Instrument Controlled Microscopy for Neurosurgical Applications

Final Report
December 9, 2009

Mason Jellings: Team Leader
Sarah Reichert: Communicator
John Byce: BSAC
Justin Gearing: BWIG

Client: Dr. Joshua Medow
Advisor: Professor Thomas Yen

Table of Contents

Design Alternatives

Abstract

Our client, Dr. Joshua Medow, is a neurosurgeon at the UW Hospital. While operating, he uses a microscope to view the inside of a surgical cavity. Although the microscope has autofocus capabilities, they tend to be more of a hindrance than an aid and therefore are not utilized during a procedure. The goal of this project is to design a new autofocus system that is capable of refocusing to a certain depth based on the location of a surgical instrument in the opening. It is necessary that the system does not interfere with other instruments or impede the surgeon's ability to perform necessary maneuvers. Our team created three potential design ideas based on these guidelines. One incorporates a linear Hall Effect sensor to detect a magnetic field, another uses infrared sensing with a light emitting diode, and the last employs a sliding mechanism. Based on the results of our design matrix, we chose to pursue the option that utilizes infrared sensing. Our final design uses a Sharp GP2D12 infrared sensor to detect the position of a surgical aspirator. This position data is relayed to an Arduino microcontroller. The microcontroller outputs corresponding signals to a Vexta stepper motor, mounted to the microscope, in order to adjust the focus to the desired depth. The final design was tested by altering the distance read by the sensor, documenting the change in the sensor's outputs, and comparing these results to the actual change in distance enacted by the system. This testing showed that the average difference between the expected and actual change of microscope depth was 0.163 cm with a standard deviation of 0.093 cm.

Background

Neurosurgery is a specialized division of surgery that functions to diagnose and treat disorders and injuries that occur in the central and peripheral nervous systems [1]. Dr. Joshua Medow is a neurosurgeon at the UW Hospital and an assistant professor of the Department of Neurological Surgery in the School of Medicine and Public Health. During a neurosurgical operation, Dr. Medow uses a microscope to magnify the cavity created to perform the procedure. This microscope is typically located at a distance between eight and sixteen inches (twenty to forty centimeters) from the incision opening. There currently is an autofocus system integrated into the microscope, but because of its limitations Dr. Medow usually chooses to deactivate it and manually control the focus of the microscope. Manually refocusing is cumbersome and requires that the surgeon halt the surgery, focus to the correct depth, and then resume operating. Therefore finding a more convenient method is desirable [2].

Various approaches have been explored to overcome this problem in the past. These alternative methods include a mouth switch and a foot-pedal control system. The mouth switch is currently implemented in some neurosurgical procedures. It functions by having the surgeon lightly hold a lever that is integrated into the microscope between his or her lips and teeth. The microscope can then be manipulated to focus up or down by orally shifting the lever up or down, respectively. Similarly, to zoom in or out the lever is shifted left or right [3]. The foot-pedal mechanism is also considered to be more efficient than refocusing by hand. In order to control the focus in this way, the surgeon places a foot on the pedal and moves it upward to focus up or downward to focus down. The pedal is capable of detecting which direction is indicated from a neutral position, and then sends signals for the microscope to adjust its focus accordingly [4].

Current Design

Currently, surgical microscopes have passive autofocus systems that operate similarly to those of single-lens reflex cameras. Light from the subject scene is directed to a pixel strip known as a charge-coupled device, or CCD. This sensor then provides input to a microprocessor that contains algorithms capable of computing how much contrast exists between different elements within the picture. When the subject is out of focus, the intensities of neighboring pixels on the CCD will be similar. If this is the case, the microprocessor will refocus the lens and make new calculations to determine if the intensities became more alike or distinct. It will continue to move the lens until it locates the position at which there is a maximum difference between the intensities of adjacent pixels. In order for this process to function properly, the scene must provide sufficient light and contrast. An alternative method of auto focusing operates through analysis of the resolution of neighboring pixels. Resolution is measured through the application of a high-pass or gradient filter that can isolate high frequencies. Optimal focus, which corresponds to maximum resolution, is measured by the magnitude of the high frequencies [5]. Both of these systems, however, are often ineffective in surgical applications because a clear distinction between neighboring pixels of a magnified portion of the brain or spine rarely exists. Also, the system is incapable of refocusing to a different location at the surgeon's discretion.

Problem Statement and Motivation

In conducting a complex neurosurgery, a surgeon operates within a cavity up to six inches in depth. Navigation within this cavity is aided by an operating microscope, which exhibits a depth of field of approximately 6 mm. Consequently, the surgeon may only observe, and therefore operate within, a relatively small fraction of the cavity at any given time. The auto-focus feature included in the existing microscope further complicates procedures as it rarely focuses to the desired depth, nor does it stay at the desired depth while conducting surgical maneuvers. To correct this inconvenience, the surgeon must adjust the focus depth manually. In order to do so the surgeon must halt the surgery and refocus the microscope by hand to the desired depth. This is extremely problematic because standard surgical procedure calls for frequent changes in depth and therefore the surgeon is required to frequently refocus the microscope. The refocusing process not only consumes valuable time, but also disrupts the surgeon's train of thought and interrupts any maneuvers requiring continuous movement of an instrument from one depth to another. An estimated 40% of total surgery time is attributed to this refocusing of the microscope[6]. In order to simplify and improve the use of a frameless microscope in stereoscopic neurosurgery, our client Dr. Joshua Medow has challenged us to devise an auto-focusing device that may be integrated on to his existing microscopic interface. Our product will function to eliminate the need to manually refocus the microscope as the surgeon transitions from one depth of field to the next, by allowing the microscope to alter its focus in response to changes in the position of the surgeon's instrument tip.

Design Requirements

The design requirements for this project are outlined in the PDS in the Appendix, and explained in more detail here. As with any surgical instrument, the design constraints of this project are strict and precise. Failure to abide by these requirements may result in extreme harm to a patient, as well as complicating standard surgical procedures. The first of these strict requirements is safety. The design of this device must, at all costs, not interfere with any surgical materials nor may it hinder the performance of the surgeon. This device must also have the capability to be easily sterilized following procedures without disrupting functionality. Any electrical components included in the design must be housed appropriately, and materials used must not release any substances that are harmful when exposed to the human brain.

