VisuAid

School of Engineering Science
Simon Fraser University
8888 University Drive
Burnaby, BC V5A 1S6

March 10, 2014

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser Univeristy
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440 Design Specifications for **Now I See**, a Travel Aid for the Visually Impaired

Dear Dr. Rawicz,

Please find the design specification documentation for our travel aid for the visually impaired, **Now I See**, enclosed herein. Our device examines the frontal proximity of the user using a depth camera and informs if obstacles are present through a vibratory user interface worn on the forehead. With this device, we aim to assist our clients towards a more independent lifestyle by providing a novel way to perceive their environment and navigate it with increased safety.

This design specification documentation presents a high-level design for the proof-of-concept device of **Now I See**, as well as justification and detail specification for each design scheme. Design provisions and speculations for our device are examined in detail, including its hardware and software components, as well as testing considerations. This document will serve as a comprehensive reference during the development phases of our device.

VisuAid is a vibrant company consisting of three enthusiastic biomedical students: Anita Kadkhodayan, Steven YM Lee, and Daria Numvar. We very much look forward to your comments on this documentation. Please feel free to contact us for any concerns or questions. You can reach us by phone at 604-763-4010 or by email at akadkhod@sfu.ca.

Sincerely,

Anita Kadkhodayan
Vice President
VisuAid

Enclosed: Design Specification for **Now I See**, a Travel Aid for the Visually Impaired

VISUAID

DESIGN SPECIFICATION for **NOW I SEE**

A Travel Aid for the Visually Impaired

Project Team: Anita Kadkhodayan
Steven Lee
Darya Namvar

Contact person: Anita Kadkhodayan
akadkhod@sfu.ca

Submitted to: Dr. Andrew Rawicz – ENSC 440W
Dr. Steve Whitmore – ENSC 305W
School of Engineering Science
Simon Fraser University
March 10th, 2014

EXECUTIVE SUMMARY

The ability to live independently is a significant and desirable quality for the visually impaired; however, moving around without direct help from another individual can be a difficult feat. At **VisuAid**, we are developing **Now I See**, our new travel aid device for the visually impaired, that will provide our clients with a novel way to perceive their environment.

Through a depth camera, the user's surroundings will be examined and any obstacles (including pits) will be reported through a vibratory interface worn on the forehead. With this information, the user will be able to 'sense' the obstacles and their relative positions, thereby acknowledging and avoiding them. Also, by moving their head they will be able to estimate the size of an obstacle as the camera will detect its edges.

There has been a large amount of effort devoted to help visually impaired to achieve independent mobility. Travel aid devices for visually impaired has taken many shapes from a simple white cane to complicated robots, however, among all, closest parallel to our work would be the study done by a group from university of Tokyo[3]. In their work they suggest translating the image taken by the camera to an array of electrodes placed on the forehead. The outline of the objects would be converted to tactile sensation by electrical stimulation. Another group from MIT suggested mapping the image onto vibrating sensors placed inside a jacket that can be worn over the user's clothes[6]. Conversely, none of these researches' product has been commercially available yet.

This document outlines the details of the design specifications and provides in depth calculation and analysis of the different aspects of the project. First, an overview of the system is explained, introducing these aspects and materials used for this project. Next, we discuss in detail the hardware design and offer specifications of the parts used, justifying all functionalities. Lastly, factors and components of the software design of current POC device is outlined and discussed.

These specifications outline the detailed design of our POC device. Access to this documentation will be provided to all members of **VisuAid**, and engineers are highly encouraged to refer to this documentation for during the implementation the **Now I See** travel aid.

TABLE OF CONTENTS

GLOSSARY

POC	Proof of Concept: a prototype device created to exhibit core functionalities of the **Now I See** travel aid
FOV	Field of View: may also refer to the artificial frontal view (since the user is visually impaired) in perspective of the user's face
UI	User Interface: unless otherwise specified, this refers to the forehead interface of the **Now I See** travel aid device
CEHW	Core Electronic Hardware Component: refers to the following components of the **Now I See** travel aid device: depth camera, accelerometer, central processor, and micro processer
SWP	Software Package: refers to the software components of the **Now I See** travel aid device
PSU	Power Supply Unit: the module used for the **Now I See** travel aid device
Mount	Refers to device mount onto which some or all parts of the CEHW and PSU of the **Now I See** travel aid device are secured
DOF	Degree of Freedom
SBC	Single-board-computer
GPIO Pins	general purpose input/output pints; generally used to interface with electronic components
UICC	User Interface Control Circuitry; refers to the circuitry prepared to interface the UI and the microcontroller; unless said otherwise, includes the microcontroller
ROS	Robot Operating System

LIST OF FIGURES AND TALBES

1. INTRODUCTION

Now I See is a travel aid designed for the visually impaired to assist them towards a more independent lifestyle. **Now I See** scans and detects obstacles in the user's surroundings and alerts them through a vibratory user interface worn on the user's forehead. This document records the design specifications of **Now I See** to illustrate current design of our proof-of-concept device (POC) and provide justifications for those design choices. This document also provides test plans for the POC device and its functions.

1.1. Scope

This documentation prescribes the design specifications that were chosen for the POC of the **Now I See** travel aid. These specifications correspond to the 'high priority' functional requirements listed in the Functional Specification document of the **Now I See** travel aid. Each design subject is presented, and the chosen solution is explained. Then each decision is explained and justified in comparison to other possible solutions. Furthermore, a list of functional test plans is presented, which will be used to evaluate the functionalities of the POC device of the **Now I See** travel aid. Low level and detailed Figures and tables are numbered and included in the appendix.

1.2. Intended Audience

This documentation is intended for use by all members of **VisuAid**, and was created to serve as a reference during the development of the POC device of the **Now I See** travel aid. Development engineers should refer to this documentation for the implementation objectives, and testing engineers for test conditions. This document will be used as a measure of compliance and progress of the current project.

2. SYSTEM SPECIFICATIONS

2.1. System Overview

The system overview of **Now I See** travel aid is illustrated in the diagram below.

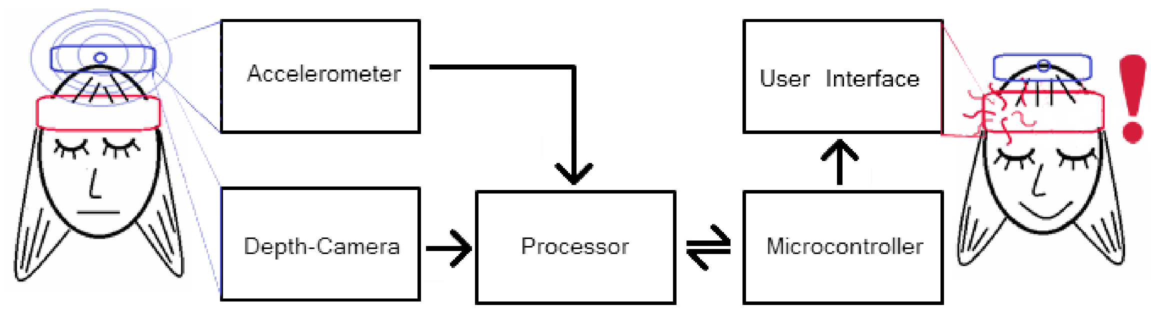


Figure 1: Now I See Travel Aid System Overview

The **Now I See** travel aid consists of a hardware device and software package. The hardware device consists of a number of electronic devices that can obtain information about the environment and a user interface module to communicate relevant aspects of the environment to the user. The software package includes image processing, decision making, and device control programs to handle data and produce suitable instructions to UI device.

The hardware device includes a depth camera, an accelerometer, a central processor, a microcontroller, a set of vibration motors, a power supply module, a wearable device mount and UI headband. The depth camera will provide the depth image of the frontal FOV, which will be examined in the processor, and appropriate alerts will be delivered to the user interface through the microcontroller.

Most of device, excluding the processor and the battery, will be mounted on the user's head, and a battery will be used to power the device for mobility. The initial design had the whole device mounted on the user's head, but this had to be compromised due to technical issues, discussed in the CEHW section and the PSU section. The user interface consists of a 3x5 array of vibration motors that are implanted into a comfortable head band.

