# Continuous Motion Morphing

Johannes Mezger, Winfried Ilg, Martin Giese

email: mezger@gris.uni-tuebingen.de
{winfried.ilg,martin.giese}@uni-tuebingen.de

WSI-2004-14
December 2004

Graphisch-Interaktive Systeme
Wilhelm-Schickard-Institut
Universität Tübingen
D-72076 Tübingen, Germany
WWW: http://www.gris.uni-tuebingen.de

ISSN 0946-3852

# Continuous Motion Morphing

Johannes Mezger[1], Winfried Ilg[2], Martin Giese[2]

[1]Graphical-Interactive Systems (GRIS)
Wilhelm Schickard Institute for Computer Science, Tübingen, Germany
mezger@gris.uni-tuebingen.de

[2]Laboratory for Action, Representation and Learning (ARL)
Department of Cognitive Neurology, University Clinic Tübingen, Germany
{winfried.ilg,martin.giese}@uni-tuebingen.de

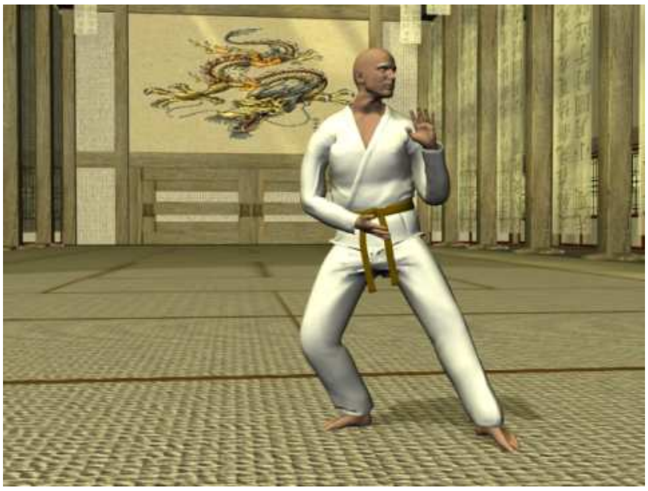

**Figure 1. Rendering of a morphed Karate Kata.**

## Abstract

*We present an extension to a previous morphing method for human motion. It works on motion capture data that is segmented into movement elements. Our new timewarping algorithm accepts time-dependent continuous functions as input for the morphing coefficients without introducing foot sliding. It is designed for creating new natural looking motions from given prototype motions. We employ a zero-moment-point criterion to analyze the physical correctness of the morphed motions.*

**CR Categories**
I.3.7 [3D Graphics and Realism]: Animation

**Keywords**
Character Animation, Motion Capture, Motion Morphing.

# 1. Introduction

In the recent years, the need for the entertainment industry to generate realistic looking virtual characters became more and more obvious. The rendering quality in animated films and computer games improved steadily, but the naturalness of the character motion could not yet satisfy the human perception. As the human idea of natural motion is heavily trained by everyday impressions of real human and animal movements, it can hardly be tricked by artificially created movements. Even if a simulated motion is physically correct, it often appears unnatural and robot-like.

Today, basically three different methods are employed for computer assisted motion generation: keyframe editing, physics based simulation, and motion capture. The first one, *keyframe editing*, demands great skill and experience on the animator. Additionally, as with the second method, it is also difficult to add all the dynamic expressions in the upper frequency domain that play an important role in natural motion. Another difficulty in *physics based simulation* is the tuning of the simulation until the desired movements are achieved, because the result of the simulation can only be controlled indirectly. The last method, *motion capture*, became favorable when infrared cameras and computer hardware were able to provide the required performance to capture and process 3D marker motion at sufficient temporal and spatial resolution. With motion capture, all important details of the movements are preserved, but every single motion sequence has to be captured separately and cleaned from artifacts, which is a time consuming and expensive process.

Hence, for the generation of realistic movements, we aim at the reuse of motion capture data to synthesize new motions that hold the physical and natural correctness of the captured motion examples. We first give an overview of related work from the past years. A brief introduction to morphable models for motion follows in Section 3. In Section 4, the new method for continuous morphing is presented. Finally, in Section 5 we compare the correctness of the zero-moment-point with the previous method.