Along with strict safety guidelines, our design must be ergonomically sound. A surgeon with a limited background in electronics must be able to easily use it. Furthermore, it is necessary for the device to be integrated with other surgical instruments without difficulty.

This device must also satisfy strict performance requirements, as it is used in highly technical surgical procedures. Mainly, it must possess the ability to track the position of the tip of a surgical instrument in an efficient manner. In doing so, our device should refocus with a lag time of less than one second, and should be compatible with any surgical implements. The device must also have a small margin of error, and high reliability. Due to the 6 mm depth of field of the microscope, and the necessity for extreme precision during surgery, our device should accurately track the tip on an instrument to within 1 cm of its actual position, and refocus the microscope with the same degree of accuracy. This degree of accuracy must be maintained throughout the lifetime of the device, to ensure its reliability meets the standards of the other instruments in the operating room.

Another important design constraint is the durability of the device. This device must have a life in service (provided it is well maintained and serviced) of three to five years. Along with this life of service, the device must have a long shelf life; meaning long periods of inactivity would not affect the usage and performance of the device.

This device will function in a meticulous operating environment. It will not be exposed to extreme high or low temperatures, as most procedures are conducted at or near room temperature. The operating room is also held to a high standard of cleanliness, which implies the device will be operated in a dust-free environment. This device must also withstand exposure to UV light, as it is prevalent during neurosurgical procedures.

Finally, the physical specifications of the device must not obstruct the surgical procedure in any manner. As a result, the entire device must be less than 2.25 kg (5 pounds). The weight of components mounted on the surgical device, however must have minimal weight (no more than 15-20 grams), as it may not disrupt the balance of the instrument. The maximum size of the device should be 6” X 6”, however the size of the device should also be minimized, as space is limited in the operating room. Provided that they are safe for the operating room and surgical procedures, any materials may be used in this device.

Design Alternatives

Hall Effect Sensor

When moving charges, also known as current, encounter a magnetic field, they experience a force perpendicular to both the field’s direction and the direction of the charges’ velocity. When current passing through a metal plate encounters a magnetic field, the force experienced by the individual moving charged particles causes these particles to deflect toward one end of the plate. The degree of deflection depends on the magnitude and direction of the applied magnetic field with larger deflections corresponding to greater magnitudes in directions more perpendicular to the plane of the plate.

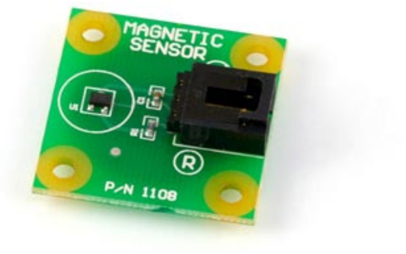

Fig. 1: Phidget 1108 Magnetic Sensor[7]
The figure shows the Phidget Hall Effect sensor, which uses angle and intensity of magnetic field to determine position.

As deflection occurs, a buildup of charged particles on one end of the plate establishes a potential difference across the plate proportional to the degree of deflection. This potential difference is known as the Hall Potential and the phenomenon to which it leads is known as the Hall Effect. Hall Effect magnetic sensors, like the one shown in Figure 1, measure the Hall Potential across an internal plate and, from that, calculate the magnitude of the perpendicular component of an applied magnetic field.

Our design incorporating a Hall Effect sensor consists of two components, an emitter mounted on a surgical utensil and a detector mounted on the microscope. A simple rare earth magnet serves as the emitter. It mounts on the utensil within a removable cap, which is interchangeable among as many utensils as is feasible via a screw clamp system. Specialized caps for more eccentrically shaped utensils will be designed if time allows. To detect the magnetic field emitted by the magnet, a Phidget - 1108 Hall Effect based magnetic sensor mounts above the surgeon’s workspace and reads the magnitude of the perpendicular component of the emitted magnetic field. Because this data alone is insufficient to calculate the position of the emitter, let alone that of the utensil’s tip, we must constrain the configuration of the utensil while the system focuses. In other words, while the focusing system is engaged, the surgeon must hold the utensil parallel to the vertical axis with the tip at the desired depth. With a known utensil configuration, and therefore a known magnet configuration, the data from the detector is transformed, via microprocessor, to the distance between the detector and the emitter. From this distance, the known length of the utensil and information about the utensil’s configuration, the position of the utensil’s tip is calculated and the microscope is focused accordingly.

While this design’s wireless aspect makes it attractive at first glance, the practical considerations of its implementation limit its applicability as a solution to our problem. For instance, the Phidget – 1108 exhibits a maximum range of detection of a sizeable rare earth magnet of approximately four inches and a range of approximately two inches in which fourth inch variations in the magnet's position are differentiable. Furthermore, the 1108 only detects magnets in these ranges provided the magnets are positioned directly below the sensor. This alone makes our design unviable as surgeons typically hold their utensils around nine inches from the microscope during an operation. However, even in the event that we modified our design to accommodate the short range of the detector, the weight of a rare earth magnet of strength sufficient to reach the detector would tend to disrupt the precise balance of most neurosurgical utensils.

Infrared Light Sensor

The GP2D12 is manufactured by Sharp as an infrared (IR) range finder. Behind one of its lenses is positioned an 850(+/- 70)nm light emitting diode (LED)[4]. Under normal operating circumstances the GP2D12 focuses light from this LED into a concentrated beam, which travels in a straight line until it encounters an object. Upon striking most objects' microscopically erratic surfaces, the beam scatters in all directions. Some of the reflected rays strike the GP2D12's second, infrared filtered lens and refract onto a charge coupled device (CCD). CCD's transform incident light into a voltage which depends on the portion of the CCD's surface the light strikes. The surface of the CCD in the GP2D12 exhibits raised portions that allow the voltage produced by the CCD to correspond to the angles of the incident light relative to an axis perpendicular to the CCD. From the angle at which reflected light strikes the CCD and the known angle at which the light originally leaves the

Figure 2: Sharp GP2D12 IR Range Finder[8]
The Sharp infrared sensor would detect a infrared LED mounted on the surgeon's tool.

GP2D12, the distance between the GP2D12 and the object from which the light reflected can be calculated through triangulation.

