February 22, 2006

Dr Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia

Re: ENSC 340 Post-Mortem report

Dear Dr. Rawicz,

The following document presents the post-mortem report for the Voiceture System as part of out project for ENSC 340. Our goal was to create a device that will translate sign language into speech using a commercially available glove to recognize hand gestures. The system demonstrated the ability to translate the American Sign Language alphabet using our glove concept device, sensor, software algorithms, and user interface designs.

This document details the current state of the project, a comparison against original functional specifications, and future work recommendations. Also summarized is our project scheduling and cost compared to previous projections. In conclusion, we describe our personal experience and what we learned while working together on this project. This document completes the requirements for ENSC 305 and 340 for the fall 2005 semester

Fivepoint Technologies is comprised of four motivated Engineering students from Simon Fraser University. These are Ganesh Swami (President and CEO), David Brayden (Vice President, Research and Development), Phoenix Yuan (Chief Operating Officer), and Kjell Eggen (Chief Technical Officer). Questions and concerns can be directed to me at 604-992-1404 or by e-mail at ensc340-Fivepoint@sfu.ca.

Sincerely,

Ganesh Swami

Ganesh Swami
President and CEO
Fivepoint Technologies
Vancouver, BC

Enclosure: Post-Mortem report for the Voiceture System

FIVEPOINT technologies

Post-Mortem Report for the Voiceture system

Translation of American Sign Language into Speech

February 27, 2006

Project Team:

David Brayden
Kjell Eggen
Phoenix Yuan
Ganesh Swami

Submitted to:

Andrew Rawicz
Steve Whitmore
Mike Sjoerdsma
Brad Oldham

Simon Fraser University
School of Engineering Science
ENSC 305/340

Table of Contents

List of Figures

List of Tables

Introduction

The Voiceture is a motion capture system to translate American Sign Language (ASL) signs and gestures into readable English on a computer screen. The device consists of two components, the motion capture device and the gesture capturing and translation software.

Communicating with English speaking people has never been easy for a deaf or mute person. To date, no economical and robust system exists outside of the traditional translator. The Voiceture device aims to bridge that gap, proving to be a milestone towards independent real-time communication for the deaf and hard of hearing.

The development done between September 2005 and February 2006 resulted in the creation of the first working prototype that translated the ASL alphabet. The prototype was presented at a project demonstration on February 6 to showcase the core technology and determine its feasibility.

This report examines the process that took place leading up to the project demonstration and concludes with the group members' individual experiences.

Current State of the Device

Overall System

As described in the project proposal, the Voiceture system detects a user's hand orientation and matches the sampled data with the corresponding trained ASL letter. The data is first captured through a P5 Glove with custom sensor add-ons, and sequentially processed by the ASL detection software engine. The final result is output to a host computer through a GUI interface. A system overview block diagram is given in the Figure below.

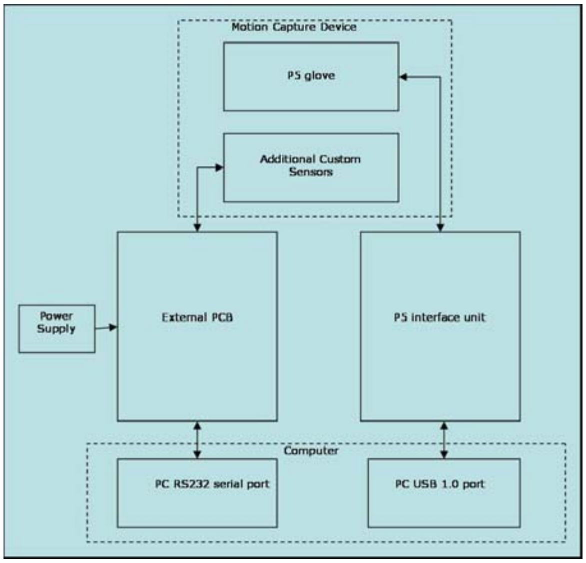

Figure 1: Voiceture prototype

P5 Glove

The P5 glove provided a great opportunity to begin the design of our system. It translated five finger measurements independently through USB to the computer system. The glove is lightweight and comfortable when worn for long periods of time. Also, it is very non-invasive during signing and daily activities.