# 2. Related Work

Physics based methods early became popular to improve keyframe animations. The artist specifies key poses or actions to be performed ("spacetime constraints"), and the system searches for matching movements that satisfy Newton's laws based on the boundary conditions [WK88, Gle97, FvdPT01, FP03]. In particular, the zero-moment-point (ZMP, the point where all moments acting on the body cancel out each other) is useful to determine correct motions [TSK00, SKG03, TK03].

Motion blending helps to combine existing motions into new, intermediate motions using linear [BW95, WP95, GP99] or radial [RCB98] basis functions. In order to interpolate joint rotations, the rotations are linearized [Gra98, LS02] and end-effector positions have to be ensured by inverse kinematics [PSS02]. Kovar and Gleicher [KG03, KG04] introduced registration curves to align the coordinate frames of the input motions before the actual morph.

Giese and Ilg introduced hierarchical movement correspondences in order to identify movement primitives [IG02] and to synthesize human motion sequences [GKB02]. Several authors propose methods to generate long motions from short sequences by concatenation, either by learning methods [LWS02] or by searching appropriate transitions in a motion database [KGP02, CLS03, AFO03].

In this work, we try to optimize the morphing process itself with the objective of generating realistic, natural motion.

# 3. Morphable Models for Motion

First we give a brief introduction to the idea of motion correspondence, combinations of motions and hierarchical models as described by Giese et al. [GP99, GKB02]. As motion capture anyway acquires only a part of the body's degrees of freedom, any further dimension reduction or working in the frequency domain removes even more important details from the input motions. Therefore, the morphable models work on motion capture data in spatial domain.

## 3.1. Motion Correspondence

We define the trajectory $x_p$ of a prototypical motion in relation to a reference trajectory $x_r$ at time $t$ as

$$\mathbb{R} \to \mathbb{R}^N : \; x_p(t) = x_r(t + T_p(t)) + X_p(t) \tag{1}$$

with temporal displacements $T_p \in \mathbb{R}$ and spatial displacements $X_p \in \mathbb{R}^n$. The trajectories are given as $N$ time-dependent values and describe variable parameters of a motion like marker coordinates and joint rotation angles. The displacements give the correspondence of two motions and are not unique.

We call the motion correspondence good if the function $g(t) = t + T_p(t)$ is monotonic and the norms $|T_p|$ and $|X_p|$ are small. If the prototype and the reference motion are identical, $T_p$ and $X_p$ should be zero. The more the two motions differ, the higher the amplitude of the displacement fields will be. Optimal displacements can be found by solving a constrained optimization problem [BW95] that minimizes the energy function

$$E = \int \left[ |X_p(t)|^2 + \lambda\, T_p(t)^2 \right] dt \qquad (2)$$

with a constant $\lambda$ that controls the trade-off between optimal time-alignment and optimal spatial proximity.

### 3.2. Motion Combination

Having several prototypical motions that are mapped to a single reference motion by good correspondence, we can express the trajectories of new motions as linear combinations of the displacement fields. The new displacements are the weighted sums

$$X(t) = \sum_{p=1}^{N} w_p X_p(t) \quad \text{and} \quad T(t) = \sum_{p=1}^{N} w_p T_p(t)$$

with weights $w_p \in \mathbb{R}$. For character motion, typically only convex combinations are used. As the prototypes, also the new motion is now defined by Equation (1) in terms of the reference motion, which gives

$$\begin{aligned} x(t) &= x_r(t + T(t)) + X(t) \\ &= x_r(t + \sum_{p=1}^{N} w_p T_p(t)) + \sum_{p=1}^{N} w_p X_p(t). \qquad (3) \end{aligned}$$

The reference motion $x_r$ can be chosen as the average of all input motions.

### 3.3. Hierarchical Morphable Models

The morphable model so far is limited to only short movement sequences, since the minimization of Equation (2) for discretized motions has quadratic complexity in the number of time-steps. Furthermore, it makes sense to split long movement sequences into smaller ones that can be handled separately in order to achieve more flexibility in the reuse of the motion.

For this reason, Giese and Ilg introduced the *hierarchical spatio-temporal morphable models, HSTMM* [GKB02, IG02], that morph individual movement elements of a decomposed motion according to the following steps:

**1) Decomposition** Each motion $x_p(t)$ is split into $M$ time intervals

$$J_{p,i} = [t_{p,i},\ t_{p,(i+1)}],\ i = 0 \dots M-1.$$

This decomposition is performed manually or automatically, e.g. based on local minima of the velocities.

