

# Morphable models for the analysis and synthesis of complex motion pattern

M.A. Giese and T. Poggio

Center for Biological and Computational Learning  
Massachusetts Institute of Technology, E25-206 / 218  
Cambridge, MA 02142, USA  
Tel.: 617 253 0549 / 5230, FAX: 617 253 2964

E-mail: giese@ai.mit.edu  
tp@ai.mit.edu

Paper in  
*International Journal of Computer Vision*,  
**38** (1), 59-73 (2000)

# Morphable models for the analysis and synthesis of complex motion patterns

Martin A. Giese and Tomaso Poggio\*

*Center for Biological and Computational Learning  
Massachusetts Institute of Technology, E25-206 / 218  
Cambridge, MA 02142, USA  
Email: giese@mit.edu, tp@ai.mit.edu*

October 12, 2001

**Abstract.** It has been shown that the linear combination of prototypical views provides a powerful approach for the recognition and the synthesis of images of stationary three-dimensional objects. In this article, we present initial results that demonstrate that similar ideas can be developed for the recognition and synthesis of complex motion patterns. We present a technique that permits to represent complex motion or action patterns by linear combinations of a small number of prototypical image sequences. We demonstrate the applicability of this new approach for the synthesis and analysis of biological motion using simulated and real video data from different locomotion patterns. Our results show that complex motion patterns are embedded in pattern spaces with a defined topological structure, which can be uncovered with our methods. The underlying pattern space seems to have locally, but not globally, the properties of a linear vector space. It is shown how the knowledge about the topology of the pattern space can be exploited during pattern recognition. Our method may provide a new interesting approach for the analysis and synthesis of video sequences and complex movements.

**Keywords:** morphing, action recognition, nonrigid motion, animation, prototype, linear superposition, correspondence, Structural Risk Minimization

## 1. Introduction

The analysis and synthesis of complex movement patterns are significant topics in both, computer vision and computer graphics. Video sequences must be processed for many applications, like multi-media interfaces, animation, and surveillance. Meanwhile, a broad spectrum of methods exists for the analysis of movements in video data, and

---

\* Martin Giese is supported by the Deutsche Forschungsgemeinschaft Gi 305 1-1. The Center for Computational and Biological Learning is supported by a grant from Office of Naval Research under contract No. N00014-93-1-3085, Office of Naval Research under contract No. N00014-95-1-0600, and National Science Foundation under contract No. IIS-9800032. Additional support is provided by: AT & T, Central Research Institute of Electric Power Industry, Eastman Kodak Company, Daimler-Benz AG, Digital Equipment Corporation, Honda R & D Co., Ltd., NEC Fund, Nippon Telegraph & Telephone, and Siemens Corporate Research, Inc.



many different methods for the recognition of gestures and actions have been proposed (see Gavrilu, 1999 for a review). Some of these methods extract directly spatio-temporal features from image sequences (e.g. Niyogi and Adelson, 1994; Davis and Bobick 1996; Essa and Pentland, 1997). Other methods use models for the human body in combination with predictive filtering techniques (O'Rourke and Badler, 1982; Essa and Pentland, 1997; Yacoob and Black, 1999) or Hidden Markov Models (e.g. Starnier and Pentland, 1995). Some of these approaches include learning of the underlying models from sets of example images (e.g. Black and Jepson, 1996; Ahmad et al., 1997; Blake and Isard, 1998).

In computer graphics, the synthesis of complex motion patterns for animation is an important subject. Most available methods are based on models for the human body geometry and its dynamics (cf. e.g. Badler, 1993). Other methods use tracking data from real moving agents that is edited using different techniques, e.g. for modifying the movement (Lee and Shin, 1999) or for blending over between different movements (Bruderlin and Williams, 1995). In spite of this great interest in the analysis and generation of video data and the application of learning techniques in this field, few is known about the properties of the underlying spatio-temporal pattern spaces. What are the limits of learning-based representations ? How are their generalization properties, and how can they be exploited ?