Printed Circuit Board – External Sensors

Data from the add-on sensors are interfaced into the PC through a custom-designed board. The board (see figure 2) debounces input sensor signals to minimize performance degradation from external noise. Then, a PIC micro-controller uses real-time interrupts to periodically check its input port for sensor triggers. Upon detection of user input, the micro-controller encodes the sensor value and updates its output routine. The output port sends out a serial data stream to a MAX232 chip, which conditions the input/output voltage as specified in the RS232 interface. The host PC picks up the data through a standard serial cable. Using a standard USB-serial adaptor, the data stream can also be captured through a USB port. The board is powered by a standard 9V power adaptor. Once completed, the board has been very reliable and consistent in its operation.

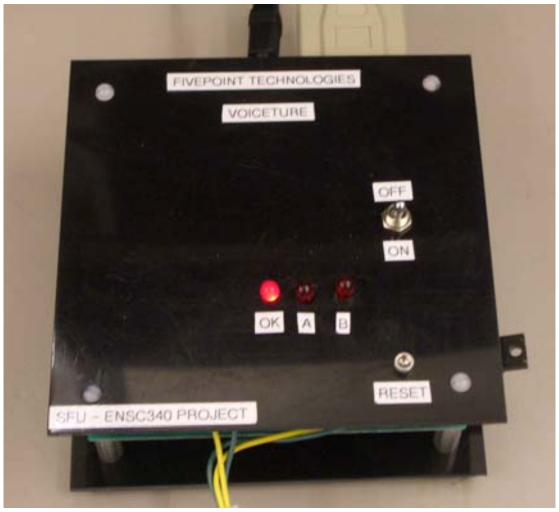

Figure 2: Custom Designed Board

External Sensors

The extra sensors added to the glove worked very well as they were effectively contact sensors. They greatly helped translation, adding significant accuracy to many letters. The metallic contacts were hand made from nickel and exposed wire to provide a discreet look and effective operation. Several different materials and sensor designs were considered, but the simplicity of hand made contacts was chosen.

Data Analysis Engine

The system was able to adapt to a users specific gestures, and avoided most intermediate letters while signing. Also, it was capable of translating at least two well signed letters a second. However, the system was not able to understand the entire ASL alphabet, and incorrect letters were frequently transmitted. As described later, this is mostly attributed to hardware issues.

Profile Creation & Training

User profiles were implemented as an integrated Java class with the necessary functions to create, save, manage, open, and list profiles. The individually profiles were managed using files that stored all the user gesture information. This meant that the main engine would only require the user profile as an argument to begin running.

The training program was another Java class that simply existed to initialize the hardware and run through the alphabet training procedure while doing the appropriate data analysis for a new user. The advantage of creating this class is that the main program can ignore hardware implementations and specifics regarding training and user profile creation.

The decision to keep these two classes detached was made to separate all file I/O from the Voiecture electronics I/O system. In the future, this would allow software threading to be implemented.

Hardware Interface

The Voiceture hardware I/O system was separated into two Java classes, one for the PCB connected via serial and another for the P5 glove connected via the USB port. Here Java had the distinct advantage of incorporating packages to handle serial and USB interfaces.

The external PCB I/O interface was created as a serial port class to handle initialization, handshaking, and sampling of the serial port using buffers. This class was slightly tricky to program since Java no longer officially supports serial port programming due to security and legacy issues. However, a work around was developed that required a fake handshake in order to “trick” java to initialize the ports.