**2) Normalization** Each motion segment is resampled to $\tau$ time-steps. Afterwards, each segment $x_{p,i}$ is transformed by a linear function that zeroes the trajectories at the segment borders, giving the normalized trajectories

$$\begin{aligned} \tilde{x}_{p,i}(t) &= x_{p,i}(t) - x_p(t_i) \\ &\quad - t/\tau\ (x_p(t_{i+1}) - x_p(t_i)), \qquad (4) \\ t &\in [0, \tau]. \end{aligned}$$

**3) Correspondences** For each normalized movement element, the energy function (2) is minimized separately, which turns the non-hierarchical complexity $O({t_M}^2)$ for discretized motions into $O(M \cdot \tau^2)$.

These first three steps are independent from the morph weights and need to be computed only once as long as the prototype motions do not change.

**4) Linear combination** The normalized trajectories now can be morphed separately for each segment with Equation (3). To be able to reverse the linear transformation in step (5), also the segment transition points and the new segment lengths are combined by weighted sums.

**5) Concatenation** Using the new transition points and segment lengths, the normalization is reversed and the global coordinates are obtained. It has to be ensured that the transitions are continuous, which is problematic if different weights are used for the motion segments. The approach depicted in the next section resolves this problem.

## 4. Continuous Motion Morphing

The application for the morphable models originally was not motion synthesis, but analysis: If an unknown motion can be represented as a combination of example motions, the weights of the particular examples give information about the kind of the motion. This feature is successfully exploited to estimate skill-levels in sports [IMG03]. Moreover, movements can be transferred to robots by means of imitation learning [IBMG04].

However, for generating naturally looking character animation, the discontinuous morphing weights are visually not acceptable because of foot-sliding. As a solution we introduce smooth weights.

### 4.1. Smooth Weights

Replacing the constant weights $w_p$ in Equation (3) by time-dependent weights $w_p(t)$, the motion can be morphed continuously between the prototypes. Unfortunately, this approach causes foot-sliding when the prototypes are spatially distant. The reason is that the weight change induces an acceleration to the trajectories that depends linearly on the displacement field amplitudes.

For illustration, let the temporal displacements $T_p^0(t)$ be zero, which means that all movements of the prototypes have the same timing. Under this assumption, applying the product rule on Equation (3) we get for the velocity $\frac{dx}{dt}$ of the morphed trajectories

$$\dot{x}^0(t) = \dot{x}_r(t) + \sum_{p=1}^{N} \left[ \dot{w}_p(t) X_p(t) + w_p(t) \dot{X}_p(t) \right].$$

With the term $\dot{w}_p(t) X_p(t)$ the velocity depends on the weight change rates at time $t$. However, we would expect the correct velocity $\dot{x}_c^0$ to be only dependent on the current weights,

$$\dot{x}_c^0(t) = \dot{x}_r(t) + \sum_{p=1}^{N} w_p(t) \dot{X}_p(t),$$

or with any $T_p(t)$,

$$\dot{x}_c(t) = \dot{x}_r(t + \sum_{p=1}^{N} w_p(t) T_p(t)) + \sum_{p=1}^{N} w_p(t) \dot{X}_p(t). \quad (5)$$

Integration of the modified velocity $\dot{x}_c$ yields the correct trajectories

$$\begin{aligned} x_c(t) &= \int_{t_0}^{t} \dot{x}_c(u) du \; + \; x_s \\ &= x_r(t + T(t)) + X_c(t) + \; x_s \end{aligned}$$

with the modified spatial displacements

$$X_c(t) = \int_{t_0}^{t} \sum_{p=1}^{N} w_p(u) \dot{X}_p(u) \, du \quad (6)$$

and the starting point

$$x_s = \sum_{p=1}^{N} w_p(t_0) X_p(t_0) \; \in \mathbb{R}^N.$$

Hence, the correct morph is obtained by first differentiating the spatial displacements, then timewarping the velocities, and finally integrating. Figure 2 shows a simple scheme for the new morphing method.