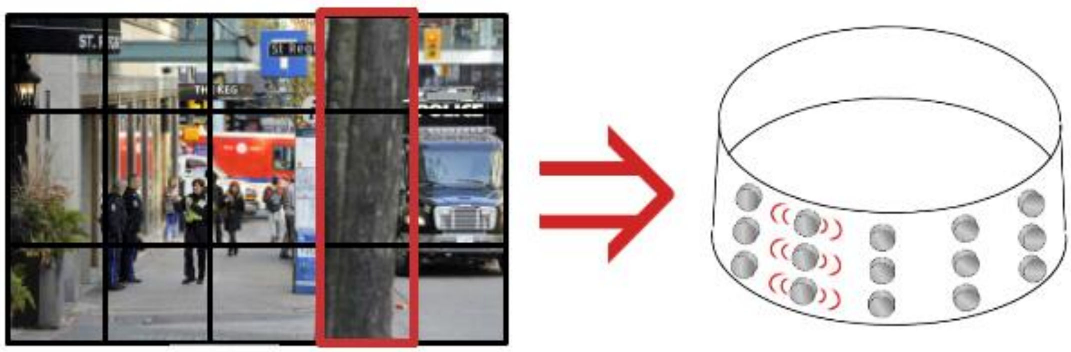


Figure 2 UI Illustration – Mapping of Subsections to Corresponding Motors

Based on the depth image and accelerometer data, the floor will be detected and the FOV will be corrected to the frontal plane (as opposed to a FOV in a declining angle). The corrected FOV is split into 15 subsections (3 horizontal, 5 vertical) and the presence of obstacles in each subsection is reported to the user at the corresponding location on the UI. Sudden drops or pits in the ground are also detected as obstacles.

3. GENERAL SYSTEM DESIGN

This section of the document will discuss the general system design of the **Now I See** travel aid. More detailed design specifications of the device are discussed in the corresponding sections.

3.1. Device Placement

The production model of **Now I See** travel aid device is designed to be mounted on the user's head. While it is possible to physically separate various portions of the device and place them on different locations of the user's body, a goal was set to build the device as compact as possible, putting the whole device onto the user's head. This design goal was chosen at the conception of our project in order to simplify the equipment process of the device. However, for the current POC device, this goal has been slightly compromised due to technical difficulties and weight issues, resulting in the processor and the battery to be carried separately in a backpack.

There exist a number of devices comparable to our device, that use various locations of user's body as mounting points. A solution[1] conceived by a team at University of Southern California, uses a sunglass-like sensor mount, a wearable vest for the user interface, and a backpack for the rest of hardware. It is notable that, in comparison to our system, their software components are computationally more complicated and, therefore, require stronger hardware and significantly more electrical power. Without such need for power and hardware, our production design will target for compactness, but our current POC device resemble their design.

Another comparable solution, titled Kinecthesia[2], compacts their device into a single wearable belt. This solution serves well for their design as the user interface provides obstacle detection only in three horizontal directions, left, middle, and right. While our device includes similar hardware components as their device, some aspects of our device (e.g. ground detection) dictate our device, especially the depth camera and the UI, to be mounted on user's head.

3.2. Use of Depth Camera

Now I See travel aid uses a depth camera as the main sensor to examine the environment. This also was decided at the conception of our device, based on it's a method of sensing and the format of their output.

The main functionality required for our device is the ability to resolve the distance of objects in user's surroundings. While this choice was conclusive and final even at the beginning of design phase, other possible solutions that could have fulfilled this requirement also exist, which include: triangulation using a stereo camera, a proximity sensor (or an array thereof), and a lidar sensor. The choice for a depth camera instead of these other methods was based on the familiarity with and availability of such devices.

3.3. Use of Accelerometer

An accelerometer is included in the system design as one of the major requirements. As our device needs to resolve location of obstacles in vertical direction (i.e. high, low, and ground) as well as the horizontal, it became necessary to resolve the orientation of the depth camera. Our device was designed to be mounted on user's head, and human head has a large degree of freedom (DOF) in movement, which poses two problems.

The first is in ground detection, which is required to detect pits on ground. Our device is required to detect ground when the ground is within the FOV, but with changing orientation of camera it is quite difficult to do so with the depth data alone, especially when the ground is not even. Knowing the orientation of the camera can help, whose method is discussed in the Ground Detection section of the software design.

The second issue arises in resolving the actual distance of obstacles from the user. As illustrated in the figure on the right, the distance to an obstacle observed from the camera is not always equal to the distance from the user. The current obstacle detection method uses a threshold detection method (more details in the software design section), and therefore there is a need for correction of measured obstacle distance.

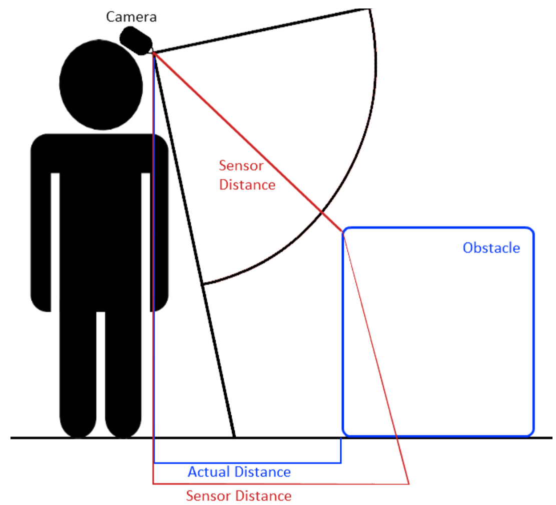


Figure 3 Sensor Distance vs Actual Distance Illustration

3.4. Vibratory User Interface

The user interface of **Now I See** is composed of an array of vibratory motors. This is another main design aspect decided at the beginning of the project. The main objective in designing of our user interface was to establish an effective method of communicating to the user which is sophisticated enough to provide as much information as possible, while simple and intuitive enough to be used easily.

The choice of having a two dimensional array (rather than 1D or single point) of actuators was made for the goal of providing a sensory substitution for lost vision. There has been a number of approaches[3][4][6] to provide a visual sensory substitution using an array of actuators with positive results. Based on these findings, we were convinced that this method of interfacing would be both feasible and effective in communicating to the user.

For the method of delivering, a number of possibilities were considered. The initial design of UI was to use a touch panel with a tactile feedback, where the user would “scan” a small touch panel with his/her finger, and a vibratory feedback would be delivered if an obstacle is detected in the corresponding FOV. This choice was discarded under the requirement that the interface must be passive, as this method requires the user to actively scan the FOV for obstacles. Utilization of electrodes in the place of vibratory motors was also considered, which would communicate through micro-shocks. In addition to its inherent hazardous nature, during research phase, it was also found to be somewhat ineffective[3]. Linear motors were also considered but their availability was too limited for the current stage of our project. With these considerations, vibratory interface using vibration motors were chosen for the harmless nature, ease of implement, and availability.

4. CORE ELECTRONIC HARDWARE DESIGN

Core Electronic Hardware (CEHW) for **Now I See** includes a depth camera, an accelerometer, a central processor, and a microcontroller.

4.1. Core Electronic Hardware Overview

These components provide raw data for the system to work on and the hardware platform to execute its high level functions. The depth camera provides the view of surrounding, the accelerometer will be used to resolve the orientation of the camera, and the central processor and the microcontroller process data and control UI hardware. Microsoft Kinect for Xbox 360 was chosen for the depth camera and the accelerometer, a generic laptop for the central processor, and an Arduino Uno board for the Microprocessor. All connections are achieved via USB. The high level overview of CEHW and their connections are shown below in Figure 4.

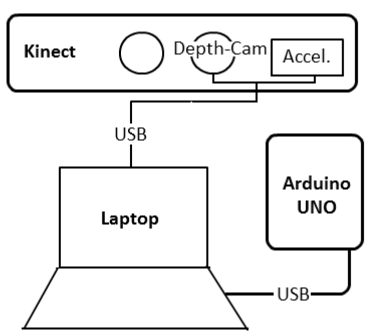


Figure 4 CEHW Overview and Interfacing

In our POC device design, it is notable that two separate devices were chosen for the central processor and the microcontroller. While it was possible to choose a single device that will serve in place for both, which would be capable of general computation and include general purpose input/output pins (PGIO), it was decided to separate the two for a couple of reasons. Which are further discussed in the following sections.