Like our design based on the Hall Effect sensor, the infrared sensor design employs an independent emitter and sensor system. In this case, two components comprise the emitter portion of the system, a utensil mounted 850nm infrared Light Emitting Diode (LED), and a wristband mounted power supply. The IR LED mounts on the utensil through a cap very similar to that used to mount the magnet of the Hall Effect sensor design. However, the size of the cap may require adjustment to accommodate any circuitry necessary to power the LED. Insulated wiring runs from the LED in the cap to a wristband, on which mounts a power cell and the circuitry required to power the LED.

The detector portion of this design incorporates two, GP2D12 IR range finders, modified such that they do not emit their own infrared light. This modification allows the GP2D12 to detect IR light from other sources, namely our utensil mounted LED, without interference from its own emissions. The 850nm LED used as the emitter in this design approximates a point source of light. Therefore, it emits rays in all directions from a small area of origin just as would the reflected beam of the GP2D12's own source. Furthermore, since the wavelength of our LED matches that of the GP2D12's filter, the GP2D12 should allow light originating from the LED to reach its CCD just as it would allow light from its own source. Therefore, the modified GP2D12s no longer detect the incident angle of light reflected from a distant object but rather the incident angle of light emitted by our LED. The angles detected by the two GP2D12s along with the known, fixed distance between the detectors, provide enough information to triangulate the position of the utensil mounted LED. However, this position does not allow for calculation of the position of the utensil's tip since many utensil configurations are possible for a single LED position. Consequently, we must limit the configuration of the utensil during focusing just as we did in the Hall Effect sensor based model. A vertical utensil configuration provides the vertical distance between the LED and the utensil tip and allows the position of the tip to be calculated from the position of the LED. The triangulation of the position of the LED, the subsequent calculation of the position of the utensil tip and finally, the focusing of the microscope based on tip position is all handled by a single microprocessor mounted on the scope with the detectors.

Despite this design's inclusion of wires, it remains a viable option. Due to the amount of slack in the wires linking the utensil to the wristband, significant rotation of the utensil with respect to the wrist should remain possible. Beyond such rotation we assume that the wrist will translate and rotate along with the utensil, so the wires will not factor into the utensil's motion. Consequently, appreciable restriction of the utensil's range of motion should not occur. In addition to preserving the utensil's range of motion, powering the LED with a wristband-based source reduces the weight of the component mounted on the utensil. This becomes important when a procedure requires a precisely balanced utensil. Finally, the factor which distinguishes this design most from the Hall Effect sensor design is the ability of light to travel long distances with minimal attenuation. Whereas the range of the Hall Effect sensor literally fell short of our requirements, the 10 to 80 centimeter normal operational range of the IR detector-emitter combination meets and exceeds our specifications.

Sliding Sensor

With the two previously mentioned design alternatives, there is a lot of difficulty associated with the three degrees of freedom that the user is allowed with their surgical tool. Using a sliding sensor like the one in Figure 3, the degrees of freedom of the system are reduced to one. The device would simply consist of a simple mechanical sliding mechanism to allow the sensor, either infrared or magnetic field, to move only in the vertical direction. As the surgeon moves to different locations in the cavity, they can simply slide the sensor to the depth they are working at. Shown if Figure 3a, the system with the sensor set at a depth nearly to the bottom of the cavity. Figure 3b shows the same setup with the sensor moved to a shallower depth in the surgical cavity. This method would work with other types of electromechanical systems that allow an object to vary the output voltage depending on its location on the slider assembly and would not necessarily have to be a wireless sensing system.

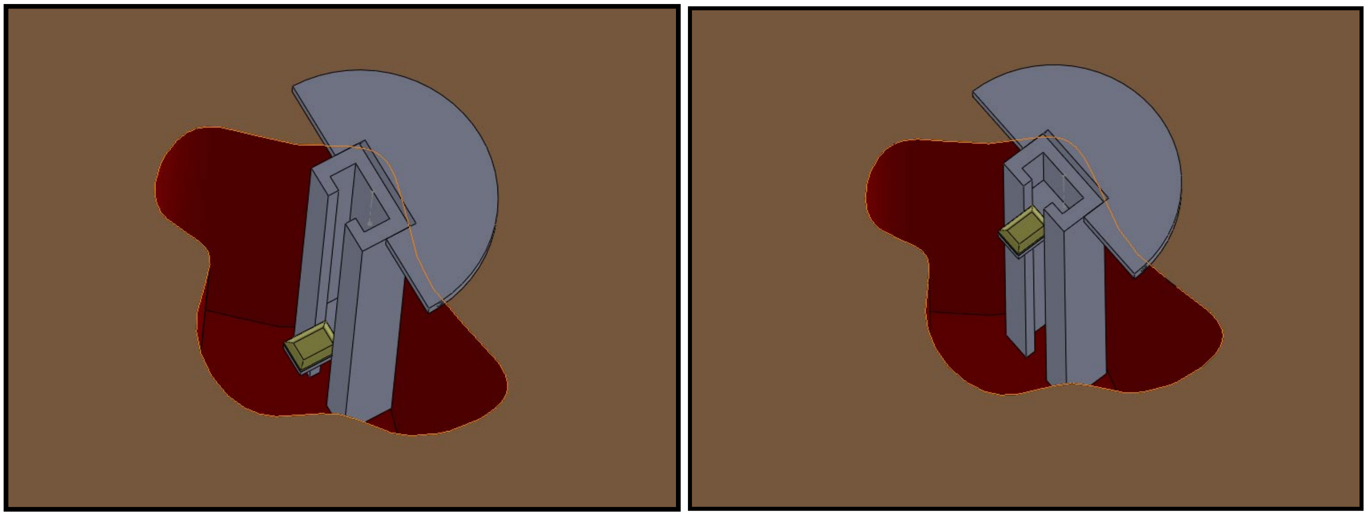

Figure 3: Sliding Sensor

Figure 3a (left) shows the sliding sensor at nearly maximum depth. The depth of the sensor is much shallower in Figure 3b (right), after the sensor was slid up by an external force (user).