For Equation (6), the velocity field

$$\dot{X}_c(t) = \sum_{p=1}^{N} w_p(t) \dot{X}_p(t)$$

is needed. Using convex weights, the computation can be further simplified. Instead of actually storing the relative velocities $\dot{X}_p(t)$ of the prototypes to the reference,

$$\dot{X}_p(t) = \frac{d}{dt} \left( x_p(t) - x_r(t) \right),$$

we may rewrite Equation (5) as

$$\begin{aligned} \hat{\dot{x}}_c(t) &= \dot{x}_r(t + T(t)) \\ &\quad + \textstyle\sum_{p=1}^{N} \left[ w_p(t) \cdot (\dot{x}_p(t) - \dot{x}_r(t)) \right] \\ &= \dot{x}_r(t + T(t)) - \dot{x}_r(t) \\ &\quad + \textstyle\sum_{p=1}^{N} w_p(t) \dot{x}_p(t) \end{aligned} \quad (7)$$

with $\sum_{p=1}^{N} w_p(t) = 1$.

Hence, with convex weights the timewarping can be performed immediately on the input velocities. If additionally the timewarping is disabled ($T_p \equiv 0$), the computation simplifies to a trivial integration of the input velocities.

### 4.2. Continuously Morphable Models

In order to be able to use the hierarchical morphable models of Section 3 for computing the correct morph $x_c(t)$, they are modified the following way.

**1) Decomposition** The splitting of the prototype motions into $M$ time intervals is performed as before. From now on we will call the trajectory segments ”positions” to be able to distinguish from the velocities.

**2) Normalization and Differentiation** The positions are resampled to $\tau$ time-steps per segment. Thus, we get $(\tau - 1)$ velocity samples as differences from the positions. Afterwards, the normalization is applied to the positions.

**3) Correspondences** The displacements are calculated as usual based on the normalized positions.

**4) Linear combination and Integration** The starting point $x_s$ is computed, and for each segment, the segment lengths and the velocities are linearly combined. Starting from $x_s$, the velocities are summed up over all segments and timewarped.

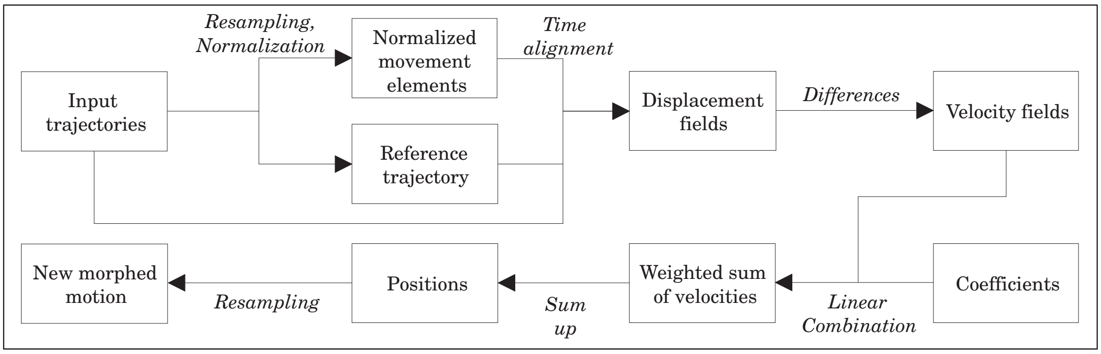

**Figure 2. Continuous morphing scheme.**

**5) Concatenation** Finally, the segments are resampled to the correct length. The concatenation of the segments is automatically continuous if the weight functions are.

It turns out that this simple time-integration does not produce noticeable errors even for considerably long sequences of 30 seconds and more. In our experiments we used $\tau = 200$ samples for movement elements of up to ten seconds in length. Linear interpolation of the samples is sufficient for the re-sampling during normalization, timewarping and concatenation.

### 4.3. Linearized Rotations

The continuous morphing is not limited to positional trajectories, but can be applied to any motion parameters that can be linearly combined. In order to morph the joint rotations of skeleton hierarchies, we employ an exponential mapping of $\mathbb{R}^3$ to the unit quaternion space $\mathbb{S}^3$ and the rotation matrices SO(3) according to [Gra98]. Spherical linear quaternion interpolation (slerp) would limit the morphing to only two prototype motions.

