School of Engineering Science
Simon Fraser University
Burnaby, BC V5A 1S6
vpw1@sfu.ca

November 26, 2012

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

Re: ENSC 440 Post-Mortem Report for a Motion-Controlled Manipulator

Dear Dr. Rawicz,

Attached to this letter is the post-mortem report, which outlines our team's design and implementation difficulties when building our system. Our project is to design an intuitive method of controlling robotic arm manipulators for various applications.

This document details the current state of our system, deviations from the original plan, and any future plans for the system. In addition, we outline some of the budgetary and time constraints we encountered and describe the inter-personal and technical experience from working in a project together.

MotiCon is composed of six enthusiastic and creative engineering students: Kevin Wong, Kay Sze, Jing Xu, Hsiu-Yang Tseng, Arnaud Martin and Vincent Wong. If you have any question, please feel free to contact us by phone at 778-889-9950 or by email at vpw1@sfu.ca.

Sincerely yours,

Vincent Wong
President and CEO
MotiCon

Enclosure: ENSC 305/440 **MotiCon** Post-Mortem Report

Motion-Controlled System For Multi-Purpose Robotic Arm Operations

Post-mortem Report

Project Team: Vincent Wong
Jing Xu
Kay Sze
Kevin Wong
Hsiu-Yang Tseng
Arnaud Martin

Contact Person: Vincent Wong
vpw1@sfu.ca

Submitted to: Dr. Andrew Rawicz – ENSC 440
Steve Whitmore – ENSC 305
School of Engineering Science

Table of Contents

List of Figures

List of Tables

Glossary

Table 1: Glossary: Definition of Terms

Black Box	Opaque enclosed system for the user, which cannot distinguish or easily access the different parts
da Vinci robot	A surgical robotic arm made by Intuitive Surgical company and designed to facilitate complex surgery using a minimally invasive method
FOV	Field of View is the extent of observable world seen at a given moment
Kinect	A motion-sensing input device by Microsoft, originally built for Xbox 360 game console, used to detect depth and RGB
MCMS	Motion Controlled Manipulator System - is the product of our company
NUI	Natural User Interface is the common jargon used by designers for human-machine interface that is natural and barely noticeable to users
OpenNI	Open Natural Interaction is the non-profit organization responsible for developing the open sourced development library that we are using [1]
Perfboard	A thin and pre-drilled standard board for prototyping electronic circuit.
PWM	Pulse Width Modulation is a way of controlling power to inertial electrical devices by modulating the pulse waves
RGB	Red Green and Blue is a color model set by adding these 3 colors to create a broad array of colors
User	An able-bodied person capable of using both arms voluntarily
UI	User Interface is an industrial design field of human-machine interaction
IR	Infrared light, an electromagnetic radiation at 0.74 µm to 300 µm
PC	Personal Computer, a general purpose commercially available computer
H/L Combo	High level (+5V) and low level (0V) voltage combinations sent to the H-bridge's input
CCD	Charge Coupled Device is a device that moves charge in capacitive bins

	to convert photons into electronic charges (bits). CCD sensors are used for digital photography.
QA	Quality Assurance is utilized to ensure our system functions as required
Framebuffer	The monitor display buffer responsible for storing pixel information
OpenCV	Open Computer Vision, library capable of providing object detection
OpenGL	Open Graphics Library, cross-language library for 2D and 3D rendering
BOM	Bill of Materials is a list of all our sourced parts, model number and cost
DC motor	Direct current motor is an electric motor running on the unidirectional flow of electric charge, or a constant voltage
Servomotor	A rotary actuator allowing precise control of angular positions with feedback
H-Bridge	An electronic circuit that enables voltage to be applied across a load in either direction, which is used in robotics to allow DC motors to run forwards and backwards
Serial Communication	A process of sending data sequentially (i.e. one bit at a time) over a communication channel or computer bus
SDRAM	Synchronous dynamic random access memory is dynamically clocked memory in sync with the system bus, which allows pipelining to occur
mm, cm	Millimeter is 10^{-3} of a meter and centimeter is 10^{-2} of a meter
Morphological Opening	An image processing technique for binary images, dilating an eroded set A by structuring element B (which is usually a circle of any radius).

1. Introduction

The **MotiCon's** Motion-Controlled Manipulator System (MCMS) is a black-box system allowing intuitive motion control of various robotic manipulators by human gesture. The system consists of hardware, software, and microcontroller which all communicate with each other to achieve a desired physical movement at the manipulator. By sensing the arm movements of the user-operator with the Kinect sensor, the MCMS can interpret the human motion, and send the processed command signals to the microcontroller which regulates the voltage to the motors incorporated in the mechanical robotics.

The **MotiCon** MCMS acts to perform as Figure 1's illustration shows.

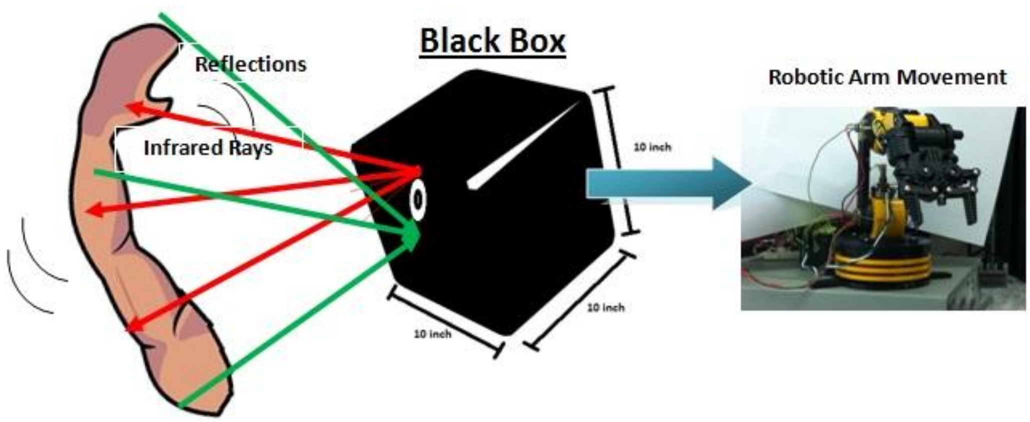

Figure 1: High-Level Model of MCMS

Over the past 13 weeks, the developers of **MotiCon** have been working vigorously to produce a proof-of-concept system. In essence, this report re-examines the process, difficulties and learning experience each member had throughout this extraordinary project. This post-mortem report finalizes our system's current status, deviation of implementation designs, future plans, budgetary and time constraints, as well as interpersonal and technical experiences of each member.

