

Moving Human Electromagnetic Scattering Simulator

ECE480 Team 2
Proposal

Air Force Research Laboratory

Sensors Directorate WPAFB, OH

Dr. Analee Miranda

Mr. Kenneth Schafer

October 19, 2012

Mark Birdsall

Will Juszczuk

Ryan Lattrel

Michael Lazar

Camden Smith

Facilitator:

Dr. Wen Li

Executive Summary: The objective of our design is to use an imaging device to collect and process data on humans detected by the system. The imaging data will be processed by a microcomputer connected to the device. The microcomputer will then forward this data over the network using Ethernet or Wi-Fi and will be further processed by a PC application in order to display real time radar results.

Table of Contents

- I. Introduction*
- II. Background*
- III. Design Specifications*
- IV. Conceptual Designs Considered*
- V. Proposed Design Solution*
- VI. Risk Analysis*
- VII. Project Management Plan*
- VIII. Budget*
- IX. References*

I. Introduction

A system is requested by the Air Force Research Laboratory (AFRL) with the purpose of quickly and automatically capturing the movement of people, along with the sizes and descriptions of those detected. The idea is to do this with a portable radar system. This system will be useful in a variety of scenarios, including military reconnaissance and search and rescue efforts. It is important to be able to discover individuals in environments where it is difficult for the naked eye to properly detect and identify targets. The main purpose of this particular project is to work on developing a way to automatically determine when a human is detected by the radar, and to determine the specific characteristics of the subject's body when he/she is detected.

The concept of the device to be designed includes something that is portable, small, lightweight and interfaces with a PC in order to provide a complete "Moving Human Electromagnetic Scattering Simulator". The result of this project will include an enclosure with a variety of components capable of directly processing radar signals detected by an imaging device.

The requirements for the project are to use any type of microcontroller or microcomputer and to interface it with any motion sensing input device. Several different options are available for the EM scattering software, which will be run by a PC connected to the microcomputer over a network interface. This project is to be used as a prototype and the main goal is to design a portable system.

II. Background

This project involves merging realistic human motion parameters measured by an input sensor with electromagnetic scattering simulation software. We have the use of EM scattering simulation software given to us by the Air Force that will output a Doppler radar real-time image of a moving human and we can utilize one of the human motion sensing input devices used in the gaming industry. Any software that models human movement will require human size parameters (i.e. arm length, leg length, total height) and typically if you want to model motions like running or walking you would need to physically measure and input those measurements into your software. Manually taking measurements of the whole population would be extremely time consuming so we will design an algorithm that will detect a human in the motion sensor data and assign the parameters that the EM scattering simulation software will need.

The input device that we will be using is Microsoft's Xbox Kinect. The Kinect is a webcam style add-on to the Xbox 360 gaming console that allows users to interact with the console through the use of spoken commands and gestures. The creators of the device did not protect the USB port purposely, which allows the Kinect to be modified and used in any industry with engineers that can design software to use it as an interface to a larger project, including we who need to use it as an automated device for capturing human movement parameters which we can use as an input for the EM scattering software so we can display human movement, real-time, on a computer screen.

Through research it had been made clear that it is possible to process the signal of an Xbox Kinect with a microcomputer.

III. Design Specifications

The goal of this project is to develop a motion sensing device that will interface with AFRL software to render an end-to-end moving human electromagnetic scattering simulator. This device is meant to be:

- Small, portable, and lightweight
- Connect to a PC via Ethernet, or 802.11
- Take advantage of the Microsoft Kinect's open USB port and use an open source driver to access it
- Accurately identify humans based off human body characteristics
- Develop an enclosure for our device that encloses the Kinect and Pandaboard ES
- Use a battery or power supply to power the device

Several different software options have been given by our sponsor. One is using Remcom XFDT, which is an advanced imaging software. The other option is more simple, requiring only a few basic parameters for the model. Based on our project outline, our own software may need to implement the following:

- An algorithm to initialize and output the AFRL EM software
- An algorithm to convert the EM software data into the time domain using an FFT transform
- Utilize a back-projection-based imaging algorithm to image the data in real time

IV. Conceptual Designs Considered

There were several different designs considered while engineering a solution to this project.

