Annotation of Human Motion Capture Data using Conditional Random Fields

Mert Degirmenci, Anil Sevim
Middle East Technical University
{ mert.degirmenci, anil.sevim } @ceng.metu.edu.tr

Abstract

Human motion classification is a challenging task since human motion lacks clear categorical structure. A reliable classifier can be used in anomaly detection, gait disease diagnosis, and content-based video querying. Moreover, human motion classifier can be used in constructing motion capture database to eliminate manual labelling phase. Most of the proposed algorithms employ Hidden Markov Models (HMMs) to classify human motion. However independence assumptions made by HMMs do not correlate with temporal dependences in human motion pattern. In this study, we propose an algorithm that annotates human motion capture data using Conditional Random Fields (CRF). Manually labeled motion sequences from CMU database have been used for both training and testing. The experimental results conducted on CMU database reveal significant improvements over previously proposed algorithms.

Introduction

The problem addressed in this study is to annotate an unobserved motion sequence previously recorded labels. It is fairly challenging task to classify a human motion data. During transitions of activities, there can be patterns of different motion classes. Human motion also exhibits ambiguous time-scales. Persons walking speeds and patterns may vary greatly. To alleviate this problem, researchers try to find the key frame between two motion sequences. After partitioning the motion sequence with correct sampling, an unobserved sequence is categorized with previously trained algorithm.

Identifying key frames in itself is a research topic. Researchers argue that key frames for human motion is ambiguous to even humans. Human perception system may select different frames of the sequence of a walking human to identify the activity. For more complex activities, the problem exhibits larger temporal dependency and thus larger information may need to be examined. Varying time slices for different motion categories makes identifying key frames even more complicated. A human may perform sequence of simple activies through the process of much complicated motion activity. Even seemingly basic categories to humans can often be decomposed into more primitive activities.

Representations of human motion is also an important issue to be addressed since the solution to the the problem differ greatly depending on the description of it. Although some researchers proposed feature set description of the motion data, it is more intuitive to represent the human motion as a sequential streaming data. Inherent dependencies between time frames can not be expressed by a set of independent features.

Another key challenge in classifying human motion capture data is its spatial flexibility. We may often classify different scales of bone orientation and magnitudes into one category simultaneously. This scaling factor is also varying for different humans. Age, gender, even culture can be a determining factor for identifying a human motion. Although there is an obvious difference between a walking child and an adult, a reliable motion classifier is expected to identify both as in the same category.

In this study, we investigate annotation of human motion sequence by set of tags. Although tagging a motion sequence is multi-label supervised learning problem in general, additional constraints imposed enables a linear-chain Conditional Random Field (CRF) to tackle the problem. Motion capture data is structured to be used as training examples to CRF. Each example supplied to CRF can be represented as (x,y) pairs, where x denote a sequence of motion data and y denote a sequence of corresponding tags. Since this representation converts motion capture labels into sequential data, usage of CRFs becomes reasonable. Next section describes the related work in human motion classification.

Current Research

Early approaches for motion classification involved low level information extraction from raw sensor data. Features such as 3D point positions, joint angles, and PCA reducted information have been used to compare and identify a new motion sequence [5-7]. Although these features are not robust to pose deformations, they produced significant improvements for specific application domains (Mul06). Algorithms from machine learning paradigm have also been applied for human motion data classification. Researchers proposed semi-automatic annotation procedures. Arikan et al. have applied SVM classifiers to allow users annotate large collections of human motion data quickly (Ari03).

Researchers tried to partition the motion sequence and classify segments afterwards. Manabe et al. proposed a hidden markov model representation of human motion to show...
the differences between each observed motion (Man06). This model is supposed to help sports player pick up the tacit aspects of different physical skills. By the help of this study, critcising between good and bad motion will be more objective. Since predicting trials and errors of the HMM structure is very time-consuming and costly, automatic optimization of HMM is used to ease the computational burden. Using the fact that human motion data is in time-series, they segmented continuous observation data to discrete symbols by using key-frame representation. Then, a nonlinear mapping from a high dimensional input space to low dimensional grid is performed by using Self Organizing Maps (SOM). Previously recorded motions are then used in forward-backward algorithm, where parameters of HMM are estimated. In HMM optimization part, generic algorithms (GA) and Baum-Welch algorithm is combined to iteratively approximate better model parameters. Their experimental results of bat swing motion pattern shows that the proposed method is capable of finding sufficiently suitable HMMs for the given class of motion patterns.