This system would also allow for exceptions to the size and wireless power problems that plagued the other devices. The hard mounting of the sensor would allow wires (not shown) to be run into the housing and power the sensor without limiting motion of the surgeon’s tool. However, it is important to note that the sliding sensor system would be placed directly into the patient’s surgical cavity, and would require the movement of the system, and recalibration of some sort (if using a sensor), if the surgeon were to require access to a portion of the cavity blocked by the sliding sensor assembly. This would be the most expensive design, as it would require the purchase of the not only the sensor components like the other devices, but also materials for the fabrication of the slider assembly.

Final Design

The design with which we decided to move forward is a somewhat simplified hybrid of the infrared sensor design and the sliding sensor design. It consists of five primary components; an instrument mounted target, a microscope mounted position sensor, an actuator, a microcontroller program and the circuitry which incorporates each component into a system.

Aspirator Assembly

Surgeons typically employ an aspirator in the same hand throughout surgery. For this reason, we employed the distance sensor to read the distances on the aspirator for the duration of the surgery. In order to use the infrared distance sensor, a uniform surface was required for infrared light deflection. For example, if a non uniform (in distance from sensor) surface was placed under the sensor; very small movements in the same plane would output different distance readings because of the inconsistency of the surface. A thin disk was mounted to the aspirator in a manner such that it would cause very little or no disturbance for the surgeon during use. The disk was mounted on a small cylinder which allowed it to create a uniform surface for distance readings, while remaining out of the way of the surgeon's hand. The disk and column are mounted to the aspirator with the aid of an internal band, which discretely wraps around the aspirator shaft and allows for secure mounting of the disk assembly to the aspirator.

Sensor

The sensor chosen is the Sharp GP2D12 sensor referenced in the design alternatives. In order to use a distance sensor to help determine the depth of the surgeon's instrument in the cavity, the sensor detects the distance to a disk mounted on an aspirator. This method works because of the constraints we have put on the system. For example, the aspirator is assumed to move only up and down perpendicular to the opening of the cavity, thus the distance read by the sensor can be used to find the depth. The system also assumes that the aspirator is held parallel to the plane of the opening of the cavity for any changes in focus. Any rotation of the tip of the aspirator about the disk will not be measured by the sensor and therefore the system will not detect any change in position.

Stepper Motor

A Vexta stepper motor was used to power the focus knob on the microscope. The motor required 12 Volts and 0.6 Amps of current to effectively turn the focus knob. For every pulse or step received by the motor, the spindle rotated 1.8°, allowing for fine control of the focus knob on the microscope. If finer tuning were required we could explore two possibilities; we could use half steps on the stepper motor, allowing for one 0.9° per step (or replace motor with one with smaller steps), or we could increase the diameter of the gear (depending on motor assembly) driving the focus knob. However, this second option would require a knob driven focus, which would not likely be used for neurosurgical applications. The stepper motor speed would be determined by the program step frequency, rather than the stepper motor itself.

Motor Mount System

The stepper motor is useful because it allows for fine and rapid control of a spindle, but in order to drive the focus knob on the microscope, the use of gears was required. Two forty five degree miter gears were used to enable the stepper motor to drive the focus knob spindle on the microscope. The gear mounted to the stepper motor has an outside diameter of 1.00 inch while the gear on the microscope spindle has an outside diameter of 1.25 inches. This allows the motor spindle to move the focus knob easier, although causing an increased time and number of steps to focus. The motor is supported by a plate mounted on the face of the stepper motor, which is mounted to four support legs that help support the weight of the motor. The motor stand sits at 6.5 inches in height, although two of the legs are shorter, to enable mounting on the raised microscope stand. In order to reduce the shear forces on the motor assembly, a plate was mounted to the stepper motor which mounted directly to the microscope. This allowed the motor stand to be fixed in space, without sliding around with stepper motor activity.

Program

The program component of our final design controls an Arduino Duemilanove[7] microcontroller and is written in the C/C++ based language of said microcontroller. The basic theory underlying the program is similar to Proportional Integral Derivative (PID) control theory in that it utilizes a proportional error term and a term similar to a derivative velocity term [8].

During normal operation, the program (shown in detail in Appendix 2) assumes that the microscope is initially configured such that the objective lens is one centimeter below the top of its focal range. If this is not the case, the program will still function until the objective lens reaches either the top or bottom of its focal range at which point it may continue to run the actuator, potentially causing damage to the actuator system or microscope, and will fail to refocus on the instrument. Therefore, when using the current system with the current microscope, this initial configuration should always be observed by the use; however, this aspect of the program is easily adjusted for other systems and microscopes.

The program also assumes that the microscope is initially in focus. If it is not, the program will maintain the same level of focus throughout operation.

Upon startup, the program looks at the value of the variable “count2.” This variable is initialized to zero. Since “count2” is zero, the program takes, as input, an analogue voltage output of the GP2D12 sensor which the program converts to a distance value in centimeters, and saves it as the variable “tot1.” After a delay of one second, the program once again takes a converted reading from the sensor and, this time, saves it as “tot2.” The program then looks at the value of the difference between “tot1” and “tot2.” This difference is equivalent to the proportional error term of PID theory. If this difference is above our one centimeter error band, that is, if the value of the difference is greater than .5 cm, and the value of “tot2” is within an acceptable range, the program begins a negative feedback loop which moves the objective lens downward. This loop sends a signal to the actuator, which depends on the current position of the actuator motor and causes the motor to rotate one step from its current position. Also within the loop, the distance traveled by the objective lens due to one step of the motor is subtracted from the initial difference between “tot1” and “tot2” and this new value is then saved as the variable “dif” and passed into the loop condition for the next iteration. The subtracted increment term is similar to the derivative velocity term in PID theory. Finally, each iteration of the loop causes a variable, labeled “danger,” and initialized to zero, to increase by one and then delays the next iteration by a set time interval. The loop terminates either when “dif” falls below .1 cm and therefore the microscope is considered in focus, or when the variable “danger” reaches 400 at which point the objective lens has reached the bottom of its focal range. When the loop terminates because “danger” has reached 400, the program causes a red LED to flash and sets the variable “count2” to 4. When the loop terminates because the microscope has focused, the program sets “count2” to 1. Regardless of how the loop terminates, the program sets the variable “save2” to “tot2” minus the original difference between “tot1” and “tot2.” This is done to compensate for the movement of the sensor with the head of the microscope. Then, at the very end of the program, outside of any conditional statements, the program sets the variable “save” to “tot2” minus the original difference between “tot1” and “tot2.”

