


Human motion analysis using wearable IMU sensors
Given a motion sequence captured by IMU sensors:

- Identify the different phases in the motion - Segmentation
- Classify different types of motion – Classification
- Compare the motion sequence to a reference motion - Identification
• High-dimensional data from the sensors
  » Number of IMU devices: $N$
  » Number of sensors per device: $S$
  » Number of axes per sensor: $A$
• Overall dimensionality: $D = N \times S \times A$
• In the case of gait analysis using 3 devices on each leg, and 1 on the hip, $D = 7 \times 3 \times 3 = 63 \text{ large!}$
• Makes motion analysis difficult
• Approach: low-dimensional representation of data ($d < D$), to ease analysis
Clinical Gait Analysis – Motivating Example

• Why? – Visual inspection and manual segmentation is tedious and error-prone
• How? – Devise objective methods for analysing the walking patterns of people wearing the IMU devices
Model-based Segmentation

- Joint angles: calculated using fusion of accelerometer and gyroscope data from individual devices attached to the connected limbs
- Frame: Collection of joint angles. For \( N \) devices, a frame at time \( t \) is the set

\[
\{\theta_i\}_t, \quad i = 1..N
\]

- Joint positions: computed by combining joint angles, lengths of the limbs and applying forward kinematics

\[
\{p_i\}_t, \quad i = 1..N, \quad p_i = [x, y, z]^T
\]

with \( x, y \) and \( z \) the Cartesian co-ordinates of \( p_i \) relative to a fixed frame of reference.

- The fixed point has been defined as the hip joint of the model, so that \( p_{\text{hip}} = [0, 0, 0]^T \) for all times \( t \).
For the evolution of joint positions over time, a motion sequence of duration $dt$ for the joint $i$ is defined as:

$$\{\{p_i\}_t\}, \; t \in [0, dt]$$

For every joint, this sequence is passed as input to the segmentation algorithm.

The three planes of motion are scanned independently for local minima and maxima in the joint positions.

Let $j \in \{\text{traverse (x-z axis), saggital (y-z axis), coronal (x-y axis)}\}$ denote the different planes of motion.

Let $\{p_i\}_t$ be the positional coordinate of joint $i$ on plane $j$ at time $t$. 

Model-based segmentation
Model-based segmentation

- For every joint $i$ and motion plane $j$, the segmentation points are a set of time instances:
  \[\{\{t_{\text{min}}\}, \{t_{\text{max}}\}\}\]
such that every $\{p_{ij}\}_{t_{\text{min}}}$ is a local minimum and $\{p_{ij}\}_{t_{\text{max}}}$ is a local maximum

- Identify distinct intervals in a periodic motion by considering pairs of successive minima and maxima

- Sensitive to sensory noise and small local discontinuities in joint positions are treated as segmentation points, leading to false positives
Model-based segmentation

• A model-based approach implies that the designer must explicitly choose the most salient joints and planes of motion which is not always easy for a non-expert

• *e.g.*, in gait analysis is motion in the transverse plane more important than on the sagittal plane? Or, are the ankle positions more important in segmentation than the knee positions?

• Conflicting segmentation sets might result for different joints, making it difficult to decide which joint produces more reliable results
Local maxima (Black) and minima (Red)
Top - motion on transverse plane; Bottom - sagittal plane
Six steps forward, turn, six steps backward
Local maxima (Black) and minima (Red)
Top - motion on transverse plane
Bottom - sagittal plane
Model-free segmentation – Latent space algorithm

- Aggregate the joint positions into a single feature vector
- For J tracked joints,
  \[ i.e., \mathbf{q} = [x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_J, y_J, z_J]^T \]
- Approach: Learn low-dimensional representation \( \{r_t\} \) of high-dimensional motion sequence \( \{q_t\} \)
- Find segmentation points using min-max procedure
- Manifold learning algorithm: Isomap (Tenenbaum et al 2000)
- Result: a single set of segmentation points for the entire motion

Normal Walk - Left Knee

(a) High-dimensional

Normal Walk Latent Representation

(b) Latent space
Dynamic Time Warping (DTW)

