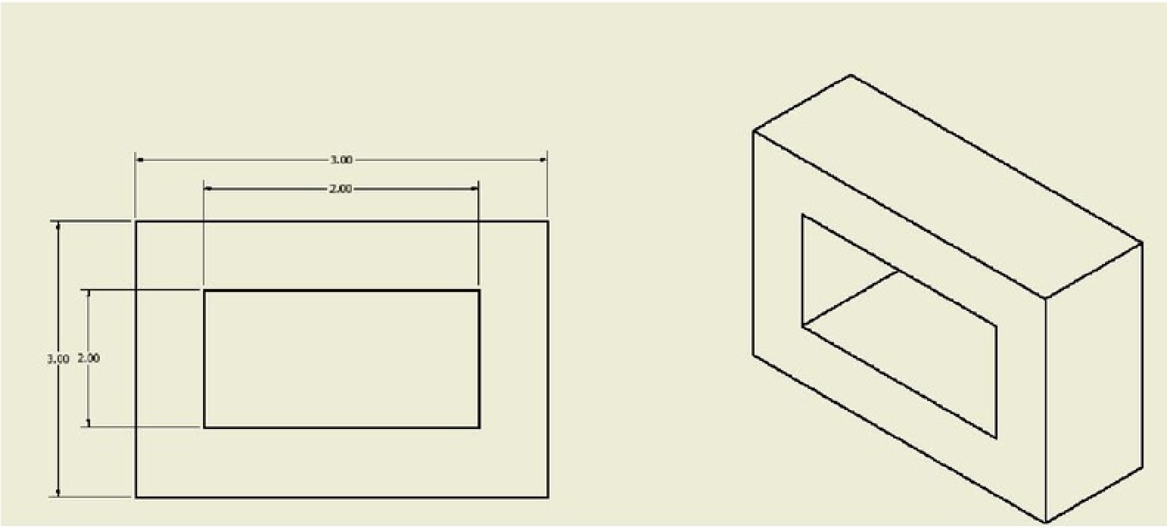

Figure A.1: Tactile sensor design

A.1 Overview

This is a sketch of dimension of the tactile sensor with the straight and curved. The thickness of the straight design will be 1 cm or 2 cm.