For detecting key events during human activity several other algorithms have been used besides self-organizing maps. On specific domains of application, namely gait disease identification, researchers have used peculiar body wearable sensor (BWS) configuration to record the motion sequence. Sabatini et al. have placed a gyroscope, and an accelerometer on the foot to find stride characteristics and determine the key events (Sab05). Main drawback of this study is its specific configuration of sensors to detect key events. More general event detectors generally involve estimating a probabilistic model for the event. Pfau et al. have used left-right HMM to determine if the horses are galloping or not (Pfa08). Although they reported significant detection rates and accurate seperation of strides, they can detect only single key event given a new motion sequence. Eric et al. have also estimated parameters of key events using HMMs (Gue09). Main contribution of their study is the fact that they can identify multiple events since their implementation involve predicting state representation of the activity.

Several sensor technologies have been used to capture human motion data. Although BWS networks can directly supply motion parameters, researchers have also tried predicting motion category from time-sequential images. Advantage of using such method is the wide availability of 2D imagery data. Although most 2D image acquisition techniques lead to ill-defined motion parameters, stochastic models generally alleviate this problem by considering the noise involved in motion data extraction. Yamato et al. has exploited both time scale invariability and robustness to error of HMMs (Yam92). They transform the images into image feature vector sequence to extract key information. Then they use vector quantization to convert this information into symbol sequence. In order to optimize parameters of HMM, they have also used Baum-Welch method. Although this method does not reach to the global optimum in general, Yamato et al. noted that experiments show significant results even with locally optimum HMM parameters.

HMM based methods are not the only techniques used in motion capture data classification. Muller et al. proposed construction of Motion Templates (MT) which can automatically mask out the variable aspects of a motion class to label unknown motion data (Mul06). Since there are certain aspects associated with a motion class that may show significant spatio-temporal variations between different executions of it, they have proposed a method that captures spatio-temporal characteristics of an entire motion class in a compact matrix representation. In their study, whole class of motion are represented in one motion template. Based on template matching techniques, a new motion sequence is categorized into one of the motion classes for which templates have been computed at the preprocessing stage. To label a related class of motion activities with a single MT, MTs of member motion sequences are averaged. However, in most cases, identifying a class of motions with a single template is impractical since one MT can hardly represent even single motion sequence.

There have also been unsupervised approaches to motion capture data classification. Barbic et al. worked on segmenting motion capture data into distinct high-level behaviours (walking,running,punching) (Bar04). Their work suggests that automatic classification of human motion data into distinct behaviours can be efficient and more robust than hand segmentation. Their segmentation incorporates three different approaches, two of which are online. The first approach chooses segments using an indication of intrinsic dimensionality from Principal Component Analysis (PCA). The second approach creates segments using a probabilistic model of motion obtained from Probabilistic PCA, and the third approach generates segments based on a Gaussian mixture model representation. Finally, they have found that very good performance can be obtained from these simple and fast techniques.

We have presented several algorithms for human motion capture data classification. First approaches were using low level feature extraction techniques in conjunction with heuristical algorithms. Although such naive approaches work well for a specific data set, they are not practical for general purpose motion identification.

Kernel based learning algorithms have also been described for human motion data classification. Although these methods achieve improvements over heuristical techniques, they miss the important caracteristical knowledge about human motion such as its time sequency and time scalability.

Hidden markov models are most widely used learning algorithms in motion classification. Since their assumptions about feature vector fit well into motion classification, HMM based techniques generally outperform other methods. However, there are important problems that remains to be resolved with these techniques. Markov processes assume strongly conditional dependent structure between hidden random variables and observed feature sets. This leads to unrealistic independence assumptions between frames separated by even small time intervals. To address this problem, conditional random fields (CRF) is used to obtain a model with sufficiently general assumptions.

Most of the previous research in human motion classification can be categorized with respect to sensors used in
motion detection. Significant amount of researchers have proposed classifying a human motion with 2D image data. With recent improvements on BWSs and releases of powerful virtual motion modeling tools, researchers started to separate capturing a motion from categorizing it. Although the first approach is unstable when noisy sensor data is used, the second approach already assumes clear identification of the motion. As researchers specialize in motion detector systems deeper, layers of abstraction are expected to increase. Although different sensor data are used throughout the literature, most of the results indicate superiority of HMM-based methods to low level feature extraction techniques for motion classification.

