Hierarchical Spatio-Temporal Morphable Models for Representation of complex movements for Imitation Learning

W. Ilg[1] **G. H. Bakır**[2] **M. O. Franz**[2] **M. A. Giese**[1]

Cognitive Neurology[1]
University Clinic Tübingen, Germany

Max Planck Institute[2]
for Biological Cybernetics,Tübingen, Germany

{winfried.ilg,martin.giese,goekhan.bakir,matthias.franz}@tuebingen.mpg.de

Abstract

Imitation learning is a promising technique for teaching robots complex movement sequences. One key problem in this area is the transfer of perceived movement characteristics from perception to action. For the solution of this problem, representations are required that are suitable for the analysis and the synthesis of complex action sequences. We describe the method of Hierarchical Spatio-Temporal Morphable Models that allows an automatic segmentation of movements sequences into movement primitives, and a modeling of these primitives by morphing between a set of prototypical trajectories. We use HSTMMs in an imitation learning task for human writing movements. The models are learned from recorded trajectories and transferred to a human-like robot arm. Due to the generalization properties of our movement representation, the arm is capable of synthesizing new writing movements with only a few learning examples.

1 Introduction

The goal of imitation learning is to teach robots by observation of movement sequences. Imitation learning has to address two fundamental problems. (1) The movement characteristics of observed movements have to be transferred from the perceptual level to the level of generated actions [20] [14]. (2) Continuous spaces of movements with variable styles have to be approximated based on a limited number of learned example sequences. This implies that the robot should be able to synthesize new movements based on the learned examples.

One method that fulfills these requirements is the technique of Spatio-Temporal Morphable Models (STMMs). This methods represents the spatio-temporal characteristics of complex movement sequences by linear combination of example trajectories with different characteristics. Linear combinations of space-time patterns can be defined efficiently by exploiting spatio-temporal correspondence, by weighted summation of spatial and temporal displacement fields that morph the prototypical movement trajectories into a reference pattern. This method has been successfully applied for the generation and analysis of complex movements in computer graphics [4, 10] as well as for the recognition of movements and movement styles from trajectories in computer vision [10, 11].

To generalize the method of linear combination for complex sequences containing multiple complex movements we have extended the basic STMM algorithm by introducing a second hierarchy level that represents motion primitives. Each movement primitive is modeled using a STMM. In this way generative models for complex sequences of movements with variable styles can be learned from example trajectories. This method of Hierarchical STMMs (HSTMMs) has been successfully applied for the automatic recognition and synthesis of sequences of complex karate techniques [11], and for the estimation of skill levels of different actors [12] using a small amount of motion capture data. This shows that our method is suitable for building models for continuous movement spaces from a small amount of training data that can be used for analysis and synthesis.

In this paper we present an application of this algorithm for the imitation learning in robots. We show how HSTMMs can be linked to a robot control architecture. We illustrate our method by imitating human-like writing movements using a robot arm. Based on a small number of prototypical examples, our robot can learn to imitate and caricature writing styles, and to synthesize new styles of writing movements.

1.1 Related Work

Our work includes the identification and the segmentation of movement primitives, and the low-dimensional representation of movements by interpolation. Various methods for the parameterization of movement styles have been proposed in computer graphics and computer vision, e.g. based on Hidden Markov Models [2][24], principal component analysis [25] [1] [3], or Fourier components [23]. Different studies on imitation learning have investigated methods for describing the spatio-temporal characteristics of movements using principal component analysis [6] and spatio-temporal isomaps [13]. In [19], a verb-adverb approach was proposed that applies a combination of radial basis functions and low-order polynomials for defining parameterized interpolations between example movements. For this approach specific key times (e.g. the foot contact with the ground) must be specified by hand. Time Warping is defined by linear interpolation between these key times. In [22] and [15], this interpolation is realized with splines.

For the identification and segmentation of movement primitives within longer movement sequences appropriate features are required that provide a robust characterization of individual movement elements. Different elementary spatio-temporal and kinematic features have been proposed in the literature, like angular velocity [16] [17], or curvature and torsion of the 3D trajectories [5].