The program then begins the process over and looks at the variable “count2” again. Since the program has set this variable to a value other than zero, it will not set the variable “tot1” to an output of the GP2D12. Instead, if the value of “count2” is one, then the program is running normally and it sets the variable “tot1” to the value of the variable “save” in order for the program to refocus to a new final position using the old final position as the new “in focus” initial position. This allows the program to adjust if the instrument position is varied during the focusing process. On the other hand, if the value of “count2” has been set to 4, then the objective lens has reached the bottom of its focal range and the program sets the variable “tot1” to 13. This value is an approximation of the distance between the sensor and the instrument-mounted target when the instrument is in focus and the objective lens is at the bottom of its focal range. Setting “tot1” to 13 allows the microscope to refocus when the instrument is moved up and the objective lens moves back within its focal range.

If the original difference between “tot1” and “tot2” is below our one centimeter error band, less than -0.5 cm, and the value of “tot2” is within an acceptable range, the program begins a negative feedback loop which moves the objective lens upward. This loop is identical to the loop which moved the lens downward except that it turns the motor in the opposite direction, adds the distance moved by the objective lens with each step to the difference between “tot1” and “tot2” to make the variable “dif” and subtracts one from “danger” for each iteration. The upward loop terminates either when the dif becomes greater than -0.1 cm or when “danger” reaches -40. When the loop terminates because “danger” has reached -40, the program causes a red LED to flash and sets the variable “count2” to 5. When the loop terminates because the microscope has focused, the program sets “count2” to 1. Regardless of how the loop terminates, the program sets the variable “save2” to “tot2” minus the original difference between “tot1” and “tot2.”

The program then restarts. If the value of “count2” has been set to 5, then the objective lens has reached the top of its focal range and the program sets the variable “tot1” to an estimate of the distance between the sensor and the target when the instrument is in focus and the lens is at the top of its focal range.

If the original difference between “tot1” and “tot2” is within our one centimeter error band, that is, if the value of the difference is greater than -0.5 cm, and less than 0.5 cm, and the value of “tot2” is within an acceptable range, then the program causes a green LED to flash and sets the variable “count2” to 1.

Finally, if “tot2” is not within an acceptable range, for instance, if the instrument is not positioned beneath the sensor, then the program causes a red LED to flash and sets the variable “count2” to 3. When the program restarts and the value of “count2” is 3, “tot1” is set to the value of “save2.” “save2” represents the most recent, valid, in focus position. Therefore, by setting “tot1” to “save2,” the program is able to refocus once the instrument is back in place beneath the sensor within an acceptable range.

Circuitry

The circuitry component of our final design includes the Arduino microcontroller, a stepper motor, a GP2D12 sensor, an L293D dual H-bridge chip, an AC to DC, 12 volt wall outlet converter, several resistors, a red LED and a green LED. The Arduino microcontroller draws power from a personal computer via a USB interface. The GP2D12 sensor draws power from a five volt output pin on the Arduino and returns an analogue voltage signal to the 0th analogue input pin on the Arduino.   The red and green LEDs draw power from digital output pins on the Arduino which provide five volts intermittently based on the Arduino program. The L293D chip draws power from the 12 volt converter. This power allows the chip to amplify signals, sent to it by four digital output pins on the Arduino, to provide sufficient current to the four pins of the stepper motor which are each powered either on or off in a specific pattern to produce uniform stepper motor steps. Each component is also connected to the ground pin on the Arduino to complete their respective circuits.

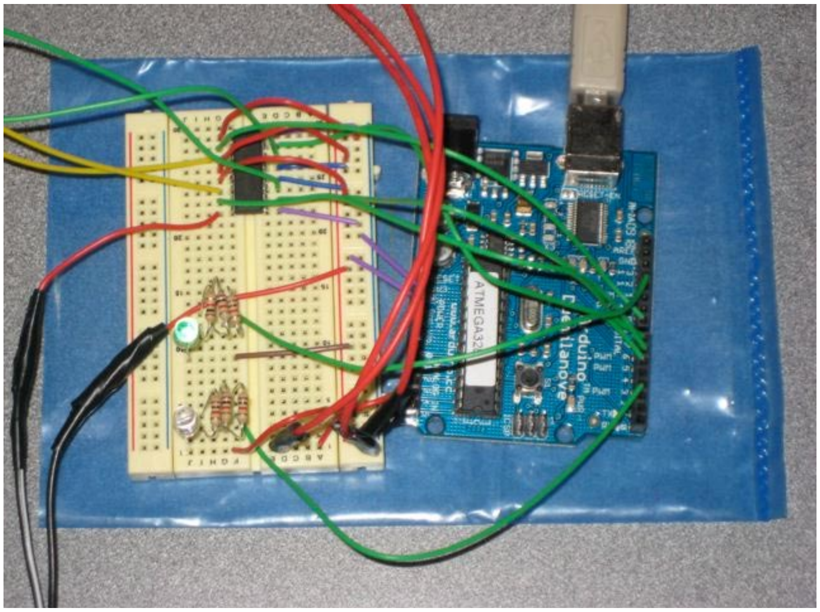

Figure 4: Circuit and Microcontroller
The figure shows the Arduino microcontroller (right) and the universal breadboard (left).

Testing

As in any design process, testing plays a large role in validating the final product. To test our final design, we compared differences in distance readings given by the Sharp GP2D12 infrared LED sensor to the actual change in displacement of the lens of the microscope. To gather distance readings from the Sharp GP2D12 we used the computer software provided with the Arduino microcontroller to display a distance reading (in centimeters) that corresponded to voltage outputs from the LED sensor. The displacement of the microscope was measured using digital calipers. This was done by subtracting initial height of the microscope lens from the final height of the lens after the program and stepper motors adjusted to the distance change detected by the sensor. The expected change based on the sensor readings and the actual change measured using calipers were then compared. A table and graphical representation of the test results are included below.