Strengths of HMM based methods can be explained from a theoretical point of view as well. Time scale invariance of HMM based methods is a major advantage over other learning algorithms. Due to this feature of HMMs, several motion capture data at different frame rates can be categorized into one class label. Frame rate is not the only factor effecting the time scale. More often, people perform their actions in different speeds. Another important advantage of HMMs is its assumption of sequential data. Since the human motion is inherently sequential at time axis, HMMs generally fit well for the real phenomena.

The Overall System Design

In our implementation of human motion capture sequence annotation, we have used the mocap database of Carnegie-Mellon University. The training and the test database was constructed from randomly drawn motion sequences in CMU database. L-BFGS algorithm has been used to train the CRF parameters. An unobserved motion sequence is partitioned into fixed size time frames and input to the CRF. Famous Vitrebi algorithm described in following sections of this document, inference and training algorithms used will be explained.

Conditional Random Fields

Conditional Random Fields (CRF) is a discriminative probabilistic model which is a special case of log-linear models used for labeling and parsing sequential data introduced by Lafferty et al. (Laf01). CRF is can be represented as an undirected graphical model as in figure 1 where X denote the observation sequence to be labeled and Y denote the label sequence. Generalizing Hidden Markov Models (HMMs), CRFs outperform HMMs and other conditional markov models on a number of fields such as natural language processing and bioinformatics. Usage of conditional model relaxes the independence assumptions made by Markov models. Additionally, label-bias problem which severely influences performance of maximum entropy Markov models (MEMMs) does not affect CRFs. In this section, linear-chain conditional random fields and their usage in the context of motion labeling has been explained.

Input motion sequence is first segmented into fixed size frame chunks. Each frame chunks contains a prespecified number of time frames captured from a human. As training examples, we assume (x,y) pairs, where x denote frame chunks and y denote a sequence of corresponding tags. Log linear models express the probability of a possible output tags \( \tilde{y} \) with corresponding input motion sequence \( \tilde{x} \) as in equation 1.

\[
p(\tilde{y}|\tilde{x}; w) = \frac{\exp(\sum_j w_j F_j(\tilde{x}, \tilde{y}))}{Z(\tilde{x}, w)}
\]

In equation 1, \( F_j \)'s are feature functions and \( w_j \)'s are weights of the feature functions to be learned from training examples. Denominator \( Z(\tilde{x}, w) \) is a normalization factor for the probability to fall into [0,1]. Linear chain CRFs introduces restriction 2 over \( F_j \)'s to ensure tractable inference.

\[
F_j(\tilde{x}, \tilde{y}) = \sum_i f_j(y_{i-1}, y_i, \tilde{x}, i)
\]

\( F_j \)'s represent high level feature functions composed by \( f_j \)'s, which are low level feature function that can depend only on current label, previous label, current position i, and motion sequence \( \tilde{x} \). This restriction on log-linear models enables polynomial training and inference in CRFs. In the following sections of this document, inference and training algorithms used will be explained.

Inference in linear-chain CRFs

For a trained CRF \( C \) with weights \( w \), following rigorous mathematical derivations show us that inference on \( C \) is possible in reasonable time with dynamic programming. Our aim of inferring a label sequence for the given and unobserved motion sequence can be mathematically expressed as computing \( \tilde{y} \) in 3.

\[
\tilde{y} = \text{argmax}_{\tilde{y}} p(\tilde{y}|\tilde{x}; w) = \text{argmax}_{\tilde{y}} \sum_j w_j F_j(\tilde{x}, \tilde{y})
\]

A naive solution is here to enumerate all possible \( \tilde{y} \) sequences and calculate the probabilities. This would result in inherently exponential algorithm. For a polynomial solution, constraints of CRFs are exploited. Equation 4 is obtained by plugging in the definition of \( F_j \)'s.

\[
\tilde{y} = \text{argmax}_{\tilde{y}} \sum_j w_j \sum_i f_j(y_{i-1}, y_i, \tilde{x}, i)
\]
Since argmax function is over y, we can interchange summations and call summation \( \sum_j w_j f_j(y_{i-1}, y_i, \bar{x}, \bar{y}) \) as \( g_i(y_{i-1}, y_i) \). The equation 4 will then reduce to equation 5.

\[
\bar{y}_i = \arg\max_y \sum_i g_i(y_{i-1}, y_i) \quad (5)
\]

The motion capture data at hand can be tagged efficiently considering the intrinsic recursion over equation 5. Define \( U(k,v) \) to be the score of the best label sequence for the motion data with tags from 1 to k, where tag k has to be v. Mathematical representation of \( U(k,v) \) is as in equation 6.