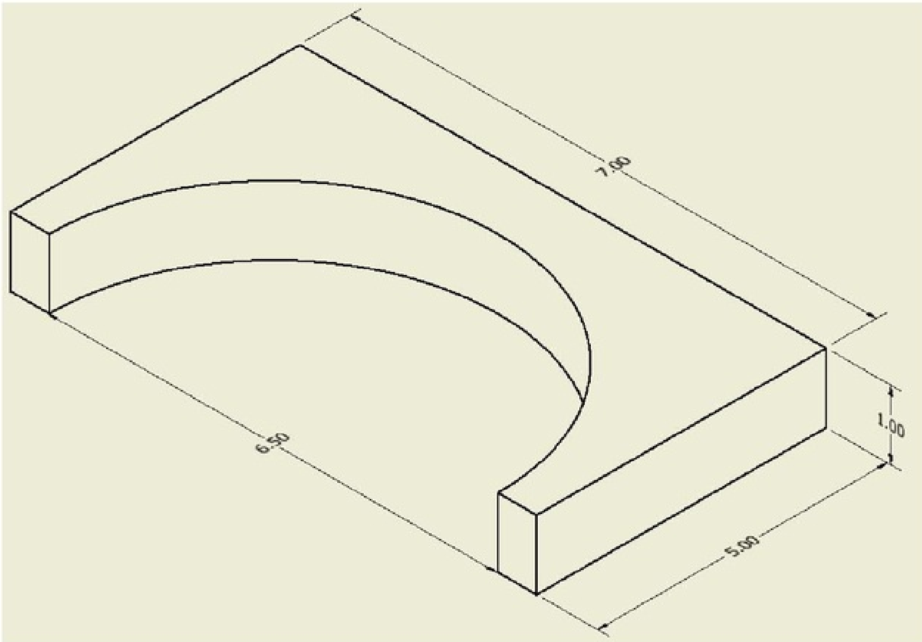

Figure A.2: Tactile sensor curved design dimensions

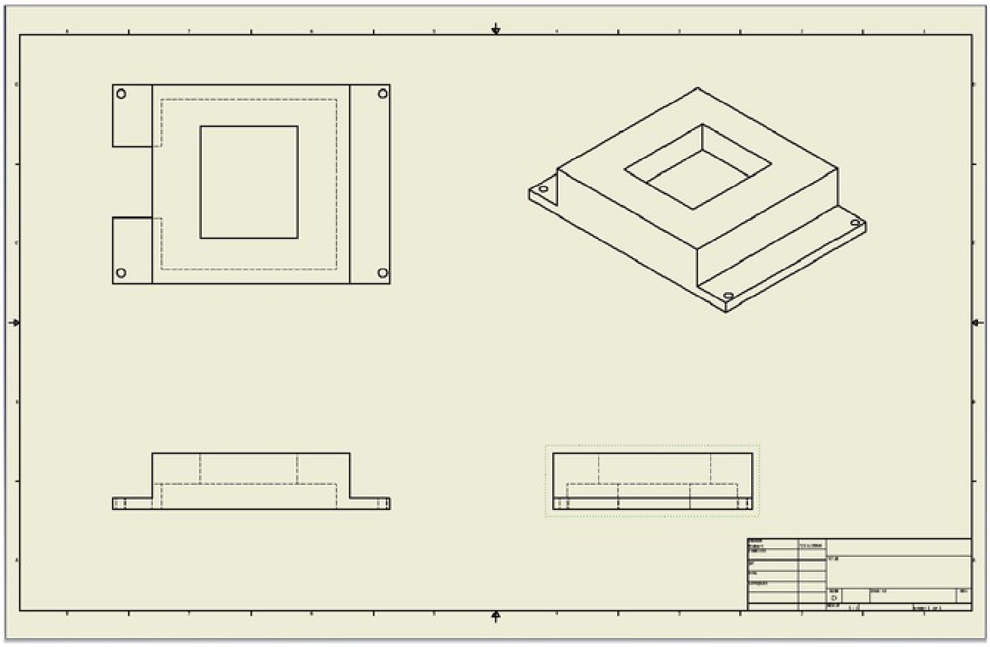

Figure A.3: 3D printed mount

Appendix B

Arduino code for ADNS9800

The following code has been edited from instructibles website for adns9800 [4]. Changes to the code will configure it to 200 cpi and converting inches to centimeters for the distance detected.

```
#include ¡SPI.h¿
#include ¡avr/pgmspace.h¿
#include ¡LiquidCrystal.h¿

LiquidCrystal lcd(9, 8, 7, 6, 5, 4);

   // Registers
#define REG_Product_ID 0x00
#define REG_Revision_ID 0x01
#define REG_Motion 0x02
#define REG_Delta_X_L 0x03
#define REG_Delta_X_H 0x04
#define REG_Delta_Y_L 0x05
#define REG_Delta_Y_H 0x06
#define REG_SQUAL 0x07 Surface quality
#define REG_Pixel_Sum 0x08
#define REG_Maximum_Pixel 0x09
#define REG_Minimum_Pixel 0x0a
#define REG_Shutter_Lower 0x0b
#define REG_Shutter_Upper 0x0c
#define REG_Frame_Period_Lower 0x0d
#define REG_Frame_Period_Upper 0x0e
#define REG_Configuration_I 0x0f
#define REG_Configuration_II 0x10
#define REG_Frame_Capture 0x12
#define REG_SROM_Enable 0x13
#define REG_Run_Downshift 0x14
```

```
#define REG_Rest1_Rate 0x15
#define REG_Rest1_Downshift 0x16
#define REG_Rest2_Rate 0x17
#define REG_Rest2_Downshift 0x18
#define REG_Rest3_Rate 0x19
#define REG_Frame_Period_Max_Bound_Lower 0x1a
#define REG_Frame_Period_Max_Bound_Upper 0x1b
#define REG_Frame_Period_Min_Bound_Lower 0x1c
#define REG_Frame_Period_Min_Bound_Upper 0x1d
#define REG_Shutter_Max_Bound_Lower 0x1e
#define REG_Shutter_Max_Bound_Upper 0x1f
#define REG_LASER_CTRL0 0x20
#define REG_Observation 0x24
#define REG_Data_Out_Lower 0x25
#define REG_Data_Out_Upper 0x26
#define REG_SROM_ID 0x2a
#define REG_Lift_Detection_Thr 0x2e //Lift detection
#define REG_Configuration_V 0x2f
#define REG_Configuration_IV 0x39
#define REG_Power_Up_Reset 0x3a
#define REG_Shutdown 0x3b
#define REG_Inverse_Product_ID 0x3f
#define REG_Motion_Burst 0x50
#define REG_SROM_Load_Burst 0x62
#define REG_Pixel_Burst 0x64
byte initComplete=0;
byte testctr=0;
unsigned long currTime;
unsigned long timer;
volatile int xydat[2];
volatile byte movementflag=0;
const int ncs = 10;
extern const unsigned short firmware_length;
extern const char firmware_data[];

void setup()
Serial.begin(9600);
lcd.begin(20, 4);
lcd.clear();
pinMode (ncs, OUTPUT);
attachInterrupt(0, UpdatePointer, FALLING);
SPI.begin();
SPI.setDataMode(SPI_MODE3);
```