The answers to such questions are available for representations of stationary images. It has been shown that sets of two-dimensional images of complex three-dimensional objects, like faces, can be represented accurately by linearly combining few prototypical images. It was shown also that it is possible to generate new (virtual) images of objects with changed pose or illumination, or with different values on more abstract dimensions, like gender, by linearly combining a relatively small set of prototypical images (Vetter and Poggio, 1997; Vetter, 1998; Blanz and Vetter, 1999). This allows an efficient learning-based representation of classes of images that encompass either the same object with different poses or different similar objects, like faces from different individuals. Furthermore, the compact parameterization of an image in terms of the weights of the linear superposition allows to represent complex image transformations, like changes of the object pose, by (typically nonlinear) transformations of the low-dimensional weight vector. The functional form of such transformations can be learned from examples for which the true pose of the object is known (Beymer and Poggio, 1996). This circumvents the reconstruction of the three-dimensional structure of the object for certain applications. The linear superposition approach has been successfully applied to object recognition (Jones and Poggio, 1997; Beymer and Poggio, 1996), as well as to computer graphics,

e.g. for animation and the synthesis of new views of an object (Jones, 1997; Ezzat and Poggio, 1999; Blanz and Vetter, 1999). Recently, the idea of a linear combination of prototypical images has been generalized to three dimensional descriptions of objects (Shelton, 1998; Vetter, 1998; Blanz and Vetter, 1999; Shelton, this issue).

Since the linear combination of prototypes has been successful in representing classes of stationary images, the question arises if it is possible to develop a similar framework for the representation of complex movement patterns in image sequences. In this case, the prototypes would be given by example image *sequences*, and a new image sequences or movement patterns would be generated by linearly combining these prototypes. The created synthetic movement patterns would represent "morphs" between the prototypical examples. In addition, the obtained learning-based representation could be used as basis for the recognition of complex movements, and for the estimation parameters which characterize them.

We show in this paper that a representation of movement patterns by a linear combination of prototypical examples is possible, and we present an evaluation of this approach. We show that the linear combination of biological motion patterns leads to new, relatively naturally looking movements. We demonstrate also that, based on a relatively small number of prototypes, classes of movements, like different types of locomotion, can be recognized and that parameters that characterize such movements, like the locomotion direction, can be estimated.

We systematically develop the theoretical considerations that underly our approach in sections 2 to 4, starting with a review of the basic ideas of the classical linear superposition approach for stationary images. In section 5, we present the results of an evaluation of the approach with simulated biological motion patterns and tracking data from real video sequences. Finally, we discuss the relationship between our work and other related approaches in section 6, and point to different interesting applications that are suggested by our results in section 7.

## 2. Linear object classes for stationary images

Gradual changes of the viewpoint of the camera or of the orientation of three-dimensional objects lead to smooth gradual changes of the two-dimensional images of the object. This makes it possible to represent images that show an object with a new orientation by linear combinations of a typically small number of prototypical views of the object with different orientations in space. The underlying basic idea

was first pointed out by Ullman and Basri (1991) who showed that, under certain conditions, a two-dimensional image of an object can be represented exactly by the linear combination of two images of the same object. This idea was later generalized (Vetter and Poggio, 1995; Vetter and Poggio, 1997), showing that for object classes, like faces, often a small number of prototypical examples is sufficient for a relatively accurate approximation. They demonstrated also that the same technique can be used to represent changes in the illumination or the texture of three-dimensional objects.

How can the linear combination of two prototypical images of an object meaningfully defined? To be useful, the linear combination must approximate another natural image of the object, like the object seen with a different orientation. The simple linear superposition of the gray value images showing the object with different view angles would not fulfill this requirement. It rather would look like a transparent superposition of different objects. A way to avoid this problem is to establish correspondence between the superpositioned images using an optic flow algorithm, and then to combine linearly the resulting correspondence vector fields<sup>1</sup> (Beymer et al., 1993; Beymer and Poggio, 1996; Vetter and Poggio, 1997). The resulting linearly combined correspondence vector fields can be used to warp the original image, resulting in new images that look relatively natural. Dependent on the weights in the linear superposition, the resulting new image is more or less similar to each individual prototype. The same technique can be used to linearly combine images from objects with slightly different shape, like faces from different individuals. In this case, the linear combination warps between the different example faces (e.g. Beymer and Poggio, 1996).

