Segmenting Human Motion for Automated Rehabilitation Exercise Analysis

Jonathan Feng-Shun Lin and Dana Kulić

Abstract— This paper proposes an approach for the automated segmentation and identification of movement segments from continuous time series data of human movement, collected through motion capture of ambulatory sensors. The proposed approach uses a two stage identification and recognition process, based on velocity and stochastic modeling of each motion to be identified. In the first stage, motion segment candidates are identified based on a unique sequence of velocity features such as velocity peaks and zero velocity crossings. In the second stage, Hidden Markov models are used to accurately identify segment locations from the identified candidates. The approach is capable of on-line segmentation and identification, enabling interactive feedback in rehabilitation applications. The approach is validated on a rehabilitation movement dataset, and achieves a segmentation accuracy of 89%.

I. INTRODUCTION

Physical rehabilitation is a branch of modern health care that focuses on the development, maintenance and restoration of body movement and function, particularly after injury or surgery. A rehabilitation session consists of a physiotherapist's assessment of the patient's current condition, as well as the performance of physical exercises recommended by the physiotherapist. Typically, the physiotherapist will supervise the performance of the exercises to determine patient progress, as well as provide corrective feedback. Technology to measure and analyze human motion has the potential to provide the physiotherapist with more accurate tools for assessment and progress measurement, as well as to provide the patient with real-time feedback.

To enable automated measurement and analysis, the system must measure the human movement and identify exercise movement segments from the time series data. Human movement can be measured via either motion capture systems [1], [2] or ambulatory sensors such as inertial measurement units (IMUs) [3]–[5]. Given the measured time series data, *segmentation* is the process of identifying the starting and ending locations of each movement of interest. If the patient is performing more than one type of exercise in a given recording session, *identification* (i.e., labeling) of each segment with the appropriate exercise type is also required. Both segmentation and identification are made more difficult due to the variability observed in human movement. Motion can vary between individuals due to differing kinematic or dynamic characteristics between individuals, and as well within a single individual over time, due to short term factors such as fatigue, or long term factors such as recovery or disease progression. Moreover, these factors can introduce both spatial and temporal variability. Thus, for rehabilitation, a good segmentation algorithm must be able to handle both temporal and spatial variations. A computationally light algorithm is also desired, in order to perform segmentation on-line, to allow real-time feedback to the patients.

Time series data segmentation for human movement is an active area of research. A common approach is to rely on known motion templates to assist in the identification of motions. Dynamic Time Warping (DTW) is one example of a template-based method. DTW [6] identifies the temporal variations between the observed motion and the motion template by selectively warping the time scale of an observation sequence to the template, based on a distance metric. Therefore segments can be accurately segmented and identified even with significant spatial and temporal variation between the observation and the template. However, DTW becomes very computationally expensive with higher dimensionality, preventing it from being utilized on-line.

If the motions to be observed are not known *a priori*, then non-template based methods are required. The Zero-Velocity Crossing (ZVC) approach identifies points where the velocity value changes sign, i.e., when a joint segment changes directions [7], as segment points. Although a fast algorithm, ZVC tends to over-segment, particularly with noisy data or as the number of DoFs increases. Since ZVC methods do not consider motion templates, it is difficult to tell which crossing points can be safely ignored.

An alternative method is to employ probabilistic algorithms, and look at changes in signal variance [8] or probability distribution [9] to indicate segmentation points. On-line template construction has also been proposed [10], by clustering together previously segmented sequences to generate new templates on the fly.

In clinical settings, the patient's exercises are prescribed by the therapist, and thus templates can be utilized. Templates encoded as Hidden Markov models (HMMs) can be used as to assess the similarity between a template and an observation sequence. The observation data can be split into smaller components, and recombined based on principle component analysis coefficients, and compared against a HMM [11]. Alternatively, the templates could be used as a weak classifier in a larger algorithm, such as AdaBoost [2].