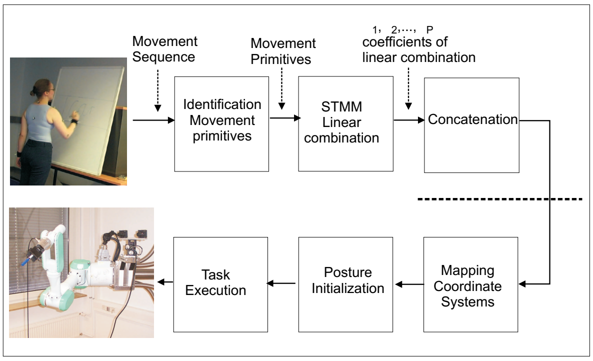

Figure 1: Schematic description of the algorithm for synthesizing and imitation of complex movement sequences. In the first step the sequence is decomposed into movement primitives. These movement primitives can be analyzed and changed in style by defining linear combinations of prototypes with different linear weight combinations. Afterward, the individual movement primitives are concatenated into longer movement sequences. This technique allows to generate sequences containing movements with multiple styles. The mapping of these movement sequences onto the robot arm is done in three steps: mapping of coordinates, posture initialization, and task execution.

2 Hierarchical Spatio-Temporal Morphable Models as representation for Imitation Learning

The process of establishing spatio-temporal correspondence between complex movement sequences consists of two steps. First, the sequences are segmented into movement primitives. Second, these movement primitives are modeled by STMMs. The following sections describe the extraction of the movement primitives, the modeling by STMMs and the transfer of the synthesized movement sequences onto the robot arm. An overview of the algorithm is shown in figure 1.

2.1 Identification of movement primitives

For the identification of movement primitives within a complex movement sequence an appropriate description of the spatio-temporal characteristics of the individual movement elements must be found that is suitable for a robust matching with stored prototypical movement primitives[1].

Based on such features spatio-temporal correspondence between new movement sequences and stored example sequences can be established on a coarse level.

The underlying features must be invariant against changes of the style of the individual movements elements. The key features of our algorithm are zeros of the velocity in few ”characteristic coordinates “of the trajectory $\boldsymbol{\zeta}(t)$, which are important for the identification of the movement primitive. Let m be the index of the motion primitive and $\boldsymbol{\kappa}(t)$ be the ”reduced trajectory” of the characteristic coordinates that has the values $\boldsymbol{\kappa}_i^m$ at the velocity zeros[2]. The movement primitive is then characterized by the vector differences $\Delta\boldsymbol{\kappa}_i^m = \boldsymbol{\kappa}_i^m - \boldsymbol{\kappa}_{i-1}^m$ of subsequent velocity zeros contained in the primitive (see figure 2).

A robust identification of movement primitives in noisy data with additional or missing zero-velocity points κ_i^s can be achieved with dynamic programming. Purpose of the dynamic programming is an optimal sequence alignment between the key features of the prototypical movement primitive $\kappa_0^m \ldots \kappa_q^m$ and the key features of a search window $\kappa_0^s \ldots \kappa_p^s$. This is accomplished by minimizing a cost function δ that is given by the sum of $||\Delta\kappa_i^s - \Delta\kappa_j^m||$ over all matched key features. A formal description of the algorithm is given in [11].

[1]The prototypical movement primitives can be specified for example by manual segmentation of prototypical trajectories

[2]Zero-velocity is defined by a a zero of the velocity in at least one coordinate of the reduced trajectory.

