METHOD FOR THE ACQUISITION OF ARM MOVEMENT DATA USING ACCELEROMETERS

by

Allison L. Hall

Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2005

Signature of Author...
Department of Mechanical Engineering
May 23, 2005

Certified by ....
Hermano Igo Krebs
Principal Research Scientist
Thesis Supervisor

Accepted by ....
Professor Ernest G. Cravalho
Chairman of the Undergraduate Thesis Committee

METHOD FOR THE ACQUISITION OF ARM MOVEMENT DATA USING ACCELEROMETERS

by

ALLISON L. HALL

Submitted to the Department of Mechanical Engineering
On May 23, 2005 in partial fulfillment of the
requirements for the Degree of Bachelor of Science in
Mechanical Engineering

ABSTRACT

Partial paralysis is one of the most common problems that affect stroke survivors. Many different rehabilitation therapies are available to stroke patients, including robot-aided rehabilitation, immobilization therapy, and electrical stimulation. Regardless of the choice of therapy, it is beneficial for the therapist to know whether the therapy is improving the patient's functional use of the impaired limb in daily activity. The goal of this project is to develop a method for using accelerometers to monitor and quantify the amount of motion in the arm, for the application of monitoring limb use in stroke patients outside of therapy sessions.

Two analysis methods were designed. The first was based on the kinematics of the arm. The second was based on angular accelerations and the related forces applied to the shoulder and elbow joints. The two methods were tested on samples of different movement, which were chosen to represent the general motion of daily activities. The methods were tested to determine their accuracy at counting the number of movements that occurred, and their ability to produce activity values as an indication of the amplitude of the movements.

The two analysis methods which were developed can identify movement of the arm under the conditions which were tested. Thus, it appears that acceleration values can be processed to monitor and quantify arm motion. With future investigation into analyzing areas that were not tested by this project, these methods hold potential to be applied to using accelerometers to monitor arm use of patients while they are receiving rehabilitation therapy.

Thesis Supervisor: Hermano Igo Krebs

Title: Principal Research Scientist

Table of Contents

List of Figures

List of Tables

1.0 Introduction

1.1Motivation

The American Stroke Association estimates that there are about 4.8 million stroke survivors alive in the United States today[1]. One of the most common effects of a stroke is partial paralysis. A study done by the National Heart, Lung and Blood Institute found that 50% of stroke survivors suffer from some one-sided paralysis[1]. In the United States stroke is the leading cause of serious, long-term disability[1]. Partial paralysis affects the survivor's ability to care for themselves and can result in high direct and indirect costs (therapy, nurses, lost work). It is important that a stroke survivor go through rehabilitation therapy in order to regain at least partial use of their affected limb and thus regain independence and control in their life.

Current rehabilitation techniques include immobilization therapy (where the functional limb is restrained, forcing use of the affected limb), repetitive exercising of the limb and electrical stimulation, among others[2]. The physiological mechanisms through which these therapies cause a beneficial improvement in limb function are not well understood. However, these therapies have all been shown to be effective in some patients[3, 4, 5]. No matter which rehabilitation technique is used, it is helpful for the therapist to know whether the patient is making progress outside of therapy sessions. This knowledge can help the therapist fine-tune therapy techniques, or change to a new therapy if the current one is not contributing to the patient's progress.

The goal of this project is to develop a method for using accelerometers to monitor and quantify the amount of motion in the arm, for the application of monitoring limb use in stroke patients outside of therapy sessions. The method should be able to detect the number of movements that have occurred over the testing period and how much physical activity corresponds to the movements.

1.2 Background

The Newman Laboratory at MIT has been developing robots, which can aid in the rehabilitation of a paralyzed limb following a stroke. These robots use computer programs that involve the patient moving a cursor on the screen using their affected limb. The robots can help to physically guide the patient's limb, ensuring successful repetition of an exercise routine of the paralyzed limb. The robots track a patient's improvements in performing the required tasks, and have been shown to have a positive effect on promoting limb use. The development of a method to analyze limb movement outside of therapy sessions would provide complementary data to the data collected during therapy sessions. This data would also allow for investigation of whether an improvement during therapy generally corresponds to an improvement in the ability to perform normal daily tasks.

Accelerometers can be used to measure acceleration in three dimensions. Thus acceleration data can be analyzed and converted to quantify the amount that the limb has moved in space. Technological advances have made it possible to have very small accelerometers, which can wirelessly transmit data to a computer. Accelerometers can be easily attached to a limb and used to collect data on the motion of that limb. Thus, motion can now be quantitatively analyzed without being inhibited by wires or bulky electronics. Studies have been done by Uswatte et al[6] as well as Haeuber et al[7], which confirm that the use of accelerometers shows potential for use in this manner.

2.0 Determination of System Requirements

2.1 Measurement System

The "Flock of Birds" system, produced by Ascension Technology Corporation was used to collect position and orientation data. The Flock of Birds system is an electromagnetic tracking system, which can operate with as many as thirty sensors (referred to as birds)[8]. Position and orientation of the birds are determined by a pulsed DC magnetic field, which is sent from a transmitter and then measured by the birds in the flock. Based on the magnetic field characteristics of the signal received by the bird, it can calculate its position and orientation relative to the transmitter.

In its standard operating mode the Flock of Birds system can accurately locate birds within a range of four feet from the transmitter[8]. The Flock of Birds system is designed such that the bird closest to the transmitter controls the magnitude of the signal being transmitted, in order to prevent saturation of the bird's electronics[8]. This results in more noise in the bird further from the transmitter (smaller signal-to-noise ratio). When all of the birds are further than nine inches from the transmitter it transmits at full power and none of the birds' output will be compromised. When one bird moves within seven inches of the transmitter, the noise in the output from the farther bird will double[8]. Thus, the optimal operating range for all birds is between 9 inches and four feet from the transmitter.

Since the Flock of Birds system is an electromagnetic tracking system, the signal can be distorted in the presence of metal. Thus, the transmitter must be mounted on a non-metallic surface. The transmitter should also be at least twelve inches away from any metallic objects and computer monitors[8].

2.2 Measurement Rate

When running the Flock of Birds system along with Microsoft Windows, the maximum measurement rate drops from approximately 100 measurements/sec to approximately 20 measurements/sec. This is due to a large amount of overhead in the Windows operating system. The measurement is further reduced in proportion to the number of birds in the flock. The frequency of human motion is generally less than 5 Hz[9]. Since this method of analyzing human motion would be applied to a population that is undergoing rehabilitation therapy for paralysis due to a stroke, it will be assumed that their frequency of upper limb movement will be significantly less than 5Hz. Thus, the corresponding Nyquist frequency will be less than 10 Hz. Therefore, a sampling rate of 5 Hz or greater satisfies the Nyquist criterion. Based on this information one bird sampling at a frequency of 20 Hz, or 2 birds sampling at a frequency of 10Hz can be used for this application.

2.3 Number and Location of Sensors

In order to determine arm movement it is necessary to have at least two sensors. It is important to be able to separate motion of the arm from motion related to whole body movement such as walking, sitting or motion generated by exogenous agents such as a car or elevator. Thus, one sensor should be placed on a part of the body that is not affected by arm motion such as the shoulder or chest. The other sensors can be placed at any convenient location along the length of the arm. Because the sampling rate limits the number of sensors to two, one sensor was used as a reference point and one was placed on the arm.

Position data was collected at three points along the arm: the wrist, elbow and shoulder. These locations are shown in Figure 1. Analysis of the position output data indicated that changes in position are clearly recorded at the wrist and elbow locations. There is little change in position recorded by the bird at the shoulder location. Thus, the shoulder is an acceptable point to be used as the origin of a reference frame that removes general body movement. The shoulder position was chosen over a location on the chest because it is easier to access and demonstrated a low level of movement relative to the arm. The wrist was chosen as the best sensor location on the arm because movement at the wrist is a combination of both shoulder and elbow movement. Thus, this choice allows both motion at the shoulder and elbow to be estimated using only one sensor on the arm. In contrast, a sensor at the elbow would only record changes in position due to movement at the shoulder.

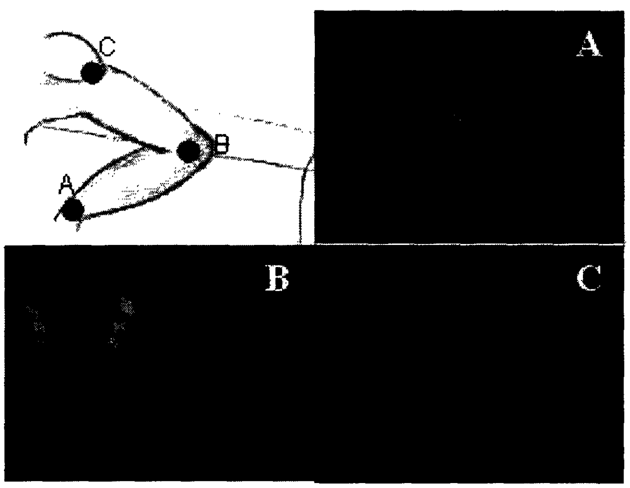


Figure 1: Locations of Three Points Considered for Bird Placement. Points at the wrist (A), elbow (B) and shoulder (C) were tested to determine the best points for bird placement. The shoulder and wrist locations were determined to be the optimal positions for all subsequent testing.

2.4 Conversion of Position Output to Acceleration

Since the methods developed were meant to be applicable to processing accelerometer output, it was necessary to convert the position data from the Flock of Birds to acceleration values. A Matlab script was written that converted the position values in each dimension to velocity components using differentiation combined with a low pass filter (filter described in section 2.5). The velocity components were differentiated a second time in the same manner to give acceleration components. Filtering caused up to ten points to be cut from the beginning and end of the data file. This loss of data corresponds to roughly one second of data. This was deemed to be an acceptable level of loss, provided the sensors are held at a constant position for at least one second at the beginning and end of data collection.

2.5 Filtering

During the process of differentiating the position values to acceleration components, it was necessary to filter the data so that noise could be removed. This would allow for easier processing of the data. When differentiating the position values, an interval that extended to either side of the point of interest by a certain number of data points was used to calculate the derivative at that point; essentially applying a moving average filter. Four different filters were developed with interval sizes of 11, 9, 7 and 5 points. These filters were analyzed to determine what their cutoff frequency was. This analysis involved performing a discrete Fourier transform using the PERIODOGRAM function in Matlab. The value used for the sampling frequency was 10 Hz. Equation 1 was used to calculate h, where M1 and M2 are the number of points to either side of the point of interest that the interval extends to[10].

$$h = \frac{1}{(M_1 + M_2 + 1)} \tag{1}$$

The cut-off frequency was approximated using the plot of the power spectral density that is output from the PERIODOGRAM function. The cut-off frequency was approximated as occurring at the frequency which corresponds to 3db below the first peak. The cut-off frequencies for the four filters are shown in Table 1.

	Interval Size (points)	**Cut-off Frequency (Hz)**
Filter 5	11	0.43
Filter 4	9	0.59
Filter 3	7	0.78
Filter 2	5	1.13

Table 1: Cut-off Frequencies. The corresponding cut-off frequencies for a moving average filter with the given interval size.