The logarithm of the unit quaternion $(v, w)$ is defined as

$$\mathbb{S}^3 \to \mathbb{R}^3 : \; log(v, w) = \begin{cases} (0,0,0)^T, & |v| = 0 \\ 2 \; arccos(w) \cdot v/|v|, & |v| > 0, \end{cases}$$

which corresponds to a rotation around the axis $v$ by the angle $\alpha$,

$$\alpha = (2z\pi + 2 \; arccos(w)), \; z \in \mathbb{Z}.$$

Since this mapping is not unique, at each time-step we pick the $\alpha$ that minimizes the angular difference between the joint rotations.

## 5. Validation of Morphs

While it is rather easy for a human observer to quantify the correctness and naturalness, it is comparably difficult to define a quality measure for morphed motions.

### 5.1. Zero-moment-point

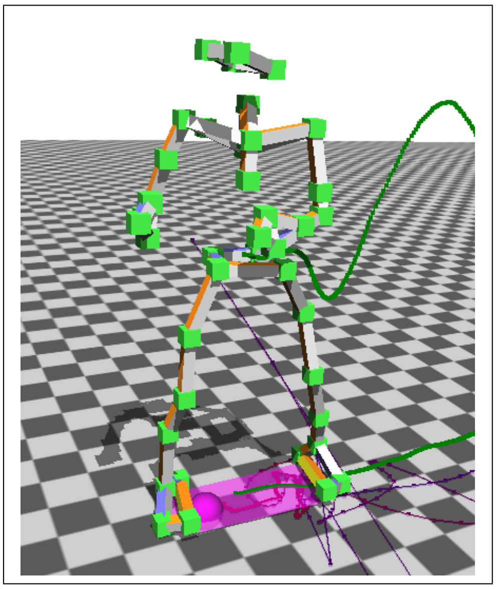

**Figure 3. ZMP and support polygon (magenta) compared to the center of mass (green trajectory).**

One possibility is to validate Newton's Laws for an approximated mass distribution on the character. We restrict to the case where the character is in non-sliding contact with the environment. In this case, forces that affect the motion of the character have to act on the contact points. For a physically correct walking motion, this means that the zero-moment-point (ZMP) always stays inside the support polygon of the feet on the ground, otherwise the law of action and reaction would be violated. When the character is in rest, the ZMP is identical with the projection of the center of mass onto the ground. However, whether the character is balanced or not, or even falls, the ZMP will always stay inside the support polygon. For this reason, the ZMP is a good measure to proof a physically incorrect motion during the support-phases.

### 5.2. Results

In our experiments, we use the mass distribution and formulas given in [TSK00] to calculate the ZMP. Figure 3 shows the ZMP (magenta ball) for a frame where the character stops its motion with the right leg after a jump. The trajectory of the center of mass (green dots) is given for comparison. During the flight phase, the ZMP was meaningless (magenta lines).

The distance of the ZMP to the support polygon for two Karate captures is depicted in Figure 4. The motions were captured at 120Hz from a beginner and a master in Karate. Figure 5 shows the distance of the ZMP for a morph between the two captures. The result of the previous model (HSTMM, dotted line) is plotted against the result of the new continuous model. During the morph, the weight coefficients change linearly from the beginner to the master. For the ZMP computation, the character height is assumed to be 1.75m and the weight 79kg, which is close to the actual measures of the two captured persons. Figure 6 shows some pictures of the generated morph.

As the calculation of the ZMP involves the calculation of accelerations, any noise in the trajectories is amplified. Therefore, we have to apply low-pass filters before each differentiation. The noise generally is higher at the end of the morph, as the better karateka moves faster. On the one hand, the difference between the two curves is only minimal, which reveals that, with the ZMP, it is impossible to quantify the foot-sliding caused by the HSTMM.

On the other hand, it shows that the continuous model does not significantly increase the noise or deteriorate the physical correctness otherwise. Note that the input motions already show a rather bad ZMP trajectory (Figure 4), and naturally the morphed motion cannot achieve a better ZMP quality.

## 6. Conclusions

Our approach for continuous motion morphing extends the hierarchical morphable models by arbitrary blending weights. The generated motion inherits the continuity of the input trajectories and the user-defined weight function. Consequently, the flexibility and the wide range of applications of the models are extended by the advantage of being able to create naturally looking motions for Computer Graphics needs.

We applied the new models to long Karate motions that pose a hard challenge to morphing algorithms since they contain a wide dynamic range of movements, and showed, that the ZMP trajectory is not adversely affected by the new timewarping method.