2. Current State of the System

2.1 Overall System

Currently our high level system follows the description of the "Black Box" and prototype's data flow chart is shown in Figure 2. Our current system has the following components: the Kinect sensor, the PC or laptop, Arduino UNO microcontroller, the circuit, a DC power supply and the DC robotic arm and the servo motor arm. We have successfully implemented 3-DOF inverse kinematics algorithm with the correct frame of reference of the user to the robotic manipulator [2]. In addition, we have also implemented 4-DOF inverse kinematics, hence activating the robotic wrist joint. We have successfully implemented the gripper to grip and release using image segmentation of the hand and fingers. Also we have re-integrated our system to the DC arm with the functions that work on the servo-arm. All the components meet the functional requirements for the proof-of-concept system.

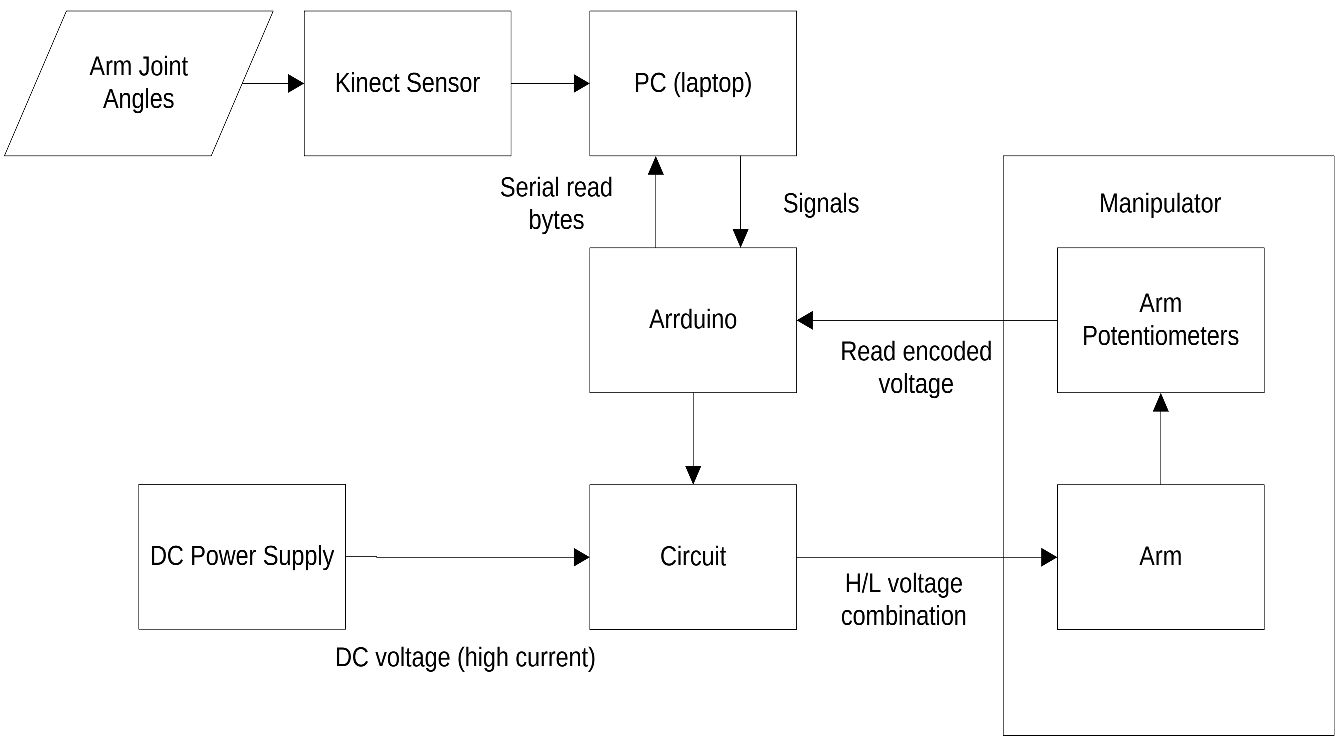

Figure 2: MCMS Black Box Dataflow Diagram

2.2 Hardware

The hardware content within our packaged system consists of an ATmega328-based Arduino UNO and an H-bridge circuit design. The microcontroller outputs controlling voltage signals to the DC or servo motors and receives feedback signal from the robotic arm joints (using potentiometers for the DC motors only). The H-bridge circuit acts as a selective switch to turn motors in clockwise or counter-clockwise rotation according to a desired angle. Our hardware is powered by an external high-current DC power supply, and the Arduino 5V (maximum 450 mA) output pin.

For the DC-motor arm prototype, the Arduino microcontroller receives the target angle positions of the 3 or 4 robotic joint angles (depending on the DOF inverse kinematics algorithm used) and one gripper angle from the potentiometers. These analog inputs are compared to desired angles from image processing, which provides feedback for accurate response when determining arm positions.

However for the servo-motor arm, we do not need analog angle feedback into the Arduino, instead a desired angle can be sent directly to the servo motor, which use discrete gear steps to achieve the desired angle, and holds this angle more accurately than DC motors.