Other important characteristics of the power spectrum density plots, which are related to how the data is filtered, are the steepness with which the first peak drops off and the location of the second peak. A steep drop from the first peak indicates a lower cut-off frequency. The second peak will be 13 db below the first peak for all moving average filters[10]. The location of the second peak can indicate the frequency of any residual frequencies that may be present in the filtered signal. As shown in Figure 2, the eleven-point filter has a steep drop from the first peak resulting in a low cut-off frequency. However it most effectively removes higher frequencies because the second peak occurs at approximately 1.5 Hz, which is still below the range of frequencies which mostly contain noise. In comparison the five-point filter has a higher cut-off frequency, but has a second peak at between 3.5 and 4 Hz which can allow a small amount of noise to remain in the data. Thus, a filter with a higher cut-off frequency will have more noise from higher frequencies in the resulting signal, making it more difficult to analyze. A filter with a lower cut-off frequency may remove significant data, but also reduces noise effectively, thus allowing for easy analysis of the signal.

While the cut-off frequencies in Table 1 seem low compared to the average frequencies of human motion, it was assumed that for the motion being studied that, much of it would still be retained. A low cut-off frequency will cut out unwanted motion such as tremors such as those caused by Parkinson's disease. These tremors generally occur at frequencies between 4-5.3

Hz[11]. All four filters were used in the analysis of the movement data in order to compare their relative effectiveness at capturing the number of movements and activity of the arm. The related Matlab scripts for all of the filters tested can be found in Appendix C.

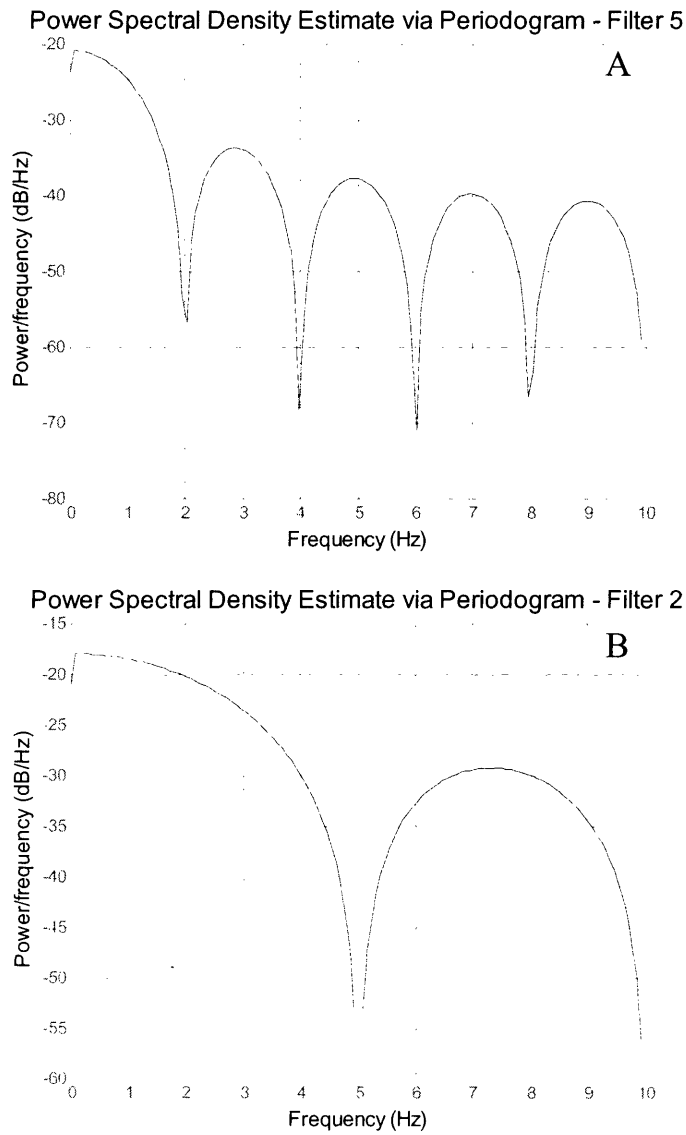


Figure 2: Power Spectral Density Plots for Filter 5 & Filter 2. (A) Filter 5 corresponds to an interval size of 11 points and has a cut-off frequency of 0.43 Hz. (B) Filter 2 corresponds to an interval size of 5 and a cut-off frequency of 1.13 Hz.

3.0 Analytical Methods

3.1 Method 1- Kinematic Approach

This method first integrates the acceleration components twice in order to calculate the velocity and position relative to the starting position of each bird. The distance between the shoulder and wrist sensor was calculated for each point in time. Changes in this distance were attributed to movement at the elbow. The change in the distance between the shoulder and elbow with respect to time was thus referred to as V_{elbow}. A movement at the elbow was defined as a pair of motions which change the distance between the shoulder and wrist and then return it to its original value. This correlates to a positive V_{elbow} followed by and negative V_{elbow}. Thus, all of the points where V_{elbow} crossed zero were found. Every movement will have two points where V_{elbow} is equal to zero, once when the forearm has been displaced and once when it returns to its original position. Thus, the number of elbow movements was defined as half of the number of points where V_{elbow} is equal to zero. This number was then rounded down to the nearest integer to account for the initial point where V_{elbow} is equal to zero at the beginning of data collection.

In addition to the number of movements made by the elbow, the magnitude of these movements is also helpful information in determining the level of activity at the elbow. To determine a value for the activity at the elbow, the corresponding position values were utilized. The change in the distance between the wrist and shoulder corresponding to the time when the velocity crossed zero and the previous crossing time was used as an indication of the magnitude of the movement. Thus, the activity level of the elbow is a measure of distance traveled by the wrist during all recorded elbow movements.

To determine movement attributed to motion at the shoulder, the different components of velocity which act on the wrist were examined. Figure 3 shows that the velocity at the wrist can be broken down into three vectors, velocity due to whole body movement (V_{body}), elbow movement (V_{elbow}) and shoulder movement ($V_{shoulder}$). Thus, $V_{shoulder}$ can be determined by subtracting the values for V_{elbow} and V_{body} from the total velocity at the wrist relative to the transmitter, as converted from the acceleration values. Since this process resulted in the magnitude of $V_{shoulder}$ it was not possible to use the zeros as a marker of movement. Thus, the points where the acceleration ($a_{shoulder}$) crossed zero were calculated and divided in half to give the number of shoulder movements. The changes in velocity at these points were used to calculate the level of activity of the shoulder movements. The Matlab script used to analyze the data for Method 1 can be found in Appendix A.

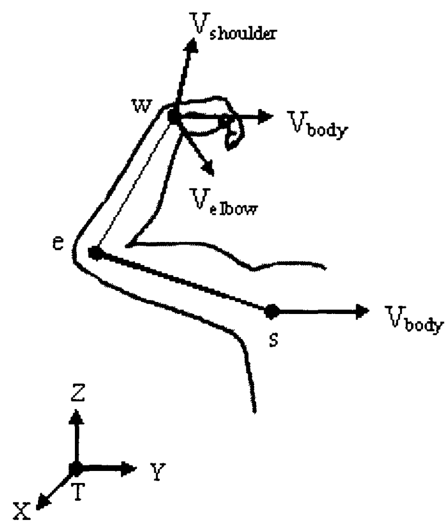


Figure 3: Diagram of the Velocity Vectors Acting on the Arm. The velocity at the wrist can be broken down into three vectors, the velocity due to whole body motion (V_{body}), elbow motion (V_{elbow}), and shoulder movement ($V_{shoulder}$). All velocities are relative to the reference frame of the transmitter (T).

3.2 Method 2 – Angular Acceleration & Forces

This method uses the angles recorded by the sensor on the shoulder as well as the acceleration values at the wrist and shoulder. These values are used to determine the relative force being exerted on the shoulder and elbow joints by the connecting muscles during movement. First the shoulder was made the origin of a new reference frame within the reference frame of the transmitter. The tangential acceleration at the elbow was calculated by finding the cross product of the components of the length of the arm segment between the shoulder and elbow (r_{se}) and the angular acceleration recorded at the shoulder. The resulting value is the acceleration of the elbow in the reference frame of the shoulder (a_{es}). Then, the elbow was modeled as the origin of a second reference frame which was rotating in relation to the reference frame of the shoulder. Equation 2 gives the acceleration of a point (P) in reference frame B, which is moving relative to the original reference frame A[12, 13]. The value for r is the distance from the origin of reference frame A to the origin of reference frame B.

$$a_{PA} = a_{PB} + a_{BA} + \alpha \times r + \omega \times (\omega \times r) \quad (2)$$

Equation 2 can be applied to the arm giving Equation 3, for the acceleration of the wrist relative to the reference frame of the elbow. The acceleration of the wrist relative to the shoulder (a_{ws}) is given by the difference in the acceleration of the wrist in the reference frame of the transmitter (a_{wT}) and the acceleration of the shoulder in the reference frame of the transmitter (a_{sT}). The length between the shoulder and elbow is given by r_{se}. The angular velocity (ω) and acceleration (α), refers to the angular velocity and acceleration of the shoulder-elbow segment of the arm.

$$a_{we} = (a_{wT} - a_{sT}) - [a_{es} + \alpha \times r_{se} + \omega \times (\omega \times r_{se})] \quad (3)$$

These accelerations can then be applied to force diagrams of the shoulder and elbow joints, as shown in Figure 4. If the upper arm and forearm are modeled as two rigid bars then it is possible to solve for the moment which is applied at each joint, which balances the forces being applied to the arm.

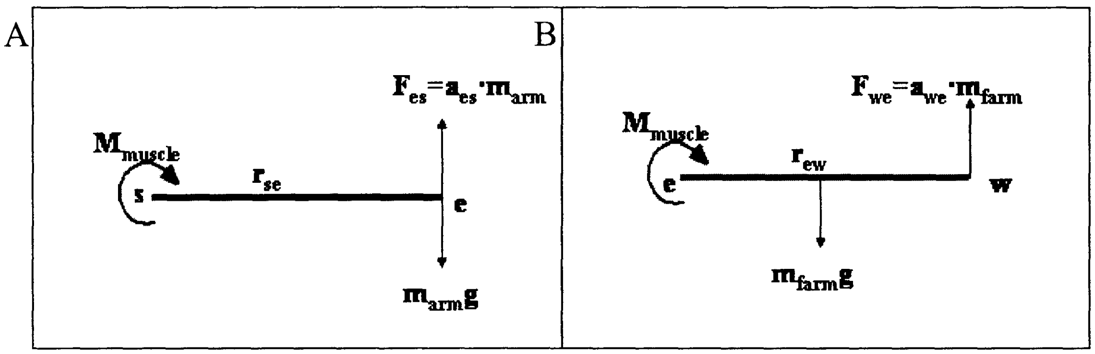


Figure 4: Force Diagrams of the Upper Arm and Forearm. (A) The moment at the shoulder can be determined from the force due to the mass of the entire arm (m_{arm}) applied at the elbow, and the force due to acceleration at the elbow (a_{es}). (B) The moment at the elbow can be determined from the mass of the forearm (m_{farm}) applied to the midpoint of the forearm, and the force due to the acceleration of the wrist relative to the elbow (a_{we}).