4.2. Depth Camera

The depth camera serves as the main sensor of our device, and the Microsoft Kinect for Xbox 360 (Kinect) was chosen for our POC device. This choice was primarily based on the familiarity of our members with the device and its availability, and was made at the very beginning of conception of our project. While the top level design of our device was already constructed based on Kinect's functional capabilities, the primary considerations for our POC device design were: ability to provide depth information in a 2D array, sensor range applicable to our application, ease of obtaining and manipulation of raw data, affordability and availability, and physical characteristics including weight and size.

Some key specifications of Kinect sensor relevant to our project are documented in the table following the illustration of Kinect's Hardware layout figure Figure 12.

Specification	Capability
Frame Rate	30 frames per second (FPS)
Viewing Angle	43° vertical by 57° horizontal field of view
Output Range Dimensions	640 X 480 pixels
Depth Range	0.4 m ~ 3.0 m
Spatial Resolution	~1.8 mm (at 1.5 m from camera)
Depth Resolution	12 bits, >1 cm
Power Requirement	12V DC, 1.08A, 13W

Table 1 Microsoft Kinect Specifications[13][14]

It is notable that some sensors that are available in the Kinect will not be utilized, while they are nevertheless still powered. Considering our device runs on a finite battery power source, it would be best to remove them, but doing so was deemed too risky, and a decision was made to use the device as it is with a minimal modification. However, the base of the sensor, including the tilt motor, is removed as it considerably interferes with our design; it is in the way of mounting the camera and makes it sit too high and unstable.

In addition, also included in the Kinect is a 3-axis accelerometer. This component is expected to fulfill the accelerometer requirement as well, which is further discussed in the Accelerometer section. Inclusion of this component also positively supports the choice of the Kinect sensor for the current POC device.

As indicated in Table 1, the frame rate at which the depth information is captured was well above our desired rate, which was 5Hz. More detailed analysis and justification regarding the frame rate is discussed in the Software Design section of this document. With regard to the current section, this value of 30Hz or FPS fulfills our requirement as down sampling can safely be applied.

With the viewing angle specified, it was decided that the camera will be mounted at an angle, such that its vertical FOV will span roughly the user's height at a reasonable distance, arbitrarily chosen to be 2.0 m. This was due to a limited vertical view angle of the device and its process is detailed in the Device Mount section of this document. While this limitation reduces the functionality of our POC device, it had to be tolerated in consideration of other benefitting characteristics of the Kinect device.

The output dimensions of the raw data and its spatial and depth resolutions also were considered adequate for our design. With simple calculation (Appendix 2), for our 3X5 user interface, 640x480 image will provide each subsection roughly (since the size of each subsection may vary) 20,000 samples or 0.13 m^2 at 1.5 m distance (~2X letter size paper) to detect obstacles. More detailed analysis of object detection algorithm can be found in the Software Design section of this document.

The Kinect sensor uses a proprietary connector and requires a 12V DC connection drawing 1.08A. A commercially available proprietary to USB2.0 and AC adaptor couple is separately purchased to resolve this issue. This adaptor is modified by removing the AC adaptor portion and creating a direct 12V DC connection to the battery. This modification process is justified by one of our engineers who had a previous encounter with this device involving the same issue. The result of this modification is illustrated in the illustration to the right:

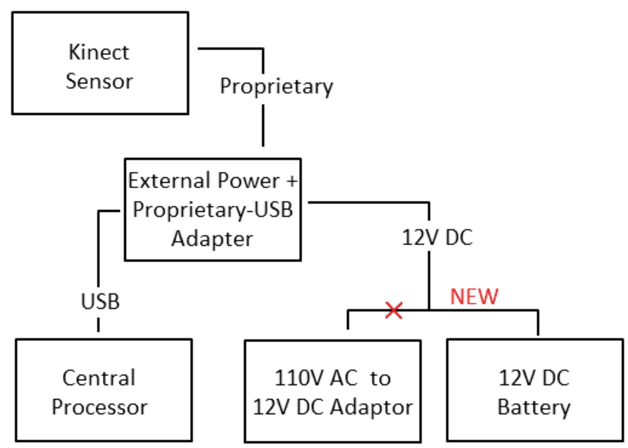


Figure 5 Kinect Connectivity and Modification

4.3. Central Processor

The central processor used for the current POC device is a laptop PC running an Ubuntu Linux environment, and serves as the main hardware platform to perform most of software activities of the POC device. The need for a central processor arose early on during the conception of the project, owing to the large computational power required to handle the depth image generated by the depth camera. Some requirements considered in choosing of a SBC were: capability to run a Linux-based Operating System, availability of USB 2.0 connection, size and reasonable power requirements.

In order to achieve a compact design, a single board computer was (SBC) initially considered, and Raspberry PI Model B (Rasp PI) was chosen from two candidates for the POC device over the alternative, the BeagleBone, for its availability and superior graphic processing capability[16]. However, it was found that the Kinect sensor chosen as the depth camera has critical problems when running on such SBCs. As a result, a compromise was made on the compact design factor, and more stable laptop platform was chosen.

The laptop will be running on its own battery, therefore independent of our battery, and therefore will not be included in the power considerations. It will interface with the camera and microcontroller via USB 2.0 ports as initially planned for this component.

4.4. Microcontroller

An Arduino Uno microcontroller board was chosen as the microcontroller component of our POC device. Inclusion of a microcontroller was necessary in order to interface with the accelerometer and the vibration motors of the UI. It was brought in early in the conception of the project, and the choice of an Arduino board was instinctive as all of our engineers had direct experience with the platform. While this decision was final, some items of consideration were: ease of use, availability of the device itself and online resources, and the number of available GPIO pins.

As our UI design requires minimum of 16 connections, our POC device needed an Arduino board with enough number of GPIO pins. An Arduino Uno board provides 14 digital I/O pins, and therefore an Arduino Mega board was initially considered. However, upon research, it was found that 6 analog input pins can also be used as digital out pins, totalling 20. Since one unit of Uno was already available to us, it was selected as our microcontroller for the POC device.

Specification	Value
Input Power Voltage Range	6 – 20V DC (12V DC used)
Digital I/O Pins	14
Analog I Pins	6 (can also be used as Digi I/O)
Interface	USB 2.0 type B
Power Connection	USB or 2.1mm coaxial jack
Est. Power Consumption	1.0W (5V @ 200mA)

Table 2 Arduino Specification[18][19]

This board is conveniently, owing to the onboard voltage regulator, powered directly from the battery via a 2.1mm coaxial DC connector, and interfaced with the central processor via a USB 2.0 B port. Some relevant characteristics of Arduino Uno board are listed in Table 2, following the Figure 13, which shows the board layout and planned connections.

While the power usage varies largely depending on the usage, an estimate was calculated based on current design of POC device. This board is also part of the user interface control circuitry (UICC), and most of power is consumed in computation, which is quite hard to measure. It was not possible to find power usage details, but based on some values that were found, estimation was made. It was found that the board roughly consumes about 50mA @ 5V when idle [19], and power consumption of the UICC, which is the only other activity taking place on the microcontroller, was insignificant (see UICC section). The final estimation was arbitrarily chosen at four times the idle consumption.

4.5. Accelerometer

For the accelerometer of our POC device, the internal accelerometer included in the Kinect device is chosen for obvious benefits, including savings in budget and energy expenditure, and the fact that it is readily calibrated with the camera's FOV. Initially an external accelerometer unit was included in the design, but upon further research, a method was found that allows for use of the embedded accelerometer of the Kinect under the current software environment design choice.

In consideration of an accelerometer, it was initially required that the acceleration is measured at least on two horizontal axes so that the orientation (more precisely the pitch and the roll thereof) of the camera can be resolved. The accelerometer of Kinect provides acceleration measurement within +/- 2g with 1° accuracy **[13]**, which is enough resolution for our purpose. Eliminating the need for an extraneous component and providing a sufficient data, it was sensible to choose this internal accelerometer for our POC design. No extra power consumption is created by this component. Resolving the orientation of the camera is further discussed in the Software section.