The P5 glove I/O was relatively simple once we learned the open-sourced driver supplied with the glove. This driver had all the necessary function classes required to retrieve data. With this, only the sample routines needed to be implemented

We incorporated these two interfaces by creating a driver wrapper that simply initialized both devices, read the latest data from both devices, and determined the state of the sensors, which was output.

Graphical User Interface

In theory, Java should be development environment agnostic. We first began developing the Java classes in Eclipse. But then Netbeans came out with an excellent “point and click” GUI builder. Unfortunately, building the UI with Netbeans made the entire software development Netbeans dependent.

The interface can currently add, remove and modify profiles. It can also train new users – which currently consists of going through the twenty six letters. All of this was built on the classes described previous – maintaining good software reuse. The main interface shows letters as they are detected. It also has the ability to turn the glove off.

Our final goal is to completely get rid of the user interface, so this worked extremely well for the effort that was put into it.

Functional Deviations

Overall System Function

The Voiceture System met some functional requirements and missed others. The system was a good start to the problem of sign language translation. However, mostly due to the imperfect P5 gloves, the overall system did not function as well as it could have. The system met three of four linguistic requirements, translating user specific letters quickly and without intermediates, but it was not able to handle the entire ASL alphabet.

PCB Electronics

In the functional specification, we specified an interrupt based solution for serial data acquisition. The idea behind is that the host computer will only be notified about the change of input state. Later on, upon a group decision, the actual implementation was changed such that the host computer will continuously poll an incoming serial data stream. The reason for this change is mainly to do with the way the sensor data is acquired using Java’s serial port communication protocol. The change affected the usage of the timer sub-system of the PIC.

P5 Glove

The P5 glove was a fast and cheap glove for this project. Unfortunately it didn’t meet some of the manufacturer’s specifications. It was originally intended for computer gaming, where the exact bend of a finger was not as important as a simple open or closed.

The requirement *“The sensors for finger positions must be accurate to 0.5 degrees of finger bend.”* was not met. Towards the end of the project the original glove had lost approximately two thirds of its resolution, as the individual bend sensors were no longer giving voltage signals across the full motion spectrum.

We have used three separate gloves during this project, each of which has worn down a significant amount. Although some of this breakdown could be fixed through driver recalibration of the glove, it greatly impaired translation accuracy.

The glove also was not *"adjustable to fit 25th to 75th percentile size hands"*. Removing and putting the glove back on caused a large variation in data. The glove also moved around during the signing process and had to be constantly adjusted. This was due to unadjustable finger rings that held the glove on the hand.

Data Analysis Engine

The data recognition algorithm required constant modification during this project. It was a very frustrating task attempting to correct imperfections with the hardware from within the software algorithm. The algorithm had to be implemented using many if…then statements that checked for errors depending on the detected letter.

Overall, our algorithm worked well for reasonably accurate incoming data. However, when letters become closer together the algorithm performance degrades. This is a problem with any pattern recognition software.

User Profile Creation, Management, and Training

We met all of the requirements in the specifications. The user can add, delete and modify user profiles. Adding profiles automatically takes the user to the training screen. The training currently consists of running through the 26 ASL alphabets one after the other.

Audio outputs and word recognition

Our original plan was to spell out each letter as it was recognized. We did find free software packages to do this, but during the development stage, our algorithm hit too many false letters making the audio very annoying. This is something we'll definitely look at for future versions. Our other plan was to suggest to the user possible completions of the word based on context, similar to the way cell phones prompt text. We didn't have time to implement this feature.

Future work & Recommendations

Overall System Function

Our eventual goal is to make the device completely portable. Since the engine was written in Java, it could be moved to any machine supporting a virtual machine. The user interface has to be reworked to fit the small screen – the details are dependent on the type of device we decide to use.

PCB Electronics

The serial communication can be improved to incorporate two-way communication with a host PC. Doing so allows the possibility of attaching actuators to the system and permits greater utilization of the PIC micro-controller. However, future work will get rid of the external PCB altogether.