The magnitude of the moment at the shoulder was calculated at every point in time using the length of the upper arm and assuming that two forces are acting at the elbow, the force due to the acceleration of the elbow and the force due to the mass of the entire arm. To remove noise, moments of less than 0.25 N·m were removed from the data. One movement was defined as occurring at each peak moment value. To find these values, the change in moment with time was calculated. The points where these values crossed zero were marked as potential indicators of a movement. In order to remove multiple points that were due to the same movement, points that occurred within 0.3 seconds of each other were examined and the point corresponding to the larger moment value was kept. The time interval of 0.3 seconds was chosen because the average reaction time for a person to visually interpret information, react, and move is approximately 300 ms. Therefore, we assume that it is not possible for two or more movements to occur within the time span of 0.3 seconds. To further reduce noise, values that were less than the median peak moment value were also removed. Thus, the resulting value for the number of shoulder movements was the number of peaks in moment values; filtered to remove small movements, noise and repeat values for the same movement. The activity of the shoulder during movement was calculated by summing the moment values that corresponded with each movement.

The number of movements occurring at the elbow was calculated in a similar way. The magnitude of the moment occurring at the elbow joint was calculated at every point in time using the length of the forearm, the force due to acceleration at the wrist relative to the elbow, and the force due to the mass of the forearm. The peaks in the values of the moment magnitude were filtered in the same manner as described when determining the shoulder movement. Thus, the value for the number of elbow movements was the number of peaks in moment value at the elbow, filtered to remove noise, small movements and points that represented the same movement. The activity of the elbow was calculated by summing the moment values that corresponded with each movement at the elbow. The Matlab script used to analyze the data using Method 2 can be found in Appendix B.

4.0 Experimental Protocol

4.1 General Setup

The Flock of Birds system was configured to gather position and angle data for two birds. Data was collected at 9600 bauds and transmitted over an RS232 serial port. The birds were taped to the wrist and shoulder locations as shown in Figure 1 using first aid all-purpose cloth tape produced by Johnson & Johnson. The distance from the wrist sensor to the elbow joint and the distance from the shoulder sensor to the elbow joint were measured. The mass of the arm and forearm were calculated using a public domain program based on measurements of height and weight[14].

4.2 Types of Movements

Movements were chosen which were representative of general components of motion used in daily living. Eight different movements were tested as described in Table 2. Movements 7 and 8, tremor and rotating elbow respectively, were meant to test the limitations of the system. The tremor test was to determine whether the analysis method recorded movement for a non-functional tremor movement, such as would be common in a patient with Parkinson's disease. The stationary wrist test was to determine whether the analysis method could identify a movement during which neither the shoulder nor wrist were moving, but the elbow was. A few additional control tests were also done where the arm was kept completely still.

Combination movements, which combined an arm motion with general body motion, were also tested. These tests all used movement #5, touching the face, as described in Table 2. This arm motion was done in combination with sitting down in a chair, walking forward and back two steps in each direction, and twisting around an axis through the center of the body.

Movement	Description
#1. Forward Reach	Start with the elbow at the side of the body and extend arm straight out slightly lower than shoulder height, return to starting position
#2. Upward Reach	Start with the elbow at the side of the body with the wrist near the shoulder and extend arm straight up, return to starting position.
#3. Wiping Motion	With the elbow near the side of the body and the arm bent, rotate forearm from roughly 45° inside the frame of the body to 45° pointing away from the body, return to starting position.
#4. Sawing Motion	Start with the elbow bent and near the side of the body, move arm forward & back roughly 6 inches.
#5. Touching Face	Start with the elbow near the side of the body, arm bent, palm up, then touch the palm to the forehead, return to starting position.
#6. Lifting Using Forearm	Keeping the elbow stationary on a table, lift the forearm from parallel to the table to a 90° angle, return to starting position.
#7. Vibration	Whole arm was shaken at a small amplitude to simulate a tremor.
#8. Rotating Elbow	With the arm bend and keeping the wrist at a stationary point, the elbow was rotated from straight down to 90° pointing away from the body, return to starting position

Table 2: Description of Movements. Eight different movements were tested which were representative of general components of motion used in daily living.

4.3 Variation of Movement.

Each movement test was performed at three different speeds. The slow speed was slower than the tester would normally perform the motion, and had a frequency of movement less 0.2 Hz. The medium speed was the speed with which the tester was most comfortable performing the motion, and had a frequency of approximately 0.5 Hz. The fast speed was faster than the tester would normally perform the motion, and had a frequency of 1 Hz or higher.

Each movement was also performed in 5 different sets of repetitions. Each movement was done in sets of 10, 15, 20, 21, and 30 repetitions. The 21-repetition test was performed in order to determine whether the analysis methods could accurately identify the number of movements to the resolution of one movement.

Each combination of movement, speed and repetition number was tested five times. The vibration test was done twice at three different time lengths: 15, 30 and 45 seconds. The control test was done once at time segments of 30, 60, and 90 seconds. For the combination movements, each of the general body motions were tested twice at two different repetition lengths with no arm motion. Then the body motion was combined with arm motion, and tested for the same repetition lengths as the body motion alone.

5.0 Results

5.1 Effects of Filtering

All four of the filters were used when processing the movement files using each analysis method. The values for the average difference from the actual number of movements for each filter are shown in Table 3. For both methods, Filter 4, which had a total interval size of 9 points, gave the most accurate number of movements. The average difference of the values calculated using Filter 4 from the actual number of movements that occurred was 3.39 ± 0.21 movements using Method 1. Using Method 2, the average difference was 9.16 ± 0.41 movements. For Method 1, Filter 5 and Filter 3 had similar differences from the actual number of movements, making them the next most accurate filters. Using Method 2, Filter 5 was the second most accurate filter, but still had an average difference from the actual number of movements of 9.29 ± 0.47 movements. Filter 2 had the least accurate results for both methods.

	Method 1				Method 2			
	Filter 5	Filter 4	Filter 3	Filter 2	Filter 5	Filter 4	Filter 3	Filter 2
Average Difference	3.98	3.39	4.11	5.91	9.29	9.16	11.24	14.16
Standard Error (N =462)	0.25	0.21	0.24	0.37	0.47	0.41	0.75	1.03

Table 3: Comparison of four different filters when used by Method 1 and Method 2. Values include all types and frequencies of movement tested and all repetition numbers (10, 15, 20, 21 & 30 repetitions)

5.2 Effects of Frequency of Motion

Each movement and repetition number was tested at three different frequencies in order to determine what the frequency limits where for the analysis methods and filters. The average differences from the actual number of movements at each frequency are shown for both methods using Filter 4 in Table 4. The medium frequency gave the most accurate results when using Method 1. The average difference when motion occurred at medium frequencies was 1.77 ± 0.28 movements. When analyzing the motion with Method 2 the fast frequency gave the most accurate results, with an average difference of 7.52 ± 0.53 movements. The slow frequency gave the least accurate results when analyzed with both methods, with an average difference from the actual value of 4.35 ± 0.46 movements for Method 1 and 15.47 ± 1.33 movements for Method 2.

	Method 1			Method 2		
	Slow (<0.2 Hz)	Medium (~0.5 Hz)	Fast (> 1Hz)	Slow (<0.2 Hz)	Medium (~0.5 Hz)	Fast (> 1Hz)
Average Difference	4.35	1.77	4.10	15.47	9.76	7.52
Standard Error (N = 144)	0.46	0.28	0.36	1.33	0.61	0.53

Table 4: Comparison of Slow, Medium and Fast Frequencies. Accuracy of movement count when movement is occurring at slow, medium and fast frequencies when analyzed using Method 1 and Method 2. Comparison is with Filter 4 and includes all types of movements and repetition numbers (10, 15, 20, 21 & 30).

5.3 Movement Type

The type of movement being tested had an effect on the accuracy of each method in counting the number of movements that occurred. The accuracy of each method in identifying the number of movements that occurred is shown in Figure 5. Method 1 was the most accurate at identifying the number of movements that had occurred for the forward reach (movement #1), with an average difference from the actual number of 1.95 ±0.42 movements. The upward reach (movement #2) and touching the face (movement #5) movements were the second most accurately counted movements when analyzed with Method 1. Method 1 was least able to identify the number of movements that had occurred for the tremor (movement #7) and rotating elbow (movement #8) tests. For all movements tested, Method 1 was more accurate at identifying the number of movements that occurred at the elbow than at the shoulder ($p < 0.025$, Student's t-test). For example, the sawing movement had an average difference from the actual number of movements recorded at the elbow of 1.76 ± 0.44 movements, compared to 4.78 ± 0.61 movements recorded at the shoulder.

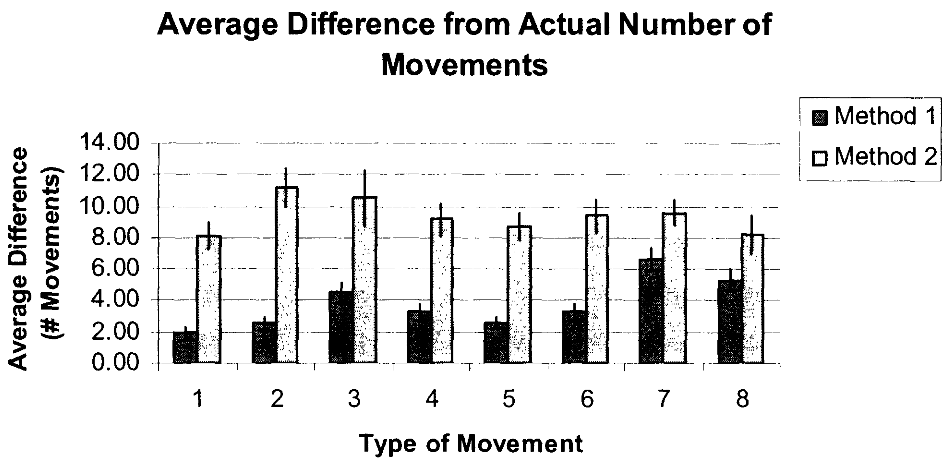


Figure 5: Average Difference from the Actual Number of Movements by Movement Type. Each movement was analyzed using Method 1 and Method 2, results shown are using Filter 4 and include all frequencies of movement and repetition numbers (10, 15, 20, 21 & 30). Error bars represent standard error.

When movement was analyzed using Method 2, the forward reach movement and the touching the face movement were the most accurately counted. Method 2 had an accuracy of 8.09 ± 0.85 movements for the forward reach movement and 8.70 ± 0.89 movements for the touching the face movement. Method 2 was least able to count the movements for the upward reach and wiping (movement #3) movements. The upward reach had an accuracy of 11.14 ± 1.20 movements.

Overall, both methods were most accurately able to count the number of movements that occurred for the forward reach and touching the face movements. They both were also able to identify moderately well the number of movements that occurred in the sawing and lifting using the forearm movements.

5.4 Activity Values

Activity values were divided by the number of recorded movements to examine whether their values could be used to indicate relative use of each joint during the movement. The elbow activity values yield a clear correlation with the amplitude of each movement tested. If the average displacement of the wrist during each movement is considered the elbow activity values seem like a reasonable estimate of this value. For example, in the experimental protocol for performing the sawing movement (movement #4), it called for a movement of roughly six inches. The average activity value for movement #4 was 6.0 ± 0.45 inches/movement. Thus, this is a very close approximation of the amplitude of this movement. The shoulder values for activity did not seem to give any clear trends relating the mount of shoulder involvement in any given movement. This is most likely due to the fact that velocity values were used for the calculation of shoulder activity. Changing the analysis method to compute the activity value for the shoulder in terms of displacement, may indicate a more clearly identifiable trend. Performing the same analysis on the results from Method 2 yielded no clear trends or indications of significance associated with any of the activity values.