The concepts were not all adequate to for our needs, as can be seen in our selection matrix in Figure 1.

Solution Selection Matrix

Criteria	Importance	Design Concepts		
		Arduino	BeagleBoard	PandaBoard
Portable	5	5	4	4
Ethernet	5	3	5	5
USB	5	3	5	5
Runs Linux	4	0	5	5
		55	90	90

Figure 1

The first concept that was considered was using an Arduino. An Arduino is a microcontroller that is well known and extremely simple to program. There are different shields that are readily available which can be attached to an Arduino to add functionality. The shields this project required were both an Ethernet shield, and one that has a Wi-Fi module. Also required would be a USB host shield to attach the Kinect. The main reason this design is not feasible is that an Arduino does not have the required processing power, and cannot utilize the open source drivers required to process Xbox Kinect data.

The second design concept was to use a BeagleBoard. The BeagleBoard would have fulfilled the minimum requirements for the project. The board can run Linux, so open source drivers can be used, and it also has an Ethernet connection. The reason it was not selected was due to the PandaBoard being a better alternative, and the PandaBoard has Wi-Fi included, which was not available for the BeagleBoard. The PandaBoard also has a faster processor and

is not prohibitively more expensive. The reasons for the PandaBoard can also be seen in Figure 2.

Feasible Design Matrix

Criteria	Importance	Design Concepts	
		BeagleBoard	PandaBoard
Portable	5	4	4
Lightweight	4	4	4
Wifi	2	0	5
Ethernet	5	5	5
USB	5	5	5
Runs Linux	4	5	5
Powerful	4	3	5
Power usage	1	5	3
Cost Effective	3	4	3
		135	148

Figure 2

V. Proposed Design Solution

The overall design includes a microcomputer running a linux kernel, with custom written software running on top of a Linux kernel to read and process data from the X-box Kinect. The custom software will Kinect to a PC via Ethernet or 802.11 and provide data to a PC program which will then process and plot the given data in real time using both an EM program provided by the AFRL as well as another custom written program that utilizes the provided software.

The design consists of three different fundamental pieces:

1. The hardware configuration: A Kinect, a microcomputer, and a power source all enclosed in a lightweight box. A single power supply will be used to power both the Kinect at 12V and the microcomputer at 5V. A diagram is shown in figure 3.

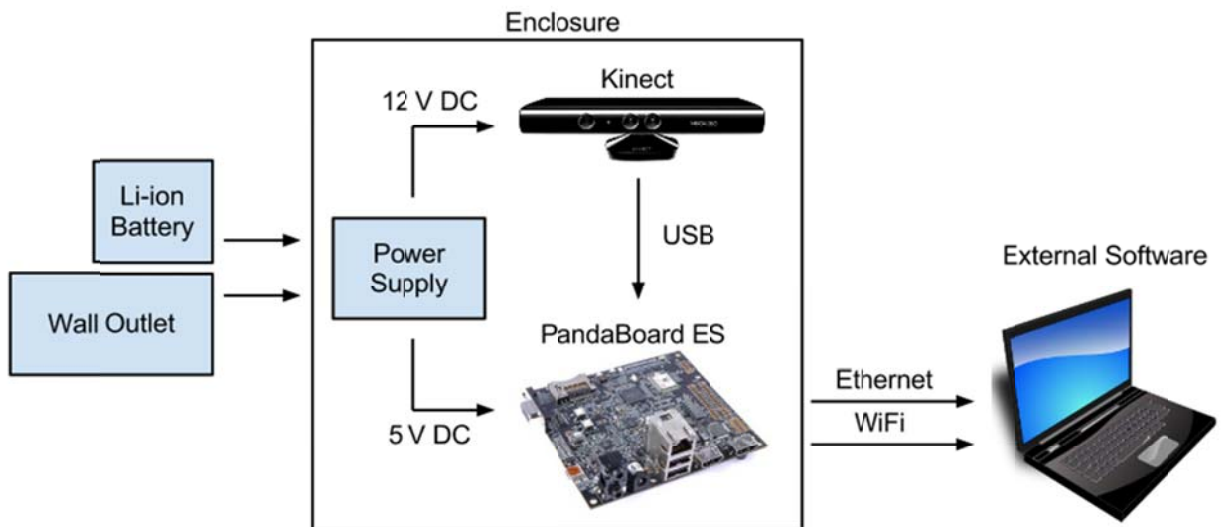


Figure 3

2. The software running on the microcomputer, which includes custom written imaging processing software running on top of a linux operating system. This will be coded in C++. The