One drawback of morphable models is that the space of producible motions is strongly limited by the range of the prototype motions. Therefore, motion databases still have to be provided for applications in Computer Animation. Another drawback is that the linear combination smoothes out more details the more prototype motions are used. A modified combination method would be needed to save specific details of one prototype while it is combined with several other ones. However, the physical and natural correctness of the generated motion then could not be ensured anymore.

**Acknowledgements**
We thank Bernhard Eberhardt for his support during the motion capture, and Lucas Kovar and Hyun Joon Shin for the helpful discussions.

## References

[AFO03] Arikan, O., Forsyth, D. A., and O'Brien, J. F. Motion synthesis from annotations. In *Proceedings of ACM SIGGRAPH*, 2003.

[BW95] Bruderlin, A., and Williams, L. Motion signal processing. In *Proceedings of ACM SIGGRAPH*, 1995.

[CLS03] Choi, M. G., Lee, J., and Shin, S. Y. Planning biped locomotion using motion capture data and probabilistic roadmaps. *ACM Transactions on Graphics*, 22(2), 2003.

[FP03] Fang, A. C., and Pollard, N. S. Efficient synthesis of physically valid human motion. In *Proceedings of ACM SIGGRAPH*, 2003.

[FvdPT01] Faloutsos, P., van de Panne, M., and Terzopoulos, D. Composable controllers for

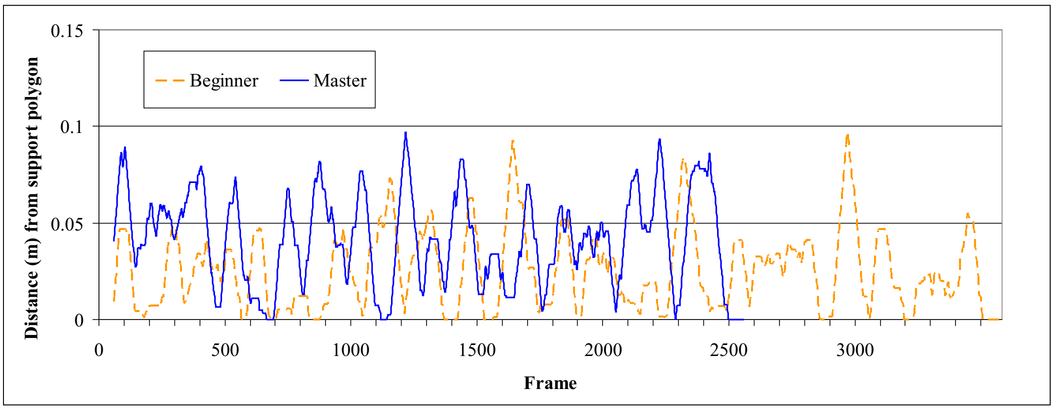

**Figure 4. Distance of ZMP from support polygon for the prototype motions (moving average over 60 frames).**

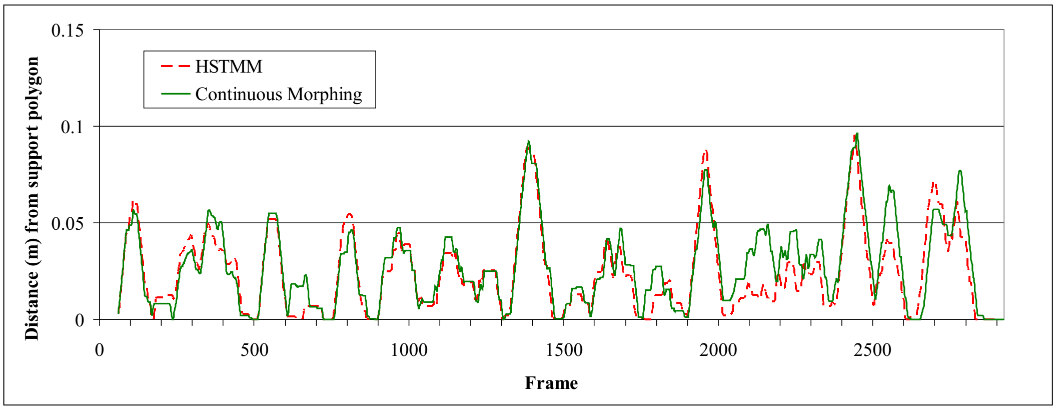

**Figure 5. Distance of ZMP from support polygon for the morph (moving average over 60 frames).**

physics-based character animation. In *Proceedings of ACM SIGGRAPH*, 2001.