This paper proposes a template-based on-line technique that combines the ZVC and the HMM techniques. The system consists of a training phase, where exemplar data are used to create feature and HMM motion templates, and a segmentation phase, where the observed data is swept for

J. F. Lin and D. Kulić are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2L 3G1, Canada (email: jf2lin@uwaterloo.ca; dkulic@ece.uwaterloo.ca)

characteristic features that matches the feature templates. When a feature match is found, the HMM template is used to determine the closeness of match and remove similar looking motions. The initial framework in [12] is expanded with a generalized approach for constructing templates capable of handling arbitrary DoF motion, where any number of DoFs may be moving simultaneously. The algorithm is tested on the motions of five subjects, performing three typical rehabilitation motions.

II. PROPOSED APPROACH

In order to meet the time constraints of a real-time system, the segmentation and identification algorithm should be as computationally efficient as possible. The feature-guided HMM [12] is a fast algorithm that employs a 2-tier approach to motion identification. The observation data is first scanned for characteristic features, consisting of velocity peaks and ZVCs, to estimate the locations of segment candidates. This reduces the number of times that the more computationally expensive but accurate HMM algorithm is required to run, thus reducing the computational costs of the algorithm.

A. Hidden Markov model

The Hidden Markov model [13] is a stochastic model where the process being modeled is represented by an evolving unobservable state. This underlying state is inferred by the probabilistic relationship between the hidden state and its corresponding observable output. The state has the Markov property, i.e., the next state depends only on the current state. The HMM stores its model information as a set of three variables: the initial state distribution, the state transition matrix, and the observation distribution. For continuous observation variables, such as human motion data, Gaussian or mixture of Gaussians distributions are used.

To assess the similarity between a trained model and a new observation sequence, the forward algorithm [13] is applied. The forward algorithm calculates the likelihood that the observation data could have been generated by the model.

B. Template training

To reduce the need to apply the forward algorithm to each segment candidate, a two stage recognition process is applied. For each motion, two types of templates are prepared: a feature profile and an HMM profile. Given an exemplar motion from which a template is to be created, the feature extraction notes the locations of velocity crossings and velocity peaks of the exemplars. Since the motions examined in this paper are rehabilitation exercises and thus exhibit regular patterns of flexion and extension cycles, the feature template would be expected to consist of a ZVC, then a positive or negative peak, then another ZVC, the opposite peak, and a final ZVC. The magnitudes of the peaks (v_p) and peak-to-peak times (t_{pp}) are also stored in order to reject insignificant motions from triggering a feature match. Only significant DoFs of the motion are used for template matching at the feature stage. Significant DoFs are identified by calculating the standard deviation of each of the joints in

the template and grouping them via 2-means clustering, and selecting the DoFs grouped with the higher centroid.

Instead of expecting a specific set of ZVCs and velocity peaks, as was proposed in [12], the template training sequence now searches for ZVCs in the template, and characterizes the peaks between each ZVC. If the velocity peaks between two ZVCs are small, defined as a percentage of the maximum or minimum peak over the whole exemplar, that peak is rejected as a feature. Each of the exemplar motions will have its features extracted in this manner, and the template characteristics, that is, its sequence of velocity peak directions and ZVCs, over each of the exemplars are compared. If a majority of the exemplars have identical template characteristics, then that characteristic sequence is used. Next, the velocities of the significant DoFs for a given template are multiplied together, to create a scalar estimate of the overall velocity and estimate the ZVCs and velocity peaks. With this approach, the same algorithm can be used regardless of the dimensionality of the input. It is also simpler to threshold on a single aggregated velocity signal instead of on several independent DoFs.

The exemplars are also used to train HMMs with the Baum-Welch algorithm [13]. An 8-state left-right model is used. The Gaussian observation functions are initialized by k-means clustering. The threshold for recognition (T_R) is also determined, via leave-one-out cross-validation (LOOCV).

C. Feature Guided Segmentation

During the on-line segmentation phase, a small sliding window is passed over the observation data, noting the local peak values and ZVCs of each of the DoFs. A ZVC is declared if the velocity makes a zero-crossing, or if is very low for several timesteps. Local peak values are tracked by an internal buffer. If the current window has a peak value higher then the stored peak value, the peak value in the buffer is updated accordingly. To avoid noise spikes in the velocity data from affecting the template matching, the peak buffer value is attenuated if it does not contribute to a match after several seconds, to prevent a large spike in the velocity from preventing feature matches. If a given DoF observes a sequence of ZVCs and peaks that matches a known template, then the algorithm has located a potential segment point.