SPI.setBitOrder(MSBFIRST);
SPI.setClockDivider(8);
performStartup();
dispRegisters();
delay(100);
initComplete=9;

void adns_com_begin()
digitalWrite(ncs, LOW);

void adns_com_end()
digitalWrite(ncs, HIGH);

byte adns_read_reg(byte reg_addr)
adns_com_begin();
// send adress of the register, with MSBit = 0 to indicate it's a read
SPI.transfer(reg_addr & 0x7f);
delayMicroseconds(100); // tSRAD
// read data
byte data = SPI.transfer(0);

delayMicroseconds(1); // tSCLK-NCS for read operation is 120ns
adns_com_end();
delayMicroseconds(19); // tSRW/tSRR (=20us) minus tSCLK-NCS
return data;

void adns_write_reg(byte reg_addr, byte data)
adns_com_begin();
//send adress of the register, with MSBit = 1 to indicate it's a write
SPI.transfer(reg_addr — 0x80);
//sent data
SPI.transfer(data);
delayMicroseconds(20); // tSCLK-NCS for write operation
adns_com_end();
delayMicroseconds(100); // tSWW/tSWR (=120us) minus tSCLK-NCS. Could be shortened, but is looks like a safe lower bound

void adns_upload_firmware()
// send the firmware to the chip, cf p.18 of the datasheet
Serial.println("Uploading firmware...");
// set the configuration_IV register in 3k firmware mode
adns_write_reg(REG_Configuration_IV, 0x02); // bit 1 = 1 for 3k mode, other bits are reserved

```
// write 0x1d in SROM_enable reg for initializing
adns_write_reg(REG_SROM_Enable, 0x1d);
// wait for more than one frame period
delay(1); // assume that the frame rate is as low as 100fps... even if it should never be
that low // write 0x18 to SROM_enable to start SROM download
adns_write_reg(REG_SROM_Enable, 0x18);

    // write the SROM file (=firmware data)
adns_com_begin();
SPI.transfer(REG_SROM_Load_Burst — 0x80); // write burst destination adress
delayMicroseconds(15);
// send all bytes of the firmware
unsigned char c;
for(int i = 0; i ¡ firmware_length; i++)
c = (unsigned char)pgm_read_byte(firmware_data + i);
SPI.transfer(c);
delayMicroseconds(15);

adns_com_end();


    void performStartup(void)
adns_com_end(); // ensure that the serial port is reset
adns_com_begin(); // ensure that the serial port is reset
adns_com_end(); // ensure that the serial port is reset
adns_write_reg(REG_Power_Up_Reset, 0x5a); // force reset
delay(50); // wait for it to reboot
// read registers 0x02 to 0x06 (and discard the data)
adns_read_reg(REG_Motion);
adns_read_reg(REG_Delta_X_L);
adns_read_reg(REG_Delta_X_H);
adns_read_reg(REG_Delta_Y_L);
adns_read_reg(REG_Delta_Y_H);
// upload the firmware
adns_upload_firmware();
delay(10);
//enable laser(bit 0 = 0b), in normal mode (bits 3,2,1 = 000b)
// reading the actual value of the register is important because the real
// default value is different from what is said in the datasheet, and if you
// change the reserved bytes (like by writing 0x00...) it would not work.
byte laser_ctrl0 = adns_read_reg(REG_LASER_CTRL0);
adns_write_reg(REG_LASER_CTRL0, laser_ctrl0 & 0xf0 );
delay(1);
```

```
Serial.println(”Optical Chip Initialized”);

void UpdatePointer(void)
if(initComplete==9)
digitalWrite(ncs,LOW);
xydat[0] = (int)adns_read_reg(REG_Delta_X_L);
xydat[1] = (int)adns_read_reg(REG_Delta_Y_L);
digitalWrite(ncs,HIGH);
movementflag=1;
}
}
void dispRegisters(void)
int oreg[7] = {
0x00,0x3F,0x2A,0x02 };
char* oregname[ ] = {
”Product_ID”,”Inverse_Product_ID”,”SROM_Version”,”Motion” ;
byte regres;
digitalWrite(ncs,LOW);
int rctr=0;
for(rctr=0; rctr¡4; rctr++){
SPI.transfer(oreg[rctr]);
delay(1);
Serial.println(”—”);
Serial.println(oregname[rctr]);
Serial.println(oreg[rctr],HEX);
regres = SPI.transfer(0);
Serial.println(regres,BIN);
Serial.println(regres,HEX);
delay(1);
}
digitalWrite(ncs,HIGH); }
    int convTwosComp(int b){
//Convert from 2's complement
if(b & 0x80){
b = -1 * ((b ˆ0xff) + 1);
} return b; }
    int tdistance = 0;
void loop(){
if(movementflag){
lcd.clear();
tdistance = tdistance + convTwosComp(xydat[0]); //REG_Delta_X_L
Serial.println(”Distance = ” + String(tdistance));
movementflag=0;
```