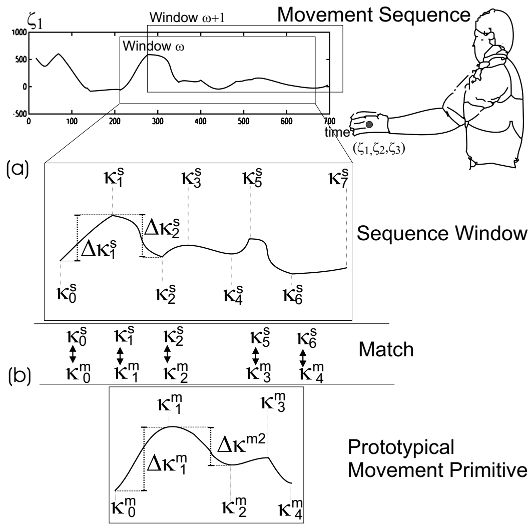

Figure 2: Illustration of the method for the automatic identification of movement primitives: (a) In a first step all key features $\boldsymbol{\kappa}_i^s$ are determined. (b) Sequences of key features from the sequences (s) are matched with sequences of key features from the prototypical movement primitives (m) using dynamic programming. A search window is moved over the sequence. The length of the window is two times the number of key features of the learned motor primitive. The best matching trajectory segment is defined by the sequence of feature vectors that minimizes $\sum_j ||\Delta\kappa_i^s - \Delta\kappa_j^m||$ over all matched key features. With this method spatio-temporal correspondence at a coarse level is established.

2.2 Morphable Models for modeling and concatenation of movement primitives

The technique of *spatio-temporal morphable models* [8, 10] is based on linearly combining the movement trajectories of prototypical motion patterns in space-time. Linear combinations of movement patterns are defined on the basis of spatio-temporal correspondences that are computed by dynamic programming [4]. Complex movement patterns can be characterized by trajectories of feature points. The trajectories of the prototypical movement pattern p can be characterized by the time-dependent vector $\boldsymbol{\zeta}_p(t)$. The correspondence field between two trajectories $\boldsymbol{\zeta}_1$ and $\boldsymbol{\zeta}_2$ is defined by the spatial shifts $\boldsymbol{\xi}(t)$ and the temporal shifts $\tau(t)$ that transform the first trajectory into the second. The transformation is specified mathematically by the equation:

$$\boldsymbol{\zeta}_2(t) = \boldsymbol{\zeta}_1(t + \tau(t)) + \boldsymbol{\xi}(t) \qquad (1)$$

By linear combination of spatial and temporal shifts the spatio-temporal morphable model allows to interpolate smoothly between motion patterns with significantly different spatial structure, but also between patterns that differ with respect to their timing.

The correspondence algorithm determines the temporal and spatial shifts by minimizing the weighted sum of the quadratic spatial and temporal displacements over the whole image sequence. In the time-continuous case, this error is given by the integral:

$$E_c[\boldsymbol{\xi}, \tau] = \int \left[|\boldsymbol{\xi}(t)|^2 + \lambda\, \tau(t)^2\right] \mathrm{d}t \qquad (2)$$

The error has to be minimized under the additional constraint that the mapping between the time variable t and the modified time $t' = t + \tau(t)$ for the trajectory $\zeta_1(t')$ must be continuous, one-to-one, and monotonically increasing, in order to define unique temporal warping of the sequence ζ_1. For further details about the underlying algorithm we refer to [8][10].

Signifying the spatial and temporal shifts between prototype p and the reference pattern by $\boldsymbol{\xi}_p(t)$ and $\tau_p(t)$, linearly combined spatial and temporal shifts can be defined by the two equations:

$$\boldsymbol{\xi}(t) = \sum_{p=1}^{P} w_p\, \boldsymbol{\xi}_p(t) \qquad \tau(t) = \sum_{p=1}^{P} w_p\, \tau_p(t) \qquad (3)$$

The weights w_p define the contributions of the individual prototypes to the linear combination. We always assume convex combinations with $0 \leq w_p \leq 1$ and $\sum_p w_p = 1$. After linearly combining the spatial and temporal shifts the trajectories of the morphed pattern can be recovered by morphing the reference pattern in space time using the spatial and temporal shifts $\boldsymbol{\xi}(t)$ and $\tau(t)$. The space-time morph is defined by equation (1) where $\boldsymbol{\zeta}_1$ has to be identified with the reference pattern and $\boldsymbol{\zeta}_2$ with the resulting space-time morph.