- Distance metric for two time series data
- Aligns two time series and returns the distance accumulating the euclidean distances between each pair of aligned points

**DTW distance: 103953.393988 for Horizontal spin vs Vertical spin**

![Graph showing time series comparison with DTW distance](image-url)
DTW

- Curves are aligned recursively using dynamic programming for nonlinearly stretching and squashing an input motion, so that it most closely matches a reference motion
- All possible monotonic, continuous alignments re-computed, and the one which minimises the remaining distance between the two time series is selected
- Running time of DTW $O(n^2)$ with an input of size $n$
- Higher dimensional versions of DTW grow exponentially with the number of dimensions $O(n^d)$ – impractical!
- DTW distances between corresponding dimensions of two motions are calculated and averaged.
- Weights assigned to each dimension when calculating the average, e.g., importance of gyroscope over magnetometer data in classifying motions.
Dimensionality Reduction Algorithm – Principal Component Analysis

- Given a high dimensionality data set, PCA projects to a new set of dimensions such that the first dimension explains the largest possible amount of variance in the original data.
Sensor data streams from IMU devices pre-PCA
Graph of 1\textsuperscript{st} and 2\textsuperscript{nd} after PCA analysis showing cycles in the data
Principal Component Analysis

- Given a high dimensional data set, project to a new set of dimensions such that the first dimension explains the largest possible variations in the original data.
- Subsequent dimensions explain the next largest possible variations in the original data while remaining orthogonal to previous dimensions.
- Highlights most pertinent information and reduces noise.
- Preparation for applying DTW as this is performance bottle neck $O(n^2)$. 
Averages of distances between matching sensor readings

MOTION 1: each cell represents an array of sensor readings

<table>
<thead>
<tr>
<th>MAG X</th>
<th>MAG Y</th>
<th>MAG Z</th>
<th>GYRO X</th>
<th>GYRO Y</th>
<th>GYRO Z</th>
<th>ACCEL X</th>
<th>ACCEL Y</th>
<th>ACCEL Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
<td>DTW</td>
</tr>
</tbody>
</table>

Average distance

MOTION 2: each cell represents an array of sensor readings

| MAG X | MAG Y | MAG Z | GYRO X | GYRO Y | GYRO Z | ACCEL X | ACCEL Y | ACCEL Z |
Each cell corresponds to Averages of distances between matching sensor readings
PCA based segmentation

Orient motion capture data

| MAG X | MAG Y | MAG Z | GYRO X | GYRO Y | GYRO Z | ACCEL X | ACCEL Y | ACCEL Z |

Principal Component Analysis

Principal components 1..N

1st Principal Component used to find segmentation points

Sensor data segment 1

MAG  GYRO  ACCEL

Sensor data segment N

MAG  GYRO  ACCEL

PCA data segment 1

Principal components 1..N

PCA data segment N

Principal components 1..N
PCA based segmentation of vertical arm exercises (right arm)
Similarity Matrices for the two methods for arm exercises at different angles (right arm)

(a) Arm exercise comparisons using high dimensional, raw sensor data

(b) Arm exercise comparisons using principal components after dimensionality reduction
Pearson’s Product-Moment Coefficient

• A measure of the linear correlation between two data sets or random variables
• positive correlation - values of the data are directly proportional,
• negative correlation - inversely proportional
• zero - no linear dependency between the variables
• For two random variables $X, Y$, mean - $\mu_X; \mu_Y$ and standard deviations - $\sigma_X; \sigma_Y$, and $E$ is the expected value (Expected value of a discrete random variable is the probability-weighted average of all possible values)

$$\rho_{X,Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$
Analysis of variance (ANOVA) Test

• ANOVA is a particular form of statistical hypothesis testing used extensively for the analysis of experimental data.

• It tests the null hypothesis that samples in two or more groups actually come from distributions with the same means.

• The test returns an F-statistic: the ratio of the population variance within all samples to the variance between the means of each sample group.