P5 Glove

The P5 Reality Glove was a good teaching tool with this project. However, it was not accurate enough for our measurements, and did not meet our specifications (or the manufacturer's specifications). Future designs of our system will work towards making our own glove, or signing an agreement with a hardware manufacturer to do so.

Packaging and Custom IC

The external PCB board is impractical from a usability perspective. Therefore, the next iteration will need to integrate all the sensors into a single package. This can be done in two stages. The existing programmable PCI card within the glove can be configured to handle additional inputs and would require significant reprogramming and some rewiring. However, this would only be a short term solution. A long term solution would be to create a micro-controller based PCB that fits within the actually glove that wouldn't require a PCI interface and therefore be much smaller. This would be a significant amount of work, but a more practical solution.

Data Analysis Engine

As this project relies on very sophisticated software, there is lots of work to be done in this department. However, the amount depends heavily on the updated glove and hardware. The future gesture recognition algorithm will likely be quite different from what it is today as further modifications are done to the glove. Also, as more gestures are added, the complexity will increase enormously. An entire dictionary would need to be programmed!

User Profile Creation, Management, and Training

If the device were to be made portable, one recommendation is to use an actual computer for the training stage. This way, the environment is not constrained and allows the use of rich media to enhance the user's experience.

Audio outputs and word recognition

We will incorporate certain natural language techniques (like text completion) to make the product more enjoyable to use. The product can also speak out the most recently recognized letter/word.

Portability

Our decision to use Java worked well, since platform independence is quite important for the project. Our code base can be easily port to another operating system such as Linux, MacOS, or PalmOS. The only constraint on the selection of portable device is that the device must contain USB or other input interfaces.

Java also saves tremendous design cost by using tools from free and Open Source Software, especially IDEs, to develop our system. Due to the popularity of Java, we can afford to choose the best tools among a large pool of free programs. Most modern portable devices are natively developed with Java compatibility.

Budget and Time Constraints

Budget

Table 1 displays our estimated cost and the actual cost for the project between September 2005 and February 2006. The “other” costs associated with the project were due to presentation material for the Western Engineering Competition 2006 in Calgary.

Item	Proposed Cost	Actual Cost
P5 Gloves	$140 (2 gloves)	$108 (3 gloves)
Electronics components	$40	$52
Textbooks/Journals	$40	$5
Stationary	$10	$3
Web space	$20	$0
Other (contingency)	$75	$18
Total	**$325**	**$186**

Table 1: Cost Analysis

The table shows that our actual cost was a little over half of our proposed cost. This can be mostly attributed to the fact that we obtained the P5 gloves at a lower cost than expected ($35 vs. $70 each). We were also able to borrow various items such as library material, cables, electronics, and PCB packaging. Also, the micro-controllers used in the electronics were obtained as samples for free. Overall, after the $140 endowment funding each group member contributed about $10.

Schedule

Table 2 presents a Gantt chart of our actual project process timeline along with the proposed time in hash marks.

