Colorado School of Mines

Pommel Horse Training

Authors: Evan Balogh, Travis Johnson, Austin Kauffman, Zac McClain

Clients: William Hoff, Brian Reily, Mounir Zok

16 June 2015

Table of Contents

Table of Figures

1. Introduction

1.1 Client Description

William Hoff, Brian Reily, and Mounir Zok each played a different role in this project. Dr. William Hoff, an electrical engineering professor at CSM, advised Brian Reily, a computer science student. Mr. Reily created the original C++ code base that used OpenCV and the Microsoft Kinect to track the Elite Athlete's feet on the pommel horse. Mr. Reily also provided the requirements and guidance throughout the project.

Dr. Mounir Zok, the Senior Sports Technologist at the United States Olympic Committee, was the main consultant given his expertise in sports, technology, and user-centered design. Dr. Zok, along with the gymnastics coaches, helped make design decisions for the features of the program.

1.2 Product Vision

Create a user-friendly coaching tool for the pommel horse event in elite male gymnastics. Using the Microsoft Kinect v2 with a Windows 8 platform, the system will record video of a pommel horse performance while tracking the feet locations of a gymnast on the pommel horse. The data collected will be analyzed so that useful statistics can be compiled and displayed. Primarily, this project aims to provide useful visualizations for the athletes and coaches to adjust training techniques to quantitative data about routines and trends.

2. Requirements

2.1 Functional Requirements

The User Must be Able to:

- Select the Performing Athlete from an Expandable List of Profiles
- Record a Performance
- Visualize Useful Data From the Video Analysis

The User Would Find it Useful to:

- Save Analysis and Video From a Performance to an Associated Profile
- Compare an Individual Athlete's Routine History
- View Video in Slow Motion and Frame-by-Frame

2.2 Non-Functional Requirements

This Program Will Require the Target Platform to:

- Run Windows 8 (64-bit)
- Connect to a Microsoft Kinect via USB 3.0
- Download OpenCV, Qt, Kinect SDK, XVid Codec, and Microsoft Visual Studio

3. System Architecture

3.1 System Diagram

As Figure 1 details, a Windows 8 C++ application abstracts the athlete database and kinect interface while providing effective GUIs for a gymnastics coach.