• A p-value can be derived from the ANOVA test for comparison with other statistical techniques.
Sensitivity (Recall)

- Sensitivity is the number of correct positive results divided by the number of positive results that should have been returned.
- It is a measure of the true positive rate of a binary classification test, *i.e.*, the proportion of positives that were correctly identified. This is calculated from the counts of correct and incorrect classifications for each class - the number of true positives, false positives, true negatives and false negatives, according to the formula:

\[
\text{Sensitivity} := \frac{\text{True positive count}}{\text{True positive count} + \text{False negative count}}
\]

E.g., Consider a motion sequence with 10 sec. of activity and 90 sec. of inactivity, and a detector which classified 5 sec. of activity correctly and mistakenly classified 10 seconds of inactivity as activity, would have a sensitivity of \( \frac{5}{5+5} = 50\% \).
Specificity

• A measure of the true negative rate of a binary classification test. i.e., the proportion of negatives that were correctly classified.

\[
\text{Specificity} := \frac{\text{True negative count}}{\text{True negative count} + \text{False positive count}}
\]

E.g., Consider a motion sequence with 10 seconds of activity and 90 seconds of inactivity, a detector which classified 5 seconds of the activity correctly and mistakenly classified 10 seconds of the inactivity would have a specificity of \( \frac{80}{80+10} = 88.9\% \).
• In a binary classification test the precision is the proportion of positive results that are true positives. It is the number of correct positive results divided by the number of all positive results.

\[
\text{Precision} := \frac{\text{True positive count}}{\text{True positive count} + \text{False positive count}}
\]

E.g., Consider a motion sequence with 10 seconds of activity and 90 seconds of inactivity, and a detector which classified 5 seconds of the activity correctly and mistakenly classified 10 seconds of the inactivity, would have a specificity of \(\frac{5}{5+10} = 33.33\%\).
F-score (F-measure)

- In binary classification, F-score is a measure of a test’s accuracy.
- F-score is the harmonic average of the precision and sensitivity, where an F1 score ranges between 1 (perfect precision and sensitivity) and worst at 0.

\[
F\text{-score} := 2 \cdot \frac{\text{Precision} \cdot \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}
\]
K-Fold Cross Validation

• A method to estimate how well a statistical model will generalise to independent data.

• Cross validation is performed by splitting a dataset into two subsets: one to train the model and the other to test it.

• The portion of the data used as the test set is rotated in successive ‘folds’ of the cross validation, until all data has been in the test set in some fold.

• ’Leave-one-out’ cross validation is a particular variant where, in each fold, a single data item is used to test while all others are used to train.
Location of 7 Orients and synchronised motion data capture from Orient-reflector pairs
Synchronised motion capture from Vicon and Orient motion capture systems

- A marker-based Vicon gait analysis system captured the motion of the human subjects during normal walking, and provided the ground truth for the experimental validation of the FNN+HMM methods on the Orient data.

- The data capture on the Vicon system was triggered from the Orients which ensured that both systems were synchronised.

- The two data frames were matched exactly for comparison in the temporal domain by interpolating the Orient data sampled at 42.67 Hz with the Vicon data sampled at 100 Hz.

- After a brief calibration period lasting 15 seconds, the subjects were invited to walk the length of a 30 foot level walkway in a clinical gait laboratory.

- Five subjects (two male, three female) with normal gaits were chosen for the experiments and four trials per subject were recorded.
Feedforward Neural Networks + Hidden Markov Model Classifier

FNN
• Class of networks with multiple layers of computational units connected in a feed-forward fashion
• FNN are more suited for recognising patterns in high dimensional sensor data input of fixed sizes
• Less able to handle events with variable time spans
• No explicit treatment of inputs as a sequence being the case of the IMU units generating continuous streams of motion capture data
• HMMs model sequences explicitly

HMM
• The HMM transition probabilities encode the sequential ordering of human gait phases (this ordering is not encoded by FNN which considers windows of sensor values in isolation)
• HMM emission probabilities encode the average error rates of the FNN classifications for each gait phase, thus providing contextual information unavailable to the FNN on its own

**HMM complements the FNN by providing sequential context to its pattern recognition ability**
HMMs model sequences explicitly and observed sequences are modelled as emissions from a set of unobservable states.