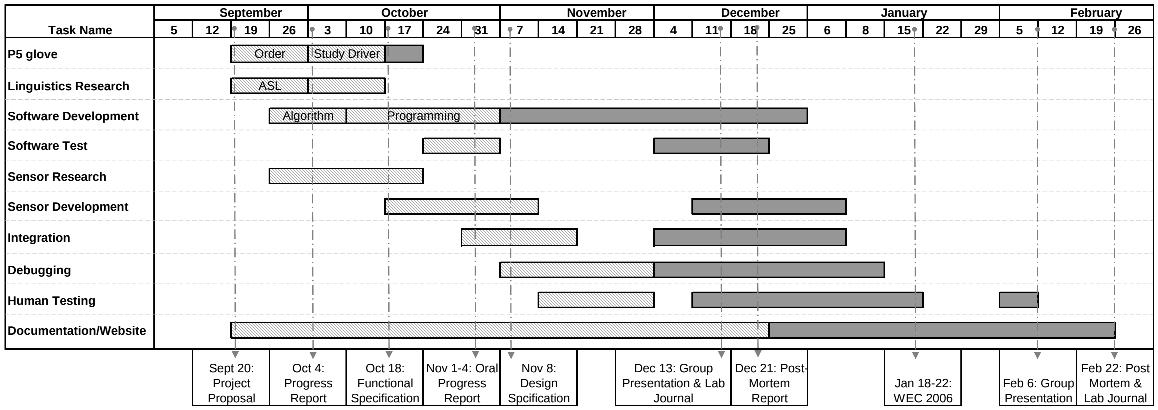

Table 2: Project Schedule

It is evident that we were far past deadlines in many situations. This was a decision the group made in early December to go to WEC 2006 and therefore extend our project past the original 4 month timeframe. From an engineering standpoint, the micro-controller was difficult to program and caused a delay in having a completed PCB together by about a month. The P5 glove bend sensors problems caused our programming timeline to extend beyond our initial expectations.

These problems translated in a working prototype that wasn't ready for testing until late December. By this date the team members were into final exams and it was decided to postpone our project demo until after WEC 2006. This relieved some project pressure and allowed the team to receive feedback about our project and presentation at WEC before the ENSC340 demo.

Independent Experience

David Brayden

My experience with this project was overall positive. I enjoyed our team and thought we worked well together. I learned the two key components of a successful partnership are patience and understanding. While we all might not agree on a particular solution, it was extremely important to take the time to listen to everyone's perspective and evaluate the benefits of each.

Along with discussing the merits of each others ideas goes finding and criticizing the flaws. At first, we were too quick to pick on a negative part of an idea, and shoot it down quickly. Eventually, we learned to evaluate the pros and cons of each idea, and brainstorm together to improve it as much as possible.

I learned an enormous amount about data analysis algorithms. Working with an imperfect data gathering system was very frustrating. Most of my time was spent trying to solve hardware issues and imperfections from within the software. The time required to gather and analyze data was enormous … especially when the data was completely different from the last set you gathered!

As engineers, we sometimes get trapped into thinking things must be done a certain way, and that no alternative exists. During the course of this project I learned to appreciate new and novel problem solving techniques, and to think more outside of the box.

This project greatly increased my understanding of the design cycle. Countless modifications were necessary to get the system in working condition. Unlike a regular school project, designed to work accurately when assembled correctly, certain parts of this project could not even be started until others were finished and working properly. This imposes tight deadlines on the early stages of the project, which we thankfully were able to meet.

We had meetings as needed throughout the semester. At first we tried to schedule these when everyone had time off. But towards the end of the project we realized that not everyone needed to be present all the time. As a result, more was accomplished, and there was less waiting around during meetings.

Kjell Eggen

This project proved to be both fun and challenging as our group set an ambitious goal to engineer a novel device. From the start I thought that our group had a great balance of experience, technical expertise, and creativity to design something new and exciting. Everyone fit into their perspective roles nicely, focusing where their strengths and interests lay. I thought we worked well as a group and enjoyed ourselves even when decision making processes became a little heated.

Everyone worked so well in their respective roles that sometimes I found that the group seemed disjointed due to lack of communication regarding each others work. However, this is expected when every group member has their own priorities and schedules to work with. This, at times, made it hard to know where to start sections of the project. This individualizing of parts also hindered some aspects, such as electronic debugging. It was difficult to help when problems arose that could have been solved faster with more man power.

The project didn't turn out as well as I hoped or go as quickly. But had it turned out as planned, our goals would have been too small. The issues that evolved due to the P5 glove bend sensors wearing down proved to be a make or break situation for this project. A lot of accuracy and stability was lost due to this issue. During the design stages, more preliminary testing with the glove should have been done to ensure the glove worked to the specifications. By the time the issue surfaced, the project was too far developed to have time for a complete design change. Another point learned.