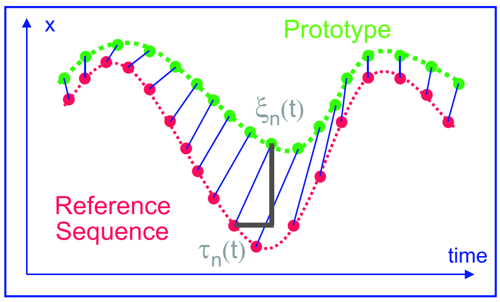

Figure 3: Illustration of the established spatio-temporal correspondence between a prototypical trajectory and a reference sequence with the correspondence vector fields τ and ξ.

The synthesized movement elements were concatenated using an algorithm described in [9]. This algorithm first normalizes the trajectories of the movement primitives and then combines linearly the trajectories and there start and end points.

3 Transferring human-like movements to a robot arm

The transfer of the trajectories to the robot is performed in three stages: 1) The HSTMM synthesizes trajectories in the same space as the prototype trajectories. Therefore, one has to transform synthesized trajectories from the prototype space into the task space of the robot. Also the trajectory is scaled appropriately. 2) The second stage initializes the robot posture to a specific recorded (and appropriately transformed) initial human arm posture. 3) The task execution is performed by reproducing the exact end-effector trajectory and approximating the human arm posture.

3.1 Mapping of the coordinate systems

In the investigated task of writing movements the end effect trajectories are approximately planar. The drawing area of the synthesized writing movements has to be transformed into a drawing area in task space. The drawing plane is given[3] by two vectors $\mathbf{u}$ and $\mathbf{v}$, which define a task orientation frame

$$\mathbf{T}_t = [\ \mathbf{u} \quad \mathbf{v} \quad \mathbf{u} \times \mathbf{v}\ ]. \tag{4}$$

The starting point of the movement is given by the position vector $\mathbf{p}$. Since the task space is planar, we can use the first principal components $\mathbf{e}_1, \mathbf{e}_2$ of the HSTMM output sequence $\zeta(t)$, to define an orientation frame of the trajectory as

$$\mathbf{T}_d = [\ \mathbf{e}_1 \quad \mathbf{e}_2 \quad \mathbf{e}_1 \times \mathbf{e}_2\ ]. \tag{5}$$

Note that $\mathbf{e}_1, \mathbf{e}_2$ span the whole task space for our application. The trajectory $\zeta(t)$ is then first centered

$$\hat{\zeta}(t_i) = \zeta(t_i) - \frac{1}{N}\sum_{k=1}^{N} \zeta(t_k), \tag{6}$$

where we assume that the trajectory is given in a discretized form $\zeta(t_1), \ldots, \zeta(t_N)$ with $t_1 = 0$. The centered trajectory $\hat{\zeta}(t_i)$ can be scaled to avoid violation of task space constraints. The final target trajectory $\boldsymbol{\zeta}^\star(t)$ is given by

$$\boldsymbol{\zeta}^\star(t) = \mathbf{p} + \mathbf{T}_t \mathbf{T}_d^{-1} \left(\hat{\zeta}(t) - \hat{\zeta}(0)\right). \tag{7}$$

[3]The plane could be determined by a stereo vision system.

3.2 Initialization of robot posture

The kinematic structure of humans and robots are usually different. Therefore, marker positions can usually not be transferred to the robot directly. Only if the robot is humanoid and has an equivalent kinematic structure the marker positions can be used directly [18]. Otherwise one has to define ”posture specifiers” that are applicable to humans as well as to robots. Imitation of posture is achieved by transferring these posture specifiers from the human to the robot.