Trial #	Actual Change (cm)	Expected Change(cm)
1	0.83	1.03
2	1.08	1.18
3	1.79	1.4
4	1.83	1.9
5	1.95	2.08
6	1.98	2.15
7	2.04	2.18
8	2.05	2.2
9	2.81	2.93

Table 1: Final Test Results
The table shows the data obtained by nine separate trials. The expected change represents the change expected from the sensor output. The actual change is the height change witnessed by the microscope.

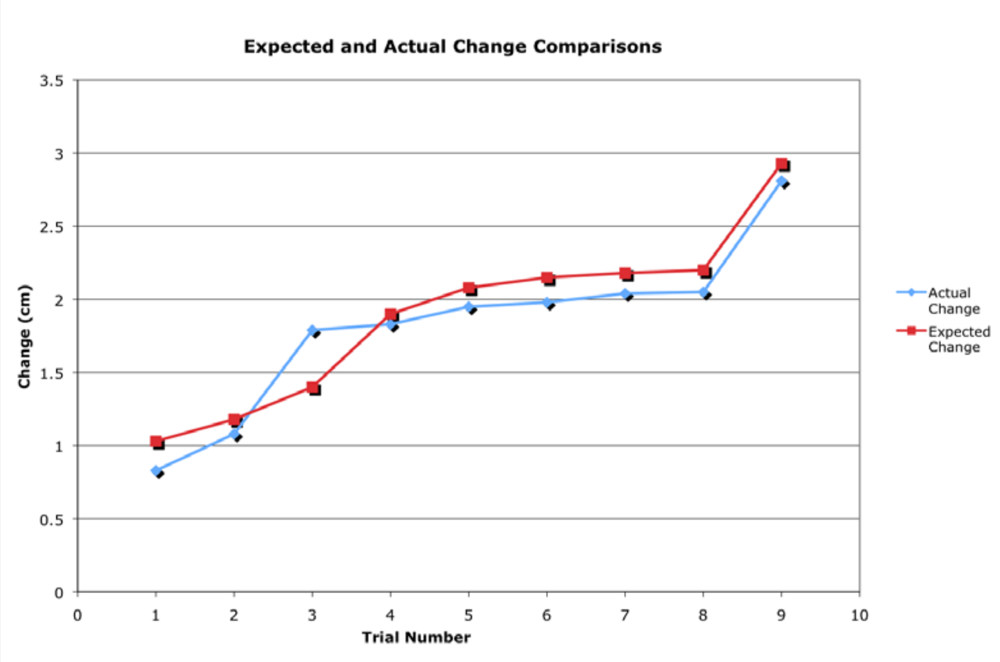

Figure 4: Expected and Actual Change Comparisons
The graph shows two scatter plots of the data obtained in testing.

The testing showed that the average difference in expected and actual change of microscope depth was 0.163 cm with a standard deviation of 0.093cm. Close proximity of the curves along with small average difference and standard deviation values depict that our design moves consistently and accurately as the sensor readings change. These test results validate that our program and motor setup function as desired, in that the margin of error in of the actual and expected change is quite small. The slight discrepancy in the expected and actual change is partially attributed to the margin of error written into the program. The program controlling the stepper motors is designed to run continuously until the height has changed to within 0.1 cm of the target value. This was chosen as it is greatly smaller than the depth of field of a typical surgical microscope which exhibits depth of field of approximately 0.6 cm. Clear focus will be achieved when the lens is within this margin of error of the desired value. The testing done on the final design of our program validates our design's accuracy and precision with its consistent values and small error margins.

Ergonomics

There were many considerations to be made regarding the ergonomics of our final design. The aspirator disk itself was constructed in such a manner that it can be used on different sized instruments without any adjustments or modifications, and can easily be used by any surgeon regardless of background with brief training. The disk was fabricated as small as feasibly possible in order to minimize obstructions caused by the disk, and does not have any sharp corners or edges that could harm the patient or surgeon during use. The aspirator disk assembly allows for wireless use, which contrary to other design alternatives, requires no power or wires that could inhibit the surgeon's flexibility or cause safety concerns for the patient or the surgeon. In order to enable the most efficient placement and user placement feedback, LEDs were incorporated into the circuit board and program. One LED blinks green when the distance is being read correctly, but there is no change (above threshold) in the distance of the aspirator disk. Another LED blinks red when the aspirator disk is not under the sensor, notifying the user that their placement is incorrect. This allows real time feedback to the user, resulting in increased efficiency of use of the device. The program actually has its own ergonomics considerations to enable efficient use by the surgeon. If the sensor reads values larger than a certain threshold (varies depending on sensor location) the program will not run through further loops to make focus adjustments. This is extremely helpful because it allows the surgeon to quickly remove the aspirator out from under the sensor in case of emergency or accidents without signaling the adjustment to focus on the depth of the background.

Costs

There was a preliminary budget of around $300 for the fabrication of a working prototype. The final cost of the prototype constructed is $287.11, showing that we were a bit under budget with the construction of our project. This number is actually lower than it otherwise could have been for a few reasons. The stepper motor, monitor and microscope used were not actually purchased, but instead borrowed from the bioinstrumentation lab. These items were important for the proof of design, but were mainly used as visual aids to show that our design worked as intended and would not need to be purchased for implementation into a neurosurgical microscope.

Item	**Total Item Cost**
Sharp GP2D12 IR Sensor (2)	$19.98
PVC	$1.84
Steel Plate for Motor	$10.61
Magnetic Sensor	$8.97
Miscellaneous Supplies	$50.00
Rare Earth Magnet	$2.00
L293D Chip (3)	$10.00
45° Miter Gears	$10.58
Arduino Microcontroller	$30.00
USB Cable	$2.42
SIRC-01 Sensor Cables	$3.90
Jumper Wire Kit	$5.75
5mm IR LED 940nm (2)	$3.98
Resistors (5)	$1.00
Universal Breadboard	$8.99
Steel Rod	$6.00
Epoxy	$25.00
Poster	$43.75
Shipping/Tax	$42.34
TOTAL	**$ 287.11**

Table 2: Cost Table
The table shows the costs for the entire semester, from initial testing to the final prototype and poster.

Time Management

The majority of time exhausted on this design project was divided into two main categories. The first of which, background research and analyzing design alternatives, consumed the entire first half of our semester. This research and analysis involved looking into position sensing systems, selecting the appropriate system for our application, and ordering materials.