I learned a lot about group dynamics, project planning, project development, and presenting. It was vital that our group worked well together and we're able to voice issues and have composure to work through the issues. Project planning proved to be slightly tricky as we attempted to define an area that none of were directly involved in and were trying to quickly learn. My major task was software programming and the project documentation, so I gained the most experience in those areas. I also opted to be the 'integrator' for the project parts, which caused a few very late nights to meet deadlines. Technically, I learned about electronic and computer system interfaces and the project process.

Overall, I enjoyed working on this project and with my three talented group members. But, I'm glad it's done. Now for some sleep.

Ganesh Swami

"If you aren't trying to solve hard problems, you are just wasting your time."

First off, I would like to congratulate and thank my teammates Dave, Kjell and Phoenix for a job well done. I've never had so much fun learning.

From a technical standpoint, we were approaching a completely new field outside of our expertise. There were too many unknowns and a steep learning curve, but we still stuck together. Machine learning has always intrigued me (and still does), and this project gave us the opportunity to familiarize ourselves with the field. We took some initial guidance from Dr. Anoop Sarkar, who is an expert in the field of natural language processing.

From a group dynamics standpoint, I think we worked quite well. Each member volunteered to do what he felt most capable of. No micromanagement was needed. I was surprised at how easy it was to integrate the various pieces. This is probably because of our use of proper engineering practices. The most enlightening moment for me was in Calgary when our computer died about an hour or so before the demo. Crisis management at its finest – we hacked our way through the software to make the hardware work. It probably gave me a few strands of gray hair, but these are stories I can tell my kids …

I volunteered to take lead of the business and marketing side of things. I got to learn a lot about how startup companies bootstrap themselves. Words like "market segmentation" are now part of my everyday vocabulary. Something I realized early into the project was that there are people around you who are infinitely better than you at certain things. We sought advice on the literature for machine learning, the business plan, the design of posters. Even for something as simple as interface design, I asked a friend to demo the product – that way I learnt to use the tool much quicker than I would have otherwise. Of course, Dr. Rawicz, Mike and Brad were always around to help us out.

I'm a little disappointed that the glove didn't work out as well as it was advertised – maybe that's why the P5 glove company has gone out of business.

The message to take home is that ideas are dime a dozen – it's the team and execution of the plan that is key.

Phoenix Yuan

One of my main goals for this project is to acquire greater knowledge in the firmware design area. Fortunately enough, my task involved dealing with the widely acclaimed

PIC micro-controller. I really enjoyed assembling a custom-design circuit board piece by piece and figuring out all the problems in the process. I learned that even the most trivial "non-issues" can produce the most bizarre, pseudo random outcome. The experience greatly increased my appreciation for the hardware design profession.

As dictated in the engineering course legends, workload from 340 can be quite a burden. I was severely burnt out at one point due to the sheer number of tasks that were still at large. Thankfully, a team member was grateful enough lighten my load by taking over part of the software/hardware interface design. This move helped me to focus on the hardware issues that were more imminent.

Being the only one working with the hardware electronic design, I learned that the implementation could have been a lot smoother if someone was able to look over my shoulder and check my work. In most cases, it is easier to describe the solution of a bizarre problem than the problem itself. Thus, a workload organization with greater amount of design overlaps could have saved time for everyone.

One area of improvement that I think could benefit the project is to have all the component interfaces clearly defined in a design document before the actual design implementation. We wasted quite a bit of time bring each other up to speed on the latest integration scheme.

Project Conclusion

Overall, the group worked well together and enjoyed this challenging project. Even though problems arose, we are satisfied with the performance of the project. In the future we hope that this technology will be implemented further and eventually make it into the lives of those who will benefit from this translator. We wish to thank Andrew Rawicz, Mike Sjoerdsma, and Brad Oldham for their guidance during the course of this project.