[GKB02] Giese, M., Knappmeyer, B., and Bülthoff, H. Automatic synthesis of sequences of human movements by linear combination of learned example patterns. In *Biologically motivated Computer Vision*, H. H. Bülthoff, S. W. Lee, T. Poggio, and C. Wallraven, Eds. Springer, 2002.

[Gle97] Gleicher, M. Motion editing with spacetime constraints. In *Proceedings of the Symposium on Interactive 3D Graphics*, 1997.

[GP99] Giese, M., and Poggio, T. Synthesis and recognition of biological motion patterns based on linear superposition of prototypical motion sequences. In *Proceedings of the IEEE Workshop on Multi-View Modeling and Analysis of Visual Scene*, 1999.

[Gra98] Grassia, S. Practical parameterization of rotations using the exponential map. *The Journal of Graphics Tools*, 3(3), 1998.

[IBMG04] Ilg, W., Bakir, G. H., Mezger, J., and Giese, M. On the Representation, Learning and Transfer of Spatio-Temporal Movement Characteristics. In *International Journal of Humanoid Robotics*, 2004.

[IG02] Ilg, W., and Giese, M. Modeling of movement sequences based on hierarchical spatio-temporal correspondences of movement primitives. In *Biologically motivated Computer Vision*, H. H. Bülthoff, S. W. Lee, T. Poggio, and C. Wallraven, Eds. Springer, 2002.

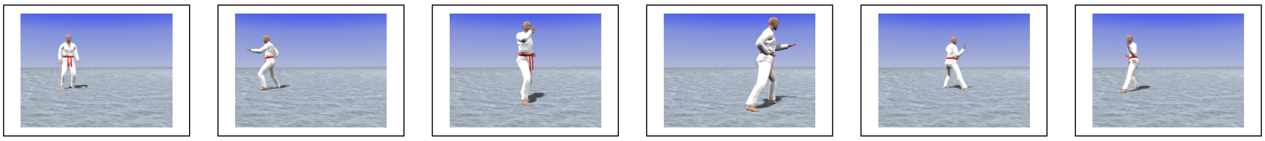

**Figure 6. Pictures from the morphed Karate sequence (about 30s overall).**

[IMG03] Ilg, W., Mezger, J., and Giese, M. Estimation of skill levels in sports based on hierarchical spatio-temporal correspondences. In *Proccedings of the 25th DAGM Pattern Recognition Symposium*, 2003.

[KG03] Kovar, L., and Gleicher, M. Flexible automatic motion blending with registration curves. In *Proceedings of Symposium on Computer Animation*, 2003.

[KG04] Kovar, L., and Gleicher, M. Automated extraction and parameterization of motions in large data sets. In *Proceedings of ACM SIGGRAPH*, 2004.

[KGP02] Kovar, L., Gleicher, M., and Pighin, F. Motion graphs. In *Proceedings of ACM SIGGRAPH*, 2002.

[LS02] Lee, J., and Shin, S. General construction of timedomain filters for orientation data, 2002.

[LWS02] Li, Y., Wang, T., and Shum, H. Motion texture: A two-level statistical model for character motion synthesis. In *Proceedings of ACM SIGGRAPH*, 2002.

[PSS02] Park, S. I., Shin, H. J., and Shin, S. Y. On-line locomotion generation based on motion blending. In *In Proceedings of Symposium on Computer Animation*. ACM SIGGRAPH, 2002.

[RCB98] Rose, C., Cohen, M. F., and Bodenheimer, B. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5), 1998.

[SKG03] Shin, H. J., Kovar, L., and Gleicher, M. Physical touch-up of human motions. In *Proceedings of Pacific Graphics*, 2003.

[TK03] Tak, S., and Ko, H. A physically-based motion retargeting filter. In *Proceedings of ACM SIGGRAPH*, 2003.

[TSK00] Tak, S., Song, O., and Ko, H. Motion balance filtering. *Computer Graphics Forum*, 19(3), 2000.

[WK88] Witkin, A., and Kass, M. Spacetime constraints. *Computer Graphics*, 22(4), 1988.

[WP95] Witkin, A., and Popović, Z. Motion warping. In *Proceedings of ACM SIGGRAPH*, 1995.