During the second half of our semester, after the materials had arrived, the time dedicated to the project increased markedly. This time was spent fabricating the final design. Fabrication of the final design consisted of three major subdivisions. The first, programming the Arduino microcontroller and designing the control circuit, consumed approximately one third of the total fabrication time. The second, fabricating the stepper motor actuator assembly and the aspirator target, consumed approximately another third of the total fabrication time. The third and final allotment of time was directed toward the testing and refinement of our final design. Good time management and delegation was essential to the success of this project.

Ethical Considerations

There were several main ethical considerations that were taken into account in the design and testing of our prototype. Specifically, the testing of our device was conducted in a manner to show the capabilities of our program, not our prototype as a whole. This could be misleading to someone who initially reads our testing data. This was provoked due to incorrect readings that were constantly being obtained by our sensor. In order to show the legitimacy of what we were in control of designing and fabricating, the data was taken from changes of the microscope focus based on readings from the sensor. This means that even if readings from the sensor were incorrect, our testing results would still show prototype accuracy if it followed the sensor readings correctly. This is still honest data, but it may be interpreted incorrectly if not mentioned. The inaccuracy of the sensor also contributed to the lack of the ability to conduct long term tests on the prototype. In order to improve our testing, we would need to find a sensor that detects distance much more accurately and reliably.

In order to get a fundamental knowledge of autofocus technology and other techniques used in current neurosurgeries, literature was searched and evaluated. Several publications were used and cited in this paper, although most were used exclusively for background knowledge. As far as we could determine, there is nothing currently existing that addresses the problem the way that we did with our prototype.

Future Work

In upcoming semesters we hope to make some modifications in order to be able to integrate our design into a neurosurgical microscope. Although the circuitry will primarily remain the same, connections will need to be made so that the Arduino microcontroller controls the motors of an operating microscope instead of the Vexta stepper motor. Along with this, some slight modifications will need to made to the microcontroller's program to reduce the time it takes for the motor to adjust the focus from one depth to another. This will be important in guarantying that the design is as efficient as possible.

Although we were able to test the short-term accuracy of our final design, in the future we hope to perform more advanced testing. This will include ensuring that our system is capable of delivering accurate results over several hours of consistent use, as it will need to be implemented in surgeries that last for these lengths of time. Also, once integrated into a neurosurgical microscope it will be necessary to retest the accuracy of our Arduino program, as the signals it sends will be controlling a different type of motor.

In order to explore all possible ways of incorporating our design into a neurosurgical microscope, we feel that it would be advantageous to do further research into the existing methods for controlling focus. In-depth knowledge of their benefits and disadvantages would assist us in optimizing the ability for our design to function in a surgical environment. Another option that could be considered is the possibility of combining our system with one of these techniques or integrating it into the autofocus that currently exists in the microscopes.

As our current sensor is rather inconsistent in the readings that it relays to the microcontroller’s program, we would like to replace it with a more accurate sensor. Our client suggested revisiting the idea of using magnetic sensing, because it would eliminate the need for the instrument to be in the direct path of the sensor. If a strip of a neodymium magnet is attached to a surgical instrument, then a compass in the presence of its magnetic field will orientate itself accordingly. Either using three plane compasses or a single three-dimensional compass could accomplish the necessary direction. A further way of detecting this orientation and sending the appropriate signals to the focus-control program would then be needed.

References

[1] “Neurosurgery.” c2009. Encyclopedia of Surgery. <http://www.surgeryencyclopedia.com/La-Pa/Neurosurgery.html> 14 October 2009.

[2] Medow, Dr. Joshua. Personal interview. 07 September 2009.

[3] Nakamura, K. (2008). In Mitaka Kohki Co. L. (Ed.), Mouth switch mechanism for operation microscope (385/384 ed.). Tokyo,Japan: G02B 21/00.

[4] Thorlakson, R. G. (1998). Method and foot pedal apparatus for operating a microscope (74/512 ed.). WA/ USA: G05G 1/14.

[5] Price, J. H., & Gough, D. A. (1994). Comparison of phase-contrast and fluorescence digital autofocus for scanning microscopy. Cytometry, 16(4), 283-297

[6] Uluç, K. M. D., Kujoth, G. C. P. D., & BaşKaya, M. K. M. D. (2009). Operating microscopes: Past, present, and future. Nuerosurgical Focus, 27(3), E4.

[7] http://www.phidgets.com/products.php?product_id=1108

[8] http://www.inexglobal.com/products.php?type=addon&cat=app_sensors&model=gp2d12

[9] http://arduino.cc/en/Main/ArduinoBoardDuemilanove

[10] http://www.societyofrobots.com/programming_PID.shtml

Appendix

Product Design Specifications
Autofocus Microscopy
Group Members: John Byce, Justin Gearing, Mason Jellings, Sarah Reichert
Advisor: Prof. Yen

Function:

During a neurosurgery, a surgeon operates in a cavity which may measure up to six inches in depth. Navigation within this cavity is aided by an operating microscope which exhibits a depth of field of approximately half an inch. Consequently, the surgeon may only observe, and therefore operate within, a relatively small fraction of the cavity at any given time. In order to change the depth of operation, a task which most surgeries require be carried out frequently, the surgeon must halt the surgery and refocus the microscope by hand to the desired depth. The refocusing process not only consumes valuable time but also disrupts the surgeon's train of thought and any maneuvers which require continuous movement of an instrument from one depth to another. Our product will function to eliminate the need to refocus the microscope as the surgeon transitions from one depth of field to the next by allowing the microscope to alter its focus in response to changes in the position of the surgeon's instrument tip.