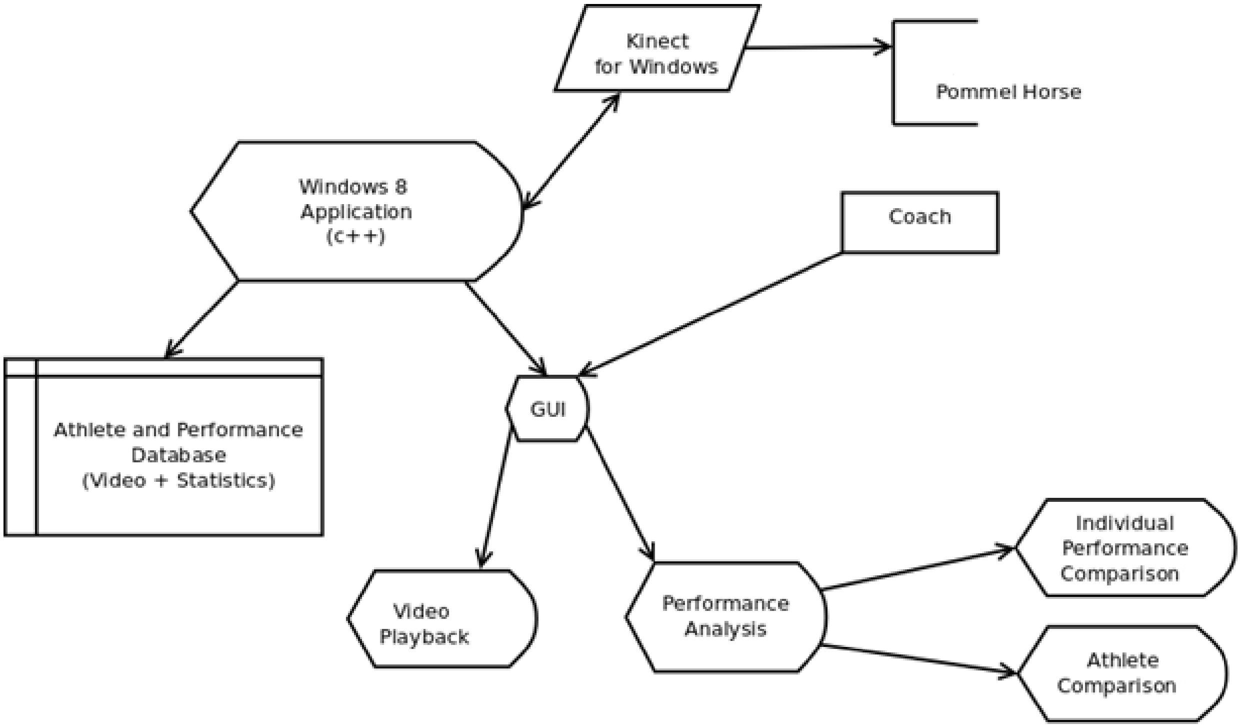


Figure 1

3.2 Explanation of Components

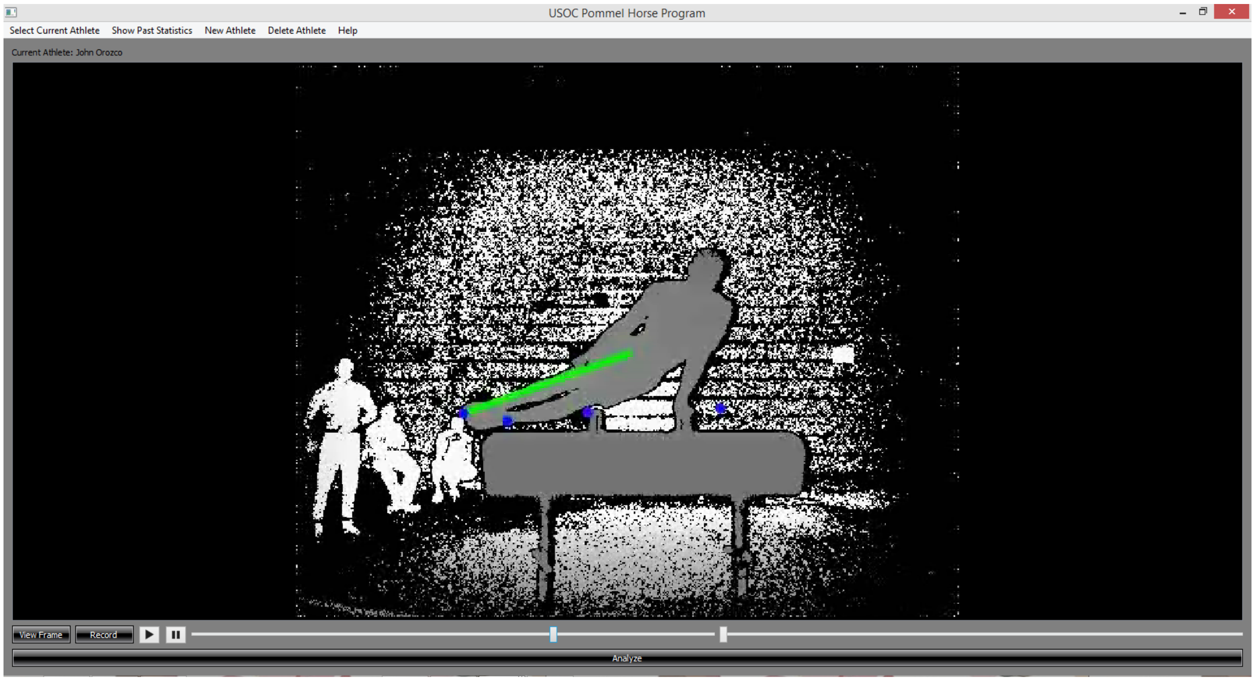
The features of the program are easy to see in the Pommel Horse Program main window. The menu bar’s only submenus are the “Select Current Athlete” and “Delete Athlete” buttons, which open a dynamic list of athlete profiles. The menu bar also includes a ”Show Past Statistics” button that allows the user to open the selected athlete’s profile, a “New Athlete” button that creates a new athlete profile, and a “Help” button that opens a pdf including FAQs and instructions about program use.