delay(3);

Appendix C

Matlab code

The following software to control the SG gripper is found on robtis [21]
%Program Initialisation————————————————————
loadlibrary('dynamixel','dynamixel.h');
%libfunctions('dynamixel'); %lists lib functions

DEFAULT_PORTNUM = 5; %COM Port Number
DEFAULT_BAUDNUM = 1; %1 for 1mbps, 34 for 57142 kbs

%Servo Limits
%global variables are shared with the gui
global s1max; s1max = 890;
global s1min; s1min = 30;
global s2max; s2max = 890;
global s2min; s2min = 440;
global s3max; s3max = s2max;
global s3min; s3min = s2min;
global s4max; s4max = 890;
global s4min; s4min = 250;
global s5max; s5max = s4max;
global s5min; s5min = s4min;
global s6max; s6max = 780;
global s6min; s6min = 160;
global s7max; s7max = 850;
global s7min; s7min = 495;

Limits = [s1max s1min;s2max s2min;s3max s3min;s4max s4min;s5max s5min;s6max s6min;s7max s7min]; %hard limits on servos, to prevent damage

%AX-12a function addresses
P_GOAL_POSITION = 30;

P_PRESENT_POSITION = 36;
P_MOVE_SPEED = 32;
P_MOVING = 46;
P_CW_ANGLE_LIMIT = 6;
P_CCW_ANGLE_LIMIT = 8;
P_LED = 25; %0/1
P_CW_COMP_MAR = 26; %error from goal pos allowed
P_CCW_COMP_MAR = 27;
P_CW_COMP_SLOPE = 28; %lvl of torque near goal pos
P_CCW_COMP_SLOPE = 29;
P_TORQUE_ENABLE = 24;
P_TORQUE_LIMIT_L = 34;
P_TORQUE_LIMIT_H = 35;
P_VOLTAGE = 42;

ProgramLoop = true;
CurrentPos = [0, 0, 0, 0, 0, 0, 0];
TorqueLimit = [0, 0, 0, 0, 0, 0, 0];
CurrentTorque = [0, 0, 0, 0, 0, 0, 0];
MoveSpeed = 75;
TorqueEnable =0;

%Servo Setup ——————————————————

%initialises the connection res = calllib('dynamixel', 'dxl_initialize', DEFAULT_PORTNUM, DEFAULT_BAUDNUM);

%applies max torque limit for the gripper
%calllib('dynamixel','dxl_write_word',7,P_TORQUE_LIMIT_L,1023);