Client Requirements:

- The device must not interfere with any surgical maneuvers
- Must be lightweight
- Must be compatible with current microscopes and other surgical equipment
- Must hold up to sterilization
- Must keep instrument in depth of field at all times
- Must not interfere with other instrumentation
- Must not harm patient or medical personnel

1. Physical and Operational Characteristics

a.) Performance Requirements: The device must track the position of the tip of a surgical implement and refocus accordingly. It should refocus without any significant lag time (<1sec.) and should work with any surgical implements.
b.) Safety Requirements: The device must not interfere with the surgeon's ability to perform to his or her best abilities. It must not release any harmful substances during surgery. All electronics must be housed appropriately. Any components which will come in contact with the patient during surgery must be easily sterilized.
c.) Accuracy and Reliability: The device must track the position of the implement tip to within one centimeter (initially) of its actual position and refocus the microscope to the same degree of accuracy. It must retain this degree of accuracy throughout its lifetime.
d.) Life in Service: Provided it is regularly serviced along with the microscope and does not undergo unnecessary abuse, the device should last three to five years.
e.) Shelf Life: Long periods of inactivity should have no effect on the performance of the device.
f.) Operating Environment: The device will function in an operating environment. This suggests it will not encounter extreme temperatures or humidity. The device is intended to be used in a clean, dust-free environment in order to optimize the performance of electronics. It will also be designed to withstand continuous UV light exposure.
g.) Ergonomics: The device should be easy to use for any surgeon with minimal electronics background. It will be user friendly such that someone skilled in the art of surgery could use it without problems.
h.) Size: The device will be used in an operating room, where space is at a premium. For this reason, the footprint of the entire device should be minimized, with a maximum of 6" X 6".
i.) Weight: The component mounted on the surgical instrument should not inhibit fine adjustments by the surgeon. The entire device must not exceed 5 pounds.
j.) Materials: Any materials are welcome, provided they are safe for use in an operating room.
k.) Aesthetics: Aesthetics should not affect any aspect of our design as our client prefers function over appearance.

2. Production Characteristics

a.) Quantity: One complete prototype will be fabricated.

b.) Target Cost: Firm guidelines for cost have yet to be established, but there will be refinance allowance of $500.00.

Appendix 2

```
float tot1 = 0;
float tot2 = 0;
float dif = 0;
float difsave = 0;
float save = 0;
float save2 = 0;
int danger = 0;
int count2 = 0;
int one = 5;
int two = 6;
int three = 7;
int four = 8;
int count = 1;
void setup()
 {
 Serial.begin(9600);
 pinMode(one, OUTPUT);
 pinMode(two, OUTPUT);
 pinMode(three, OUTPUT);
 pinMode(four, OUTPUT);
 pinMode(3, OUTPUT);
 pinMode(12, OUTPUT);
 }

float read_ir(byte pin) {
		float temptot;
		temptot = analogRead(pin);
		if (temptot < 3)
			return -1; // invalid value

		return (6787.0 /(temptot - 3.0)) - 4.0;
}
void loop()
 {
   if (count2 == 0)
   {
     tot1 = read_ir(0);
   }
   else if (count2 == 3)
   {
     tot1 = save2;
   }
   else if (count2 == 4)
   {
     tot1 = 13;
   }
   else if (count2 == 5)
   {
     tot1 = 13;
   }
   else
   {
     tot1 = save;
```

```
   }
Serial.println(tot1);

delay(1000);
tot2 = read_ir(0);
Serial.println(tot2);
if (tot2 - tot1 > .5 && tot2 < 25 && tot2 != -1)
{
  dif = tot2 - tot1;
  difsave = dif;
          while (dif > .1 && danger <= 400)
        {
          if (count == 1)
         {
              digitalWrite(one, HIGH);
              digitalWrite(two, LOW);
              digitalWrite(three, HIGH);
              digitalWrite(four, LOW);
             count = 2;
         }
          else if (count == 2)
         {
              digitalWrite(one, HIGH);
              digitalWrite(two, LOW);
              digitalWrite(three, LOW);
              digitalWrite(four, HIGH);
              count = 3;
         }
           else if (count == 3)
         {
              digitalWrite(one, LOW);
              digitalWrite(two, HIGH);
              digitalWrite(three, LOW);
              digitalWrite(four, HIGH);
              count = 4;
         }
            else if (count == 4)
         {
              digitalWrite(one, LOW);
              digitalWrite(two, HIGH);
              digitalWrite(three, HIGH);
              digitalWrite(four, LOW);
              count = 1;
         }
           dif = dif - .007875;
           danger = danger + 1;
           delay(50);
        }
        if (danger >= 400)
        {
          digitalWrite(3, HIGH);
          delay(500);
          digitalWrite(3, LOW);
          count2 = 4;
        }
        else
```

```
        {
          count2 = 1;
        }
        save2 = tot2 - difsave;
}
else if (tot2 - tot1 < -.5 && tot2 < 25 && tot2 != -1)
{
        dif = tot2 - tot1;
        difsave = dif;
          while (dif < -.1 && danger >= -40)
        {
          if (count == 1)
         {
              digitalWrite(one, LOW);
              digitalWrite(two, HIGH);
              digitalWrite(three, LOW);
              digitalWrite(four, HIGH);
             count = 4;
         }
          else if (count == 2)
         {
              digitalWrite(one, LOW);
              digitalWrite(two, HIGH);
              digitalWrite(three, HIGH);
              digitalWrite(four, LOW);
              count = 1;
         }
           else if (count == 3)
         {
              digitalWrite(one, HIGH);
              digitalWrite(two, LOW);
              digitalWrite(three, HIGH);
              digitalWrite(four, LOW);
              count = 2;
         }
            else if (count == 4)
         {
              digitalWrite(one, HIGH);
              digitalWrite(two, LOW);
              digitalWrite(three, LOW);
              digitalWrite(four, HIGH);
              count = 3;
         }
           dif = dif + .007875;
           danger = danger - 1;
           delay(50);
        }
        if (danger <= -40)
        {
          digitalWrite(3, HIGH);
          delay(500);
          digitalWrite(3, LOW);
          count2 = 5;
        }
        else
        {
```

```
          count2 = 1;
        }
        save2 = tot2 - difsave;
}
else if (tot2 - tot1 <= .5 && tot2 - tot1 >= -.5 && tot2 < 25 && tot2 != -1)
{
  difsave = 0;
  digitalWrite(12, HIGH);
  delay(500);
  digitalWrite(12, LOW);
  count2 = 1;
}
else
{
  digitalWrite(3, HIGH);
  delay(500);
  digitalWrite(3, LOW);
  count2 = 3;
}
save = tot2 - difsave;
}
```