In the main window (Figure 2), you can also use the “View Frame” button to set up your Kinect camera. This button opens a live feed from the camera without saving. The “Record” button will bring up the same window as the “View Frame” button, but the routine will be saved. To use the “Record” button, you must first select an athlete. After a routine has been recorded, two dialog boxes will appear. The first asks if the user would like to save the video and the second asks if the user would like to save a note that describes the routine. If the video is saved, the video will be shown in the main window and the statistics can be accessed by clicking the “Analyze” button.

In the Analytical Data window (Figure 3), two donut charts are shown with the average spin consistency and average body angle consistency. Below the donut chart is a combined bar/line graph of the individual spin times for the routine. This graph can be customized with checkboxes that hide/show the bar graph and the line graph. The estimated accuracy of the Kinect (calculated by analyzing the frame rate consistency of the camera during the routine) displays next to the checkboxes.

The Past Statistical Data window (Figure 4) includes the Date, Time, Body Angle Consistency, Spin Time Consistency, Number of Spins, Kinect Accuracy, and Notes for each saved routine. The user may double click on a routine to view the video in the main window and pull up the statistics in the Analytical Data window. The “Graph Historical Trends” button at the bottom will bring up similar graphs as the Analyze window. The “Delete Old Performances” button brings up a dialog box that performs a batch delete. It shows how many performances are in the profile and asks how many of the oldest routines the user would like to delete.