The velocity magnitudes and peak-to-peak distance must exceed v_p and t_{pp} respectively in order for a potential segment point to be declared. This prevents noise, such as when the subject is stationary, from triggering the feature match. The template and window edge combination that results in the highest likelihood value over the threshold T_R is declared a segment. All window edge combinations are resampled so they are all equal length, to prevent the forward algorithm from favouring shorter sequences. Following HMM template matching, the recorded peak magnitude and ZVCs are reset, and the feature search resumes at the next time step.

III. EXPERIMENTS AND RESULTS

The algorithm was tested on five subjects performing three types of physiotherapy exercises: knee extensions while
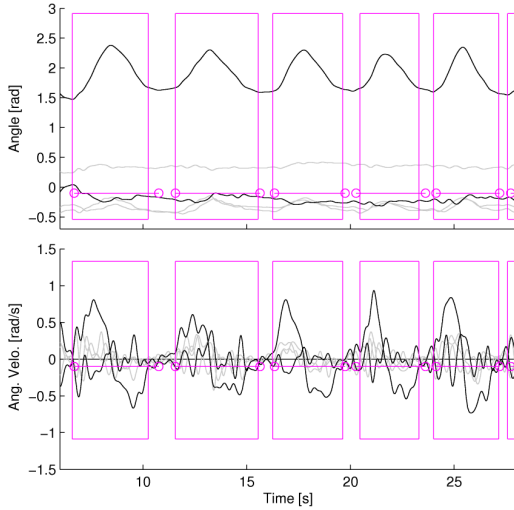
Fig. 1. Joint angles (top) and angular velocity (bottom) of a subject performing hip flexions. Despite the amount of noise in the other DoFs, the algorithm is successfully segmentation the motion under examination.
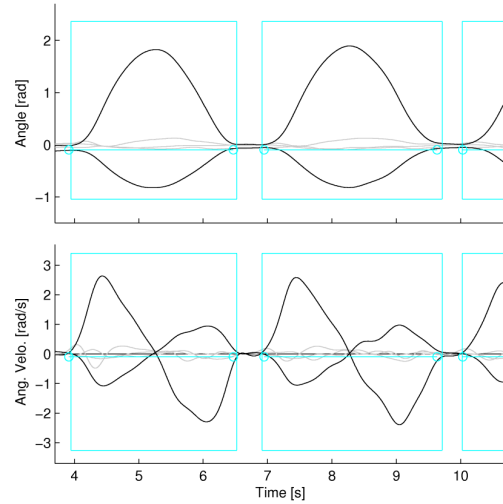

Fig. 2. Joint angles (top) and angular velocity (bottom) of a subject performing squats. The boxes indicate the algorithmic segmentation points, whereas the circles indicate manual segmentation points. The black lines indicate significant DoFs, whereas the gray lines are not significant DoFs.

TABLE I
SEGMENTATION RESULTS, WITH $t_{error} = 0.20s$

	Total	Correct	False positive	False negative
Knee extension	206	188 (91%)	2	18
Hip flexion	218	187 (86%)	2	31
Squats	200	179 (90%)	0	21
Total	624	554 (89%)	4	70

seated, squats, and hip flexions while supine. The average age of the participants was 22 years old. The experiment was approved by the University of Waterloo Research Ethics Board, and signed consent was obtained from all participants.

Each motion was performed twice, with ten repetitions each. For each subject, the first motion set was used for the template training, then both motion sets were used for testing the proposed algorithm. The templates used were subject specific, so the templates generated from one subject's motions was used to segment only their own motion.

The data was recorded with a set of IMUs, and translated to joint angles via an extended Kalman filter. The subjects were modeled as a 5 DoF system. Motion capture data was collected simultaneously; ground truth manual segmentation was determined by a human observer using video playback of the motion capture data.