software will allow any number of PCs to connect and receive data from the connected Xbox Kinect. The software will also utilize open source frameworks for the Kinect. The main framework we will be using is OpenNI (Open Natural Interaction).

A basic overview of our program can be seen in Figure 4.

Microcomputer Software

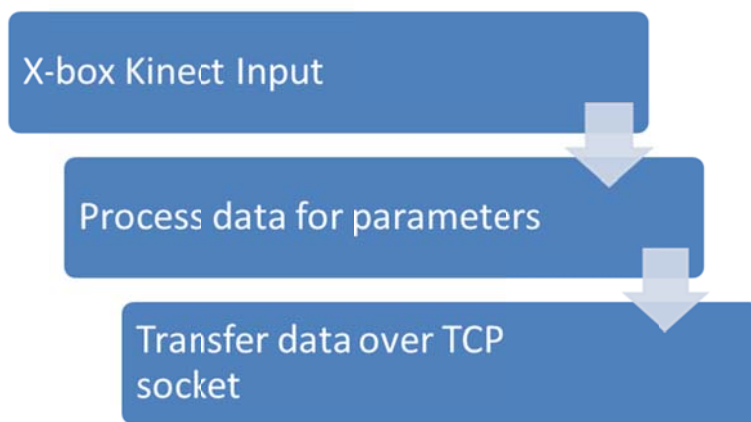


Figure 4

3. PC software that connects to the microcomputer via wifi or ethernet, and processes the incoming data from the microcomputer in order to plot real time radar information. This will be coded in either C# or C++.

Figure 5 shows the flow of our PC software program.

PC Software

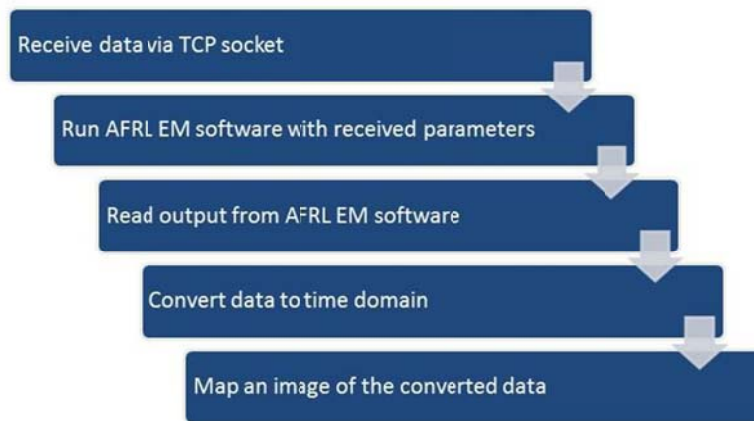


Figure 5

VI. Risk Analysis

The risks in our design are minimal. Due to prior research it is clear that a microcomputer will be able to both run the Linux kernel as well as read data from an X-box Kinect over USB. It is also clear that this device will be able to connect to a PC over the requested Ethernet or 802.11 protocols. The rate at which the data can be processed is yet to be determined, although the microcomputer selected will include multiple cores to allow parallel processing of the incoming image data.

Risks involved include the potential cost of our product being very near our budget, and therefore any missteps could potentially put us over our limit. The microcomputer itself is between \$150 and \$200 and so it would be costly to replace if anything were to happen to it. The risk to the computer will come from the power supply we will be working with, as the computer only takes 5V, though we will be using a modifying a 12V supply.