3.3 Main Window Screenshot

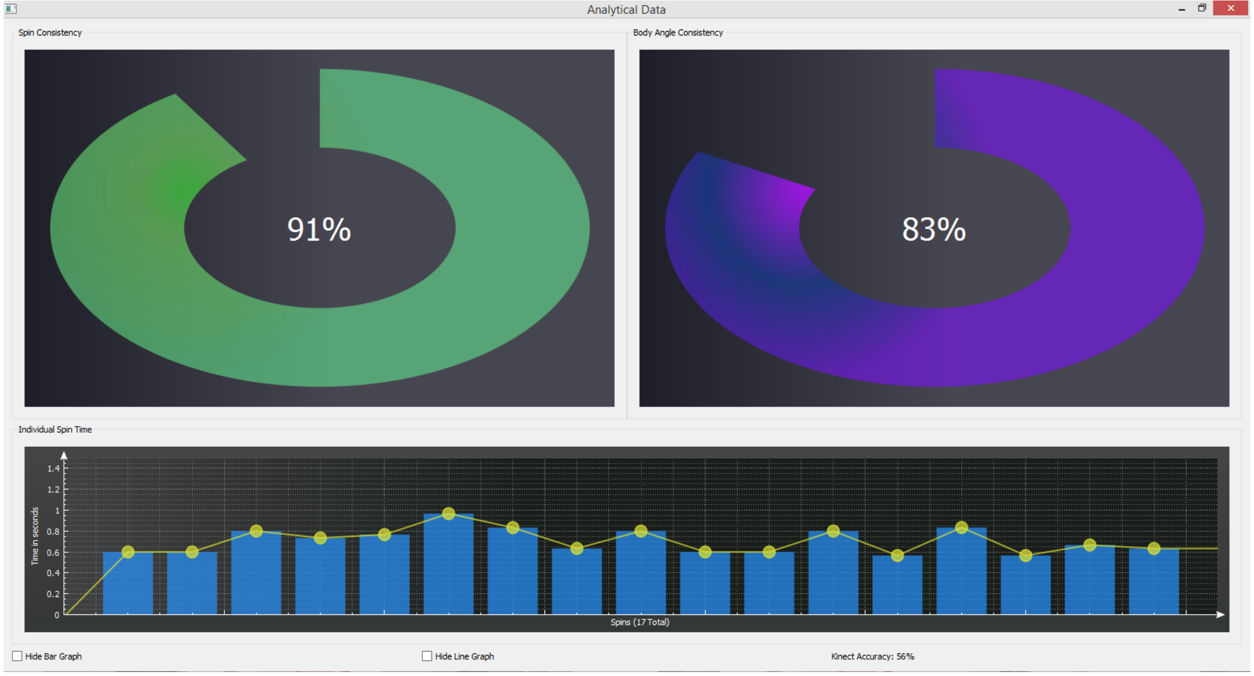


USOC Pommel Horse Program

Current Athlete: John Orozco

Figure 2

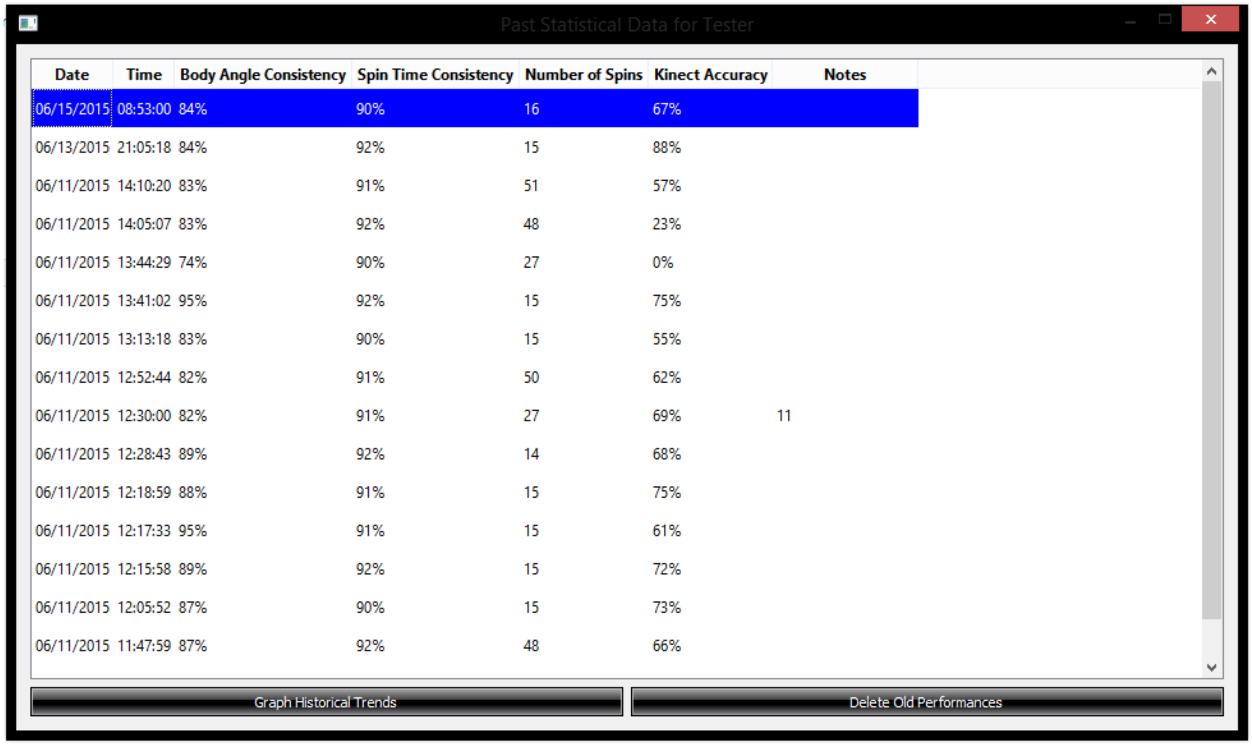
3.4 Analytical Data Window



Kinect Accuracy: 56%

Figure 3

3.5 Athlete Profile Screenshot



Past Statistical Data for Tester

Date	Time	Body Angle Consistency	Spin Time Consistency	Number of Spins	Kinect Accuracy	Notes
06/15/2015	08:53:00	84%	90%	16	67%	
06/13/2015	21:05:18	84%	92%	15	88%	
06/11/2015	14:10:20	83%	91%	51	57%	
06/11/2015	14:05:07	83%	92%	48	23%	
06/11/2015	13:44:29	74%	90%	27	0%	
06/11/2015	13:41:02	95%	92%	15	75%	
06/11/2015	13:13:18	83%	90%	15	55%	
06/11/2015	12:52:44	82%	91%	50	62%	
06/11/2015	12:30:00	82%	91%	27	69%	11
06/11/2015	12:28:43	89%	92%	14	68%	
06/11/2015	12:18:59	88%	91%	15	75%	
06/11/2015	12:17:33	95%	91%	15	61%	
06/11/2015	12:15:58	89%	92%	15	72%	
06/11/2015	12:05:52	87%	90%	15	73%	
06/11/2015	11:47:59	87%	92%	48	66%	