Let $\mathbf{L}_d, \mathbf{R}_d, \mathbf{E}_d$ and $\mathbf{F}_d$ denote left shoulder, right shoulder, elbow and finger marker in transformed prototype space. As posture specifiers we chose orientation normals of two planes. The normal vector of the first plane is defined as

$$\mathbf{e}_d = \frac{(\mathbf{L}_d - \zeta^\star(t_1)) \times (\mathbf{E}_d - \zeta^\star(t_1))}{\|\,(\mathbf{L}_d - \zeta^\star(t_1)) \times (\mathbf{E}_d - \zeta^\star(t_1))\,\|}. \tag{8}$$

This plane is spanned by the left shoulder, the elbow and an arbitrary reference point. In our case we chose the starting point $\zeta^\star(t_1)$ of the trajectory $\zeta^\star(t)$. Equivalently let

$$\mathbf{f}_d = \frac{(\mathbf{R}_d - \zeta^\star(t_1)) \times (\mathbf{F}_d - \zeta^\star(t_1))}{\|\,(\mathbf{R}_d - \zeta^\star(t_1)) \times (\mathbf{F}_d - \zeta^\star(t_1))\,\|} \tag{9}$$

be the normal of the second plane which is spanned by finger, right shoulder and $\zeta^\star(t_1)$. Let $\mathbf{q} = [\mathbf{q}_1, \mathbf{q}_2]$ be the joint values of the robot, where $\mathbf{q}_1$ influences the elbow position and $\mathbf{q}_2$ does not. The corresponding plane normals $\mathbf{e}_r(\mathbf{q}_1)$, $\mathbf{f}_r(\mathbf{q}_2)$ of the robot are calculated in an equivalent way (see fig. 4). For this purpose we use the a-priori specified position vector $\mathbf{p}$ from 3.1 instead of $\zeta^\star(t_1)$[4]. In addition a virtual left shoulder position has to be specified to determine the relative orientation of robot arm to the robot basis.

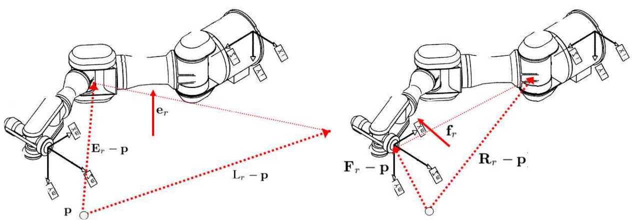

Figure 4: Illustration of the plane normals $\mathbf{e}_r$ and $\mathbf{f}_r$. A virtual left shoulder $\mathbf{L}_r$ position of the robot is defined a-priori.

The initial posture of the robot is adjusted to the initial human posture by first minimizing

$$\min_{\mathbf{q}_1} \ \|\mathbf{e}_d - \mathbf{e}_r(\mathbf{q}_1)\|. \tag{10}$$

[4]The reference point $\zeta(t_1)$ must ensure that $\mathbf{e}_d \neq f_d \forall t$. Otherwise another reference point has to be chosen.

over the joints $\mathbf{q}_1$, and subsequently minimizing

$$\min_{\mathbf{q}_2} \; \|\mathbf{f}_d - \mathbf{f}_r(\mathbf{q}_2)\| \tag{11}$$

over $\mathbf{q}_2$. The solution minimizes the angles between $\mathbf{e}_r, \mathbf{e}_d$ and $\mathbf{f}_r, \mathbf{f}_d$ respectively.

3.3 Task Execution

Starting from its initial posture, the trajectory of the robot is planned by solving the following optimization problem that depends on the discretely sampled joint variables $\mathbf{q}(t_i)$:

$$\min_{\mathbf{q}(t_i)} \; \rho(\mathbf{q}(t_i)) = \|\mathbf{e}_d - \mathbf{e}_r\|^2 + \alpha\|\mathbf{q}(t_i) - \mathbf{q}(t_{i-1})\|^2 \tag{12}$$

subject to

$$\mathbf{P}_r(\mathbf{q}(t_i)) - \zeta^\star(t_i) = 0 \tag{13}$$

where $\mathbf{P}_r(\mathbf{q}(t_i))$ describes the end-effector position. This problem is solved for each time step t_i of the trajectory separately. The objective function $\rho(\mathbf{q}(t_i))$ measures the euclidian distance between the normals $\mathbf{e}_d$ and $\mathbf{e}_r$. An additional regularization term is added to penalize high joint velocities. This term depends on the difference between the new joint configuration $\mathbf{q}(t_i)$ and the previous configuration $\mathbf{q}(t_{i-1})$. The scalar α determines the tradeoff between smoothness of obtained joint trajectories and the quality of imitation. As a starting point, we use the joint values obtained by classical inverse kinematics. The joint trajectories were computed off-line. [5]

4 Experiments

We demonstrate the application of the proposed method by imitation and synthesis of human writing movements. In the following we desribe the technical details and results of the performed experiments.