To control the DC motors in the prototype, H-Bridge SN754410NE circuits that are limited to sustaining very low power consumption at the signal input pins are used to regulate current throughput of up to 1A from external power source [4]. Those bridges are not required for the control of the servo-motor arm; instead a small circuit is built to simplify signal connections and cable management.

2.3 Software

The primary functionality of the software is to compute and deliver accurate values representing the positioning of the user, which includes the inverse kinematics computation for the robotic arm joint angles to correspond with each of the user hand movements.

The high-level design of our software consists of three main libraries which construct the foundation of the whole program:

- Open Graphics Library (OpenGL) [5]
- Open Natural Interaction (OpenNI)
- Serial Communication Library

Using these three libraries, we are able to produce a stable communication between the Kinect, the computer, the Arduino, and the robotic arm.

We achieved 3-DOF inverse kinematics because it provides an intuitive navigation of the end effector with respect to the user's hand instead of movements on user's joint angles [4]. In addition, we have achieved the 4-DOF with wrist joint maneuver at the robotic arm.

Significantly, although OpenNI library does not have wrist or finger tracking, we developed our own image segmentation based on the "morphological opening" image processing technique (refer to Glossary) to differentiate an opened or closed fist. With this technique, we are able to control the robotic gripper to grab objects. To connect our code to control the robotic arm, the Arduino serial communication library is used to connect our calculated angles to send to the servo-motor and the DC-motor robotic arms.

2.4 Mechanical Actuator

Currently, our mechanical DC-motor arm and servo-motor arm functions as intended. For the OWI DC-motor arm, there are five degrees of freedom in this robotic arm; these are the elbow pitch rotation, the base yaw and pitch rotation, the wrist pitch rotation, and the gripper clench and open [6]. We have utilized 4 degrees of freedom, with only the wrist being inactive in movement. For the Lynxmotion servo-motor arm, we have a fully functioning robotic arm with all 5 degree of freedom implemented with the 4-DOF algorithm and the gripper-hand segmentation for opening or clenching the gripper [7].

2.5 User Interface

The Main Menu will display three options, Start, Stop, and Exit. "Start" will start the processing the movements into robotic arm movements. "Stop" holds the current position of the robotic arm, and stop processing any user movements until "Start" is pressed again. "Exit" will exit the program and reset the robotic arm back to default position. The Menu is displayed on the left side of the user's screen and is controlled by user's left hand, and each button can be selected by left hand clenching and opening to select the buttons.

The Figure 3 below shows our current UI with one of our fellow engineers in the field of view of the Kinect IR sensor.

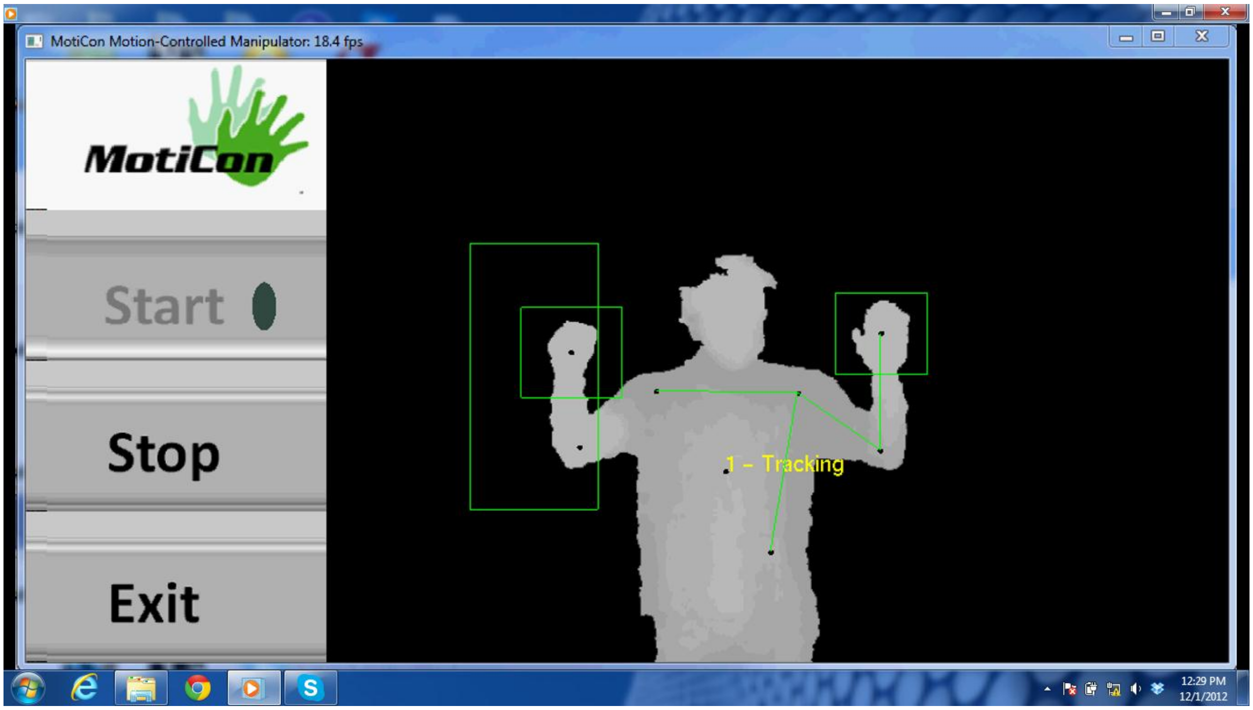

Figure 3: GUI for MCMS

3. Deviation of the System

3.1 Overall System

Our goal from the beginning is to build a functional proof-of-concept system; hence the overall current system meets our proof-of-concept functional requirements. Due to time constraint, our "Black Box" packaging is not as well-designed as we expect and is not light enough to be portable easily. Most of the deviations can be attributed to the overestimation of complexity in the intersystem connections. Currently, our system consists of two Arduino microcontrollers due to a lack of digital outputs for controlling both arms. Additionally the circuits are built on perfboards but not surface-mount assembled on PCBs due to the simplicity of the circuits.