5. DEVICE MOUNT AND USER INTERFACE DESIGN

The device mount for the **Now I See** POC device includes a physical apparatus in the form of a helmet, one enclosure containing the UICC, and another enclosure containing the PSU, excluding the battery. The user interface module design pertains to the physical apparatus including the vibration motors, the control circuitry, and the controlling software. A power switch is also included as part of UI.

5.1. Device Mount Design

The main advantage of **Now I See** arises from the fact that it moves with the user's head, and the representation follows the user's attention reflected by the direction of face. To achieve this, inclusion of a device mount was necessary, which would secure, amongst other components but most importantly, the camera onto the user's head. For obvious reasons, helmets were considered as the main apparatus, and a bike helmet that was available has been chosen for the role in this POC device. This bike helmet is relatively comfortable and has large breathing holes for air circulation, onto which the components can be fastened.

5.2. Mounting of Depth Camera

As briefly discussed in the CEHW section, the depth camera is required to be installed at an angle due to its viewing angle. This makes it possible to detect obstacles in the lower area, especially on the ground, at a much closer distance. However, a consequent complication arises in turn at the implantation level as well, one of which is the need to resolve the camera's orientation. These topics are further discussed in the Software section of this documentation.

Based on calculation shown in Appendix 3, our current POC design will have the camera mounted at 20° degrees inclining downwards, allowing for ground detection at 2.0 m distance from the user. These values were arbitrarily chosen at this point and will be finalised upon the calibration phase of our project.

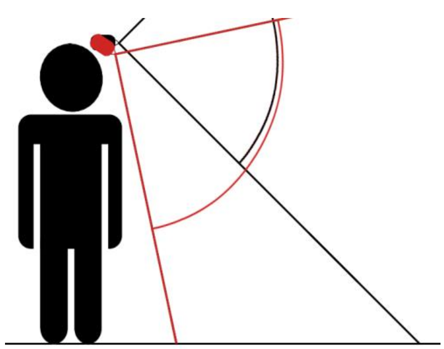


Figure 6 Camera Mount Illustration

A comparison chart showing the view range is included in the same Appendix item. An illustration in Figure 6 below shows the result of such installation, denoted with the red camera, where its vertical range spanning the height of user is observable at a much closer distance, in comparison to a camera installed levelly.

5.3. UICC Enclosure and PSU Enclosure

As specified in the functional specification of the current POC device, all electronic contact points must be concealed from physical contact, for safety and durability reasons. For the current POC device, one rectangular enclosure will be created to house components that need to be concealed. In the following figure, the layout inside the enclosure is illustrated. Its design specifications are listed in the Figure 14 and Table 6.

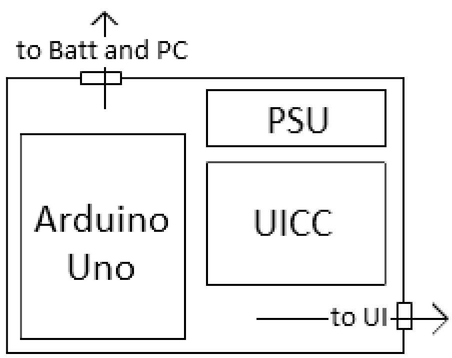


Figure 7 Component Layout inside Enclosure

Plastic was chosen for the material based on the availability and ease of manipulation. Some further modification of these enclosures is expected. Plastic enclosures are widely available in various shapes and sizes at reasonable prices. It was sensible to make this choice, for plastics are light, durable, electrically insulating. As the finalized circuitries have not yet been implemented, the actual enclosure also has not yet been purchased, in case of design changes.

5.4. UI Design

The user interface module of our POC device contains 15 vibration motors and a head band onto which those motors are placed. A head band was chosen for some important functional capabilities, while being easily available, these capabilities include: to absorb moisture, to push the motors onto the skin of user for improved perception, and to structurally isolate the motors from each other.

Small coin type vibration motors were selected for the current POC design. Another alternative was a cylindrical capsule type, but it was decided that the coin shape fits our design better, as it offers larger surface area. Upon research, it was found that the perception of vibration on human skin is optimal in the 100Hz – 300Hz range[20].

We sampled measured our engineers' forehead sizes and decided to span the vibration motors roughly over a 15cm X 5cm space, whose illustration is given in the Figure 15.

Motors are placed at locations indicated by the green grid. It should be noted these values are not final and may be changed during the calibration phase if better design were to be conceived. However, this design can serve as a bench mark in choosing the vibration motors. With these information and consideration, a vibration motor was chosen, which is shown in the figure below and its relevant information is listed in the following table.

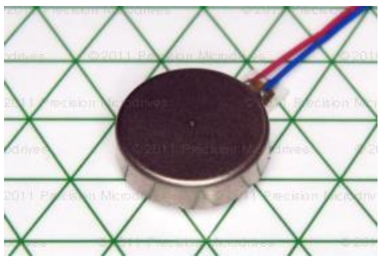


Figure 8 Coin-type Vibration Motor Used in POC Device[22]

Specification	Value
Body Diameter	12 mm
Body Height	3.4 mm
Operating Voltage	3.0 V DC
Rated Operating Current	53 mA
Nominal Power Consumption	160 mW
Nominal Vibration Frequency	180 Hz
Typical Start Current	115 mA
Typical Rise Time	110 ms
Typical Stop Time	210 ms

Table 3 Vibration Motor (312-101) Specification[22]

We concluded that this component meets our requirements based on the information provided; the physical dimensions are adequate, power consumption is reasonable, and the response times were acceptable.

3V operating voltage made it necessary for a voltage regulation circuit to be created specifically for these motors but it was deemed more suitable than high power high intensity alternatives for the current design. Response times were slightly on the long side, but they were short enough for our targeted refresh rate of 5Hz; while it is possible for a scan to be missed due to long response time, we deduced that it may help in coping with false spiky impulses in data.

In controlling these motors, use of multiplexing was considered, but was rejected as activation of any combination of these motors is viable. Multiplexing would only allow one or a limited number of combinations depending on the implementation – multiplexing of all combination has no benefit over parallel control. An accompanying control circuit is designed due to power issues, which is discussed in the following section.

A power switch was chosen to be rocking button switch and located where the user can easily reach, on the side of the device. A rocking button switch was chosen as it is easier to tell without looking if the switch is on, in comparison to other alternatives, such as a push button switch.

5.5. UICC Design

While the vibration motors were suitable in terms of their size and availability, they introduced some complications. The motors draw significant amount of current (53 mA steady state, 115 mA start current [22]) that powering them directly from the Arduino microcontroller was not possible (output pins max out at 40 mA[18]) , the current surges created due to the brushing in the motor core posed threat to the microcontroller's circuitry, and the vibration when fully active was too strong. To resolve these issues, a control circuit was designed to provide sufficient current circumventing the Arduino board, to protect other devices, and also to reduce the intensity of vibration. The vibration motor control circuit was constructed as shown in Figure 16, and its components are listed in the Table 7. 15 of these circuits will be prepared, one for each of 15 vibration motors.

Utilizing a transistor as a switch, the current source is redirected and the I/O pins are protected. The diode, D1, redirects the current surges from the motors, protecting the transistor. This control circuit also provides high input resistance (with correspondingly chosen R1 and R2) seen from the microprocessor's output pins and thus limits current drawn and puts almost no pressure on the Arduino board's output.

The intensity of the vibration will be manually tested by varying the input voltage to find a comfortable intensity and the corresponding current value through the motor. Then the input voltage that corresponds to the desired vibration intensity is traced back to obtain the voltage divider's resister ratio based on the spec sheet details[22]. This will dictate the values of R1 and R2.

5.6. UI Protocol

The current design of UI protocol of the POC device is a 3 way operation on the motors, off, blinking, and on. Obstacles in distance will be indicated with a blinking vibration, whose frequency will be decided at the calibration phase, and those close by will be with a constant vibration. The threshold values in distances will be finalised during the calibration phase as well, but are arbitrarily chosen at 2.0 m for far obstacles and 1.0m for close ones. It is planned for the production model to implement more variety of output types, to inform user of different kinds of obstacles. Based on the obstacle detection method implemented in the decisive software algorithm, the motors corresponding to the subsections with obstacles detected will vibrate to inform the user.

Commands are passed down from the processor in a single string, containing 15 comma separated numbers, each of which corresponds to a subsection. Current implementation design involves only zeros and ones, using a string as the message, any alphanumeric value can be passed on, allowing this protocol grow its variety of interfaces.