	Movement							
	1	2	3	4	5	6	7	8
More Elbow or Shoulder Movement	Elbow	Equal	Shoulder	Equal	Equal	Elbow	Shoulder	Neither
Ellbow Average Activity (in/movement)	12.3 +/- 0.52	14.6+/- 0.83	3.6+/- 0.49	6.0+/- 0.45	11.3+/ 0.86	11.1+/- 0.68	1.1+/- 0.12	.5+/- 0.33
Shoulder Average Activity (in/s·movement)	5.3 +/- 0.33	8.2+/- 0.33	7.6+/- 0.29	3.9+/- 0.15	9.1+/- 0.52	6.0+/- 0.29	2.3+/- 0.10	1.1+/- 0.23

Table 5: Values for Activity for Each Movement Tested. All activity values calculated by analysis with Method 1 and Filter 4, these values include all frequencies and repetition numbers.

5.5 Combination Movements

Combinations of whole body motion and arm movement #5 (touching the face) were tested to see if the analysis methods could accurately separate the whole body motion from the motion of the arm. The results of Method 1 are shown in Table 6. Method 1 was better able to subtract out the whole body motion when arm motion was occurring simultaneously. On average, when only body motion was occurring Method 1 gave 4.4 ± 2.08 false elbow movements and 4.5 ± 1.36 false shoulder movements. However, when arm motion was performed simultaneously with the body motion, the average difference from the actual number of movements was reduced to 0.56 ± 0.18 elbow movements and 1.56 ± 0.37 shoulder movements. Thus, it appears that when arm motion is absent Method 1 counts noise and small errors in the subtraction of the body motion as movements. However, in the presence of arm motion the magnitude of the arm motion is large enough that these smaller values get filtered out. The average value for the elbow activity for the body motion and arm movement tests was 7.37 ± 1.51 inches/movement, which is within the range of the value for the elbow activity found when analyzing movement #5 alone.

	Average Difference from Actual Number of Movements (for repetition sets of 2, 4, 8, 10 & 20 movements)	
Body Motion	**Elbow**	**Shoulder**
Sitting	8.00	9.00
Walking	3.00	1.50
Twisting	0.00	1.50
Average Difference	**4.40 ± 2.08**	**4.5 ± 1.36**
Sitting + Movement	1.00	2.33
Walking + movement	1.00	1.50
Twisting + Movement	0.00	1.00
Average Difference	**0.56 ±0.18**	**1.56 ± 0.37**

Table 6: Average Difference from Actual Number of Movements when Combined Motions are Analyzed with Method 1. Accuracy of Method 1 in counting the number of arm movements that occurred when combined with sitting, walking and twisting, or when these body motions are performed without arm motion.

The results of analyzing combination movements with Method 2 are shown in Table 7. As with Method 1, these results show that Method 2 is more accurate when body motion is combined with arm motion, than when body motion alone is analyzed. When body motion alone is analyzed, Method 2 records on average 5.7 ± 1.54 non-functional movements for the elbow and 4.0 ± 1.11 non-functional movements for the shoulder. When arm movement is included the average difference from the actual number of movements is 3.7 ± 1.62 elbow movements and 2.44 ± 0.67 shoulder movements. The activity values determined by Method 1 were not found to have meaningful significance. Therefore, no conclusions could be gathered from the activity values for the combination movements using Method 2.

	Average Difference from Actual Number of Movements (for repetition sets of 2, 4, 8, 10 & 20 movements)	
Body Motion	**Elbow**	**Shoulder**
Sitting	11.00	5.75
Walking	1.75	2.75
Twisting	3.00	3.00
Average Difference	**5.7 ± 1.54**	**4.0 ± 1.11**
Sitting + Movement	9.33	3.67
Walking + movement	1.25	1.75
Twisting + Movement	0.00	2.00
Average Difference	**3.7 ± 1.62**	**2.44 ± 0.67**

Table 7: Average Difference from Actual Number of Movements when Combined Motions are Analyzed with Method 2. Accuracy of Method 2 in counting the number of arm movements that occurred when combined with sitting, walking and twisting, or when these body motions are performed without arm motion.

6.0 Discussion

6.1 Testing Parameters

When used with Method 1 and Method 2, Filter 4 was found to be the most accurate filter at counting the number of arm movements that occurred during a testing period. This result corresponds to expectations based on the qualities of the moving average filter used for this test. The results of filtering are a compromise between how low the cut-off frequency is and how well the filter removes higher frequencies. Therefore, it is expected that a filter in the middle of the range of those tested would be the most accurate. Filter 5, which had the lowest cut-off frequency, was the second most accurate filter. This indicates that for the frequencies tested, removal of higher frequency noise is more important than a higher cut-off frequency.

Method 1 was most accurate at counting the number of movements when tests were performed at the medium frequency (~0.5 Hz). Method 2 gave the most accurate values when the tests were performed at the fast frequency (> 1 Hz). In both cases, the tests performed at the slowest frequency (<0.2 Hz) were the least accurate at counting the number of movements that had occurred. This was surprising as it was expected that faster movement would be filtered out of the data due to the relatively low cut-off frequencies of the filters. However, this shows that quicker movements are not removed by the filters and can be identified by the analysis methods.

The reason that the slowest movements resulted in the least accurate movement counts may be due to small changes in position near the points of maximum displacement during the course of movement. One movement at the slow frequency would contain more sampling points than at the two faster frequencies. Thus, it is more likely that the system would record these small inconsistencies in position. These small inconsistencies could be then counted as additional movements, leading to an over-prediction of the number movements. When applying Method 1 to all movements tested at the slow frequency, only 5.6% of elbow and 9.0% of shoulder values were lower than the actual value. In comparison, 6.2% of elbow and 47.0% of shoulder values were lower than the actual value when the tests were performed at the medium frequency. Thus, it appears that at slow frequencies there is an over-estimation of the number of movements.

6.2 Method 1

Method 1 was able to count the number of movements that occurred during a testing period with an overall accuracy of 3.39 ± 0.21. This method was shown to work best when movements occurred at a frequency of approximately 0.5 Hz. This is a convenient frequency at which to have optimal accuracy, as it is the frequency at which the tester would normally have performed the movements.

Method 1 was most accurate at counting motions such as the forward reach, upward reach and touching the face movements. It had moderate results in counting movements such as sawing, wiping and lifting using the forearm. Method 1 was not able to accurately remove tremor motions and it could not correctly identify movements that involved a rotation at the shoulder where the wrist is held stationary. These results make sense based on how motion is calculated using Method 1. The motions that were the most accurately counted were motions that involved at least equal elbow motion in relation to shoulder motion. Movements such as wiping or sawing have limited elbow motion relative to shoulder movement. The lifting with the forearm motion was placed in this moderately accurate group based on its overall ability to accurately identify movements. However, since it is entirely elbow motion one would expect it to be in the category of the most accurately measured movements. If the elbow and shoulder accuracies are separated, the value for elbow movement had an average difference of 1.48 ± 0.33 from the actual number of movements, as compared to the overall value of 3.29 ± 0.58. Thus, Method 1 was able to accurately count the number of elbow movements that occurred, but the overall value can be compromised by less accurate shoulder values. Method 1 has no mechanism to measure movements such as the rotating elbow movement. Method 1 is entirely based on accelerations of the wrist and shoulder. For movements such as movement #7 where neither point is moving, Method 1 will not be able to correctly identify motion.

For every movement tested, the number of movements calculated at the elbow was more accurate than the shoulder value. This is most likely due to the fact that the elbow value for the number of movements is solely based on the change in distance between the shoulder and wrist. Thus, only errors in this calculation will affect the elbow value. The shoulder value is the velocity at the wrist after V_{elbow} and V_{body} have been removed. Thus, $V_{shoulder}$ will be affected by errors in both V_{elbow} and V_{body} as well as any noise in the over all data set.

Method 1 was able to count movements that occurred simultaneously with whole body motion with an accuracy of 0.56 ± 0.18 movements at the elbow and 1.56 ± 0.37 movements at the elbow. However, when no arm motion was occurring, Method 1 had a higher error. This could be due to small errors in the subtraction of velocities and noise. These errors could be counted as movements when there are no larger magnitude arm movements occurring simultaneously. The false positive movements that are recorded in this case all had a low activity value. Thus, with a better method for filtering noise out of the final data, or by determining a cut-off point for activity per movement below which the movement is not counted, these false movements could be eliminated.

6.3 Method 2

Method 2 was able to calculate the number of movements with an overall accuracy of 9.16 ± 0.41 movements using Filter 4. Method 2 was able to most accurately count the number of movements at frequencies greater than 1 Hz. This frequency is faster than the tester would have normally performed the movements. The accuracy at the medium frequency was only slightly lower, so Method 2 could still be used for people who are moving at relatively normal frequencies. However, if Method 2 was used with a person who has difficulties performing movements at a normal speed, the accuracy of the number of movements would be compromised.

Method 2 was most accurate at counting the number of movements for the forward reach and touching the face movements. It was able to count the number of movements moderately well for motions such as sawing, lifting with the forearm, rotating the elbow and the tremor test. Method 2 was not able to accurately identify the number of movements for the upward reach and wiping motion. Method 2 is based mainly on the change in angle at the shoulder and the relationship between that value and the acceleration at the wrist. Thus, it is reasonable that Method 2 would be most accurate when the shoulder motion is greater than or equal to the elbow motion. This is the case with the forward reach and touching the face movements (movements #1 and 5). Method 2 is able to identify motion occurring when the elbow is rotated and the wrist is kept stationary because the change in the angle of the shoulder is measured by Method 2. Method 2 is unable to accurately count movement when the movement occurs around a vertical axis which goes through the center of the upper arm and shoulder, such as the wiping motion. In this case, the angle recorded by the sensor is not changing enough to register the movement at the shoulder. Method 2 also had a low accuracy identifying the upward reach movement (movement #2). This may be due to the shoulder and elbow joint opening in the same plane in a compound motion where that the acceleration of the wrist could be the same regardless of whether the arm was straight or bent.