In order to compare the proposed approach to existing work, a ZVC method, described in [14] was implemented. Segmentation points are declared when the velocity crosses zero. To reduce the amount of spurious ZVCs, only ZVCs that occurred on the significant DoFs were declared. A fixed-sliding window HMM was also implemented. HMM construction for the fixed sliding window was identical to the feature guided HMM. The fixed window length was computed from the lengths of the exemplar templates. Segmentation points were declared on local maximas of the likelihood, as long as it is above some likelihood threshold.

All processing and algorithmic implementation were done in MATLAB 7.12. The HMM functions were implemented with Murphy's HMM MATLAB Toolbox [15].

A. Segmentation results

A algorithmic segmentation point was declared correct if it falls within $\pm t_{error}$ of a manual segment point. It is a false positive if an algorithmic segment point was declared when there is not one. It is a false negative if a segment point was not declared when there should be one. Each segment's two segment points are awarded points separately. That is, a completely correct segment could receive +2 correct, whereas a partially correct segment could receive +1 correct and +1 false negative. The segmentation results for the feature-guided HMM with $t_{error} = 0.2s$ can be found in Table I, and show that the algorithm correctly determined 89% of the segments. If $t_{error} = 0.3s$, the accuracy increases to 93%, implying that a number of the algorithmic segments are just outside of the manual segment error bounds.

Fig. 1 and 2 show examples of the segmentation algorithm. The boxes indicate the algorithmic segmentation points, whereas the circles indicate manual segmentation points. Fig. 1, showing hip flexions, demonstrates that the algorithm is able to produce good segmentation results even when large

TABLE II
SEGMENTATION RESULTS COMPARED TO OTHER ALGORITHMS. TOTAL
SEGMENT POINTS = 624

t_{error}	Algorithm	Correct	False Pos.	False negative
0.2	ZVC	303 (49%)	1032	429
	Fixed HMM	78 (13%)	408	546
	Feature HMM	554 (89%)	4	70
0.3	ZVC	309 (50%)	1080	365
	Fixed HMM	95 (15%)	392	529
	Feature HMM	580 (93%)	0	44

motions are occurring in the other DoFs. It can be noted that the manual segmentation points here exhibit significant offset from the algorithmic segmentation points, causing these points to be declared as false negatives. Several different factors contribute to this. Since there are no standardized guidelines on how a motion sequence should be segmented, manual segmentation is typically left up to the perception of the human observer. Although manual segment points are typically used as ground truth, there may be inconsistencies between observers. It is difficult to visually determine when a segment has started or ended, due to stray motion or tremors at segment ends, causing manual segments to be declared after the actual segmentation point. Fig. 2 shows that sometimes the algorithm overestimates the bounds of the segments, as the motion may have been perceived to have ended due to slow velocity before the actual ZVC.

The feature-guided HMM compares favourably against previous algorithms. Table II shows the feature-guided HMM's error metrics compared against ZVC and fixed-window HMM, at different t_{error}. Although feature-guided HMM segments along ZVC points, its ability to reject spurious crossings as opposed to the ZVC algorithm's naive approach greatly improves the segmentation accuracy and reduces the number of false positives. Fixed-window HMM performed poorly. Its inability to change its window size, which is the average length of the templates, means that even though it is good at identifying the underlying motion, it is not able to suggest accurate segmentation bounds.

B. Timing results

Table III shows the timing results for each of the examined algorithms. The exemplar length is the average length of each observation sequence. The template construction time is the average time taken to construct all relevant template data for a given subject. The segmentation time is the average time required to segment one set of observation data.

Due to its simplicity, the ZVC algorithm requires no template training time and very little segmentation time. However, as noted in the previous section, this algorithm is very inaccurate. Fixed-window HMM uses the Baum-Welch algorithm to train the HMMs based on exemplar data, and requires a significant amount of training time. Its segmentation time is also very long, as it needs to run the forward algorithm numerous times at each time step, once for each template available. The feature-guided HMM requires the most training time. Although the HMM training component is identical to the one utilized the fixed-window HMM, the additional training time comes mainly from the LOOCV method to determine likelihood thresholds. The feature extraction component takes very little time. With the feature-guiding, the proposed algorithm is able to more intelligently determine when to apply the forward algorithm, and decrease the segmentation runtime significantly.

