Towards Lifelong Learning and Organization of Whole Body Motion Patterns

Dana Kulić, Wataru Takano and Yoshihiko Nakamura[1]

Department of Mechano-Informatics, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan
{dana,takano,nakamura}@ynl.t.u-tokyo.ac.jp

Summary. This paper describes a novel approach for incremental learning of motion pattern primitives through long-term observation of human motion. Human motion patterns are abstracted into a stochastic model representation, which can be used for both subsequent motion recognition and generation. The model size is adaptable based on the discrimination requirements in the associated region of the current knowledge base. As new motion patterns are observed, they are incrementally grouped together based on their relative distance in the model space. The resulting representation of the knowledge domain is a tree structure, with specialized motions at the tree leaves, and generalized motions closer to the root. Tests with motion capture data for a variety of motion primitives demonstrate the efficacy of the algorithm.

1 Introduction

Learning from observation is an attractive proposition for humanoid robots, as the similar body structure to humans can be utilized to bootstrap learning. A variety of algorithms have been proposed for learning human motions through demonstration and imitation [3, 16]. However, most of these approaches consider the case where the number of actions to be learned are specified by the designer, the demonstrated actions are observed and clustered a priori, and the learning is a one shot, off-line process. In this case, there is no need to classify the observed motions. Alternatively, if all the training examples are available off-line, a global clustering method can be applied once all the data has been acquired, to determine the optimum number of motion primitives, and the allocation of the data to each primitive. However, a robot which is an inhabitant of the human environment should be capable of continuous learning over its' entire lifespan. The robot should be able to observe, segment and classify demonstrated actions on-line during co-location and interaction with the (human) teacher. During this type of learning, the number of motion primitives is not known in advance and may be continuously growing, and must be determined autonomously by the robot, as it is observing the motions. In addition, as the number of observed motions and learned motion primitives increases, the robot must be able to organize the acquired knowledge in an efficient and easily searchable manner.

Due to the fact that motions are being sorted incrementally, before all the motions are known, there will be a tradeoff between classification accuracy and the number of training examples. As more examples become available, a more accurate clustering can be achieved. However, if the robot is able to extract appropriate motion primitives quickly, after only a few examples are shown, the learned motion primitives can immediately be used for motion recognition and generation. Once the robot can generate a learned motion, the learned motion can be further refined through other learning modalities, such as practice [1] and feedback from the teacher [13], which may be more effective than repeated observation alone. Therefore, we seek an algorithm which will accurately cluster with few examples. The ability to cluster quickly and accurately will depend on the similarity between the motions to be clustered, and the accuracy of the model used to represent each motion. For dissimilar motions, even a simple model can be successfully used, while for similar motions, a higher accuracy model is required to distinguish between motions. However, a higher accuracy model requires more memory and time resources. Therefore, it would be preferable if the model choice could be adaptable to the current level of similarity of the motions in the relevant region of the knowledge base.

In order to extract motion primitives during on-line observation, several key issues must be addressed by the learning system: automated motion segmentation, recognition of previously learned motions, automatic clustering and learning of new motions, and the organization of the learned data into a storage system which allows for easy data retrieval. In this paper, our focus is on the last three items.

Breazeal and Scasellati [3] and Schaal et al. [16] provide reviews on motion learning by imitation. As noted by Breazeal and Scasellati, the majority of algorithms discussed in the literature assume that the motions to be learned are segmented a-priori, and that the model training takes place off-line. For example, Billard et al. [2] use HMM models for motion recognition and generation. The Bayesian Information Criterion (BIC) is used to select the optimal number of states for the HMM. However, all the exemplar motion patterns are acquired and grouped before the training begins, and the number of motions to be learned is specified a priori.

Ogata et al. [14] develop a connectionist architecture suitable for long term, incremental learning. In their work, a neural network is used to learn a navigation task during cooperative task execution with a human partner. However, in their implementation, the robot learns only one task, and no hierarchical organization of knowledge takes place.

Kadone and Nakamura [8, 9] describe a system for automated segmentation, memorization, recognition and abstraction of human motions based on associative neural networks with non-monotonic sigmoid functions. However, the abstracted motion representation can only be used for subsequent motion recognition, and cannot be used for motion generation.