Motion Capture: We recorded writing movements of two human actors who wrote the word “ICAR” (fig. 5) using a commercial motion capture system (VICON 612, Oxford) with 6 cameras. We used 10 (passive) markers that included the shoulders, 2 front and one rear torso, upper arm, elbow, front arm, hand and index finger of the writing arm.

[5] A computational faster implementation to solve eq. (12) is obtained by using explicit information about the null space of the manipulator jacobian (see [7]).

Identification of movement primitives: Individual letters were defined as movement primitives. The automatic segmentation of the movement primitives was based on the index finger trajectories. The segmentation algorithm was trained with one example for each movement primitive that was obtained by manual segmentation of the trajectory of one of the actors. Figure 6 shows the result of the automatic segmentation for the actor that was not used for training.

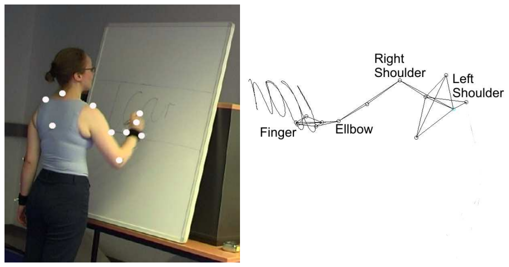

Figure 5: Left panel: Motion capturing of writing movements on a board. White dots indicate the positions of the recorded markers. Right panel: Illustration of the marker set and the trace of the finger marker during the wrting of the word ”ICAR”.

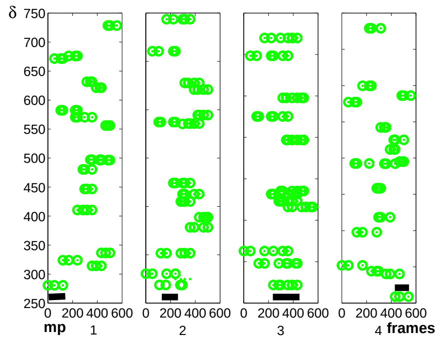

Figure 6: Results of the automatic segmentation for one “ICAR” sequence identifying movement primitives (letters) based on previously learned prototypical movement primitives. The diagrams shows the distance measure of the dynamic programming method, δ, for different matches of the corresponding movement primitive over the whole sequence. The circles mark the times of the matched key feature κ_i^m in the sequence. Each match of a whole movement primitive is illustrated by a row of circles with the same δ. The number of circles corresponds to the number of key features of the movement primitive. The black bars at the bottom describe the result of a manual segmentation of the same four movement primitives (mp 1-4).

Syntheses of writing movements: Continuous spaces of individual movements are generated by linear combinations of the segmented movement primitives. These movements are then automatically concatenated into longer sequences including multiple movement primitives. Figure 7 shows the synthesized pen trajectories of the writing movements. The method allows to morph continuously between the writing sequences of the two actors (left panel). In addition we can synthesize caricatures of the specific writing styles of each actor (right panel). Also, the individual movement primitives can be reassembled in a different sequential order, e.g. in order to write the word "'IACR" (middle row). All movement sequences were synthesized based on only two prototypical example trajectories.

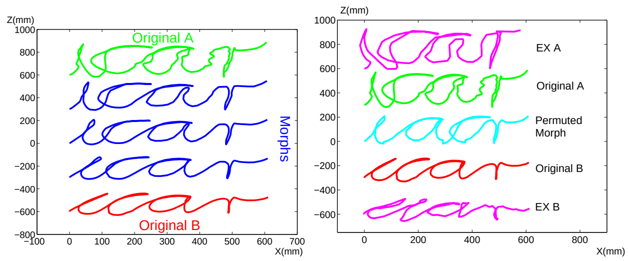

Figure 7: Left panel: Recorded pen trajectories and morphs between the original writing movements. The morphs interpolate continuously in space-time between the prototypes. Right panel: Original pen trajectories and exaggerations of the writing styles of the two actors. The middle row shows synthesis of a new word "IACR" by reassembling the movement primitives in a different sequential order.