\[
U(k, v) = \max_{y_{k-1}} \sum_{i=1}^{k-1} g_i(y_{i-1}, y_i) + g_k(y_{k-1}, v) \quad (6)
\]

Noticing that equation 6 can be realized as a recurrent equation 7, \( \bar{y} \) can be efficiently computed with dynamic programming.

\[
U(k, v) = \max_{y_{k-1}} U(k-1, y_{k-1}) + g_k(y_{k-1}, v) \quad (7)
\]

If length of \( \bar{x} \) is \( n \), and cardinality of set of labels is \( m \), \( \bar{y} \) can be computed in \( O(m^2n) \) time. This dynamic programming algorithm turns out to belong Viterbi class of algorithms which are also utilized in HMMs. There is also a polynomial algorithm for computing \( Z(\bar{x}, y) = \sum_{\bar{y}} \exp \sum_j w_j f_j(\bar{x}, \bar{y}) \) without enumerating all \( \bar{y} \).

**Training CRFs**

There are various algorithms used for training conditional random fields. Common training techniques for CRFs are gradient ascent algorithms, and Quasi-Newton methods. Following remarks explain the former approach relying on differentiating the Conditional Log-Likelihood (CLL). The stochastic gradient ascent considers the error with respect to only one training example at a time to converge the maximum faster. Therefore, the partial derivative for each training example all \( w_j \)’s are updated according to error of the model with respect to that training example. Partial derivative of each \( w_j \) over CLL is given in equation 8.

\[
\frac{\partial}{\partial w_j} \log(p(y|x; w)) = F_j(x, y) - \frac{\partial}{\partial w_j} \log(Z(x, w)) \quad (8)
\]

Following equation 8 by plugging in definition of \( Z(x,w) \), equation 9 has been reached.

\[
\frac{\partial}{\partial w_j} \log(p(y|x; w)) = F_j(x, y) - E_{y' \sim p(y'|x;w)}[F_j(x, y')] \quad (9)
\]

Since at global maximum of CLL gradient is zero, equation 9 gives us a rule for training CRFs. All \( w_j \)’s are updated with rule 10 where \( \alpha \) is the learning rate.

\[
w_j := w_j + \alpha(F_j(x, y) - E_{y' \sim p(y'|x;w)}[F_j(x, y')]) \quad (10)
\]

**Hypothesis Space**

Only parameters in both log-linear models and CRFs are \( w_j \)’s. Dimensionality of hypothesis space obviously depends on number of feature functions. Mathematically it is \( \mathbb{R}^d \), if the cardinality of feature function set is \( d \). In order to elaborate on features, structured input data for the CRF should be specified. Following sections discuss the data-set characteristics and feature functions.

**Dataset Selection**

The raw training set at CMU database can be represented as in table 1. Each motion sequence is simply a sequence of real numbers representing 3D outputs of the sensors placed at humans.

Notice that for a moderate frame count (fc) and sensor count (sc), there are potentially large number of feature functions for our discriminative model to be trained. In order to alleviate this problem, adjacent sensor outputs are subtracted to produce vectorial change in positions. By this operation, our implementation also gains translation invariance. Then, dot product between adjacent vectors has been computed to find the angle change between successive vectors. Figure 2 shows how the motion capture data obtained is used to calculate the angle between adjacent sensor outputs. Equation 11, shows the calculation of the angle between two motion vectors \( \bar{a} \), where \( \bar{a} = < x_2 - x_1, y_2 - y_1, z_2 - z_1 > \) and \( \bar{b} = < x_3 - x_2, y_3 - y_2, z_3 - z_2 > \). The angles obtained obviously do not represent whole characteristic information about 3D motion sequence at hand. However, scale
Two major groups, which we call single label and inter-label feature functions. As its name suggests single label feature functions can be observed by considering the current frame and the current label only. Inter-label feature functions, on the other hand, also incorporate the label assigned to the previous frame. The later type of feature functions allow dependency between adjacent labels in the label sequence. Thus the border constraint of conditional random fields is utilized without affecting tractability of the problem. Furthermore, inter-label dependency is often possible considering temporal dependencies in the activity pattern of a human. Humans next actions are generally predictable considering the action currently being performed. This section describes types of feature functions and their corresponding mathematical formulas in detail.