Takano and Nakamura [17] develop a system for automated segmentation, recognition and generation of human motions based on Hidden Markov Models. In their approach, a set of HMMs is trained incrementally, based on automatically segmented data. Each new motion sequence is added to the HMM which has the highest likelihood of generating the motion. However, no mechanism is proposed for the emergence of a hierarchy among different levels of abstraction.

Another important issue during model learning is the selection of the mode size. The use of the BIC [2] or Akaike criterion [10] have been proposed, however, both these criteria are based on a tradeoff between model performance at characterizing the observations, and the number of parameters, and both require a time-consuming search of the model space to find the best matching model. In the motion recognition domain, the model size required depends not only on the model performance for the current observation, but also on the structure of the knowledge database itself. If there are many similar motions in the database, a more accurate model is required, so that they can be easier discriminated. On the other hand, if motions are dissimilar, a very simple model can easily discriminate between them.

In this paper, a variable structure Hidden Markov Model based representation is used to abstract motion patterns as they are perceived. Individual motion patterns are then clustered in an incremental fashion, based on intra model distances. The resulting clusters are then used to form a group model, which can be used for motion generation. The model size is adjusted automatically on-line, based on the accuracy requirements in the given region of the knowledge space.

2 Incremental Behavior Learning

In the proposed approach, a hierarchical tree structure is incrementally formed representing the motions learned by the robot. Each node in the tree represents a motion primitive, which can be used to recognize a similar motion, and also to generate the corresponding motion for the robot. Within each local area of the motion space, a standard clustering technique is used to subdivide motion primitives. This component of our approach is based on the long standing area of research in clustering analysis [7], in particular to dynamic cluster algorithms [5] and adaptive dynamic clusters [4]. A Hidden Markov Model is used to abstract the observation sequences. The parameters of the model form the feature set of the data. These features are then used to define a distance measure between observation sequences, which is used for clustering. Rather than using a fixed size model for the motions, the model accuracy is adjusted based on the recognition requirements in the given region of the knowledge database. Initially, each motion and motion group is encoded as a simple

Hidden Markov model, with few states. As the required model complexity increases, additional dynamic chains are added to the model, to form Factorial Hidden Markov Models [6](See Fig. 1). FHMMs can provide a more accurate representation of the motion, and enable better discrimination ability in areas of the knowledge base where there are many similar motions, without being prone to overfitting [6, 11]. In addition, it has been shown that the FHMM and single chain HMM models of the same motion remain sufficiently similar, so that FHMM models may be used simultaneously with HMM models, by using FHMM models only in dense areas of the motion model space where better discriminative ability is required. This is illustrated schematically in Fig. 2.

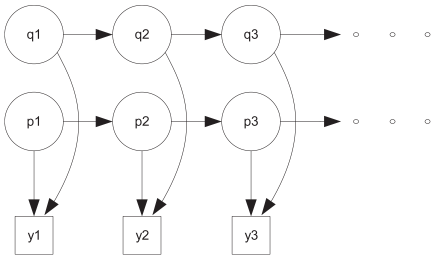


Fig. 1. Factorial Hidden Markov Model

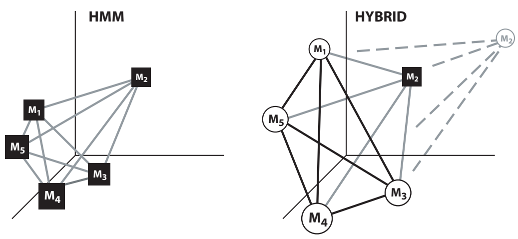
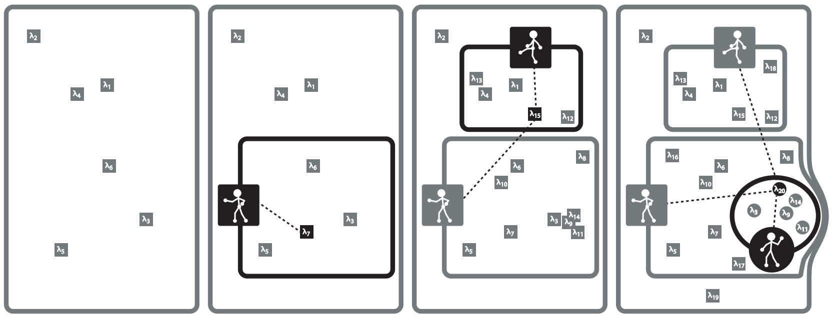