Figure 4

4. Technical Design

4.1 System Design Flowchart

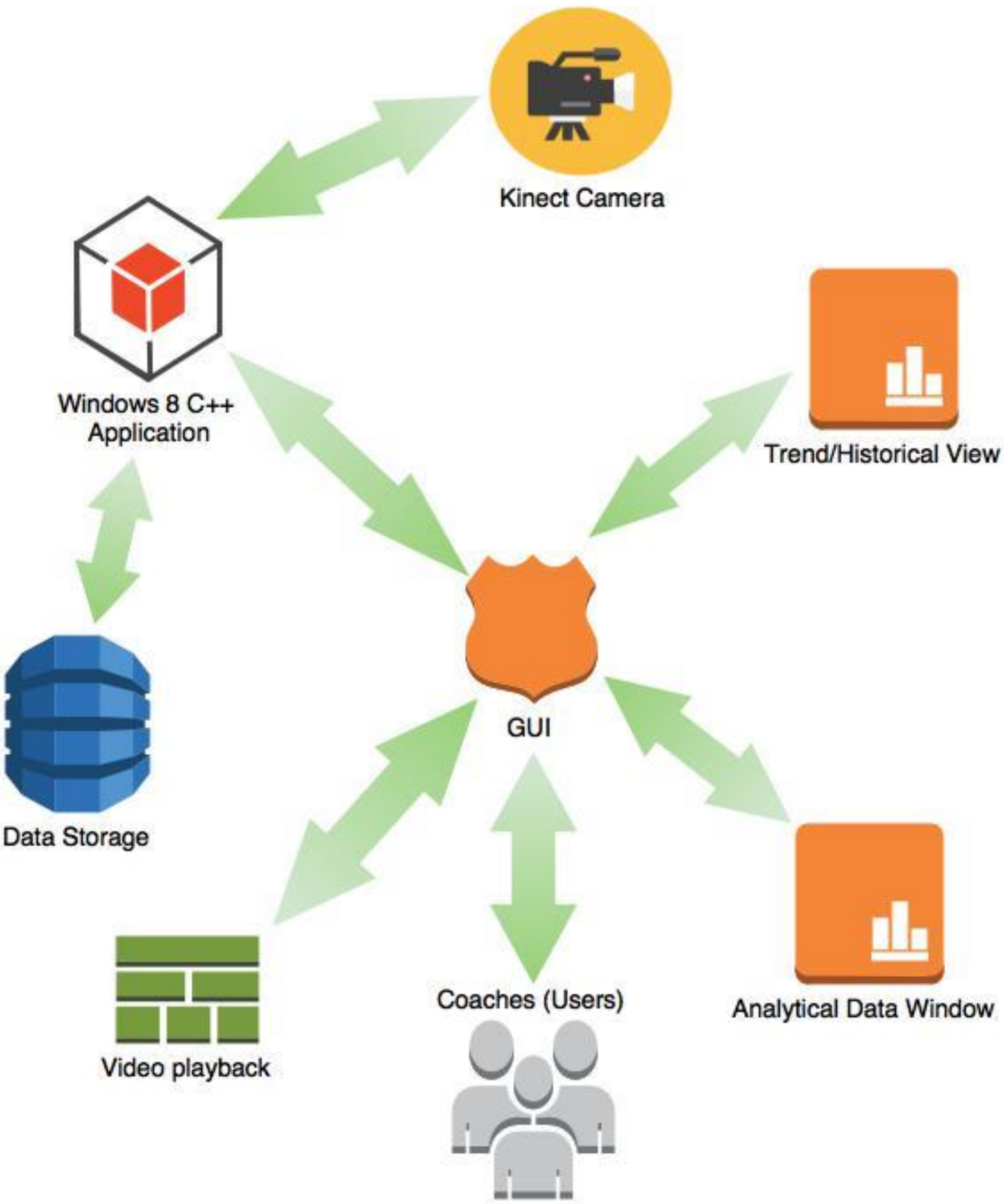


Figure 5

4.2 User Friendly Design

The target users (athletes and coaches) require that the program have easy interfaces. This is because the users will not have a background in technology and the program will be used in the midst of training; so the steps to record, analyze, and view data cannot be complicated. This requirement was the main focus of GUI design for this project.