 » A set of transition probabilities between hidden states;
 » a set of emission probabilities for each state;
 » and the initial state probability distribution

Given a sequence of observed emissions and a HMM, the Viterbi algorithm can infer the most likely sequence of hidden states that could have produced the emissions.

HMMs are less suited for high dimensional data as the number of possible states and emissions grows exponentially

The five gait phases of interest become the hidden states of the HMM

The classifications of sensor data by the neural network are modelled as emissions from the hidden states
FNN+HMM Classifier

• The FNN uses a sliding window approach to classify the continuous stream of sensor data:
  – each frame of data contains all the sensor values for a given instant
  – values from all frames in a local window are concatenated to form an input vector for the neural network.

• The FNN assigns a class label to each window.
  – Class labels are learned from annotated training data, where each window’s class is the phase of gait at the time of the central frame (e.g., the swing phase).

• The FNN sliding window classifications represent the emissions of the HMM

• The Viterbi algorithm is used to infer the most likely sequence of hidden states (gait phases) given the observed sequence of emissions (FNN classifications) and a trained HMM.

• The most likely sequence inferred is taken as the final output of the classifier.
STANCE: Heel Strike $\rightarrow$ Foot Flat $\rightarrow$ Heel Off $\rightarrow$ Toe Off.
SWING: Toe Off $\rightarrow$ Mid-swing $\rightarrow$ Heel Strike.
FNN+HMM Classifier

• FNN consisted of three hidden layers of ten units each, and an output layer consisting of five units corresponding to the five gait phases.

• The input vector to the FNN was obtained by using a sliding window of fifteen frames, with a step size of one.

• Each window had 630 data points obtained from seven Orients;
  – Six dimensions of data from the three axis accelerometer and gyroscope (6 sensors x 7 Orients x 15 frames).
  – each frame of data contains all the sensor values for a given instant
  – values from all frames in a local window are concatenated to form an input vector for the neural network.

• The network was trained using a scaled conjugate gradient back propagation algorithm until it converged, using cross-entropy as the performance measure during training
FNN+HMM Classifier

• Leave-one-out cross validation:
  – For each fold of the cross validation, the data was split into mutually exclusive test and training sets: the test set consisting of a single trial from one subject (77 frames of data, on average), while all the other data formed the training set (1471 frames, on average).
  – The test set was rotated with each fold so that all the data was eventually used for testing and training, and the average performance of all tests is reported.

• HMM was configured as follows:
  – uniform distribution was used for the probability of the initial state
  – The transition matrix was estimated by counting the transitions between ground truth classes of each sliding window
  – HMM emission probabilities - the trained FNN was used to classify all of the windows in the training set, and the correspondences between true states and classified states were used to estimate the emission matrix of the HMM.

• The phases in the tables are named after the initiating event, i.e., the ‘Heel Strike’ phase lasts from the ‘Heel Strike’ event to the ‘Foot Flat’ event.
• The HMM complements the FNN by providing context to the latter’s pattern recognition
• The transition probabilities encode the sequential ordering of human gait phases
• (Recall that this ordering is not encoded by the FNN, which just looks at windows of sensor values in isolation)
• The emission probabilities encode the error rates of the FNN classifications given the current state, which provides contextual information unavailable to the FNN in isolation.
• The ground truth data in the results is obtained from the Vicon camera-based system
HMM hidden state transition probabilities

<table>
<thead>
<tr>
<th>Phase Transition</th>
<th>Heel Strike</th>
<th>Foot Flat</th>
<th>Heel Off</th>
<th>Toe Off</th>
<th>Midswing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heel Strike</td>
<td>0.75</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Foot Flat</td>
<td>0</td>
<td>0.95</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Heel Off</td>
<td>0</td>
<td>0</td>
<td>0.86</td>
<td>0.14</td>
<td>0</td>
</tr>
<tr>
<td>Toe Off</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.89</td>
<td>0.11</td>
</tr>
<tr>
<td>Mid Swing</td>
<td>0.10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.90</td>
</tr>
</tbody>
</table>
HMM emission probabilities from hidden states