1. Label and Transition functions: These feature functions allow our discriminative model to encode following two probabilities: \( P(y_t|y_{t-1}) \) and \( P(y_t) \), where \( y_t \)’s represent the label assigned to \( t \)’th motion frame \( x_t \). Note that probabilities depend on labels assigned to motion sequences, but not the motion sequence itself. These feature functions allowed our model to predict commonly assigned labels to motion sequences and label transition probabilities. For all \( v, w \in Tags \), corresponding \( f_j \)’s are defined formally as in equation 12 and 13.

\[
f(y_{t-1}, y_t, \bar{x}, i) = I(y_t = v) \quad (12)
\]

\[
f(y_{t-1}, y_t, \bar{x}, i) = I(y_t = v)I(y_{t-1} = w) \quad (13)
\]

where \( I(P) = \begin{cases} 1 & \text{if } P \text{ is true} \\ 0 & \text{if otherwise} \end{cases} \)

Notice that if \( m = |Tags| \), there are \( m^2 + m \) label and transition feature functions.

2. Angle Statistics functions: This category of feature functions allow our model to consider the angular changes in motion vectors while labeling an unobserved motion sequence. For each label and each sensor output, a feature function is constructed by using first and second order statistics obtained from the sensor output. For all \( v \in Tags \) and \( k \in \mathbb{Z} \) in the interval \( [0, (fc-2)sc - 1] \), corresponding \( f_j \)’s are given in equation 14 and 15.

\[
f(y_{t-1}, y_t, \bar{x}, i) = x_i[k]I(y_t = v) \quad (14)
\]

\[
f(y_{t-1}, y_t, \bar{x}, i) = x_i[k]^2I(y_t = v) \quad (15)
\]

where \( x_i[k] \) is the angle between motion vector \( k \) and \( k-1 \) in time frame \( i \). The motion vector is calculated as the difference between adjacent sensor outputs in \( \mathbb{R}^3 \).

If the frame and sensor count is \( fc \) and \( sc \) respectively, there are \((fc-1) \times sc \) motion vectors and \((fc-2) \times sc \) angles. Therefore, there are \((fc-2) \times sc \times 2 \) angle statistics feature functions in our implementation.

3. Cross-Label Statistics functions: Although transition functions allow our model to estimate probability density function of adjacent label assignments \( P(y_t|y_{t-1}) \), inter-label dependencies are often characterized by the correlation between corresponding motion frames. Therefore, our aim in defining cross label statistics feature functions is to use statistical information about more complex transition function parametrized by \( x_t \) and \( x_{t-1} \), namely \( P(y_t|y_{t-1}; x_t, x_{t-1}) \). However probability space of \( P(y_t|y_{t-1}; x_t, x_{t-1}) \) is huge to be characterized by a set of feature functions. To tackle this problem, we consider correlation parametrized probability space instead, namely \( P(y_t|y_{t-1}; NCC(x_t, x_{t-1})) \) where \( NCC \) denotes Normalized Cross-Correlation. By using \( NCC \), our model can label the motion sequence considering the similarity between previous sequence and its tag. The reason for

<table>
<thead>
<tr>
<th>x : motion sequence</th>
<th>y : label sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x \in Chunk^* )</td>
<td>( y \in Tag^* )</td>
</tr>
<tr>
<td>Chunk ( \in Frame^{fc-2} )</td>
<td>Tag ( \in {Walk, Run, Leap,...} )</td>
</tr>
<tr>
<td>Frame ( \in Angle^{sc} )</td>
<td></td>
</tr>
<tr>
<td>Angle ( \in \mathbb{R} )</td>
<td></td>
</tr>
<tr>
<td>( StdevM \in \mathbb{R} )</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Modified training example description with angular changes in motion vectors

<table>
<thead>
<tr>
<th>x : motion sequence</th>
<th>y : label sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x \in Chunk^* )</td>
<td>( y \in Tag^* )</td>
</tr>
<tr>
<td>Chunk ( \in Frame^{fc-2} )</td>
<td>Tag ( \in {Walk, Run, Leap,...} )</td>
</tr>
<tr>
<td>Frame ( \in Angle^{sc} \times StdevM )</td>
<td></td>
</tr>
<tr>
<td>Angle ( \in \mathbb{R} )</td>
<td></td>
</tr>
<tr>
<td>( StdevM \in \mathbb{R} )</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Final dataset description including standard deviations of magnitudes per frame
normalizing the cross correlation is to equalize the angular scalers in motion vectors. The definition of feature functions are given as in the form of equation 16 for all \( v, w \in Tags \).

\[
f(y_{i-1}, y_i, \bar{x}, i) = NCC(x_i, x_{i-1})I(y_i = v)I(y_{i-1} = w)
\]