VII. Project Management Plan

The team has a variety of skills that can be utilized to complete this project. Some team members come to the team with a solid understanding of software while others have a solid understanding of hardware, but all team members have demonstrated an ability to learn new skills and apply them as needed. We will divide the large tasks amongst sub teams. The sub teams will then break down their section and finish it together. The majority of the work in this project is software related; therefore several teams will be created to focus on specific problems. As stated in the concept there are three main components of the project, though each component is somewhat dependent upon another.

Team tasks:

A software team will look at connecting our device (Kinect) to a USB port using an open-source driver. They will also be responsible for parsing the incoming data from the device (Kinect) and assigning the parameters such as, arm and leg length to a particular human as preparation for sending the data to EM software.

A second software team will be needed in order to develop the application to run on the PC, which will receive the data via a socket connected by Ethernet or Wi-Fi to the microcomputer. The team will be responsible for launching the EM software and parsing and organizing the output of that program. This team may or may not also be responsible for performing an FFT on the output data, and then image it using a back-projection-based algorithm.

There will need to be a team or individual responsible for generating the power required to run the Xbox Kinect and the microcomputer simultaneously off of a single power supply.

Another task is to create our enclosure, and to figure out how much time and the amount of resources that will go into creating it on time.

The timeline for our project is shown in figure 6.

Resources needed:

Our team won't need many resources that are not provided for us by the ECE 480 lab. We all have access to computers and we will use a Microsoft Kinect as our device for capturing human inputs. We will use a Pandaboard ES microcontroller to process our data and use either ethernet or WIFI to send this data to a PC running EM software.

We will likely need to utilize the EGR machine shop to create a custom enclosure if we can not find a cheap one available for purchase that suits our needs.

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		Proposal	4 days			
2		Order Items	10 days			
3		Get Pandaboard set up with Linux	5 days	Mon 10/1/12	Fri 10/5/12	
4		Start working with Kinect and Pandaboard	5 days	Mon 10/8/12	Fri 10/12/12	
5		Testing and catch-up week	5 days	Mon 10/15/12	Fri 10/19/12	
6		We should have something presentable, maybe a bit buggy	0 days	Fri 10/26/12	Fri 10/26/12	
7		Testing (project will hopefully start working)	10 days	Mon 10/22/12	Fri 11/2/12	
8		Get a power supply working	5 days	Mon 10/29/12	Fri 11/2/12	
9		Design an enclosure	5 days	Mon 11/5/12	Fri 11/9/12	
10		Late round testing (hopefully it mostly works by now)	7 days	Mon 11/12/12	Tue 11/20/12	
11		Get the enclosure done	5 days	Mon 11/26/12	Fri 11/30/12	
12		Final Testing	5 days	Mon 12/3/12	Fri 12/7/12	

Figure 6

VIII. Budget

The project budget as specified by the ECE department is \$500 USD. All items will be ordered through the MSU ECE shop from reputable dealers.

Microsoft Kinect for Xbox 360 - *\$110*

Package includes a Kinect Sensor and a 12 V power supply cable.

PandaBoard ES - *\$162*

This development board is an essential part of the project. It acts both as a processor to parse the Kinect's output and as an interface between the Kinect and an external PC.

16 GB SD Card - *\$20*

An SD card is required to use a Linux Operating System on the PandaBoard ES.

Power Supply (Undecided) - *\$40*

The power supply must provide clean, regulated power simultaneously to both one 12 V and one 5 V input in some fashion. Depending on availability and time constraints, it may be custom built in the lab, ordered pre-built, or a combination of the two.

Enclosure - *\$0*

The enclosure will be designed and built in the MSU ECE shop free of charge.

Li-ion Battery (Undecided) - *\$50 - \$100*

The battery provides a means to turn the enclosure into a fully portable solution. It will only be added if it is determined feasible and/or necessary to run the system off of battery power. It may also be downgraded or cut out completely if our budget is anticipated to go over.

Other - *\$0 - \$100*

Other costs may include additional memory, cables, shipping costs, components, and software

IX. References

<http://pandaboard.org/>

<http://openni.org>

<http://beagleboard.org/>

<http://www.arduino.cc/>