<table>
<thead>
<tr>
<th>Phase Emission</th>
<th>Heel Strike</th>
<th>Foot Flat</th>
<th>Heel Off</th>
<th>Toe Off</th>
<th>Midswing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heel Strike</td>
<td>0.6923</td>
<td>0.0769</td>
<td>0.0769</td>
<td>0.0769</td>
<td>0.0769</td>
</tr>
<tr>
<td>Foot Flat</td>
<td>0.0222</td>
<td>0.8889</td>
<td>0.0222</td>
<td>0.0444</td>
<td>0.0222</td>
</tr>
<tr>
<td>Heel Off</td>
<td>0.0526</td>
<td>0.0526</td>
<td>0.7895</td>
<td>0.0526</td>
<td>0.0526</td>
</tr>
<tr>
<td>Toe Off</td>
<td>0.0667</td>
<td>0.0667</td>
<td>0.0667</td>
<td>0.7333</td>
<td>0.0667</td>
</tr>
<tr>
<td>Mid Swing</td>
<td>0.0667</td>
<td>0.0667</td>
<td>0.0667</td>
<td>0.2000</td>
<td>0.6000</td>
</tr>
</tbody>
</table>
CLASSIC GAIT TERMS

Heel Strike  Foot Flat  Heel Off  Toe Off  Midswing  Heel Strike

STANCE PHASE

% of GAIT CYCLE

STANCE PHASE

40
50
60
70
80
90
100

SWING PHASE

Acceleration  Deceleration

Left Leg Phase Detections (Capture data M19)

Sensor values

0
10
20
30
40
50
60
70
80
90
100

Detected Phases – Sensitivity: 89.5% Specificity 97.4% (Tolerance ±0 frames)

Ground Truth

Time (frames)

Right Leg Phase Detections (Capture data M19)

Sensor values

0
10
20
30
40
50
60
70
80
90
100

Detected Phases – Sensitivity: 97.7% Specificity 99.4% (Tolerance ±0 frames)

Ground Truth

Time (frames)
Phase Detection Graphs

- Panel (top) : raw inertial sensor data for both legs
- Panel (middle) : the gait phases detected by the FNN+HMM
- Panel (bottom) : ground truth gait phases derived from the Vicon camera & reflective marker based system
- The tolerance value was set to 0
- [For a given tolerance, ±N, a gait event is counted as detected even if it is missed by ±N frames]
- Impact of tolerance on the sensitivity and specificity outcomes is graphed in the next slide – performances increase up to 2 frames
- Having a window of tolerance when calculating performance is useful for determining the range over which the classifier is accurate.
- The choice of the window is application-dependant with the onus on the developer to decide on an acceptable range.
Impact of sliding window size for FNN on specificity and sensitivity

• The advantage of smaller window sizes is the reduction in the computational requirements, and can even be performed locally on the sensor.
• The performance is slightly higher for large sliding window sizes peaking at 22 frames.
• Even a window size of 1 results in a Sensitivity value just under 90%, which is due to the high dimensionality (d=42) in each frame.
• A single frame therefore contains sufficient data to infer the posture and momentum of the legs for the FNN to identify the most likely gait phase.
• In contrast, for low-dimensional sensor data, and a correspondingly larger window size is required to match the accuracy.
• Uniaxial accelerometer on each feet, resulted in the following performances:
  
  - d=2 per frame; window size = 20; Sensitivity=85%; Specificity=96%
  - d=2 per frame; window size = 1; Sensitivity=74%; Specificity=93%
Effect On Specificity And Sensitivity As Neural Network Sliding Window Size Increases

Sensitivity/Specificity (%)

Window size (frames)
Summary of specificity and sensitivity for 5 subjects (averaged over 4 trials each)

<table>
<thead>
<tr>
<th>Subject</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>88.79%</td>
<td>97.66%</td>
</tr>
<tr>
<td>2</td>
<td>96.07%</td>
<td>96.70%</td>
</tr>
<tr>
<td>3</td>
<td>87.72%</td>
<td>98.17%</td>
</tr>
<tr>
<td>4</td>
<td>89.47%</td>
<td>98.22%</td>
</tr>
<tr>
<td>5</td>
<td>88.93%</td>
<td>98.47%</td>
</tr>
</tbody>
</table>
Summary of gait analysis metrics