3.2 Hardware

The circuits were initially estimated to be complicated, however due to the nature of the servo-motors, controlling the servo-motor arm can be done directly with the Arduino microcontroller. So we deviated in our requirements thinking that we needed a separate circuit design for the servo-motor arm.

3.3 Software

The software has not deviated from the initial plan of the requirements, because we have achieved 3-DOF inverse kinematics. In fact, after achieving 4-DOF inverse kinematics, we have exceeded our requirements in the functional documentation.

3.4 Mechanical Actuator

The two robotic arms, OWI DC motor arm and the Lynxmotion Servo-motor arm, have always met our mechanical requirements of a robotic manipulator. They are sufficiently small to be easily portable, and have enough degree of freedom (5 in total) to demonstrate our system.

3.5 User Interface

The user interface that we have built has “Start”, “Stop” and “Exit” options, but we did not implement the “Recalibrate” option or the “Save and Load Previous Path” options due to lack of time. Moreover, the additional options are not critical in the demonstration of the proof of concept, and therefore will only be implemented in the future development where the customers may desire more options of functionality.

4. Future Plans

4.1 Overall System

Since our completed project represents a proof-of-concept system, there are various aspects of this project that can be improved in order to guarantee the final product a place in the market. **MotiCon** believes there is significant potential for future developments of this product.

4.2 Hardware

For the electronic side, our system should have a general multi-application circuit that is compatible for any robotic manipulator. The circuit design can be further improved to a PCB design so it can be compact enough to fit into a marketable black box product. Currently, our circuits are specific to the robotic arms that we have used.

Significant work could also be done to improve the microcontroller. By purchasing a microcontroller with more pins, such as the Arduino ATxmega series’ XMEGA, we can customize the amount of input and output pins available to ensure successful integration for robotic arms with more degree of freedom. Currently an Arduino UNO only have 14 digital output pins and 6 analog inputs, which may not be sufficient for both arms if using more than 3DOF inverse kinematics algorithms.

Since our power supply is current limited at a relatively high value of 12A for the 5V output, an additional safety consideration is to ensure power and ground lines are tightly enclosed and insulated. Also, no shorting should occur by building a fail-safe component that turns supply off if too much current is drawn. In addition, sealing the case to protect internal components need to be consider for safety purposes, such as to protect from rain or excessive heat.

4.3 Software

The current 4-DOF inverse kinematics algorithm with gripper control is sufficient for grabbing objects and mobilizing the robotic arm to user's desired coordination. However, we believe the fluidity and accuracy of controlling the arm can be further improved by optimizing the algorithm's calculation of joint angles. In addition, we believe the hand's segmentation can be further improved from using 2 Kinect sensors for more precise joint position determination, or by using the Motion Leap sensor which will provide high resolution detection of motion.

4.4 Mechanical Actuator

In order to show that our system can be integrated to multiple applications, we have to expand our MCMS integration into more than one type of robotic arm. Because servo-motor arms represent the largest field of industrial robotic manipulators used in the commercial market today, we plan to focus more on that type of motors in future development works.

4.5 User Interface

Currently, our user interface suffices to show user can start, stop and exit the program simply by using their left hand to choose. As described in section 3.5, the user interface can consist of more options and functionality, such as recording a motion and allowing the program to repeat the motion on the robotic arm.

4.6 Haptic Feedback

Currently in the Microdynamic Systems Lab in Carnegie Mellow University, magnetic levitation haptic interfaces allow users to have tactile feedback on the object being grabbed [8]. This technology can be incorporated into our intuitive controlling system to create an ideal tool for surgeons to emulate tactile feedback for simulated surgical operations. Nevertheless, this technology will be perfect in further improving our MCMS.

5. Budgetary and Time Constraints

5.1 Budget

The Table 2 below outlined the projected and actual costs of each of the components of **MotiCon** up to November 28th, 2012.

Table 2: Projected versus Realized Costs of MCMS

Required Materials	**Quantity**	**Projected Costs**	**Realized Costs**
Hardware Kit:			
Microsoft Kinect Sensor	1	$150	Sponsored by Marinko Sarunic
ATmega328 - Arduino UNO	2	$40	Owned
1x40 Header Socket	5	$5	$4.48
H-bridges SN754410NE	5	$5	$10.67
81-row Perf Boards	3	$15	$15.63
10 ***kΩ*** *Thumbwheel Trimpot*	4	$2	$1.50
10 ***kΩ*** *Rotary Potentiometer*	1	$1	$0.95
1x40 Header Socket	5	$5	$4.48
Computer Power Supply	1	$20	Owned
Mechanical Arms:			
OWI-535 DC-motor Robotic Arm	1	$100	$68.65
Lynx-AL5D Servo-motor Robotic Arm	1	$300	$302.77
Hot Glue Gun Sticks	1	$3	$2.24

Hot Glue Gun	1	$6	$5.60
Hardware packaging of MCMS:			
Case Material	2	$50	Owned
30cm x 30cm Ply Wood	1	$20	Owned
Other materials:			
Logitech HD Pro Webcam	1	$3	$10.67
TOTAL		$722	$427.64

From the table above, we see that our total cost for building the proof-of-concept is cheaper than our projected value. We initially thought the total product will cost more than the $700 generously funded from ESSEF. However, our estimated cost is approximately $450, summing to approximately $250 less than the projected cost. One of major reasons is because the Kinect sensor was sponsored by Dr. Marinko Sarunic. The second reason is due to the ownership of the Arduino UNO microcontrollers by several of **MotiCon**'s own developers: Vincent, Kevin and Kay. Lastly, the currently used power supply has been obtained from salvaging an old computer. Otherwise, this budget list is accurate down to the cost of pin sockets, connectors, black box and headers costs. Needless to say, we have met our planned budget and achieved what we desired.