Transfer to the Robot arm: The synthesized movements were executed using a Mitsubishi PA-10 7-DOF robot arm (fig. 8). Optimization has been performed for different values of α (eq. 12). Figure 9 illustrates that for small values of α a better imitation (measured by the difference $||\mathbf{e}_d - \mathbf{e}_d||$) is achieved but discontinuous joint trajectories can arise. These discontinuities disappear for large values of α at the cost of worse imitation quality.

5 Discussion

We have presented a method for imitation learning of complex movement trajectories that is based on linear combination of small sets of prototypical example movement sequences. The proposed algorithm decomposes long trajectories automatically into movement primitives, and models these primitives by linear combination of prototypical trajectories. We also have shown how such flexible

Figure 8: Left panel: The Mitsubishi PA–10 robot arm used to execute the writing movements. Right panel: Writing examples of the Originals A and B and the average morph in between (compare fig. 7).

representations of movement trajectories can be coupled with a real robot system in a way that ensures the accurate reproduction of endpoint trajectories and the imitation of the style of the human movement. The proposed method can be generalized in a straightforward way to other tasks and movement classes and is not restricted to the imitation of writing.

The method of HSTMM has the advantage that it works with very small sets of training data [10][12][11]. Many popular methods for the representation of trajectories, e.g. HMMs or unsupervised learning of manifolds [13][2] typically require substantial amounts of training data. Another advantage of HSTMMs is the relatively intuitive interpretation of the weights of the linear combinations that specify the style characteristics of the individual prototypes. The proposed method for transferring the synthesized trajectories to the robot has the advantage that it combines an exact control of the endeffector position with a more "soft" control geometric variables that characterize the style of the executed arm movements.

The presented application is only a first simple demonstration of the application of HSTMMs in imitation learning. Future work has to apply and to extend the proposed algorithms for more complex robot systems, and for more complex tasks that include additional constraints, e.g. obstacle avoidance. The successful application of HSTMMs for the synthesis and analysis of complex whole body movements in computer graphics and sports [10][12][11] suggests that the same algorithms might also perform well in imitation learning for humanoid robots.

Acknowledgments

This work is supported from the Deutsche Volkswagenstiftung. We thank H.P. Thier, B. Schölkopf, H.H. Bülthoff,

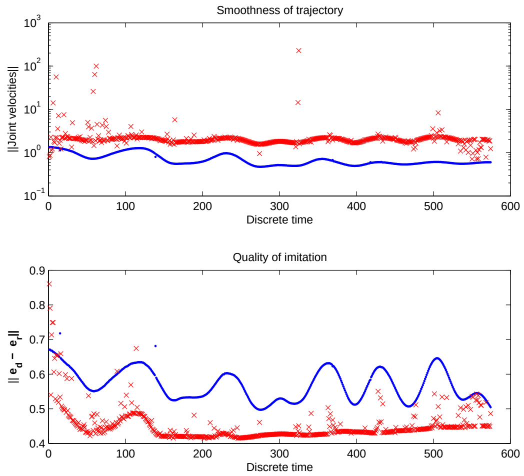

Figure 9: Joint velocities (top panel) $||\mathbf{q}(t_i) - \mathbf{q}(t_{i-1})||$ and elbow norm difference (bottom panel) $||\mathbf{e}_d - \mathbf{e}_r||$ as a function of time for $\alpha = 10^{-1}$ (dots) and $\alpha = 10^{-2}$ (crosses). One obtains continuous joint trajectories for larger α.

M. Hein and the Max Planck Institute for Biological Cybernetics for additional support.

References

[1] A. F. Bobick and J. Davis. An appearance-based representation of action. In *Proceedings of the IEEE Conference on Pattern Recognition*, pages 307–312, 1996.