6. POWER SUPPLY UNIT DESIGN

The power supply unit (PSU) of **Now I See** includes the battery which serves as the main power source for all electronic parts except the central processor and voltage regulating circuitry to create required voltage for the vibration motors in the UI. The battery had to be placed in a distance from the rest of circuitry due to weight issues, but in the production design, they will be co-located on the device mount.

6.1. Battery

As the battery of current POC device, a sealed lead-acid rechargeable battery was chosen, with nominal 12V and 1.3Ah capability. This type of battery was chosen based on affordability and availability. In the production model however, a more detailed analysis shall be performed in choosing the battery for better performance and lighter weight. The current design's specification is show in the Table 4 below:

Characteristic	Specification
Output Voltage	12 V DC (nominal)
Current Capacity	1.3 Ah
Energy Capacity	15.6 Wh
Type	Lead Acid (rechargeable)

Table 4 Battery Specification

Component	Consumes
Kinect	13W (1.08A @ 12V DC)
Arduino	2W (400mA @ 5V DC)
Vibration Motor	2.6W (53mA @ 3.3V DC X 15)
Overhead	+ 15%
Total	**20W**

Table 5 Power Consumption Summary

For the current POC design, this specific model serves well, except in its weight. With the central processor being chosen as a laptop placed in a backpack, it was sensible to place this unit in the backpack as well.

In terms of battery life, it is expected to operate for 45 minutes hours based on our current POC design. This consideration is illustrated in the table below, and the actual battery life performance will be further tested in the calibration/modification phase of development. In the case of insufficient battery life, (arbitrarily chosen to be minimum 30 minutes for POC design), it is planned to recruit an extra battery.

6.2. PSU Circuitry Design

The PSU circuitry was designed to provide each component with the required input voltage and sufficient current. With the compromised explained in the CEHW section, only two voltage values are required, and the circuitry has been greatly simplified for the current POC device.

In designing of this circuitry, combined power consumption of each group of components had to be considered, and as a result some voltage regulating components were repeated in parallel to allow for enough current draw. 12V DC devices will draw power directly from the battery with nominally no limitation in current draw, and the UICC components will receive power via voltage regulators, which have current limits.

For our POC device, to prevent issues that may arise from insufficient current, 5 L4931[23] 3V DC voltage regulators are used. Each of these have a 300mA current limit, and with four units in parallel, current of 1.5A can be drawn at once. Considering the extreme case of all motors being activated simultaneously, it would theoretically need 1.7A (115mA X 15units). This is slightly over the current design's limit, but a design decision was made to include only five as this is a rare extreme case, and it does not really matter in the user's point of view, if one or two motors doesn't vibrate for a fraction of a second when all the rest are starting at the same time. The operating current draw is calculated to be 0.8A (53mA X 15units), which must be and is met.

An overview of this circuitry is shown in the figure on the right. No circuit drawing is included here as the circuit simply just includes 5 regulators in the circuit.

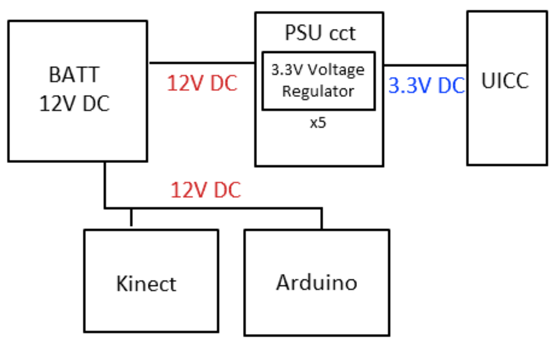


Figure 9 PSU Overview

7. SOFTWARE DESIGN

The Software Package (SWP) of **Now I See** includes the image processing and decision making algorithms to be used in the central processor and the control software for the microcontroller.

7.1. Software Design Overview

A high level overview of the software design is illustrated in the figure below:

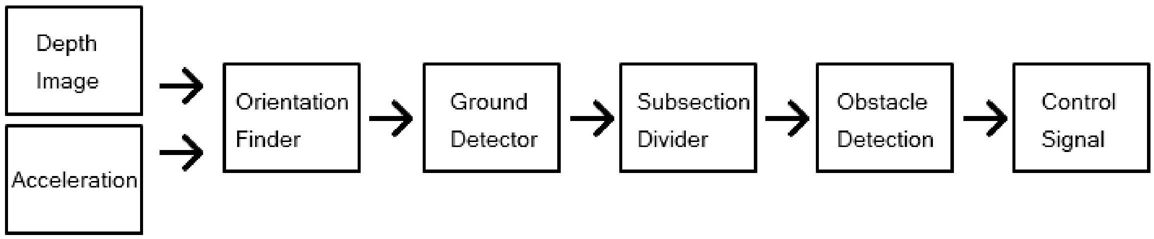


Figure 10 Software Overview

The software package handles the raw data received from the sensors and creates an output control signal to the UI. All processing takes place in the central processor, and the UI commands are passed to the microprocessor, which handles the hardware interfacing through the GPIO pins. Communication amongst processes is taken care of via the Robot Operating System framework and all hardware level communication uses USB connections.

7.2. Operating System and SW Environment

For the current POC design, Ubuntu Linux has been chosen for the operating system environment and Robot Operating System (ROS)[24] as the software framework. While Windows OS environment was also considered, it was decided to be too "heavy" for our purpose, as the initial design included a SBC which would have a very limited computational performance, as well as a small disk space and memory. Also, the production model will not be using Windows OS as its licensing issue will create many problems including increased unit cost. As ROS officially supports Ubuntu OS, working under Ubuntu OS was further justified.

During research, the use of ROS as our frame work was suggested, and while ROS is a robotic software development framework, it was agreed that some of its functional aspects will greatly benefit our project. ROS provides a message passing service between processes, which can be very useful as our software components are modular and require some kind of communication method amongst them. An overview of the software environment is illustrated in the figure to the left. As seen here, most of communication amongst devices and processes are handled by ROS.

ROS also has the hardware driver for the Kinect camera and accelerometer, which again makes working with the device much easier. One of our engineers has had experiences working with ROS as well. With these considerations, it was sensible to utilize ROS as our software frame work.

7.3. Raw Data Acquisition

Raw data in our POC device is obtained using two ROS software packages for the Kinect device. The first package, titled freenect_stack[25], includes an open source library and driver for the Kinect device. When launched, this software fetches the depth values from the Kinect and publishes them in an array of values via the ROS's message passing service, which then can be easily accessed and processed in other software components.

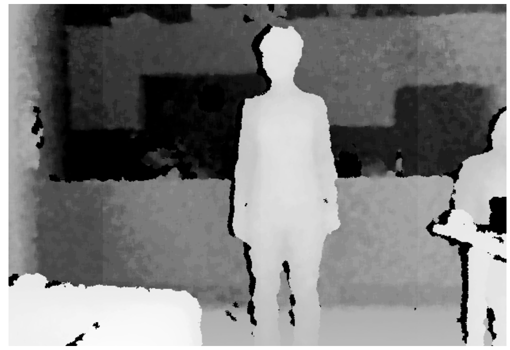


Figure 11 Sample Depth Image Captured by Kinect Depth Camera

There exists another software package that can be used for the Kinect device, titled openni_kinect[26]. Freenect_stack was chosen instead of this package as it is no longer maintained[26]. The accelerometer of Kinect can be accessed via a similar method, using a package titled kinect_aux[27]. The acceleration information is given in a vector form. The figure above right shows a visualization of raw data that is expected from the depth camera. The numeric values in depth, represented by the brightness of each pixel in the visualization.

7.4. Camera Orientation Calculation

The orientation of the camera is obtained via the equation listed in the appendix 5, which is illustrated in the Figure 17. We are ignoring the orientation on the xz plane shown in the diagram below, because it rotates with the user's view. i.e. the direction of z axis follow the camera. Based on the accelerometer data, the orientation of camera is obtained which will be used in the following sections.