5.2 Time

The following table illustrates our scheduled project milestones throughout this semester.

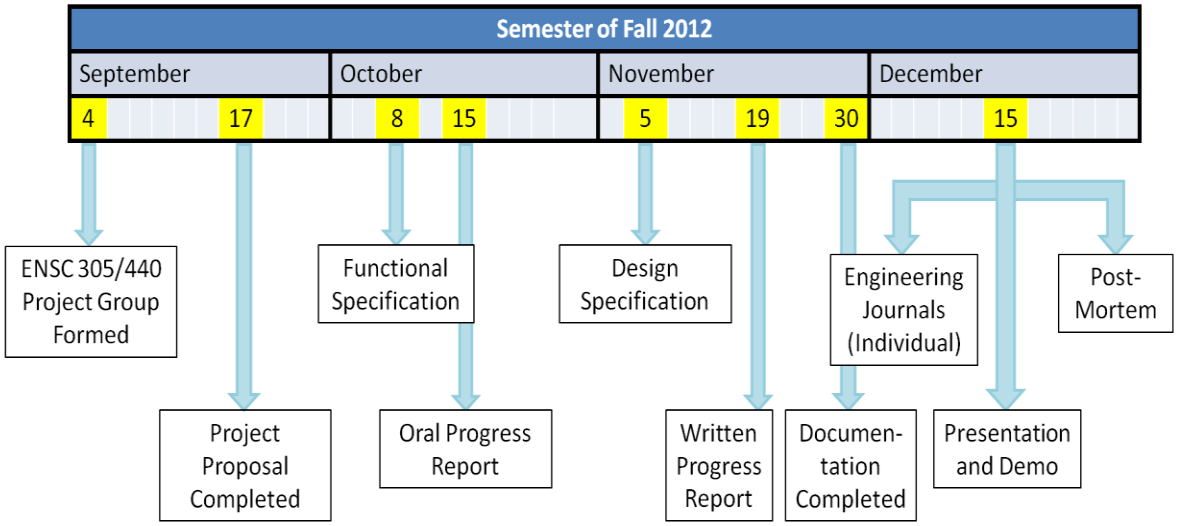

Figure 4: Milestone Chart

Here is the Gantt chart displaying the proposed timeline for each task within the project itself that we made during planning phase.