%set angle limits
for i = 1:7
id = i;
calllib('dynamixel','dxl_write_word',id,P_CCW_ANGLE_LIMIT,Limits(i,1));
calllib('dynamixel','dxl_write_word',id,P_CW_ANGLE_LIMIT,Limits(i,2));
end

%write move speed
for i = 1:7
id = i;
calllib('dynamixel','dxl_write_word',id,P_MOVE_SPEED,MoveSpeed);
end

```
%GUI Setup --------------------------------------------
%this stuff needs to only run once. The gui opening function loops.
% stuff should go here to save processing

handles = guihandles(dynamixel_gui);

%set start position
set(handles.s1_slider, 'value', 150);
set(handles.s2_slider, 'value', 250);
set(handles.s4_slider, 'value', 85);
set(handles.s6_slider, 'value', 145);
set(handles.s7_slider, 'value', 145);

%Display start position
set(handles.s1_value, 'string', 150);
set(handles.s2_value, 'string', 250);
set(handles.s4_value, 'string', 85);
set(handles.s6_value, 'string', 145);
set(handles.s7_value, 'string', 145);

dynamixel_gui
%Program Loop ------------------------------------------

while ProgramLoop %main loop, continues until ProgramLoop is false i.e. stop buton

handles = guihandles(dynamixel_gui);

s1pos = (get(handles.s1_slider, 'value'))/0.29;
s2pos = (get(handles.s2_slider, 'value'))/0.29;

s3pos = s2pos;
s4pos = (get(handles.s4_slider, 'value'))/0.29;
s5pos = s4pos;
s6pos = (get(handles.s6_slider, 'value'))/0.29;
s7pos = (get(handles.s7_slider, 'value'))/0.29;

loop = get(handles.stop_button, 'Value');
ProgramPause = get(handles.pause_button, 'Value');
ProgramLoop = loop;

while ProgramPause
%Disables torque on motors when stop is pushed
for i = 1:7
```

```
id = i;
calllib('dynamixel','dxl_write_word',id,P_TORQUE_ENABLE,0);
end
ProgramPause = get(handles.pause_button, 'Value');
pause(0.1);
end

    GoalPos = [s1pos,s2pos,s3pos,s4pos,s5pos,s6pos,s7pos];

    %write goal position
for i = 1:7
id = i;
calllib('dynamixel','dxl_write_word',id,P_GOAL_POSITION,GoalPos(id));
end

    %display current position set(handles.s1_current_pos, 'string', num2str((calllib('dynamixel','dxl_read_word
set(handles.s2_current_pos, 'string', num2str((calllib('dynamixel','dxl_read_word',2,P_PRESENT_POSITION
set(handles.s4_current_pos, 'string', num2str((calllib('dynamixel','dxl_read_word',4,P_PRESENT_POSITION
set(handles.s6_current_pos, 'string', num2str((calllib('dynamixel','dxl_read_word',6,P_PRESENT_POSITION
set(handles.s7_current_pos, 'string', num2str((calllib('dynamixel','dxl_read_word',7,P_PRESENT_POSITION

    %display s7 voltage
set(handles.s7_voltage, 'string', num2str((calllib('dynamixel','dxl_read_word',7,40))));

    %speed control
for i = 1:7
id = i;
current_pos = calllib('dynamixel','dxl_read_word',id,36); %Read current pos
goal_pos = calllib('dynamixel','dxl_read_word',id,30); %read goal pos

    diff = abs(goal_pos - current_pos); %how far from goal is it?

    if (0 ¡ diff) && (diff ¡ 50)
move_speed = 40;
elseif (51 ¡ diff) && (diff ¡ 100)
move_speed = 80;
elseif (101 ¡ diff) && (diff ¡ 300)
move_speed = 100;
elseif (301 ¡ diff) && (diff ¡ 500)
move_speed = 120;
elseif (501 ¡ diff) && (diff ¡ 1023)
move_speed = 150;
end
```

```
calllib('dynamixel','dxl_write_word',id,P_MOVE_SPEED,move_speed);
end

    end

    %Program Shutdown ——————————————————————————

    %Disables torque on motors when stop is pushed
for i = 1:7
id = i;
calllib('dynamixel','dxl_write_word',id,P_TORQUE_ENABLE,0);
end

    %Terminates connection and closes gui
calllib('dynamixel','dxl_terminate');
unloadlibrary('dynamixel')
close all force
%clearvars
```

Bibliography

[1] S. S. Adbullah Kassir, Robert Fitch, "Communication-aware information gathering with dynamic information flow," *The International Journal of Robotics Research February*, vol. 34, no. 2173, 2015.