IV. CONCLUSIONS AND FUTURE WORKS

Feature-guided HMM segments data with high accuracy. It does so by reducing the comparison space of the observation

TABLE III
TIMING RESULTS COMPARED TO OTHER ALGORITHMS

	Exemplar length	Template construction	Segmentation time
ZVC	37.44	0.00	0.13
Fixed HMM	37.44	26.28	63.30
Feature HMM	37.44	249.42	4.35

data by looking at significant DoFs, velocity peaks and ZVCs, to estimate locations of segment potentials. When such segment potentials are located, the sequence under examination is checked with against an HMM to find the most accurate segmentation and identify the motion.

We plan to apply the algorithm to a larger dataset to assess the algorithm against a wide set of motion types. A specific weakness of the current implementation is that subject-specific templates are used. This is impractical in clinical settings, as it would require that each patient provide a template *a priori*. In future work, we will test the algorithm by using a single set of templates against the motion data of other multiple users, in order to assess how well this algorithm scales against interpersonal variations.

REFERENCES

[1] W. Ilg, G. H. Bakir, J. Mezger, and M. A. Giese, "On the representation, learning and transfer of spatio-temporal movement characteristics," *Int. J. Humanoid Robotics*, vol. 1, no. 4, pp. 613–636, 2004.
[2] F. Lv and R. Nevatia, "Recognition and segmentation of 3-d human action using hmm and multi-class adaboost," in *European Conf. on Computer Vision*, vol. 4, 2006, pp. 359–372.
[3] K. Lorincz, B.-R. Chen, G. W. Challen, A. R. Chowdhury, S. P., P. Bonato, and M. Welsh, "Mercury: A wearable sensor network platform for high-fidelity motion analysis," in *Proc. ACM Conf. on Embedded Networked Sensor Systems*, 2009.
[4] D. Roetenberg, P. J. Slycke, and P. H. Veltink, "Ambulatory position and orientation tracking fusing magnetic and inertial sensing," *IEEE Trans. Biomed. Eng.*, vol. 54, pp. 883–890, 2007.
[5] H. Zhou, H. Hu, N. D. Harris, and J. Hammerton, "Applications of wearable inertial sensors in estimation of upper limb movements," *Biomedical Signal Processing and Control*, vol. 1, pp. 22–32, 2006.
[6] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.
[7] M. Pomplun and M. Matarić, "Evaluation metrics and results of human arm movement imitation," in *RAS Conf. Humanoid Robotics*, 2000.
[8] N. Koenig and M. J. Matarić, "Behaviour-based segmentation of demonstrated tasks," in *Int. Conf. Development and Learning*, 2006.
[9] J. Kohlmorgen and S. Lemm, "A dynamic hmm for on-line segmentation of sequential data," in *Adv. Neural Inform. Process. Syst.*, vol. 14, 2002, pp. 793–800.
[10] D. Kulić, W. Takano, and Y. Nakamura, "Online segmentation and clustering from continuous observation of whole body motions," *IEEE Trans. Robotics*, vol. 25, no. 5, pp. 1158–1166, 2009.
[11] F. Bashir, W. Qu, A. Khokhar, and D. Schonfeld, "Hmm-based motion recognition system using segmented pca," in *IEEE Int. Conf. Image Processing*, vol. 3, 2005, pp. 1288–91.
[12] J. F.-S. Lin and D. Kulić, "Automatic human motion segmentation and identification using feature guided hmm for physical rehabilitation exercises," in *IEEE Int. Conf. Intelligent Robots and System*, 2011.
[13] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
[14] A. Fod, M. J. Matarić, and O. C. Jenkins, "Automated derivation of primitives for movement classification," *Autonomous Robots*, vol. 12, no. 1, pp. 39–54, 2002.
[15] K. Murphy, "Bayes net toolbox for matlab," 1998, last accessed on 2012/04/27. [Online]. Available: code.google.com/p/bnt/