Fig. 2. Schematic comparing an HMM model space and a hybrid HMM-FHMM model space

The algorithm initially begins with one behavior group (the root node). Each time a motion is observed from the teacher, it is encoded into an HMM and compared to existing behavior groups via a tree search algorithm, and placed into the closest group. Each time a group is modified, local clustering is performed within the exemplars of the group. If a a cluster with sufficiently similar data is found, a child group is formed with this data subset. Therefore the algorithm incrementally learns and organizes the motion primitive space, based on the robot's lifetime observations. The algorithm pseudocode is shown in Figure 4, while a schematic of the incremental memory structure formation is shown in Fig. 3.



(a) (b) (c) (d)

Fig. 3. Schematic Illustration of the Segmenting Algorithm. (a) initial state, when only one group is present; (b) a child group forms when enough similar examples are observed; (c) new observations are located into the closest group based on the distance between the new observation and the group model; (d) a higher order model is used in dense areas of the motion space

procedure INCREMENTALCLUSTER

- **Step1** Encode observation sequence O_i into an HMM
- **Step2** Search the behavior tree for the closest group λ_{Gj} to the current observation model λ_i, based on the inter-model distance
- **Step3** Place λ_i into the closest group G_c
- **Step4** Perform clustering on all the exemplar motions within G_c
- **Step5** If a sufficiently similar subgroup of motions is found, form a new group G_n, as a child of G_c, containing the observation sequences of the subgroup
- **Step6** Using the observations sequences of the new subgroup, form the group model λ_{Gn}

end procedure

Fig. 4. Segmenting Algorithm Pseudocode

This algorithm allows the robot to incrementally learn and classify behaviors observed during continuous observation of a human demonstrator. The generation of a hierarchical structure of the learned behaviors allows for easier retrieval, and the automatic generation of the relationships between behaviors based on their similarity and inheritance. In addition, the robot's knowledge is organized based on the type of training received, so that the robot's knowledge will be most specialized in those areas of the behavior space where the most data has been observed.

2.1 Observation Sequence Encoding

Each newly acquired observation sequence is encoded into a Hidden Markov Model. In order to train the model, the HMM parameters, such as the number of states and the number of gaussian mixtures must be selected. In previous work, we have used the Akaike Information Criterion (AIC) to select the best fitting model [10]. However, this approach can be time consuming, as it requires training each model candidate, and performing an exhaustive search of the model space. Using the AIC also does not consider the need for a better model when many similar motions need to be distinguished. Instead of using a fixed size model determined by the AIC, in this paper we propose using a variable size model, where the number of dynamics chains in an FHMM model are increased based on the density of the motion exemplars in the relevant region of the motion space. With this approach, each motion is initially represented by a simple, single-chain, front-to-back HMM. If a better model is required, additional chain(s) are added as described below.

2.2 Intra-Model Distance calculation

Once the newly observed behavior is encoded as an HMM, it is compared to existing groups (if any). Here, the distance between two models can be calculated [15] by Equation 1.

$$D(\lambda_1, \lambda_2) = \frac{1}{T}[logP(O^{(2)}|\lambda_1) - logP(O^{(2)}|\lambda_2)] \tag{1}$$

where λ_1, λ_2 are two models, $O^{(2)}$ is an observation sequence *generated* by λ_2 and T is the length of the observation sequence. Since this measure is not symmetric, the average of the two intra distances is used to form a symmetric measure. This distance measure is based on the relative log likelihood that a generated sequence is generated by one model, as compared to a second model. It represents a Kullback-Leibler distance between the two models. The formulation of the distance based on the model probability means that this measure can similarly be applied to Factorial HMM models, by using the modified forward procedure [6] to calculate the log likelihood, as well as used to compare FHMM and HMM models [10].