(16)

where \( x_i \) denote the angle sequence at time frame \( i \), and \( NCC(x_i, x_{i-1}) \) is the normalized cross-correlation between sequences \( x_i \) and \( x_{i-1} \) equal to the expression 17. The terms \( \eta_{x_i} \) and \( \sigma_{x_i} \) represent mean and the standard deviation of the sequence \( x_i \), respectively.

\[
\frac{1}{(fc-2)sc} \sum_j \frac{(x_i[j] - \eta_{x_i})(x_{i-1}[j] - \eta_{x_{i-1}})}{\sigma_{x_i}\sigma_{x_{i-1}}}
\]

(17)

There are \( m^2 \) cross-label statistics feature functions to be trained in our implementation of CRF, where \( m \) is the cardinality of the tag set.

4. **Magnitude Statistics functions**: There are types of labels in our dataset that differentiates the sequence by considering the magnitude of the movement. For example 'running' and 'fast walking' sequences can hardly be recognized seperately considering only the angular variations in the motion vectors. Feature functions we have formulated up to now rely on angular changes in the motion vectors for recognizing patterns in labels. Magnitude statistics feature functions, on the other hand, allow the model to use the magnitude variations. The reason we have reduced motion vectors to angles was to decrease the size of our feature function set. Not to fall into the same problem, density of \( P(y_i; \sigma_{|x_i|}) \) has been estimated instead of \( P(y_i; |x_i|) \). The definition of the magnitude statistics feature functions is given in the equation 18 for all \( v \in Tags \). Note that there are only \( |Tags| \) magnitude statistics feature functions.

\[
f(y_{i-1}, y_i, \bar{x}, i) = \sigma_{|x_i|}I(y_i = v)
\]

(18)

where \( \sigma_{|x_i|} \) denote the standard deviation in magnitudes of motion vectors at time frame \( i \).

**Evaluation of Results**

This section describes the experiments conducted on CMU database [1]. In order to supply the structured input that the CRF requires, downloaded files are preprocessed. The training examples in CMU database are variable length mocap files associated with one label. The training examples are composed of randomly sampled labels with fixed durations. The training set is composed of 621 examples whereas the size of test set is 123. Each training example can be regarded as a set of frame chunks where each chunk is 10 frame. The frame consists of sensor outputs at a specific time instant.

For evaluating the results, a set of examples drawn independently of each other and our hypothesis \( H \) has been downloaded from manually labeled CMU database. The resultant error in this test set \( S \) is stored in \( error_S(H) \). Since \( |S| \geq 30 \), the accuracy of \( H \) with respect to the data distribution \( D \) lies in confidence interval given in expression 19 with 95%.

\[
error_S(h) = \frac{1.96}{\sqrt{|S|}} \left( \frac{error_S(h)(1 - error_S(h))}{|S|} \right)
\]

(19)

Equation 19 directly follows from random example selection and central limit theorem. \( error_S(h) \) is the average loss function on the test set \( T \) is given in equation 20, where \( T \) is a set of pairs \( < x_i, y_i > \) and \( h \) is the CRF model learned from the training set.

The figure 3 shows the result of a single experiment where the training example consists of 3 chunks. The algorithm we have proposed have successfully computed the corresponding tags, which are given in the figure.

Three algorithms proposed for motion classification have been implemented and compared to our algorithm. The two of the proposed algorithms models the human motion with HMMs. Without experiment, the intuitive outcome of the comparison with these algorithms is the prevalence of the CRF based approach we have used. The third algorithm is based on motion template representation of the labels.

When sufficiently large number of training examples are randomly sampled from the entire distribution, two types of feature functions dominates the final decision on labeling. These are angle & magnitude statistics feature functions. The information gain obtained by using these feature functions are given in figure 4 and 5. Figures show the change in information gain for labels "run" and "walk". The obvious increase in the information gain in the "run" label stems from the fact that the running activity can be characterized by considering the magnitude variation in the activity.

**Discussion & Conclusion**

In this study, we have suggested an algorithm to annotate a human motion capture data. The CRF is known to be the
best performing algorithm on sequential data labeling where there are dependencies between adjacent labels. Our proposed feature functions capture the intrinsic information in human motion data such as angular variations, label transition probabilities, and the magnitude changes in sensor outputs. The cardinality of feature function set does not interfere the tractability of the labeling process even with moderately large frame count.

The proposed method outperformed all other algorithms suggested in the literature. High detection performance can be utilized to create motion capture databases without manual tagging.

References