[2] M. Brand. Style machines. In *SIGGRAPH*, 2000.

[3] C. Bregler, L. Loeb, E. Chuang, and H. Deshpande. Turning to the masters: Motion capturing cartoons. In *SIGGRAPH*, 2002.

[4] A. Bruderlin and L. Williams. Motion signal processing. In *SIGGRAPH*, pages 97–104, 1995.

[5] T. Caelli, A. McCabe, and G. Binsted. On learning the shape of complex actions. In *International Workshop on Visual Form*, pages 24–39, 2001.

[6] A. Fod, M. J. Mataric, and O. C. Jenkins. Automated derivation of primitives for movement classification. *Autonomous Robots*, 12(1):39–54, 2002.

[7] C. Ott G. Schreiber and G. Hirzinger. Interactive redundant robotics: Control of the inverted pendulum with nullspace motion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001.

[8] M. A. Giese and T. Poggio. Synthesis and recognition of biological motion pattern based on linear superposition of prototypical motion sequences. In *Proceedings of IEEE MVIEW 99 Symposium at CVPR, Fort Collins*, pages 73–80, 1999.

[9] M.A. Giese, B. Knappmeyer, and H.H. Bülthoff. Automatic synthesis of sequences of human movements by linear combination of learned example patterns. In *Workshop on Biologically Motivated Computer Vision*, 2002. accepted.

[10] M.A. Giese and T. Poggio. Morphable models for the analysis and synthesis of complex motion patterns. *International Journal of Computer Vision*, 38(1):59–73, 2000.

[11] W. Ilg and M.A. Giese. Modeling of movement sequences based on hierarchical spatial-temporal correspondence of movement primitives. In *Workshop on Biologically Motivated Computer Vision*, pages 528–537, 2002.

[12] W. Ilg and M.A. Giese. Estimation of skill level in sports based on hierarchical spatio-temporal correspondences. 2003. submitted.

[13] O.C. Jenkins and M. J. Mataric. Deriving action and behavior primitives from human motion data. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2551–2556, 2002.

[14] M. J. Mataric. Visuo-motor primitives as a basis for learning by imitation : Linking perception to action and biology to robotics. In K. Dautenhahn and C. Nehaniv, editors, *Imitation in Animals and Artifacts*, pages 392–422. MIT Press, 2002.

[15] H. Miyamoto and M. Kawato. A tennis serve and upswing learning based on bi-directional theory. *Neural Networks*, 11:1331–1344, 1998.

[16] T. Mori and K. Uehara. Extraction of primitive motion and discovery of association rules from motion data. In *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*, pages 200–206, 2001.

[17] A. Nakazawa, S. Nakaoka, K. Ikeuchi, and K. Yokoi. Imitating human dance motions through motion structure analysis. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2539–2544, 2002.

[18] N. S. Pollard, J. Hodgins, M. J. Riley, and C. Atkeson. Adapting human motion for the control of a humanoid robot. In *IEEE International Conference on Robotics and Automation*, 2002.

[19] C. Rose, M. F. Cohen, and B. Bodenheimer. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5):32–40, 1998.

[20] S. Schaal. Is imitation learning a route to humanoid robots. *Trends in Cognitive Science*, 3:233–242, 1999.

[21] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric frame work for nonlinear dimensionality reduction. *Science*, 290(5000):2139–2323, 2000.

[22] A. Ude, C. G. Atkenson, and M. Riley. Planning of joint trajectories for humanoid robots using b-spline wavelets. In *IEEE International Conference on Robotics and Automation*, 2000.

[23] M. Unuma, K. Anjyo, and R. Takeuchi. Fourier principles for emotion-based human figure animation. In *SIGGRAPH*, pages 91–96, 1995.

[24] A. D. Wilson and A. F. Bobick. Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):884–900, 1999.

[25] Y. Yacoob and M. J. Black. Parameterized modeling and recognition of activities. *Journal of Computer Vision and Image Understanding*, 73(2):398–413, 1999.