The repository of known groups is organized in a tree structure, so that the new observation sequence does not need to be compared to all known behaviors. The comparison procedure is implemented as a tree search. At each node of the tree, the new observation sequence is compared to the leaves of that node. If the distance between the new observation sequence and one of the child nodes is sufficiently small, the search recurses to the most similar child node, otherwise, the new observation sequence is added to the current node.

$$D_{thresh} = K_{maxGD} D^{G}_{max} \tag{2}$$

D_{thresh} is the distance threshold at which a new observation sequence is considered for inclusion to a node, K_{maxGD} is the multiplication factor applied and D^G_{max} is the maximum intra observation distance for the given node. If the distance between the new observation and the cluster is larger than D_{thresh}, this cluster will not be considered as a possible match for the new observation sequence. If there are multiple candidate clusters, the new sequence is placed in the closest cluster. If there are no candidates, the new sequence is placed in the parent cluster. In the case of a new motion pattern which is completely dissimilar to any existing motion patterns, the motion pattern will be placed into the root node.

The maximum intra observation distance for the placement node D^G_{max} is also the criterion used to decide the level of model complexity required for the motion sequence. If the new motion is most similar to a node which D^G_{max} falls below a certain threshold, the FHMM model is generated by adding additional chain(s) to the current representation, to increase the discriminative ability of the model.

2.3 Clustering and New Group Formation

When a new observation sequence is added to a group, a clustering procedure is invoked on that group, to determine if a subgroup may be formed. The complete link hierarchical clustering algorithm is used to generate the hierarchical tree structure within a group [7]. Clusters are formed based on two criteria: number of leaves in the subgroup, and the maximum proximity measure of the potential subgroup. To calculate the maximum distance measure, the average and standard deviation of the inter motion distances in the cluster is calculated. The distance cutoff is then calculated as a function of the distribution function:

$$D_{cutoff} = K_{cutoff}\mu \tag{3}$$

where D_{cutoff} is the distance cutoff value (i.e., only clusters where the maximum distance is less than this value will be formed), and μ is the average distance between observations.

2.4 New Behavior Instantiation

If a new subgroup is generated in Step 5, a new group model is trained using the raw observation sequences from all the group elements. The structure of the new group model is determined based on the maximum intra observation distance for group, D^G_{max}. The generated model is subsequently used by the robot to generate behaviors. The group model replaces the individual observations in the parent node.

If one of the group elements allocated to the new cluster is already a group model, the generated motion sequence based on that model is used for the training. In this case, a modified form of the re-estimation formulas for multiple observation sequences [15] is used. The algorithm is modified by over-weighting the group models, in order to account for the fact that there are multiple observation sequences stored in the generated model, and therefore more weight should be given to the group model, as compared to the individual observation sequences.

2.5 Computational Efficiency and Memory Usage

The developed algorithm is suitable for on-line behavior acquisition, as the computational requirements are significantly lower as compared to a global clustering approach. There is no need for an exhaustive search in the model space, as the model size is adjusted on-line based on the similarity between the new motion and previously known motions. A larger model is only utilized when motions are similar to each other, and better discriminative ability is required. Each new sequence is compared to known sequences via tree search, reducing the number of comparisons required. Once the closest node is found, the computation time for node clustering is constant (only the models in the closest node are clustered). Each cluster is limited to a maximum number of observation sequences, N_{max}. If a new observation sequence is being added to a cluster which already contains N_{max}, an old observation sequence is selected for removal from the cluster, before adding the new observation. Currently, the observation sequence to be removed is selected based on FIFO (first in, first out).

In the developed algorithm, the number of motion primitives is not defined a priori, but emerges over time as the robot observes new motion data and builds the tree structure. In practice, the maximum number of motion primitives must be limited by the memory resources. To conserve memory resources while remembering learned motion primitives, it is also possible to amalgamate leaf clusters into a single observation (i.e., do not store any of the constituent observation sequences for a leaf cluster, but only the resulting group HMM). However, following amalgamation, it would no longer be possible to subdivide that cluster, i.e., it would be considered as a single observation in the parent cluster. Therefore, amalgamation could be performed once the tree hierarchy had reached a certain depth level, or once the distance between cluster elements has become sufficiently small.

2.6 Motion Generation

Once a cluster node has been formed, the group model for the node constitutes the abstraction of the motion primitive. To generate a motion trajectory for the robot from the group model, the deterministic motion generation method is used [11]. In this method, at each time step, the state duration is first estimated from the state transition model, and the subsequent state is selected by a greedy policy. The output observation vector is then generated by a greedy policy on the output model. The resulting reference trajectory is then low-pass filtered and passed to a low level controller, to ensure that dynamic and stability constraints are satisfied.