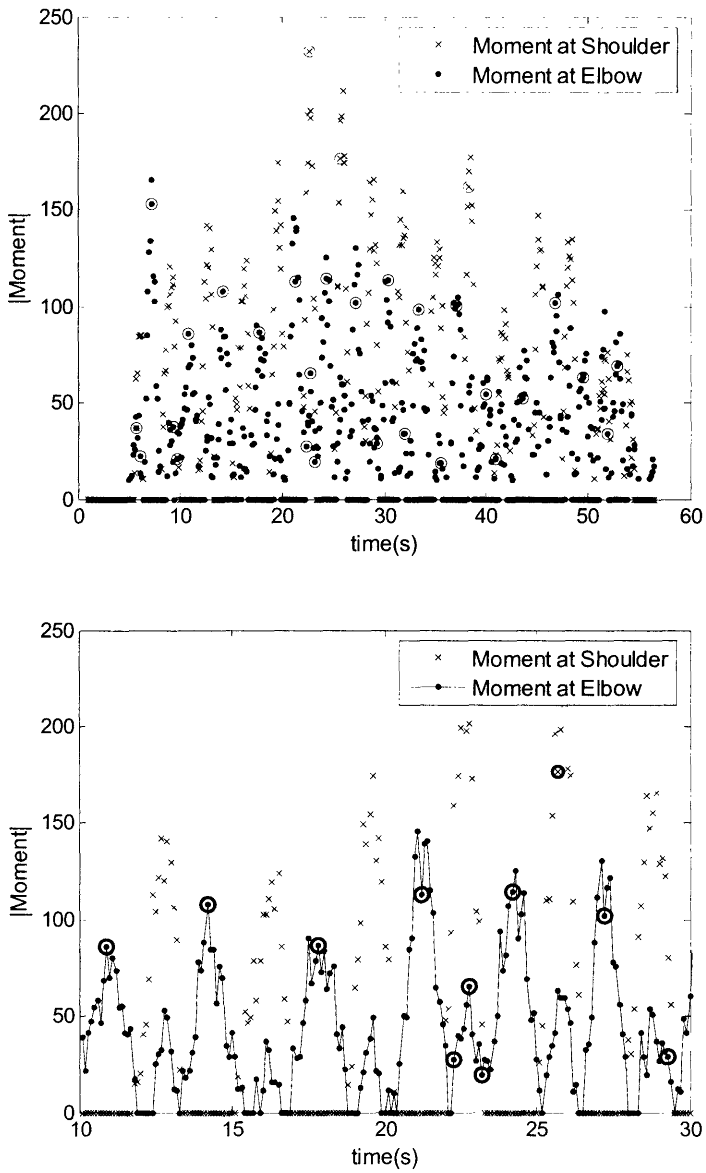
When movements occurred simultaneously with whole body motion, Method 2 had an accuracy of 3.7 ± 1.62 movements for counting elbow movements and 2.44 ± 0.67 movements for counting shoulder movements. When body motion was tested without any arm motion, Method 2 recorded an average of 5.7 ± 1.54 false elbow movements and 4.0 ± 1.11 false shoulder movements. The explanation for this is most likely the same as given for the same problem with Method 1.

The level of accuracy using Method 2 was much lower than expected. In reviewing the different steps of the analysis it seems that the problem could be mostly due to the final step of finding the peaks in the moment values. Before the filtering process to remove small movements, noise and repeated values for the same movements, there are many points which are indicated as points where the change in moment with time crosses zero. However, after the filtering step the number of points that are actually given as the number of movements is much lower than the actual number of movements. Figure 6 shows an example plot of the magnitude of the moment at the elbow and at the shoulder. Points that are being counted towards the number of movements are circled. When the number of peaks for each movement is counted by eye, the number generally corresponds to the number of movements that occurred. However, the number of circled points does not correspond with the number of movements that occurred. In this case, only 3 points are being recorded as markers of a shoulder movement out of the 15 movements that actually occurred.

Before filtering, there were 21 points flagged as points that could be peaks in magnitude of the moment at the shoulder. The final elbow movement value given is 27 movements, indicating that more points are being counted as movements than actually occurred. A visual examination of the peaks of elbow moment shows that more points are being indicated as points of movement than actually occurred. Thus, it appears that the largest source of error in Method 2 is due the filtering of the points that takes place at the end of the analysis process.

If the proper points are found to indicate when a movement has occurred, the values for activity should also be fixed. No meaningful trend was found in the activity values produced using Method 2 during these trials. However, since the process for determining the activity is based upon the summation of the moment values at all points where movement occurred this is a result of not properly counting movements. If Method 2 is able to accurately find the points which correspond to the maximum moment values, the activity values should properly correspond to the relative amplitude of the movement that occurred.

A



B

Figure 6: Example Plots of the Moment Values at the Elbow and Shoulder as Calculated Using Method 2. (A) Elbow values are indication in red and the shoulder values are indicated in blue. The actual number of movements is 15; Method 2 gave values of 3 shoulder movements and 27 elbow movements. Points being counted towards the movement count are circled. (B)A zoomed in view of the same plot showing the inappropriate indication of moment peaks.

6.4 Comparison of Method 1 and Method 2

The results of the testing of each method show that for all movements, at all frequencies of motion, Method 1 was able to more accurately identify the number of movements that occurred. Method 1 was also able to produce activity values that could be used to determine the relative involvement of the elbow during each movement. Therefore, the results of these tests indicate that Method 1 is a more accurate method for collecting information regarding the number and relative size of movement of the arm. However, the inaccuracy of Method 2 is hypothesized to be due to the filtering at the end of the analysis process. Thus, there is the possibility that if this issue is remedied Method 2 could be shown to reliably count movements.

Method 1 was able to more accurately predict the number of movements that occurred at the elbow than at the shoulder. This is because in Method 1 the movement at the elbow is determined independently of any information other than the change in the distance between the shoulder and wrist. Method 1 has limitations when analyzing motion that has no acceleration at the wrist, but movement at the shoulder; such as rotating the elbow with the wrist stationary.

Method 2 was able to more accurately predict the number of movements that occurred at the shoulder than at the elbow. This is due to the fact that Method 2 calculates the movement at the shoulder based on the changing angles of the shoulder. Therefore Method 2 was able to identify movement such as the rotation of the elbow with the wrist stationary. Method 2 was also more accurate at not reporting any functional movement during the tremor test.

If the accuracy of Method 2 was able to be improved with a different method of filtering, it may be advantageous to combine the two methods. The elbow movement calculation process of Method 1 could be combined with the shoulder movement calculation process of Method 2. This would combine the strengths of each analysis method and increase the range of motion that could be identified.

7.0 Conclusion

7.1 Accomplishments

The results of the testing of the two methods produced for this project show that using acceleration values shows promise in the development of a method to analyze motion of the arm. The filters were shown to be effective in cutting out noise without removing functional movement. The cut-off frequencies were shown to be at a frequency that can still allow normal human motion to be tracked by each analysis method. Both methods were able to capture movement that occurred at the comfortable frequencies of human movement. These results also show that even a system with a relatively low sampling frequency can be used to collect data on human movement.

7.2 Limitations

This study was limited in that it was not performed using accelerometers. While the preliminary results from this experiment show the possibility for these methods to be applied to accelerometer data, there will be slightly different issues present in the accelerometer data, which need to be taken into consideration when converting between the two measurement systems. One example of a difference between the position tracking output and an accelerometer's output is that the accelerometer will always include a gravitational acceleration, which will need to be taken in to account during analysis.

The filtering method utilized in Method 2 shows indications of being the source of the larger error in Method 2 in comparison to Method 1. Thus, results of this project cannot completely assess the accuracy of the background theory upon which Method 2 is based. An improved filtering method would quickly remedy this issue and allow for a more accurate assessment of the accuracy of Method 2. This would also allow for a new analysis into whether the method of calculating the activity value of the motion is an accurate way of producing a value, which can be used to determine the relative motion of the shoulder or elbow.

The number of tests performed on combination movement was much smaller than that of the single movement tests. Thus, the results of this project for determining whether these methods can remove the motion of the entire body are limited in their ability to show clear trends. Also, data collection did not include combinations of different types of arm movement. Human motion is not just one movement repeated at a time, but rather a combination of different types of movements, which occur with differing displacements and at different frequencies. Thus, these results cannot be shown to accurately identify arm movement when different types of arm motion are combined into one data file.

The analysis methods are limited in their ability to identify arm movement and remove body movement due to simplifications that were applied in developing the methods based on the supported physics. Method 1 does not take into account movement such as rotation of the body around a vertical axis through the center of the body. This sort of movement would produce an additional component of acceleration at the shoulder and wrist due to the angular acceleration of these points around this centrally located axis. Also, the method for finding the activity values at the shoulder in Method 1, using the points where acceleration crossed zero, was not correct. This method must be improved upon in order to give meaningful values for the activity of the shoulder during motion.

In Method 2, when analyzing the wrist motion relative to the elbow, the elbow is placed at the origin of a reference frame which rotates with respect to the reference frame of the shoulder. In this case the Coriolis acceleration of the wrist should also be included into Equation 3, resulting in Equation 4[12, 13].

$$a_{we} = (a_{wT} - a_{sT}) - [a_{es} + \alpha \times r_{se} + \omega \times (\omega \times r_{se})] - (2\omega \times v_{we}) \qquad (4)$$

Because the relative velocity of the wrist in relation to the elbow was not known, this component of the acceleration was removed from the equation for simplification.

Method 2 utilizes the values for the angular orientation of the shoulder sensor when calculating the arm movement values. An accelerometer does not have the ability to calculate angles.

Therefore, Method 2 would require an additional sensor to be combined with an accelerometer in order to be properly utilized. A gyroscope could be utilized for this purpose. Utilizing MEMS technology gyroscopes can now be manufactured at very small sizes (less than 0.15 cc and less than 0.5 g[15]). Thus, this additional sensor should not increase the size of the sensor, although it may increase the cost.

7.3 Future Directions

In further developing these methods for their application to analyzing arm movement using accelerometers, the first step should be in changing the filtering method used in Method 2. This will allow for a better understanding of the ability of Method 2 to identify and quantify arm movement. Thus, a more accurate assessment of Method 2's accuracy could be made. This could also allow for the consideration of whether it is feasible or desirable to combine the shoulder movement calculation of Method 2 with the elbow movement calculation of Method 1.

More testing should be performed, especially on motion that contains whole body motion and multiple different types of arm motion. This will allow for better judgment of each method's relative ability to accurately count the number of arm movements within realistic motion. The source of the false positives that are counted by each method when body motion is occurring without simultaneous arm motion should be analyzed. Further analysis of this situation may result in the utilization of an improved method of filtering small movements and noise. Another possibility is the determination of a cut-off point for activity per movement, below which the system doesn't not count a movement towards the total number of movements. Further testing could lead to small changes in the analysis methods which may improve each method's level of accuracy.

Finally these methods should be tested using accelerometers. The accelerometers may cause initial problems in the conversion of the methods to dealing directly with an acceleration output. The use of the accelerometer will remove the need to use the filters which were tested in this project, although it may also require a new method for initially filtering the data. Thus, it is possible that using the accelerometers may also reduce some of the inaccuracies of these methods due to the way the position output from the Flock of Birds system was processed. If a gyroscope is combined with an accelerometer in order to use Method 2, the output of these two sensors will have to be combined in an efficient manner to facilitate data collection and processing.

Once these methods have been adapted for use with accelerometers, the next step is to test them on the patient populations for which they were designed. Tests on people who are currently receiving rehabilitation therapy after a stroke may show limitations of these methods when applied to this population versus a healthy population with a full range of arm movement. Thus, further modification of these methods may be necessary to adapt to the analysis of these patients. Also, future thought could be applied to determining the best way to package the sensors and attach them to the arm. The design of the final sensor system will need to be comfortably worn by the people being tested, durable and not inhibit motion.

The results of this project show that the two analysis methods which were developed can identify movement of the arm under the simplified conditions which were tested. Thus, it appears that acceleration values can be processed to monitor and quantify arm motion. With future investigation into solving the limitations described above, these methods hold potential to being applied to using accelerometers to monitor arm use of patients with partial paralysis of an arm while they are receiving rehabilitation therapy.

References

[1] American Heart Association. *Heart Disease and Stroke Statistics — 2005 Update*. Dallas, Tex.: American Heart Association; 2004.