7.5. Ground Detection

Ground detection node is used in our POC design as our device is required to detect obstacles on the ground, a pit for example. Based on the camera's orientation information and the known height of user, it is easy to estimate where the ground should be. With this in mind, simple threshold comparison will indicate whether an obstacle is present. The ground detection node carries out this estimation part, by feeding the orientation information back to the equation introduced in Appendix 4m which was the method used to find the camera's installation angle. With this estimation, it is known (or at least estimated) which portion of FOV is the ground, which can be represented as a mask of the same size. This information is used in the subsection division node in the following.

7.6. Subsection Division

15 subsections and corresponding vibration motors is the main feature of our POC device. It is possible to simply divide the FOV into 15 equally sized subsections, but as the FOV changes with the user's head, and the nature of obstacles on the ground is somewhat different from the other ones, it is required to divide the FOV in a dynamic way.

The division in the horizontal plane is more or less uniform. In the vertical direction, the bottom row will be dedicated for the ground, and the rest will be evenly divided. The distribution will be passed on in the form of a mask, each pixel denoting the subsection number. The Figure 18 illustrates this distribution.

7.7. Obstacle Detection

With the subsections decided, presence of obstacles is evaluated. For the frontal space, two thresholds (distance thresholds) are set, arbitrarily at 1.0m and 2.0m, and the depth values are simply compared against these thresholds. Before comparing, ach pixel's value is adjusted based on the orientation of the data, using simple geometry.

For the ground pixels, obstacles will be detected in comparison to the estimated location of the ground obtained in the previous section. A threshold (ground threshold) is set arbitrarily at 5cm and will be finalised at the calibration phase.

Another threshold (count threshold) will be set for the number of pixels with values that exceeds distance thresholds, as noise may exist in the data. This value is arbitrarily set at 3 currently, and will be finalized during the calibration phase. If enough pixels have crossed the distance threshold, corresponding UI command will be assigned to a 3X5 array, which will be sent to the microcontroller at the end of each processing cycle.

The processing rate at which the depth information is evaluated is currently decided to be 5Hz. Based on the consideration that the user of our device will be walking with the device at a normal walking speed, roughly 15cm/s[15], rough calculation can show that with our initial consideration of 5Hz, each scan can cover about 5 cm of change in distance, which would give 20 scans, and therefore 20 computations or chances to alert the user, before the user collides with an obstacle at 1.0 m away.

7.8. Control Software

With the decisions made in the central processor, UI commands are sent to the microprocessor to interface the vibration motors. The software involved in this section is quite simple. The command is given in a string with 15 numeric values separated by commas. Each number corresponds to each subsection and corresponding output pin is switched on or off based on this command.

8. SYSTEM TEST PLAN

The tests for our POC device are performed on individual aspects of our design, and then on the whole device as a field test. Each test plan consists of an input condition and an expected result. Meeting of the expected result in response to the given input will approve the system's functionality.

8.1. CEHW Testing

Depth Camera + Accelerometer Unit Test

- Testing the device working

Input:

Power is plugged in with sufficient charge (>9V), and USB cable is properly connected.

Expected Result:

Using visualisation software, it is possible to look at the data stream.

Incoming data stream can be shown in its raw format.

Acceleration data can be displayed.

Microprocessor Unit Test

- Testing the device working and connections established

Input:

Power and USB cables are connected, and some UI commands are inputted via the serial connection

Expected Result

UI responds to the inputted commands

8.2. Obstacle Detection and UI Mapping

Obstacle Detection Testing

- Testing the detection working

Input:

With device turned on, bring one or more objects in the FOV

Expected Result:

Vibration goes off when closer than the threshold, according to the UI protocol, including different kinds of vibrations.

UI Mapping Testing

- Testing the correct mapping to UI

Input:

With device on, bring objects to specific locations of FOV

Expected Result:

Correctly corresponding motor goes off. For visualisation purposes, an array of LED may be used

8.3. PSU Testing

Battery Life Testing

- Testing the battery life of device

Input:

Turn on device and equip it.

Keep the device working such that the power drainage is realistic

Expected Result:

Device will be functional for minimum of 30 minutes from full charge

Power Supply Unit Unit Testing

- Testing the capability and stability of PSU

Input:

Bring obstacles in all FOV, causing all motors to fire.

Expected Result:

All motors fire for reasonable amount of time (~5 seconds)

8.4. Field Testing

<u>Field Testing of Device</u>

- Testing correct functionality of device in realistic setting

Input:

With the device equipped and blind folded, navigate indoors.

Encounter various types of obstacles, including pits, high obstacles, and low obstacles

Expected Result:

Successful navigation with nominal collisions

9. CONCLUSION

Now I See is a travel aid designed for the visually impaired. It scans and detects obstacles in the user's surroundings and alerts them through a vibratory user interface worn on the user's forehead. The **Now I See** travel aid consists of a hardware device and software package. The hardware device includes a depth camera, an accelerometer, a central processor, a microcontroller, a set of vibration motors, a power supply module, a wearable device mount and UI headband. The design specifications established in this documentation define the current proof-of-concept model of the **Now I See** travel aid. Each aspect of the POC device is presented, considered alternatives are discussed when appropriate, and the current design choice is introduced and justified. This documentation will serve as a guideline for during the creation of POC device. The development of the proof-of-concept device is well on the way, and we expect its successful completion by early April, 2014.

REFERNECES

[1] N Gerard, G Medioni. " Visual Aid for the Blind" Internet: http://iris.usc.edu/people/medioni/current_research.html, [accessed: Mar, 2014]
[2] Eric Berdinis. "Kinecthesia Heptic Belt for The Blind"Internet: http://www.kinecthesia.com/, [accessed Mar, 2014]
[3] Kajimoto, H., Kanno, Y., & Tachi, S. "Forehead electro-tactile display for vision substitution". In Proc. *EuroHaptics*. Jul 2006.
[4] K. A. Kaczmarek, "The tongue display unit (TDU) for electrotactile spatiotemporal pattern presentation" *Scientia Iranica*, 18(6), 1476-1485, 2011.
[5] N. Vuilleme, O. Chenu, A. Moreau-Gaudry, J.Demongeot, Y. Payan. "Motion Control Design for an Economical Robotic Walking Support System: The Robotic Cane", *In Engineering Letters.* Vol. 19. Issue 3. P185, 2011
[6] Massachusetts Institution of Technology. "MIT scientist designs wearable tactile displays for people with vision and hearing disabilities", *In Global Alliance on Accessible Technologies and Environment*. Internet: http://globalaccessibilitynews.com/2013/06/17/mit-scientist-designs-wearable-tactiledisplays-for-people-with-vision-and-hearing-disabilities/ updated 2013 [Accessed: Jan 2014]
[7] Samzenpus. "Hand-Mounted Sonar for The Blind", *In Slashdot*, Internet: http://tech.slashdot.org/story/11/08/22/0156201/hand-mounted-sonar-for-the-blind, 2011, [Retrieved Jan 2014]
[8] Adarfruit. "Proximity Sensor Cane". In *Adarfruit Industries*. Internet: http://www.adafruit.com/blog/2010/05/20/proximity-sensor-cane-for-the-blind/ 2010, [Retrieved Jan 2014]
[9] Y. Z. Zhang, S. S. Yeh. "Motion Control Design for an Economical Robotic Walking Support System: The Robotic Cane". *In Engineering Letters*. Vol. 19. Issue 3. P188-196. 2011
[10] M. K. Hisham. "Tech replaces Seeing Eye Dog". In *TrendHunder Tech*. 2010,[Retrieved Jan 2014]. Available: http://www.trendhunter.com/trends/mygo-cane-could-replace-seeing-eyedogs
[11] C. Brigida. "New Technology That Can Help The Blind". *In USC News*. 2013,[Retrieved Jan 2014]. Available: http://news.usc.edu/#!/article/55803/new-technology-that-can-help-the-blind /
[12] Julius." Electronic Mobility Devices for Persons Who Are Blind or Visually Impaired". *In Even Grounds*, Accessibility Consulting.2010, [Retrieved Jan 2014]. Available:

http://www.evengrounds.com/blog/electronic-mobility-devices-for-persons-who-are-blindor-visuallyimpaired?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+EvenGroundsAccessibilityBlog+(Even+Grounds+Accessibility+Blog)
[13] MSDN Microsoft. "Kinect for Windows Sensor Components and Specifications". *Microsft MSDN Library reference* 2012 [Retrieved march 2014]. Available: http://msdn.microsoft.com/en-us/library/jj131033.aspx
[14] MSDN Microsoft. "Kinect for Windows Sensor Components and Specifications". *Microsft MSDN Library reference – Depth Range* [Retrieved March 2014]. Available: http://msdn.microsoft.com/en-us/library/hh973078.aspx#Depth_Ranges
[15] Levine, R. V. and Norenzayan, A. "The Pace of Life in 31 Countries". *Journal of Cross-Cultural Psychology* 30 (2): 178–205.1999
[16] Michael H. Leonard Blog - "Raspberry vs. Beaglebone", [Retrieved March 2014], Available: http://www.michaelhleonard.com/raspberry-pi-or-beaglebone-black/
[17] "Arduino Analog Pins", [Retrieved March 2014] available: http://arduino.cc/en/Tutorial/AnalogInputPins#.Uxn2Efk2wpo
[18] "Arduino Uno", [Retrieved March 2014], Available: http://arduino.cc/en/Main/ArduinoBoardUno#.Uxpi-Pk2wpo
[19] "Arduino Power Consumption", [Retrieved March 2014], Available: http://gadgetmakersblog.com/arduino-power-consumption/
[20] "Somatosensory Systems", [Retrieved March 2014], Available: http://neuroscience.uth.tmc.edu/s2/chapter02.html
[21] "Comparison of Battery types", [Retrieved March 2014], Available: http://en.wikipedia.org/wiki/Comparison_of_battery_types
[22] "Vibration Motor Spec Sheet", [Retrieved March 2014], Available: https://catalog.precisionmicrodrives.com/order-parts/product/312-101-12mm-vibration-motor-3-4mm-type
[23] "L4931 Voltage Regulator Spec Sheet" [Retrieved March 2014], Available: http://www.st.com/web/en/catalog/sense_power/FM142/CL1015/SC312/SS1733/PF259704
[24] ROS www.ros.org
[25] ROS freenect_stack http://wiki.ros.org/freenect_stack
[26] ROS openni_kinect http://wiki.ros.org/openni_kinect
[27] ROS Kinect_aux http://wiki.ros.org/kinect_aux

Appendix

1. Figures and Tables

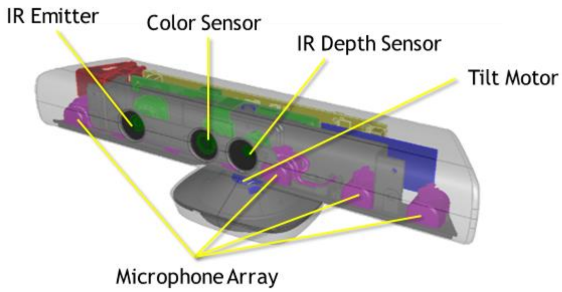


Figure 12 Microsoft Kinect Sensor[13]

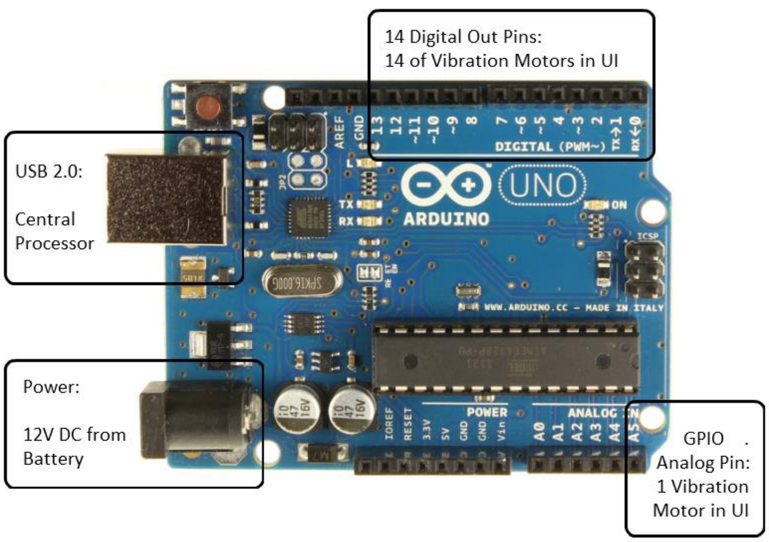


Figure 13 Arduino Uno Microcontroller Board with Planned Connections

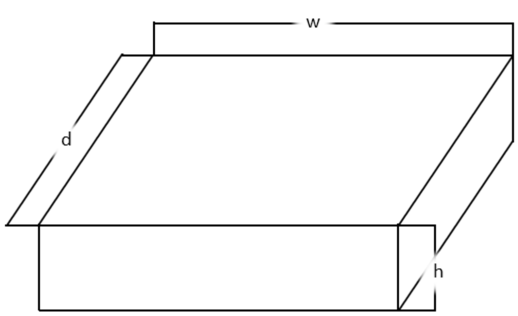


Figure 14 Enclosure Dimensions

Component	Max Width (mm)	Max Depth (mm)	Max Height (mm)
Arduino	70	55	15
PSU	70	20	20
UICC	70	40	20
Enclosure	150	100	30

Table 6 Components and Enclosure Dimensions

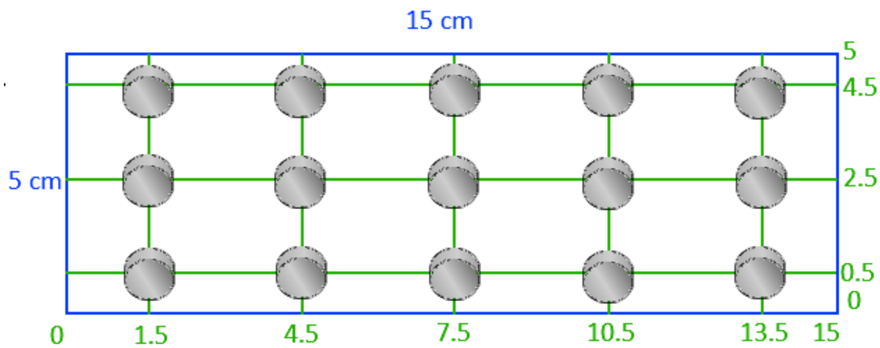


Figure 15 UI Motor Layout

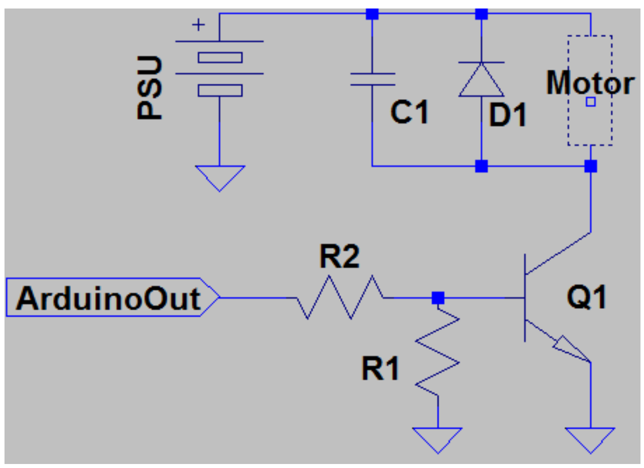


Figure 16 Vibration Motor Control Circuit

Component	Value
C1	0.1 uF
D1	1N4148
Q1	2N2222
Motor	312-101
R1, R2	TBD

Table 7 UICC Components

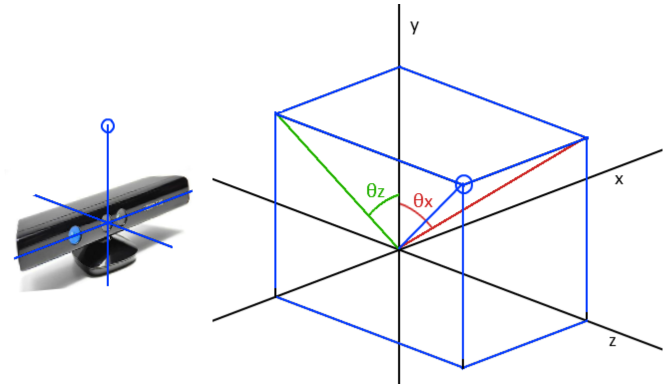


Figure 17 Resolution of Orientation of Camera

High	High	High
Low	Low	Low
Grnd	Grnd	Grnd

High	High	High
Low	Low	Low
Off	Off	Off

Figure 18 Division of Subsection Illustration – Ground Detected (left) and not Detected (Right)