The menu bar in the Pommel Horse Program main window is not complicated, with only two buttons with submenus. For the rest of the program, all other buttons are visible in the window and are laid out so that the functions of the program will flow seamlessly.

4.3 Body Angle Arc

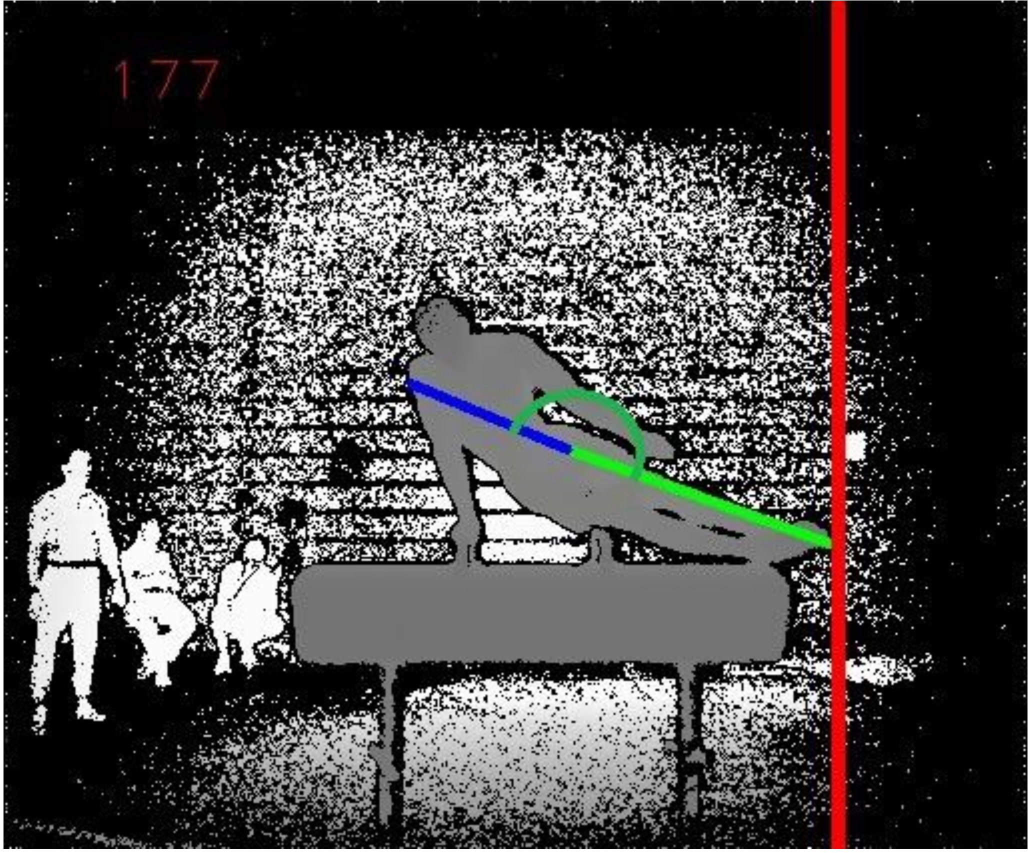


Figure 6

Brian Reily’s code finds the body contour and distinguishes between the upper body and lower body by finding the two longest vectors in the contour. The vector which has lower height coordinates is considered to track the lower body, and vice-versa. The angle between these vectors is saved at all extrema and is represented by the dark green arc.