[2] Administrator. (2015, Feburary) Ir sensor. ELECTRONICS HUB. [Online]. Available: http://www.electronicshub.org/ir-sensor/

[3] bestech. Young modulus for some common materials. [Online]. Available: www.bestech.com.au/wp-content/uploads/Modulus-of-Elasticity.pdf

[4] G. Butbaia, "Arduino tutorial: Get traveled distance using adns-9800 laser mouse sensor," September 2013. [Online]. Available: http://www.instructables.com/id/Arduino-Tutorial-ADNS-9800-Laser-Mouse-Traveled-Di/

[5] G. Causey, "Guidelines for the design of robotic gripping systems," *Assembly Automation*, vol. 23, pp. 18–28, 2003.

[6] B. Doda, "Ldv laser doppler velocimetry," *velocimetry*, 2008.

[7] edpm. Epdm specifications section. [Online]. Available: http://mykin.com/epdm-properties

[8] M. Fassler, "Force sensing technologies," *Swiss Federal Institute of Technology Zurich*, 2010.

[9] R. A. R. Greg Hellard, "A robust, sensitive and economical tactile sensor for a robotic manipulator," *Australasian Conference on Robotics and Automation*, 2002.

[10] b. R. Hani William, Yousef Ibrahim, "A tactile sensor for incipient slip detection," *International Journal of Optomechatronics*, 2007.

[11] R. C. HIBBELER, *Mechanics of Materials*, 8th ed. Prentice Hall, 2011.

[12] R. D. Howe, "Tactile sensing and control of robotic manipulation," *Harvard University*, 1994.

[13] D. Kizilarslan, "Optical versus laser mouse. whats the difference between optical and laser mouse?" 2015.

[14] J. Liu. what materials reflect infrared? [Online]. Available: http://www.techwalla.com/articles/what-materials-reflect-infrared

[15] A. R. Mazid Abdul, "A robotic opto-tactile sensor for assessing object surface texture," *Central Queensland University*, 2006.

[16] Nipun. (2015, October) Difference between yield strength and tensile strength. [Online]. Available: http://pediaa.com/difference-between-yield-strength-and-tensile-strength/

[17] printerfilling. 3d printing a 3d honeycomb infill concept. [Online]. Available: https://engineerdog.com/2015/03/08/3d-printing-a-3d-honeycomb-infill-concept/

[18] printerliteon. Ir detector data sheet. Sparkfun. [Online]. Available: https://www.sparkfun.com/datasheets/Components/LTR-301.pdf

[19] Printers. Types of 3d printers. [Online]. Available: http://3dprintingfromscratch.com/common/types-of-3d-printers-or-3d-printing-technologies-overview/

[20] Printershore. Durometer shore hardness scale. [Online]. Available: https://www.smooth-on.com/page/durometer-shore-hardness-scale/

[21] Robotis software support. robotis. http://support.robotis.com/en/software/ dynamixel_sdk/usb2dynamixel/windows/matlab.htm.

[22] T. Rogers, "Everything you need to know about abs plastic," 2015. [Online]. Available: https://www.creativemechanisms.com/blog/everything-you-need-to-know-about-abs-plastic

[23] RONZO. Ir communication. sparkfun. [Online]. Available: https://learn.sparkfun.com/tutorials/ir-communication

[24] T. Sobh, "Advances in intelligent robotics and collaborative automation," pp. 167–192.

[25] A. Technologies. Avago adns-9800 high-performance laserstream gaming sensor. Avago Technologies. [Online]. Available: https://octopart.com/adns-9800-avago-21394749

[26] A. Wilson, "Basic ir detector/emitter," *reconnsworld*, 2006. [Online]. Available: http://www.explainthatstuff.com/computermouse.html

[27] C. Woodford, "Computer mice," *explainthatstuff*, 2016. [Online]. Available: http://www.explainthatstuff.com/computermouse.html