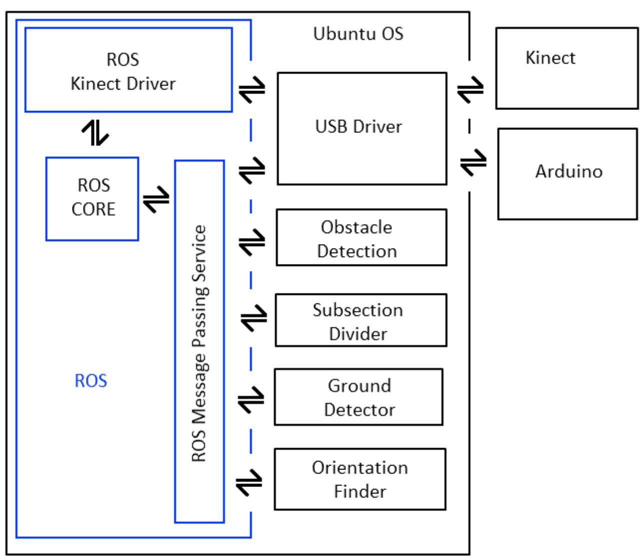


Figure 19 Software Environment Illustration

2. Samples (pixels) per subsections:

$$\frac{x\ (resolution).\,y\ (resolution)}{\#\ of\ subsections} = \#\ of\ pixels$$

Substituting 640 pixels for x resolution and 480 pixels for y resolution and 15 for number of subsections, it follows that:

$$\# \; of \; pixels = \frac{640 \; X \; 480}{15} = 20480 \; (pix)$$

3. Size of each subsection at 1.5 m away from the camera in x and y directions in meters:

$$size_x(m) = \frac{\tan\left(\frac{\theta_x}{2}\right).1.5(m).2}{15(\# \; of \; subsections)}$$

$$size_y(m) = \frac{\tan\left(\frac{\theta_y}{2}\right).1.5(m).2}{15(\# \; of \; subsections)}$$

4. Finding the appropriate camera installation angle

Kinect vertical view angle = 43°

$$\emptyset^{'} = 43^{\circ} - \emptyset$$

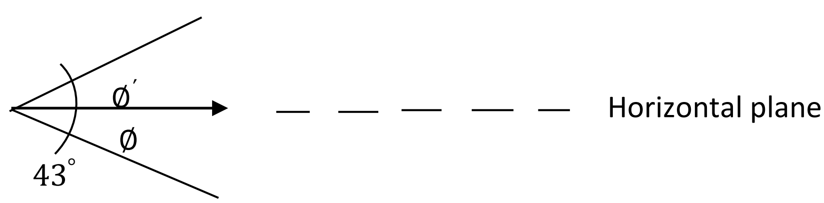
Distance of the object from the person is calculated by:

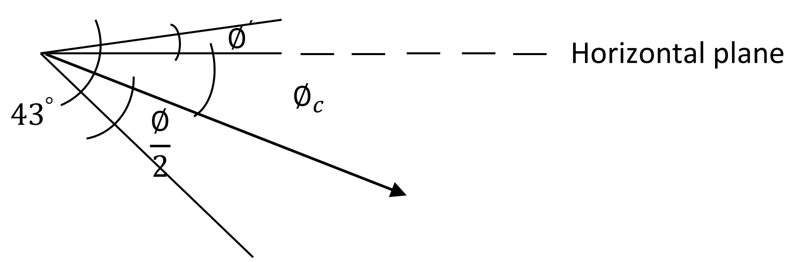
$$x \; (m) = \frac{\tan(\emptyset)}{1.75^{*} \; (m)}$$

*Here average height of a normal person is taken to be 1.75 m.

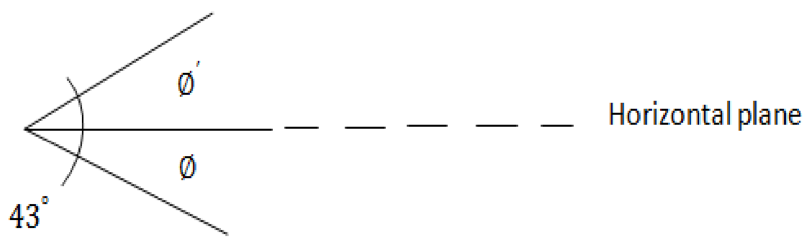
The angle of the camera:

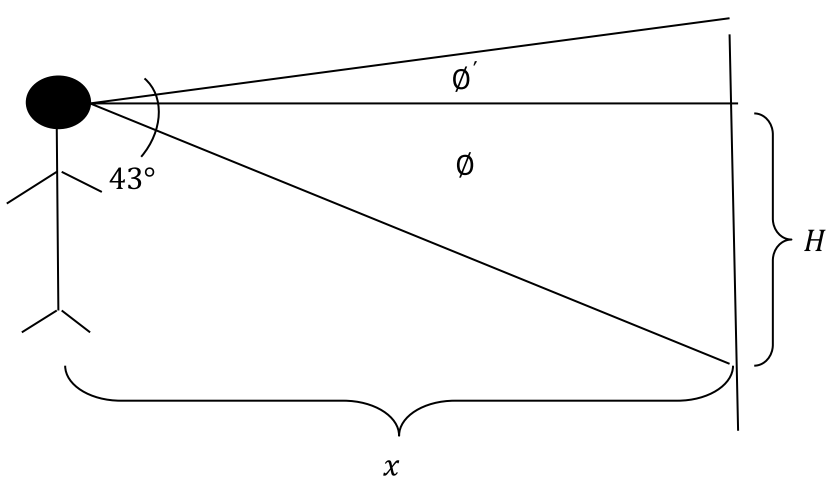
$$\emptyset_c = \frac{43^{\circ}}{2} - \emptyset^{'}$$





H(m)\x(m)	1.5	2	2.5	3
1.65	$\emptyset = 47.72^\circ$	$\emptyset = 39.52^\circ$, $\emptyset' = 3.48^\circ$ $\emptyset_c = 18.2^\circ$	$\emptyset = 33.66^\circ$, $\emptyset' = 9.34^\circ$ $\emptyset_c = 12.16^\circ$	$\emptyset = 28.81^\circ$, $\emptyset' = 14.19^\circ$ $\emptyset_c = 7.31^\circ$
1.75	$\emptyset = 49.23^\circ$	$\emptyset = 41.18^\circ$, $\emptyset' = 1.82^\circ$ $\emptyset_c = 19.68^\circ$	$\emptyset = 34.99^\circ$, $\emptyset' = 8.01^\circ$ $\emptyset_c = 13.49^\circ$	$\emptyset = 30.25^\circ$, $\emptyset' = 12.75^\circ$ $\emptyset_c = 8.75^\circ$
1.85	$\emptyset = 50.96^\circ$	$\emptyset = 42.76^\circ$, $\emptyset' = 0.24^\circ$ $\emptyset_c = 21.26^\circ$	$\emptyset = 36.50^\circ$, $\emptyset' = 6.5^\circ$ $\emptyset_c = 15^\circ$	$\emptyset = 31.66^\circ$, $\emptyset' = 11.34^\circ$ $\emptyset_c = 10.16^\circ$
1.95	$\emptyset = 52.43^\circ$	$\emptyset = 44.27^\circ$	$\emptyset = 37.95^\circ$, $\emptyset' = 5.05^\circ$ $\emptyset_c = 16.45^\circ$	$\emptyset = 33.02^\circ$, $\emptyset' = 9.98^\circ$ $\emptyset_c = 11.52^\circ$





5. Information provided by the accelerometer is in the form of : $(g_x\ ,\ g_y\ ,\ g_z)$

However we do not need to process information in z direction. To obtain information about the angel of the camera in each plane we can use the following relation:

$$g_x = g\,.\sin(\theta_x)$$
$$g_y = g\,.\cos(\theta_y)$$

From that we can deduct:

$$\theta_x = \sin^{-1}(\frac{g_x}{g})$$

And

$$\theta_y = \cos^{-1}(\frac{g_y}{g})$$