3 Experiments

The algorithm was tested on a human motion data set containing a series of 9 different human movement observation sequences obtained through a motion capture system [12, 8]. The motion capture system [12] captures the position of markers located on the body, and performs inverse kinematics computations to convert the data to joint angle positions in real time. A 20 degree of freedom humanoid model is used for the inverse kinematics computations. The data set contains joint angle data from multiple observations of walking (WA - 28 observations), cheering (CH - 15 observations), dancing (DA - 7 observations), kicking (KI - 19 observations), punching (PU - 14 observations), sumo leg raise motion (SL - 13 observations), squatting (SQ - 13 observations), throwing (TH - 13 observations) and bowing (BO - 15 observations).

The motion sequences are presented to the algorithm in random order. Motion sequences are presented one at a time, simulating on-line, sequential acquisition. After each motion is presented, the algorithm is executed, performing incremental clustering. In each simulation performed, the algorithm correctly segments the behaviors such that the resulting leaf nodes represent the grouping that would be obtained with an off-line method. Out of 100 simulation runs performed, there was no cases of misclassification at the leaf nodes, showing that the final segmentation is robust to presentation order. A comparison of the clustering performance between using only HMM models, and using adaptable models is shown in Fig. 5. The adaptable model can distinguish between similar motions TH and PU, whereas those motions cannot be distinguished given the same number of examples, when only single chain HMM models are used. Note that the actual order of node formation will vary depending on the motion presentation order.

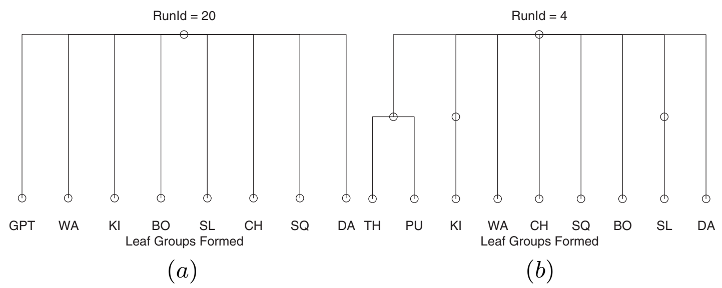


Fig. 5. Sample Segmentation Results: (a) using only single-chain model, (b) using adaptable models

Figure 6 shows a comparison between motion encoded by the two chain FHMM and the single chain HMM before any trajectory post processing has been applied, for the walk subgroup. As can be seen in Figure 6, due to the higher number of states available to represent the motion, FHMMs achieve better spacial accuracy compared to a single chain HMM model. An example of a mid-level node abstracting the punch and throw motions is shown in Figure 7. The abstracted motion is an averaging of the two motions.

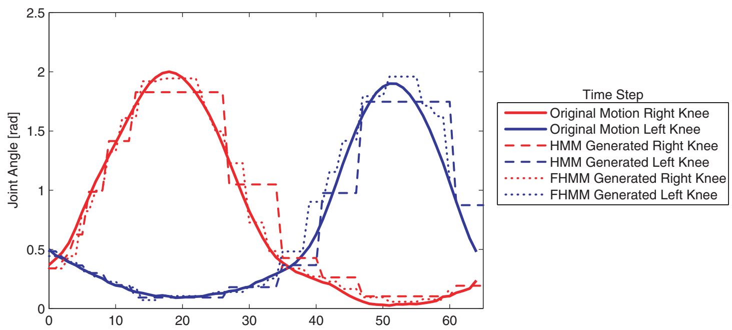


Fig. 6. Comparison of Generation Results of the HMM and FHMM (two chains) for the Knee Joints during a Walking Motion, prior to applying any post processing