<table>
<thead>
<tr>
<th>Subject</th>
<th>Average Cycle Time (seconds)</th>
<th>Steps Per Minute</th>
<th>Left Stance to Swing Ratio</th>
<th>Right Stance to Swing Ratio</th>
<th>Average Stance to Swing Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.13</td>
<td>52.89</td>
<td>0.63</td>
<td>0.65</td>
<td>0.64</td>
</tr>
<tr>
<td>2</td>
<td>0.97</td>
<td>61.82</td>
<td>0.64</td>
<td>0.64</td>
<td>0.64</td>
</tr>
<tr>
<td>3</td>
<td>1.01</td>
<td>59.34</td>
<td>0.63</td>
<td>0.63</td>
<td>0.63</td>
</tr>
<tr>
<td>4</td>
<td>0.95</td>
<td>63.48</td>
<td>0.62</td>
<td>0.62</td>
<td>0.62</td>
</tr>
<tr>
<td>5</td>
<td>0.90</td>
<td>66.71</td>
<td>0.63</td>
<td>0.64</td>
<td>0.63</td>
</tr>
</tbody>
</table>

- The ratio of the swing to stance phases is close to 60% as expected - extrinsic validation of the method
Cohort-level analysis: Gait phase durations colour-coded by subject

- Duration left-right asymmetries for all gait phases for all subjects.
- The distance from the blue line indicates the degree of asymmetry
Cohort-level analysis of left-right asymmetry in swing/stance durations colour-coded by subject

![Swing/Stance Duration Asymmetries (colour coded by subject)]
Inferences

• Fairly even distribution around the symmetry line, with some variability between subjects as would be expected

• The light blue dots subject has all the gait phase durations slightly biased to the right-hand side

• The right-hand bias of the light blue dots subject is also apparent in swing/stance durations

• Black dot subject (left-handed) -
  – asymmetry in the swing and stance biases
  – left biased swing phase and a right biased stance phase
  – This could suggest that the stronger side drives the gait while the weaker side is used to balance

• These assessments could be automated to assist the practitioners in driving their investigations
Inferences

- Fairly even distribution around the symmetry line, with some variability between subjects as would be expected
- The light blue dots subject has all the gait phase durations slightly biased to the right-hand side
- The right-hand bias of the light blue dots subject is also apparent in swing/stance durations
- Black dot subject (left-handed) -
  - asymmetry in the swing and stance biases
  - left biased swing phase and a right biased stance phase
  - This could suggest that the stronger side drives the gait while the weaker side is used to balance
- These assessments could be automated to assist the practitioners in driving their investigations
Impact of sensor numbers and placements

• Hypothesis – 1 Reducing the number of Orients would reduce the computational complexity of the FNN+HMM classifier but detrimental to detection performance.

• Hypothesis – 2 Orients in some positions would be more effective than others in detection performance, and that different positions might vary in effectiveness for different gait phases.

• The results are computed from the sum of performance counts over all folds of the cross validation for 7 Orients placed on the feet, shins, thighs and one on the pelvis at the base of the spine.
Impact of sensor placement using Gyro + Accelerometer Data

- FNN+HMM classifier trained using the Orient data from each of those locations in isolation.
- Performance of the gait phase recognition remains high in any lower body location even when using a single Orient placed at the base of the spine.
- The performance is highest when using all of the sensors in combination, but any of the sensors in isolation also produces good results.
- Feet sensors gave the best performance for heel strike and thigh sensors the worst.
Impact of sensor placement – Acclerometer-only Data

- Results from the Accl or Gyro in isolation were more sensitive to sensor placement.

- Specificity in both cases was high and consistent, with the exception of the thigh placement of the gyro data, which had relatively poor performance in detecting the foot centric events (all but the mid-swing).

- The sensitivity was more variable in part because the quantity of data used was reduced to a single pair of triaxial accelerometers, and further split by gait phase.
Impact of sensor placement – Gyro-only Data

- thigh placement of the gyro data, had relatively poor performance in detecting the foot centric events (all but the mid-swing)