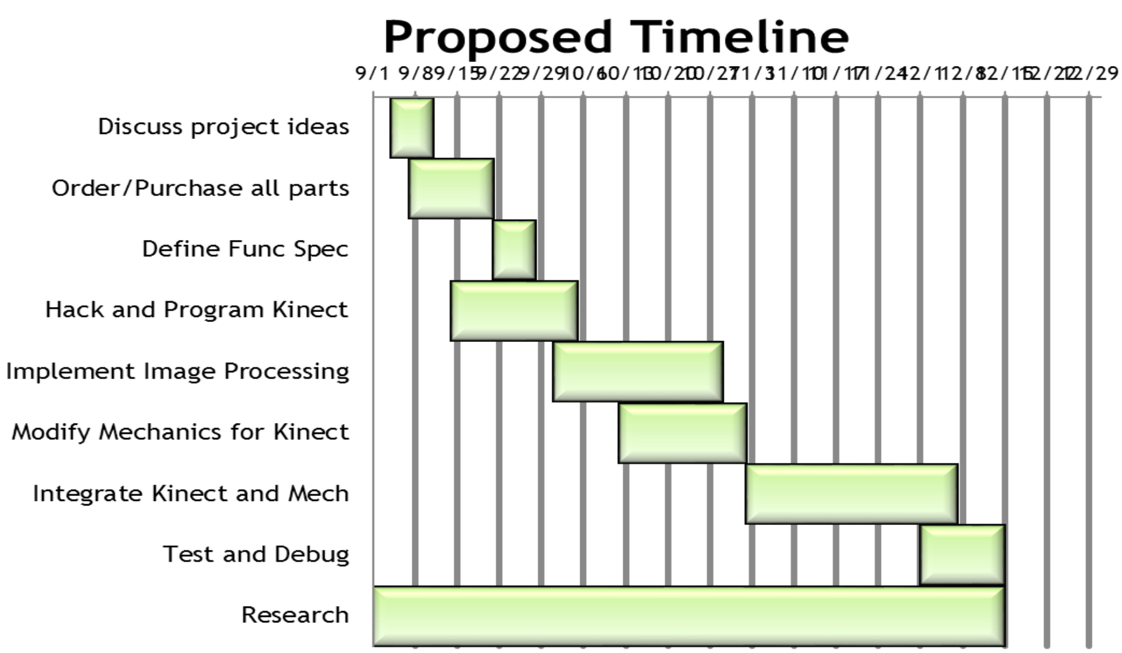

Figure 5: Gantt chart of Projected Timeline of MCMS

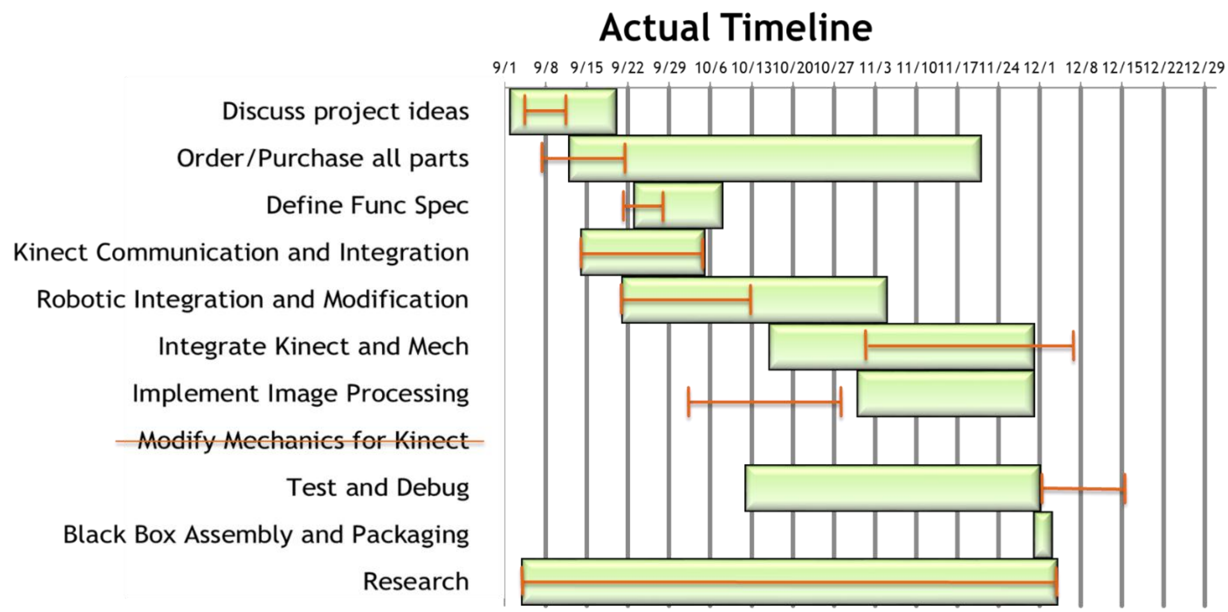

Figure 6: Gantt Chart of Actual Timeline compared to Planned Timeline in Figure 5

In Figure 6, we see that the orange bars show the original expected timeline, the green bars shows the actual timeline that we worked on MCMS for. It is interesting to note that the actual implementation and debugging stages duration deviated the most than we had anticipated. Of course, there are also deviations in other stages from the expected deadlines we have set up for the project in Figure 6. One reason is because we had underestimated the complexity for the integration of components, and also the difficulty in troubleshooting and debugging the overall system errors and flaws.

Throughout our development, our main goal is to progress one step at a time, following the infamous KISS rule (i.e. "Keep it Simple Stupid") to ensure at least everything works at all times or return to the previous working state. We begin integration of one joint on DC arm and software early in October 13th, which is one week ahead of schedule. From then on, after successful integration, it's a simple matter of improving the code for inverse kinematics and the re-integration to the servo-motor arm. Without a doubt, all group members believed the project schedule had to be followed tightly, and that deadlines for tasks were assigned every group meeting. This commitment and delivery from each member resulted in the overall success we had with the prototype.

6. Alternative Solutions to Encountered Issues

Undeniably, as a cohesive team, our approaches to overcome any challenges have proven to be quite successful for us. However, there were several aspects that we as a team agreed could have been smoother had we taken an alternative approach. These include:

6.1 Utilize familiar development approaches

We spent a lot of time in the beginning determining the method for development between Microsoft SDK and OpenNI. We ended up using quite a lot of time trying to figure out the development of Microsoft SDK and the C# language that came with it.

The usability of the Microsoft was also limited to Windows 7 and Visual Studio 10 and above. Therefore we foresee that a large learning curve would be mandatory in order to begin efficiently programming and developing the software. However, many of us had already much experience programming with C++ in the Visual Studio 2008 development environment.

OpenNI is a far more versatile library for communication with Kinect, and it offered support for the development with C++ in Visual Studio 2008. Therefore, from the beginning we should have decided to use OpenNI as a development library due to the fact that it provides us with the advantage of familiarity and saves us the time required for researching with the Microsoft SDK libraries.

6.2 Perform cable management early

The DC-motor arm became messy with wires very quickly, and if we had added more concern with cable management from the very beginning, this could have been avoided.

6.3 Further organize group meetings

Often during group meetings, members were either late or distracted when discussing the project. We should have held our meetings in more formal areas, with meeting tables, and no possible objects to cause distraction, to have a formal company-like meeting. Also, during meetings, we often debate on technical difficulties instead of focusing discussions in big picture.

6.4 Improve communication and leadership

Communication amongst each member was not always successful throughout the project. Not all members were able to always completely understand what was being discussed and described. More initiative could have been taken to prevent miscommunication amongst the team, which could result in saving even time throughout development.

6.5 Further distinguish roles and tasks

Unfortunately, we have overestimated the work load of the mechanical component. Even though the two demonstration robotic arms took time to build and understand, the team should have been separated into Hardware and Software instead.

We think that we could have reduced the project integration time, by putting one or two members on the integration, before having actually completed each team's tasks. If we had decided on this at the very beginning of the project, more time could have been saved for the two members in relearning the other components of the project, and to use this time for further development instead.

6.6 Detailing the milestones

The last biggest issue we encountered concerns the milestones. It seems like our team wrote them in a too general way, without going far enough into the details of the project. This mainly showed up for the functional and design specifications. We believe we could easily have included more specific information on the actual implementation of the prototype.

7. Inter-personal and Technical Experiences

7.1 Vincent Wong

When I first juggled with different capstone ideas back in August 2012 with fellow engineering peer Kevin Wong, we initially aimed our project to be biomedical-oriented. The idea of using a Kinect sensor to control robotic arms by simple intuitive user movements came about after multiple ideas were researched into and trialed out after the whole team was garnered.