4.2 Ensuring Accurate Data

Due to the inconsistencies with the Kinect Camera, a percentage is shown to give the user an idea of the accuracy of the analysis: determined by the number of frames dropped every second. The accuracy is heavily dependent upon the current performance of the machine. If the computer is not solely dedicated to this program while recording a performance, the program will drop frames and therefore data. Attempting to record the color video stream resulted in very poor accuracy. Currently, the program will remain as a depth image to ensure more accurate results.

5. Design & Implementation Decisions

5.1 Windows 8, OpenCV, Qt

The initial code base designed to track the feet location of an athlete was created with the open source library, OpenCV. This library provided the main functionality for processing images.

Qt was used to design the graphical user interface (GUI). The use of the Qt library allowed for a more professionally built, aesthetically pleasing GUI. Qt greatly enhanced the color schemes and the capabilities of this program.

Placing those components together, Windows 8 was chosen as the operating system in order to allow connectivity to the Kinect Camera and to allow utilization of the Kinect Software Development Kit (SDK).

5.2 Video Formatting

Due to the limitations of hard drive sizes and the video size the Kinect Camera creates, it is unrealistic to allow the functionality to save each individual depth recording. With the use of OpenCV, the video is reformatted into a smaller video size to allow the capability of mass storing of videos. Using the XVid codecs, the video is formatted into Audio Video Interleave (AVI). This video format allows the coaches to save the video files for review without worry of quickly running out of hard drive space.

5.3 Efficient Data Display

With the vast amount of ways to display the data, it was in the best interest of the coaches to view the data in graphs and charts. With donut charts, bar graphs, and line graphs, the coaches are able to glance at the screen to determine the consistency of each performance or trends over multiple performances.

6. Lessons Learned

When developing for a specific machine, ensure that the machine is able to handle the software libraries that are necessary for the program to perform as expected. If many libraries are necessary for the program to perform, build a simple application and test that it runs on the target machine. This extra testing will allow for the issues to be discovered on a smaller scale, allowing the library dependencies missing to be discovered more efficiently. If the libraries are not necessary, do not include them. Try to reduce the dependencies on external libraries to allow the release version to be created easier. The main lesson learned is to use software SDKs that allow for cross platform development environments and application deployments.

7. Results

7.1 Features

After four weeks of developing, the user of this coaching tool can:

- Select the performing athlete from a list of profiles
- Create and Delete Athlete Profiles
- View the Kinect's Frame Without Recording
- Record a Pommel Horse Performance
- Playback the Recording
- Edit the Speed of Video Playback (slow motion/frame-by-frame)
- Get data about the performance
    - Spin Times
    - Average Tempo
    - Consistency in Spin Times
    - Consistency in angle from Head-Centroid-Feet
- Save analysis and video from a performance to the athlete's profile
- Visualize the analysis of an individual performance
- Discern Consistency of a Statistic by Colors
    - green > 85% consistent whereas red < 60% consistent
- Hide the Overlaid Histogram and/or Line Plot on Analysis and Trend Views
- Sort a Table of an individual athlete's past performances based on different statistics
- Load a previous performance for playback and analysis
- Delete the n oldest performances from a profile
- Visualize performance trends over time
- Click Radio Buttons to View Different Trends in a Bar/Line Plot

7.2 Testing and Future Work

The system was tested at several stages in the development process at the Training Center in Colorado Springs with the assistance of the male gymnasts and coaching staff. During testing sessions, the coaching staff was also able to give direct input about the usability of features and data visualizations in the application.

A known issue with the system involves inconsistent frame rates from the Kinect's depth sensor feed. Attempting to record the color video feed for playback exacerbates this issue. Additionally, the Kinect services for Windows 8 are very intensive processes; therefore, running more processes on the host machine will interfere with the accuracy of the coaching tool.

In continuing development of this application, these features would enhance the usefulness of the product:

- Color Video for Playback in a Good Framerate
- Spline Interpolation in C++ for more data points
- A Consistent Frame Rate for Depth Image Data
- Comparison of Athlete-to-Athlete Trends and Performances
- Tracing the plane of rotation (path of the feet)
- Measuring the angle that the plane of rotation makes with the horizontal
- Enhance the Feet Tracking Algorithm to detect extrema for smaller athletes (kids)
- Cropping Video after Recording