[2] Krebs, H.I., "Robot-Aided Neurorehabilitation," *IEEE Trans Rehabil Eng.,* vol. 6, pp. 75-87, 1998.

[3] Butefisch, C. et al, "Repetitive Training of Isolated Movements Improves the Outcome of Motor Rehabilitation of the Centrally Paretic Hand," *J. Neurologic. Sci*, vol. 130, pp 59-68, 1995.

[4] Taub, E., "Technique to Improve Chronic Motor Deficit After Stroke," *Arch. Phys. Med. Rehab.*, vol. 74, pp.347-354, 1993.

[5] Smith, L.E., "Restoration of Volitional Limb Movement of Hemiplegics Following Patterned Functional Electrical Stimulations," *Percep. Motor Skills*, vol. 71, pp. 1393, 1983.

[6] Uswatte G., et al, "Objective Measurement of Functional Upper-Extremity Movement Using Accelerometer Recordings Transformed with a Threshold Filter," Stroke, vol. 31, pp. 662-667, 2000.

[7] Haeuber, E., "Accelerometer monitoring of home- and community-based ambulatory activity after stroke," *Arch Phys Med Rehabil.,* vol. 85, pp. 1997-2001, 2004.

[8] AcensionTechnology Corpportation., "The Flock of Birds; Installation and Operation Guide," 1998.

[9] Angeloni, Cesare, Patrick O. Reily, and David E. Krebs. "Frequency Content of Whole Body Gait Kinematic Data". IEEE Transactions on Rehabilitation Engineering, Vol. 2, No. 1, 1994.

[10] Oppenheim, Alan V. and Ronald W. Schafer. Discrete – Time Signal Processing. Prentice Hall; New Jersey, 1999.

[11] Findley LJ, Gresty MA, Halmagyi Gm. "Tremor, the cogwheel phenomenon and clonus in Parkinson's disease." *Journal of Neurology, Neurosurgery and Psychiatr,* vol 44, 1981, pp.534-46.

[12] Boresi, Arthur P. and Richard J. Schmidt, Engineering Mechanics: Dynamics. Brookes/Cole; California, 2001: pp 439-441.

[13] Bedford, Anthony and Wallace Fowler, Engineering Mechanics Dynamics, Addison-Wesley; Massachusetts, 1999: pp459-461.

[14] http://www.motco.dir.bg/Data/MassInertial.html

[15] Analog Devices. "iMEMS Gyroscopes," www.analog.com.

Appendices

Appendix A - Matlab Files for Method 1

To calculate number of movements and activity values for the entire arm:

ALL_ARM_MOTION

```
%to find relative velocity of wrist to shoulder
%add these when running more than one data file
%clear A* P* S* C* Z* a adj am* M* R b c cr* d* e* fd* f find_ones g h* i j
k* l*
%clear m* n numb* o p* q r* s* t* v* w* z*

all_elbow_motion

%subract general body motion
rel_vel_x = Zi1x-Zi2x; rel_vel_y = Zi1y-Zi2y; rel_vel_z = Zi1z- Zi2z;
rel_vel = [rel_vel_x' rel_vel_y' rel_vel_z'];
vel_vector = sqrt((rel_vel_x).^2+(rel_vel_y).^2+(rel_vel_z).^2);

magnitude_velchangedist = abs(velchangedist);
[v w] = size(vel_vector);
vel_vector_adj = vel_vector(:, 1:(w-1));
shoulder_only_velocity = vel_vector_adj - magnitude_velchangedist;

for i = [2:1:(w-2)]
    risev(i-1) = (shoulder_only_velocity(i+1)-shoulder_only_velocity(i-1));
end

second_derivative = risev/.2;
[p q] = size(second_derivative); r = q/10;

for i = [1:1:(q-1)];
    k2 = [second_derivative(i) second_derivative(i+1)];
    time2 = [time4(i) time4(i+1)];
    p2 = polyfit(time2, k2, 1); r2 = roots(p2);

        if r2>time4(i) & r2<time4(i+1);
        R=1;
        else R=0;
        end

    rootZ(i) = R;
end

crossings = find(rootZ==1); find_ones=crossings;
[length, width]=size(find_ones); number_of_ones=width;

if number_of_ones<1
    shoulder_motion = 0
    total_shoulder_motion = 0
```

```
else
how_many = [1:1:number_of_ones]; crossing_times = time4(crossings);
amount_of_velocity = shoulder_only_velocity(crossings);

Meanv = mean(amount_of_velocity); [c d] = size(amount_of_velocity);
padded_v = [1 1 amount_of_velocity 1 1]; [q w] = size(time4);
crossing_times_v_adj = [time4(1) time4(2) crossing_times time4(w-1)
time4(w)];

for i = [3:1:d+2]
    if padded_v(i)<(Meanv*1.1)
        S=0;
    elseif (crossing_times_v_adj(i+1)-crossing_times_v_adj(i))>=.3 &
(crossing_times_v_adj(i)-crossing_times_v_adj(i-1))>=.3
        S=1;
    elseif (crossing_times_v_adj(i+2)-crossing_times_v_adj(i))<=.3 &
padded_v(i)<padded_v(i+2)
        S=0;
    elseif (crossing_times_v_adj(i)-crossing_times_v_adj(i-2))<=.3 &
padded_v(i)<padded_v(i-2)
           S = 0;
    elseif (crossing_times_v_adj(i+1)-crossing_times_v_adj(i))<=.3 &
padded_v(i)<padded_v(i+1)
        S = 0;
    elseif (crossing_times_v_adj(i-1)-crossing_times_v_adj(i))<=.3 &
padded_v(i)>padded_v(i-1)
        S=1;
    else S=0;
    end
    real_peakv(i-2)=S ;
end

true_maxv = find(real_peakv==1); find_ones=true_maxv;
[length, width]=size(find_ones); number_of_ones_v=width;
crossing_times_v = time4(crossings(true_maxv));
[o p] = size(crossing_times_v);
number_shoulder_movements_p = floor(p/2);
shoulder_motion = shoulder_only_velocity(crossings(true_maxv)+5);
total_shoulder_motion_p = sum(shoulder_motion);

if total_shoulder_motion_p/number_shoulder_movements_p < 1;
   total_shoulder_motion = 0; number_shoulder_movements = 0;
else
   total_shoulder_motion = total_shoulder_motion_p;
   number_shoulder_movements = number_shoulder_movements_p;
end
end
```

To calculate the number of movements at activity values for the elbow:

ALL_ELBOW_MOTION

```
% subtracts out general body motion in all dimensions, then determines elbow
motion

%change this when running more than 1 data file
```

```
%A = load(current_file);

clear all
file = input('Select file: ', 's')
A = load(file);

All = sortrows(A, 1); [j l]=size(All);
sensor1 = All(1:1:(j/2), :); sensor2 = All((j/2+1):1:j, :);
sensor1x = sensor1(:, 2); sensor1y = sensor1(:, 3);
sensor1z = sensor1(:, 4);
Sensor1 = [sensor1x sensor1y sensor1z];
sensor2x = sensor2(:, 2); sensor2y = sensor2(:, 3);
sensor2z = sensor2(:, 4);
Sensor2 = [sensor2x sensor2y sensor2z];

[m, n] = size(sensor1);
k = m/10;
t = [0.1:0.1:k];
filter = input('choose filter: ','s');
run(filter)

%to integrate to position
Z1x=cumtrapz(t2new, sd1x); Zi1x=Z1x+fd1x(6);
P1x=cumtrapz(t2new, Zi1x); Pi1x=P1x+sensor1x(11);

Z2x=cumtrapz(t2new, sd2x); Zi2x=Z2x+fd2x(6);
P2x=cumtrapz(t2new, Zi2x); Pi2x=P2x+sensor2x(11);

Z1y=cumtrapz(t2new, sd1y); Zi1y=Z1y+fd1y(6);
P1y=cumtrapz(t2new, Zi1y); Pi1y=P1y+sensor1y(11);

Z2y=cumtrapz(t2new, sd2y); Zi2y=Z2y+fd2y(6);
P2y=cumtrapz(t2new, Zi2y); Pi2y=P2y+sensor2y(11);

Z1z=cumtrapz(t2new, sd1z); Zi1z=Z1z+fd1z(6);
P1z=cumtrapz(t2new, Zi1z); Pi1z=P1z+sensor1z(11);

Z2z=cumtrapz(t2new, sd2z); Zi2z=Z2z+fd2z(6);
P2z=cumtrapz(t2new, Zi2z); Pi2z=P2z+sensor2z(11);

Pi1x=Pi1x'; Pi1y = Pi1y'; Pi1z = Pi1z';
Pi2x=Pi2x'; Pi2y = Pi2y'; Pi2z = Pi2z';
Sensor1new = [Pi1x Pi1y Pi1z]; Sensor2new = [Pi2x Pi2y Pi2z];

dist = sqrt((Pi1x-Pi2x).^2+(Pi1y-Pi2y).^2+(Pi1z-Pi2z).^2);
[f d] = size(dist);

for i = [1:1:f-1];
    Changedist(i) = dist(i+1)-dist(i);
end

velchangedist = Changedist/0.2;

for i = [1:1:f-2];
```

```
    d = [velchangedist(i) velchangedist(i+1)];
    time = [t3new(i) t3new(i+1)];
    p = polyfit(time, d, 1);
    r = roots(p);
        if r>t3new(i) & r<t3new(i+1);
        R=1;
        else R=0;
        end
    rootz(i) = R;
end

crossing = find(rootz==1); find_ones=crossing;
[length, width]=size(find_ones); number_of_ones=width;

if  number_of_ones<1
    number_elbow_movements = 0; total_elbow_motion = 0
elseif number_of_ones == 1
    total_elbow_motion = abs(dist(crossing)-dist(1));
    number_elbow_movements = 1
else
    how_many = [1:1:number_of_ones]; crossing_times = t3new(crossing);
    amount_of_motion = dist(crossing);

        for i = [2:1:number_of_ones];
            elbow_motion(i) = amount_of_motion(i)-amount_of_motion(i-1);
        end
    mag_elbow_motion = abs(elbow_motion); [b a] = size(crossing);
    number_elbow_movements_p = floor(a/2);
    total_elbow_motion_p = sum(mag_elbow_motion)+abs(amount_of_motion(1)-
dist(1));

        if total_elbow_motion_p/number_elbow_movements_p <1
        total_elbow_motion = 0; number_elbow_movements = 0;
        else
        total_elbow_motion = total_elbow_motion_p; number_elbow_movements =
number_elbow_movements_p;
        end
end
```

Appendix B – Matlab files for Method 2

To determine the number of movements and activity values for the elbow and shoulder:

METHOD2

```
%to determine angular velocity & acceleration, and related forces
clear all
%include when running more than one data file
%clear A* H* M* R* S* ac* ad* a alp* am* b c ch* cr* d f fin* fd* fa fo* g h
i j k l*
%clear m* n numb* o* p* q r* s* t* u* v w* x* y* z*
file = input('Select file: ', 's')
A = load(file);

All = sortrows(A, 1); [j k]=size(All);
sensor1 = All(1:1:j/2, :); sensor2 = All(j/2+1:1:j, :);
position_to_acceleration;

%theta is x angle, phi is z angle, psi is y angle
sensor1_theta = (sensor1(:, 5))*(2*pi/360); sensor1_phi = (sensor1(:,
6))*(2*pi/360);
sensor1_psi = (sensor1(:, 7))*(2*pi/360); sensor1_angles = [sensor1_theta
sensor1_phi sensor1_psi];
sensor2_theta = (sensor2(:, 5))*(2*pi/360); sensor2_phi = (sensor2(:,
6))*(2*pi/360);
sensor2_psi = (sensor2(:, 7))*(2*pi/360); sensor2_angles = [sensor2_theta
sensor2_phi sensor2_psi];
f, g] = size(sensor1_angles); q = f/10; t = [0.1:0.1:q];

%sensor 1
for i = [6:1:(m-5)]
    change_theta1(i-5) = (sensor1_theta(i+5)-sensor1_theta(i-5));
    change_phi1(i-5) = (sensor1_phi(i+5)-sensor1_phi(i-5));
    change_psi1(i-5) = (sensor1_psi(i+5) - sensor1_psi(i-5));
end

omega_theta1 = change_theta1/1; omega_phi1 = change_phi1/1; omega_psi1 =
change_psi1/1;
[g h]=size(omega_theta1); tnew = [0.6:0.1:(q-0.5)];

%sensor 2
for i = [6:1:(m-5)]
    change_theta2(i-5) = (sensor2_theta(i+5)-sensor2_theta(i-5));
    change_phi2(i-5) = (sensor2_phi(i+5)-sensor2_phi(i-5));
    change_psi2(i-5) = (sensor2_psi(i+5) - sensor2_psi(i-5));
end
omega_theta2 = change_theta2/1; omega_phi2 = change_phi2/1; omega_psi2 =
change_psi2/1;

%angular acceleration
%sensor 1
for j = [6:1:h-5]
    change_omegatheta1(j-5) = (omega_theta1(j+5) - omega_theta1(j-5));
```

```
    change_omegaphi1(j-5) = (omega_phi1(j+5) - omega_phi1(j-5));
    change_omegapsi1(j-5) = (omega_psi1(j+5) - omega_psi1(j-5));
end
alpha_theta1 = change_omegatheta1/1; alpha_phi1 = change_omegaphi1/1;
alpha_psi1 = change_omegapsi1/1;
t2new = [1.1:0.1:q-1];

%sensor 2
for j = [6:1:h-5]
    change_omegatheta2(j-5) = (omega_theta2(j+5) - omega_theta2(j-5));
    change_omegaphi2(j-5) = (omega_phi2(j+5) - omega_phi2(j-5));
    change_omegapsi2(j-5) = (omega_psi2(j+5) - omega_psi2(j-5));
end
alpha_theta2 = change_omegatheta2/1; alpha_phi2 = change_omegaphi2/1;
alpha_psi2 = change_omegapsi2/1;

%velocity relative to shoulder/sensor2
omega_sensor1 = [omega_theta1' omega_psi1' omega_phi1'];
omega_sensor2 = [omega_theta2' omega_psi2' omega_phi2'];
alpha_sensor2 = [alpha_theta2' alpha_psi2' alpha_phi2'];
alpha_sensor1 = [alpha_theta1' alpha_psi1' alpha_phi1'];

relative_angular_vel = omega_sensor1-omega_sensor2;
relative_angular_acc = alpha_sensor1-alpha_sensor2;

%make shoulder reference point
acc_rel_sx = sd1x - sd2x; acc_rel_sy = sd1y - sd2y; acc_rel_sz = sd1z - sd2z;
acc_rel = [acc_rel_sx' acc_rel_sy' acc_rel_sz'];

%to determine acceleration at elbow
ua = input('distance from shoulder sensor to elbow: ');
fa = input('distance from elbow to wrist sensor: ');
y_comp_ua = ua*sin(sensor2_psi); l = ua*cos(sensor2_psi);
x_comp_ua =  l.*sin(sensor2_phi); z_comp_ua = l.*cos(sensor2_phi);
r_se = [x_comp_ua y_comp_ua z_comp_ua];
[a b] = size(r_se); r_se_adj = r_se(11:(a-10), :, :);
omega_sensor2_adj = omega_sensor2(11:(a-10),:,:);
acceleration_elbow_s = cross(alpha_sensor2, r_se_adj);
acceleration_elbow = acceleration_elbow_s - [sd2x' sd2y' sd2z'];

%%to find predicted shoulder only acceleration at wrist
% to find acceleration of wrist relative to shoulder
y_comp_fa = fa*sin(sensor2_psi); l2 = fa*cos(sensor2_psi);
x_comp_fa = l2.*sin(sensor2_phi); z_comp_fa = l2.*cos(sensor2_phi);
r_ew = [x_comp_fa y_comp_fa z_comp_fa];
r_ew_adj = r_ew(11:(a-10),:,:);
wXRew = cross(omega_sensor2_adj, r_ew_adj);
wXwXRew = cross(omega_sensor2_adj, wXRew);
alphaXRew = cross(alpha_sensor2, r_se_adj);
[l1 w1] = size(acc_rel);
acc_rel_adj = acc_rel((adj+1):(l1-adj), :);
acc_we = acc_rel_adj- acceleration_elbow- (alphaXRew-wXwXRew);

%determining muscle moment at shoulder
marm = input('mass of arm: ');
```

```
force_acc_e = acceleration_elbow.*marm;
Mxs = -force_acc_e(:,1)*ua; Mys = -force_acc_e(:,2)*ua; Mzs = marm*9.8*ua-
force_acc_e(:,3)*ua;
Moment_magnitude_s = sqrt(Mxs.^2+Mys.^2+Mzs.^2);
adj_M_mag_s = Moment_magnitude_s - Moment_magnitude_s(1);
[u v] = size(Moment_magnitude_s);
for i =[1:1:u];
if adj_M_mag_s(i)>= 10;
    pos_M_s(i) = adj_M_mag_s(i);
else
    pos_M_s(i) = 0;
end
end

%determining muscle moment at elbow
mfa= input('mass of forearm: ');
force_acc_w = acc_we.*mfa;
Mxe = -force_acc_w(:,1)*fa; Mye = -force_acc_w(:,2)*fa; Mze = mfa*9.8*fa-
force_acc_w(:,3)*fa;
Moment_magnitude_e = sqrt(Mxe.^2+Mye.^2+Mze.^2);
adj_M_mag_e = Moment_magnitude_e - Moment_magnitude_e(1);

for i =[1:1:u];
if adj_M_mag_e(i)>= 10;
    pos_M_e(i) = adj_M_mag_e(i);
else
    pos_M_e(i) = 0;
end
end

%change in moments with time
for i = [6:1:(u-5)];
   change_moment_e(i-5) = pos_M_e(i+5)-pos_M_e(i-5);
end
fdMe = change_moment_e/1;

for i = [6:1:(u-5)];
   change_moment_s(i-5) = pos_M_s(i+5)-pos_M_s(i-5);
end
fdMs = change_moment_s/1; [q r] = size(t2new); time5 = t2new(6:(r-5));

%to determine elbow motion
for i = [1:1:(u-11)];
        if pos_M_e(i)>0 & pos_M_e(i+1)>0
        He = [fdMe(i) fdMe(i+1)];
        time2 = [time5(i) time5(i+1)];
        pe = polyfit(time2, He, 1);
        re = roots(pe);
            if re>time5(i) & re<time5(i+1);
            R=1;
            else R=0;
            end
        else R=0;
        end
      rootZe(i) = R;
end
```

```
crossingse = find(rootZe==1); find_ones=crossingse;
[length, width]=size(find_ones); number_of_ones_e=width;

if number_of_ones_e<1
    total_elbow_motion = 0;
    number_elbow_movements = 0;
elseif number_of_ones_e==1
    crossing_times_e = time5(crossingse);
    amount_of_elbowmotion = pos_M_e(crossingse+5);
    total_elbow_motion = sum(amount_of_elbowmotion) ;
    number_elbow_movements = number_of_ones_e;
else
crossing_times_e = time5(crossingse);
amount_of_elbowmotion = pos_M_e(crossingse+5);
Meane = median(amount_of_elbowmotion);
[c d] = size(amount_of_elbowmotion);
padded_em = [1 1 amount_of_elbowmotion 1 1];
[q w] = size(time5);
crossing_times_e_adj = [time5(1) time5(2) crossing_times_e time5(w-1)
time5(w)];
for i = [3:1:d+2]
    if padded_em(i)<(Meane*1.1)
        S=0;
    elseif (crossing_times_e_adj(i+1)-crossing_times_e_adj(i))>=.3 &
(crossing_times_e_adj(i)-crossing_times_e_adj(i-1))>=.3
        S=1;
    elseif (crossing_times_e_adj(i+2)-crossing_times_e_adj(i))<=.3 &
padded_em(i)<padded_em(i+2)
        S=0;
    elseif (crossing_times_e_adj(i)-crossing_times_e_adj(i-2))<=.3 &
padded_em(i)<padded_em(i-2)
            S = 0;
    elseif (crossing_times_e_adj(i+1)-crossing_times_e_adj(i))<=.3 &
padded_em(i)<padded_em(i+1)
        S = 0;
    elseif (crossing_times_e_adj(i-1)-crossing_times_e_adj(i))<=.3 &
padded_em(i)>padded_em(i-1)
        S=1;
    else S=0;
    end
    real_peake(i-2)=S ;
end
true_maxe = find(real_peake==1); find_ones=true_maxe;
[length, width]=size(find_ones); number_of_ones_e=width;
crossing_times_e = time5(crossingse(true_maxe));
amount_of_elbowmotion = pos_M_e(crossingse(true_maxe)+5);
total_elbow = sum(amount_of_elbowmotion); total_elbow_motion =
double(total_elbow);
number_elbow_movements = number_of_ones_e;
end

%to determine shoulder motion
for i = [1:1:(u-11)];
    if pos_M_s(i)>0 | pos_M_s(i+1)>0
        Hs = [fdMs(i) fdMs(i+1)];
        time2 = [time5(i) time5(i+1)];
```

```
        ps = polyfit(time2, Hs, 1);
        rs = roots(ps);
        if rs>time5(i) & rs<time5(i+1);
            R=1;
        else R=0;
        end
    else R=0;
    end
    rootZs(i) = R;
end
crossingss = find(rootZs==1); [length, width]=size(crossingss);
number_of_ones_s=width;
if number_of_ones_s<1
   total_shoulder_motion = 0;
   number_shoulder_movements =0;
elseif number_of_ones_s==1
    crossing_times_s = time5(crossingss);
    amount_of_shouldermotion = pos_M_s(crossingss+5);
    total_shoulder_motion = sum(amount_of_shouldermotion) ;
    number_shoulder_movements = number_of_ones_s;
else
crossing_times_s = time5(crossingss);
amount_of_shouldermotion = pos_M_s(crossingss+5);
Means = median(amount_of_shouldermotion);
[a b] = size(amount_of_shouldermotion);
padded_ss =[1 1 amount_of_shouldermotion 1 1];
[q w] = size(time5);
crossing_times_s_adj = [time5(1) time5(2) crossing_times_s time5(w-1)
time5(w)];
for i = [3:1:b+2]
    if padded_ss(i)<(Means*1.3)
        S=0;
    elseif (crossing_times_s_adj(i+1)-crossing_times_s_adj(i))>=.3 &
(crossing_times_s_adj(i)-crossing_times_s_adj(i-1))>=.3
        S=1;
    elseif (crossing_times_s_adj(i+2)-crossing_times_s_adj(i))<=.3 &
padded_ss(i)<padded_ss(i+2)
        S=0;
    elseif (crossing_times_s_adj(i)-crossing_times_s_adj(i-2))<=.3 &
padded_ss(i)<padded_ss(i-2)
           S = 0;
    elseif (crossing_times_s_adj(i+1)-crossing_times_s_adj(i))<=.3 &
padded_ss(i)<padded_ss(i+1)
        S = 0;
    elseif (crossing_times_s_adj(i-1)-crossing_times_s_adj(i))<=.3 &
padded_ss(i)>padded_ss(i-1)
        S=1;
    else S=0;
    end
    real_peaks(i-2)=S ;
end
true_maxs = find(real_peaks==1); find_ones=true_maxs;
[length, width]=size(find_ones); number_of_ones_s=width;
crossing_times_s = time5(crossingss(true_maxs));
amount_of_shouldermotion = pos_M_s(crossingss(true_maxs)+5);
total_shoulder_motion = sum(amount_of_shouldermotion);
number_shoulder_movements = number_of_ones_s;
```

```
end
```