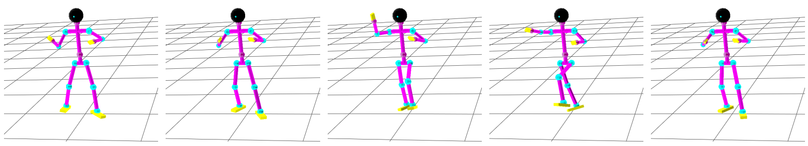


Fig. 7. Generated Hybrid Punch/Throw Motion

4 Conclusions

This paper develops a novel approach towards on-line, long term incremental learning and hierarchical organization of whole body motion primitives. Motion primitives are abstracted using adaptable Factorial Hidden Markov Models, where the model structure is adapted based on the similarity of the motions to be distinguished, such that a larger model is only used in dense regions of the knowledge base. The learned motions are aggregates of the observed motions, which have been autonomously clustered during incremental, on-line observation. Following the observation of each new motion sequence, the observation is placed into the closest motion grouping, based on the model distance between the observation and the group model. The modified group is then analyzed via clustering to extract child nodes, i.e. new, more specific motion primitives. The clustered motions are thereby incrementally organized into a hierarchical tree structure, where nodes closer to the root represent broad motion descriptors, and leaf nodes represent more specific motion patterns. The tree structure and level of specialization will be based on the history of motions observed by the robot. The resulting knowledge structure is easily searchable for recognition tasks, and can also be utilized to generate the learned robot motions.

Acknowledgment

This work is supported by the Japanese Society for the Promotion of Science grant 18.06754 and Category S Grant-in-Aid for Scientific Research 15100002.

References

1. D. C. Bentivegna, C. G. Atkeson, and G. Cheng. Learning similar tasks from observation and practice. In *International Conference on Intelligent Robots and Systems*, pages 2677–2683, 2006.
2. A. Billard, S. Calinon, and F. Guenter. Discriminative and adaptive imitation in uni-manual and bi-manual tasks. *Robotics and Autonomous Systems*, 54:370–384, 2006.
3. C. Breazeal and B. Scassellati. Robots that imitate humans. *Trends in Cognitive Sciences*, 6(11):481–487, 2002.
4. E. Diday and G. Govaert. Classification automatique avec distances adaptives. *R.A.I.R.O. Informatique Computer Science*, 11(4):329–349, 1977.
5. E. Diday and J. C. Simon. Clustering analysis. In K. S. Fu, editor, *Digital Pattern Recognition*, pages 47–92. Springer, 1976.
6. Z. Ghahramani and M. I. Jordan. Factorial hidden markov models. *Machine Learning*, 29:245–273, 1997.
7. A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.
8. H. Kadone and Y. Nakamura. Symbolic memory for humanoid robots using hierarchical bifurcations of attractors in nonmonotonic neural networks. In *International Conference on Intelligent Robots and Systems*, pages 2900–2905, 2005.
9. H. Kadone and Y. Nakamura. Segmentation, memorization, recognition and abstraction of humanoid motions based on correlations and associative memory. In *IEEE-RAS International Conference on Humanoid Robots*, pages 1–6, 2006.
10. D. Kulić, W. Takano, and Y. Nakamura. Incremental on-line hierarchical clustering of whole body motion patterns. In *IEEE International Symposium on Robot and Human Interactive Communication*, 2007.

11. D. Kulić, W. Takano, and Y. Nakamura. Representability of human motions by factorial hidden markov models. In *IEEE International Conference on Intelligent Robots and Systems*, 2007. To Appear.
12. K. Kurihara, S. Hoshino, K. Yamane, and Y. Nakamura. Optical motion capture system with pan-tilt camera tracking and realtime data processing. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1241–1248, 2002.
13. M. N. Nicolescu and M. J. Matarić. Task learning through imitation and human-robot interaction. In K. Dautenhahn and C. Nehaniv, editors, *Imitation and social learning in robots, humans and animals: behavioral, social and communicative dimensions*. Cambridge University Press, 2005.
14. T. Ogata, S. Sugano, and J. Tani. Open-end human-robot interaction from the dynamical systems perspective: mutual adaptation and incremental learning. *Advanced Robotics*, 19:651–670, 2005.
15. L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
16. S. Schaal, A. Ijspeert, and A. Billard. Computational approaches to motor learning by imitation. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 358:537 – 547, 2003.
17. W. Takano and Y. Nakamura. Humanoid robot's autonomous acquisition of proto-symbols through motion segmentation. In *IEEE-RAS International Conference on Humanoid Robots*, pages 425–431, 2006.