In search of some highly dependable and strong members, I quickly thought of Kevin Wong and Kay Sze. I have known Kevin Wong since first year, and have worked closely with him over our undergraduate years, and know him to be a very well-rounded engineer, proficient especially in OpenGL programming after working with Dr. Marinko Sarunic. I knew Kay was a highly reliable partner having worked with her in ENSC 372, and that her microcontroller programming skills with the Arduino and experience with circuit assembling would be useful to the project. Surprisingly, Kevin Wong introduced two engineering PhD students, Jing Xu and Hsiu-Yang Tseng, into our group. Jing specialized in image segmentation and has strong programming expertise. Lastly, Hsiu-Yang had a proficient mechanical and robotic theoretical background. Having already found five group members, I believe that was already substantial and the idea was first to build a linear Intracytoplasmic Sperm Injection (ICSI) controlled by Kinect. However, with the additional sixth member, Arnaud Martin, an exchange student from Belgium with robotic experience, we finalized our motion controlled system to be a multi-purpose integration system for controlling robotic manipulators.

This capstone experience was a very humbling experience and gave me the chance to experience how much effort it takes to lead a group and work in a large group. As I am a meek person, it was initially difficult to organize or voice my opinions as CEO, but during meetings, the group members have been supportive and encouraging, and never once did we have any conflicts from unsettled disagreements. Organizing the group members according to their expertise was difficult but it brought about a higher efficiency for work output because the team was smaller and communication was less difficult then.

Throughout this experience, I have also learned how to program proficiently with OpenGL and OpenNI and have experienced code management. In as well, I learned a lot about robotics with help from Arnaud, who was proficient with robotic and inverse kinematics. The most satisfying feeling, is to use my hardware skills in circuit designing and hardware development to help the team develop circuits used in the project. Lastly, as CEO, I was able to oversee many of the technical and project overview, so it was a very rewarding experience for me.

7.2 Kevin Wong

The capstone project course has provided me with many valuable experiences that will one day be beneficial in both the industrial world and my personal life. It has been a fantastic four months in the amount of technical and interpersonal wisdom that I have gained. Being able to work cooperatively in a team on a project, even when the team is composed of several close friends, is not always an easy task. However to be able to overcome these challenges together successfully I had learned to work through difficulties and setbacks together as a team, to discuss and solve issues together. This, I realized, produces a friendly environment for each group member to work in. In my perspective, our group has been able to achieve each of these characteristics, which I am truly happy about.

At the beginning of the semester, we initially started with only five members, Vincent Wong, Kay Sze, Jing Xu, and Hsiu-Yang Tseng. We initially came up with the idea of using Kinect as a basis for motion-control of robotics. However none of us had experience in robotics, and without any knowledge in robotics this project would have been a disaster. Luckily in the first week of classes, an exchange student from Belgium, Arnaud Martin, was looking for a group to be a part of. We saw his list of experience, and discovered that he would be an excellent addition to our roster. Throughout these few months, Arnaud has not only shown his competence in knowledge of robotics, he has also become a very friendly and critical member of our group.

There are three main categories in our project, software, electronics, and robotics. I was mainly responsible for the software portion along with Jing Xu. Due to my strong background in programming with C++ and OpenGL, I was able to grasp the concept behind the integration between Kinect, OpenGL, and Arduino very quickly at the beginning of the semester. Jing and Vincent were relatively new to OpenGL, but upon teaching them the general concepts of OpenGL, I found they were amazing learners, and were able to program using the OpenGL as well as I could in no time. We then tackled the image processing algorithms together, and I found that our discussions for how to tackle these issues really help the process of development of the software. At times there were some communication issues when it came to integration; luckily each of the team members were very patient with one another, and we were communicate with each other about possible issues and solutions which resulted in the successful development of the final product.

I am quite proud to be able to see the progress of our project evolve so much, from an idea to a manifestation. It really gives me a sense of accomplishment that we were able to achieve all of our pre-assigned milestones successfully and according to our planned schedule. This course has been one of the toughest courses I have taken throughout my undergraduate degree, but it is also undeniably one of the most valuable.

7.3 Jing Xu

From the very beginning, I knew we chose an extremely ambitious project and we would settle for nothing less than success. We knew we would have to face a lot of challenges, so we were committed to working on a tight set schedule. Since I was responsible for the entire software development along with Kevin, my learning experience was mainly focused in this area. Kevin and I both had programming experiences with C++, and he has been working with OpenGL. For working with Kinect, we chose to use OpenNI to communicate with Kinect which can be written in C++. This made it easier for me because this library compiled with my first choice, as I had decided on writing everything with C++. Along the way, I managed to learn OpenGL, and implement the interface menu which can be controlled by Kinect. There are two types of kinematic algorithms that we needed to implement, the 3-DOF and the 4-DOF. It is a great experience for me to revisit the math that I learned a long time ago. This course helped me expand my experience in integrating and interfacing with components made by other people. And I always love programming and debugging code.

Furthermore, I managed to gain some interpersonal skills in project management and team development as well. Every member has his or her strengths and weaknesses, and some group members are more experienced than others. Hsiu-Yang and I have done several projects from scratch during our graduate studies, and due to our rich experience with working in these start-up projects, we were able to solve issues and questions discussed in our meeting. Also, we are able to quickly update with the progress of work that is done by our lab mates. So sometimes when a question and an idea were brought up by group members, we can give quick response to save a lot of researching time. This group project is much more different that working in a lab group for a course. It requires excellent communications between every group member. Sometimes we have different approaches to solve the same problem, hence, the abilities to listen and express your own ideas are very crucial to the success of the project. These skills and experiences takes time to achieve, and this course is definitely beneficial to all of us. We do have struggling and disagreements, but we all managed to overcome them efficiently, and move on to the next topic. I do believe the interpersonal skill I learned from this project will be extremely useful in my future career.

7.4 Kay Sze

I still remember when I was a first year engineering student, I felt that it is impossible to start a project from nothing to something that we can demonstrate to professors and TAs. But I know that it would happen in my final year. However, from what the other engineering friends told me, Capstone project is always a challenging and a time consuming course in our undergrad life, and they always complain about their lazy group mates. Luckily, I have Vincent, Hsiu-Yang, Arnaud, Kevin and Jing to work with me. Since the project timeline is very tight, we tried to come up with lots of ideas before this semester starts. Yet, we were having a hard time to decide which idea we want to use. This is due to the lack of experience. Nevertheless, our PhD student members, Hsiu-Yang and Jing, have more experiences of progressing projects from their studies. The advices from them are valuable and helped us to know the possibilities of each idea. Soon after the semester started, we met up with our 6th member, Arnaud, and warmly invited him to our group.