POSITION_TO_ACCELERATION

```
% to determine resulting acceleration values
sensor1x = sensor1(:, 2); sensor1y = sensor1(:, 3);
sensor1z = sensor1(:, 4); Sensor1 = [sensor1x sensor1y sensor1z];

sensor2x = sensor2(:, 2); sensor2y = sensor2(:, 3);
sensor2z = sensor2(:, 4); Sensor2 = [sensor2x sensor2y sensor2z];

[m, n] = size(sensor1); k = m/10;

t = [0.1:0.1:k];
filter = input('choose filter: ','s');
run (filter)
```

Appendix C –Matlab files for Filters

FILTER5

```
%apply filter with a spacing of +/-5

%x values
for i = [6:1:(m-5)]
    rise1x(i-5) = (sensor1x(i+5)-sensor1x(i-5));
end
fd1x=rise1x/1; [g h]=size(fd1x);
for j = [6:1:h-5]
    rise21x(j-5) = (fd1x(j+5) - fd1x(j-5));
end
sd1x=rise21x/1; tnew = [0.6:0.1:(k-0.5)]; t2new = [1.1:0.1:(k-1)];
for i = [6:1:(m-5)]
    rise2x(i-5) = (sensor2x(i+5)-sensor2x(i-5));
end
fd2x=rise2x/1; [d f]=size(fd2x);
for j = [6:1:f-5]
    rise22x(j-5) = (fd2x(j+5) - fd2x(j-5));
end
sd2x=rise22x/1;

%y values
for i = [6:1:(m-5)]
    rise1y(i-5) = (sensor1y(i+5)-sensor1y(i-5));
end
fd1y=rise1y/1; [g h]=size(fd1y);
for j = [6:1:h-5]
    rise21y(j-5) = (fd1y(j+5) - fd1y(j-5));
end
sd1y=rise21y/1;
for i = [6:1:(m-5)]
    rise2y(i-5) = (sensor2y(i+5)-sensor2y(i-5));
end
fd2y=rise2y/1; [d f]=size(fd2y);
for j = [6:1:f-5]
    rise22y(j-5) = (fd2y(j+5) - fd2y(j-5));
end
sd2y=rise22y/1;

%z values
for i = [6:1:(m-5)]
    rise1z(i-5) = (sensor1z(i+5)-sensor1z(i-5));
end
fd1z=rise1z/1; [g h]=size(fd1z);
for j = [6:1:h-5]
    rise21z(j-5) = (fd1z(j+5) - fd1z(j-5));
end
sd1z=rise21z/1;
for i = [6:1:(m-5)]
    rise2z(i-5) = (sensor2z(i+5)-sensor2z(i-5));
end
fd2z=rise2z/1; [d f]=size(fd2z);
```

```
for j = [6:1:f-5]
    rise22z(j-5) = (fd2z(j+5) - fd2z(j-5));
end
sd2z=rise22z/1; t3new = [1.2:0.1:(k-1)];
time4 = [1.3:0.1:(k-1.1)]; adj = 0;
```

FILTER4

```
%apply filter with a spacing of +/-4

%x values
for i = [5:1:(m-4)]
    rise1x(i-4) = (sensor1x(i+4)-sensor1x(i-4));
end
fd1x=rise1x/.8; [g h]=size(fd1x);
for j = [5:1:h-4]
    rise21x(j-4) = (fd1x(j+4) - fd1x(j-4));
end
sd1x=rise21x/.8; tnew = [0.5:0.1:(k-0.4)]; t2new = [0.9:0.1:(k-.8)];
for i = [5:1:(m-4)]
    rise2x(i-4) = (sensor2x(i+4)-sensor2x(i-4));
end
fd2x=rise2x/.8; [d f]=size(fd2x);
for j = [5:1:f-4]
    rise22x(j-4) = (fd2x(j+4) - fd2x(j-4));
end
sd2x=rise22x/.8;

%y values
for i = [5:1:(m-4)]
    rise1y(i-4) = (sensor1y(i+4)-sensor1y(i-4));
end
fd1y=rise1y/.8; [g h]=size(fd1y);
for j = [5:1:h-4]
    rise21y(j-4) = (fd1y(j+4) - fd1y(j-4));
end
sd1y=rise21y/.8;
for i = [5:1:(m-4)]
    rise2y(i-4) = (sensor2y(i+4)-sensor2y(i-4));
end
fd2y=rise2y/.8; [d f]=size(fd2y);
for j = [5:1:f-4]
    rise22y(j-4) = (fd2y(j+4) - fd2y(j-4));
end
sd2y=rise22y/.8;

%z values
for i = [5:1:(m-4)]
    rise1z(i-4) = (sensor1z(i+4)-sensor1z(i-4));
end
fd1z=rise1z/.8; [g h]=size(fd1z);
for j = [5:1:h-4]
    rise21z(j-4) = (fd1z(j+4) - fd1z(j-4));
end
sd1z=rise21z/.8;
```

```
for i = [5:1:(m-4)]
    rise2z(i-4) = (sensor2z(i+4)-sensor2z(i-4));
end
fd2z=rise2z/.8; [d f]=size(fd2z);
for j = [5:1:f-4]
    rise22z(j-4) = (fd2z(j+4) - fd2z(j-4));
end
sd2z=rise22z/.8;
t3new = [1:0.1:(k-.8)]; time4 = [1.1:0.1:(k-.9)];
adj = 2;
```

FILTER3

```
%apply filter with a spacing of +/-3

%x vales
for i = [4:1:(m-3)]
    rise1x(i-3) = (sensor1x(i+3)-sensor1x(i-3));
end
fd1x=rise1x/.6; [g h]=size(fd1x);
for j = [4:1:h-3]
    rise21x(j-3) = (fd1x(j+3) - fd1x(j-3));
end
sd1x=rise21x/.6; tnew = [0.4:0.1:(k-0.3)]; t2new = [0.7:0.1:(k-.6)];
for i = [4:1:(m-3)]
    rise2x(i-3) = (sensor2x(i+3)-sensor2x(i-3));
end
fd2x=rise2x/.6; [d f]=size(fd2x);
for j = [4:1:f-3]
    rise22x(j-3) = (fd2x(j+3) - fd2x(j-3));
end
sd2x=rise22x/.6;

%y values
for i = [4:1:(m-3)]
    rise1y(i-3) = (sensor1y(i+3)-sensor1y(i-3));
end
fd1y=rise1y/.6; [g h]=size(fd1y);
for j = [4:1:h-3]
    rise21y(j-3) = (fd1y(j+3) - fd1y(j-3));
end
sd1y=rise21y/.6;
for i = [4:1:(m-3)]
    rise2y(i-3) = (sensor2y(i+3)-sensor2y(i-3));
end
fd2y=rise2y/.6; [d f]=size(fd2y);
for j = [4:1:f-3]
    rise22y(j-3) = (fd2y(j+3) - fd2y(j-3));
end
sd2y=rise22y/.6;

%z values
for i = [4:1:(m-3)]
    rise1z(i-3) = (sensor1z(i+3)-sensor1z(i-3));
end
```

```
fd1z=rise1z/.6; [g h]=size(fd1z);
for j = [4:1:h-3]
    rise21z(j-3) = (fd1z(j+3) - fd1z(j-3));
end
sd1z=rise21z/.6;
for i = [4:1:(m-3)]
    rise2z(i-3) = (sensor2z(i+3)-sensor2z(i-3));
end
fd2z=rise2z/.6; [d f]=size(fd2z);
for j = [4:1:f-3]
    rise22z(j-3) = (fd2z(j+3) - fd2z(j-3));
end
sd2z=rise22z/.6;
t3new = [0.8:0.1:(k-.6)]; time4 = [0.9:0.1:(k-.7)];
adj = 4;
```

FILTER2

```
%apply filter with a spacing of +/-2

%x values
for i = [3:1:(m-2)]
    rise1x(i-2) = (sensor1x(i+2)-sensor1x(i-2));
end
fd1x=rise1x/.4; [g h]=size(fd1x);
for j = [3:1:h-2]
    rise21x(j-2) = (fd1x(j+2) - fd1x(j-2));
end
sd1x=rise21x/.4; tnew = [0.3:0.1:(k-0.2)]; t2new = [0.5:0.1:(k-.4)];
for i = [3:1:(m-2)]
    rise2x(i-2) = (sensor2x(i+2)-sensor2x(i-2));
end
fd2x=rise2x/.4; [d f]=size(fd2x);
for j = [3:1:f-2]
    rise22x(j-2) = (fd2x(j+2) - fd2x(j-2));
end
sd2x=rise22x/.4;

%y values
for i = [3:1:(m-2)]
    rise1y(i-2) = (sensor1y(i+2)-sensor1y(i-2));
end
fd1y=rise1y/.4; [g h]=size(fd1y);
for j = [3:1:h-2]
    rise21y(j-2) = (fd1y(j+2) - fd1y(j-2));
end
sd1y=rise21y/.4;
for i = [3:1:(m-2)]
    rise2y(i-2) = (sensor2y(i+2)-sensor2y(i-2));
end
fd2y=rise2y/.4; [d f]=size(fd2y);
for j = [3:1:f-2]
    rise22y(j-2) = (fd2y(j+2) - fd2y(j-2));
end
sd2y=rise22y/.4;
```

```
%z values
for i = [3:1:(m-2)]
    rise1z(i-2) = (sensor1z(i+2)-sensor1z(i-2));
end
fd1z=rise1z/.4; [g h]=size(fd1z);
for j = [3:1:h-2]
    rise21z(j-2) = (fd1z(j+2) - fd1z(j-2));
end
sd1z=rise21z/.4;
for i = [3:1:(m-2)]
    rise2z(i-2) = (sensor2z(i+2)-sensor2z(i-2));
end
fd2z=rise2z/.4; [d f]=size(fd2z);
for j = [3:1:f-2]
    rise22z(j-2) = (fd2z(j+2) - fd2z(j-2));
end
sd2z=rise22z/.4;
t3new = [0.6:0.1:(k-.4)]; time4 = [0.7:0.1:(k-.5)];
adj = 6;
```