We are mainly divided into three groups, the software team, the hardware team as well as the microcontroller team. Hsiu-Yang and I are in the hardware team, which is responsible for the robotic arm. We thought it would work nicely as we planned everything seriously in the beginning. Yet, things sometimes do not follow our way. During the first month of the term, everything was still developing so that our hardware team did not have much to do. Therefore, we decided to split our team into the other two teams (hardware or software) for that period of time. It would help the progress of the development and it gives better work distribution. I learnt that plans have to be changed according to the progress of the work because we usually do not know what is going to happen in the next step.

7.5 Hsiu-Yang Tseng

Back in the summer of 2012, I was, as usual, busy about my research, and one day I was encouraged by Jing Xu in a conversation to join a forming group and discuss potential topics for this ENSC440 course. I right away decided to participate in this energetic and promising team. I was soon introduced to every other team members, Kevin, Vincent, and Kay. We all spent few weeks brainstorming for various possible ideas. We shared our thoughts to each other and fully discussed the feasibility and challenges of them. During that discussion, we performed an example of initiating innovations, and I realized that an open-minded environment is always critical and necessary to develop any groundbreaking ideas.

We first focused on looking for a technology need in the auto-control and image processing fields. Once I heard of the new technology, Kinect infrared depth sensor, from Kevin, I proposed to use it for controlling micromanipulators, which is currently, inconveniently operated with joysticks and buttons meanwhile requiring a second person to view through a microscope. And afterwards, we further push the idea to develop a remote control surgery tool/robot, which is one of the most promising and challenging medical instrument technology people trying to resolve. After we finished the planning, we separate the work into three parts: software, firmware, and hardware. I was assigned in the hardware group, in charge of robotics and mechanical problems with Kay. During the development, we met several challenges and technical obstacles, and thanks to our effort and mutual help we would be able to solve or improve most of the problems we met. For example, we had difficulties on finding a way to locate the robotic arm to the position desired as assigned by human arm. We, instead, utilized the inverse kinematic method to control three and even four degree of freedom motions. From these experiences, I feel this is the best opportunity to not only practice my technical acumen but to also learn from other fellow engineers.

After doing this project, I am enriched by this experience. I am also more confident than before to work on a team project especially for product development. I really appreciate to have these wonderful group members and working culture; furthermore, I especially appreciate the SFU School of Engineering Science and our course instructors and teaching assistants to provide much help, advice, and most importantly, this learning opportunity to us. I believe the experience and accomplishment gained in the ENSC440 project will be one of the most valuable assets throughout my future professional career.

7.6 Arnaud Martin

As an exchange student, the capstone project course gave me the opportunity to integrate and discover a completely new university. By having the chance of working with this excellent team, I have been able to share knowledge and experience on a technical level as well as on a personal level.

Jumping into this well-functioning team was a little bit scary at first, probably because the other members of the team already knew each other and had been thinking on the project for a long time. However, the integration period was really quick and each member of the team participated to the open-mindedness required for me to successfully bring my personal background.

The reason I loved working on this project is its diversity in the domains of development. Indeed, our team needed skills in software development, electronics, robotics and even marketing. Due to the extreme variety of background the six members of the team had, each one of those developments was managed by someone with a competent background in this domain.

But the realization of this project lies in the capacity of everyone to adapt, to learn on specific topics, to question each other's opinions and to accept challenges openly. Our team had to make a lot of decisions and none of these choices have been regretted.

To conclude, I would say that the entire time spent on this project has been an amazing experience on the inter-personal level, the technical level and the academic level. I am glad I chose this course and I am fortunate that I found this team to work with.

8. Conclusion

In retrospect, this entire project could not have been possible without any of the members. This project allowed every member to experience the difficulty and the joy of building an application from our idea to a final proof-of-concept system. Even though our current system cannot be sold, there is enormous satisfaction from seeing the achievement phases evolve from 1-DOF forward kinematics joint motion, to 3-DOF inverse kinematics and finally to 4-DOF inverse kinematic with gripper activation. What really pushed our group to the success that we have today is that we planned well ahead and ensured that simple and realizable milestones are set so that we have something working each step of the development cycle. Instead of letting the ambitious scope of the project overwhelm us, at **MotiCon**, we used the engineering tactic that we garnered from learning experiences at SFU to finish our project from start to finish.

9. References

[1] Open Natural Interaction. OpenNI User Guide. Retrieved September 10th, 2012.

[2] Grochow, Keith et al. Style-Based Inverse Kinematics. University of Washington and University of Toronto. 2004.

[3] Borenstein, Gregg. Making Things See: 3D Vision with Kinect, Processing, Adruino, and Makebot. O'Reilly. Media Inc. 2012.

[4] Texas Instrument. SN754410 Quadruple Half-H Driver. November 1995. http://www.ti.com/lit/ds/symlink/sn754410.pdf

[5] Khronos Group. The official Guide to Learning OpenGL, Version 1.1. Accessed October 1st, 2012. http://www.glprogramming.com/red/about.html

[6] OWI Inc. Robotic Arm Edge. Accessed September 20th, 2012. http://www.owirobot.com/robotic-arm-edge-1/

[7] Lynxmotion. Lynxmotion AL5d Robotic Arm. Accessed October 15th, 2012. http://www.lynxmotion.com/c-130-al5d.aspx

[8] Magnetic Levitation Haptic Device. Accessed November 20, 2012. http://www.msl.ri.cmu.edu/projects/haptic/haptic_device.php