The linear combination of images can be mathematically described in the following way: Assume that the positions of the features of an image, e.g. its pixels, are given by a vector  $\mathbf{x}$ , and that a reference image is characterized by the vector  $\mathbf{x}_0$ . The correspondence vector field that warps the reference image into the image is then given by the spatial displacement vector  $\boldsymbol{\xi} = \mathbf{x} - \mathbf{x}_0$ . Such displacement vector fields can be calculated with a usual optic flow field algorithm. If a set of prototypical images is given, characterized by the feature vectors  $\mathbf{x}_1, \dots, \mathbf{x}_P$ , the linear combination of the images with weights  $c_p$  is defined by the *linear combination* of the associated displacement vectors  $\boldsymbol{\xi}_p$ , resulting

---

<sup>1</sup> We present here a simplification of the linear combination approach by Vetter and Poggio that does not include the treatment of texture information.

in the superpositioned displacement vector:

$$\boldsymbol{\xi} = \sum_{p=1}^P c_p \boldsymbol{\xi}_p \quad (1)$$

To recover the new "superpositioned" image, the new displacement vector  $\boldsymbol{\xi}$  must be added to the feature positions of the reference image, resulting in the new feature positions  $\mathbf{x} = \boldsymbol{\xi} + \mathbf{x}_0$ .

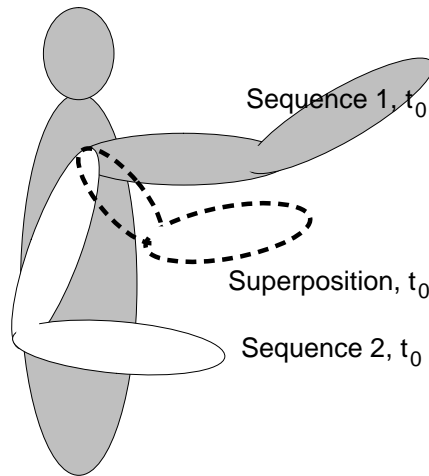
The vectors  $\boldsymbol{\xi}_p$  are the basis vectors of a linear space. When, as assumed above, all linear combinations of these basis vectors correspond to naturally looking images the method parameterizes a continuous class of images. In particular, the method imposes a topology onto this class: the topology of an *Euclidian linear space*. The weight vector  $\mathbf{c} = [c_1, \dots, c_P]$  characterizes each image of this class in a compact way. Image transformations, like view point changes, can be expressed in terms of changes of the weight vector. Preassigning the elements of the weight vector permits to create new synthetic images that are morphs between the prototypical images. Finally, a linear combination of the form (1) can be fitted to the correspondence vector field  $\boldsymbol{\xi}$  of a new image with respect to the reference image. The resulting weight vector can be used either to classify the identity of the shown object (e.g. faces of different individuals), or to estimate continuous parameters that characterize the image (like the pose of the face). The functional relationship between the weight vector and the interesting continuous parameter can be learned from examples using radial basis function networks (e.g. Giroi, Jones and Poggio, 1995; Beymer and Poggio, 1996).

### 3. Object classes for image sequences

How can the concept of a linear superposition of prototypes be made applicable to video sequences? From the analysis in the last section it follows that such a generalization is possible in a straight forward manner, when "correspondence" between image sequences can be defined in a meaningful way. In this case, it is possible to use a set of video sequences as prototypes, and to define "linear combination" again by the linear superposition of the associated correspondence fields. *The central theoretical problem is thus the adequate definition of "correspondence" between different image sequences.*

The condition for an *adequate* definition of correspondence is again that linear superpositions should result in motion patterns that have a natural appearance. Otherwise, the linear combination defines some

## Frame-by-frame Correspondence



*Figure 1.* Frame-by-frame correspondence can lead to strong spatial distortions when patterns with different timing are linearly combined.

arbitrary spatio-temporal pattern which can not be exploited for the representation of meaningful information. The definition of "correspondence" should, therefore, ensure that linear combinations approximate naturally looking motion patterns for a maximal set of prototypical image sequences.

The probably simplest definition of correspondence between two time-sequences of images is to establish correspondence between the image pairs that occur at the same points in time. This replaces the problem of determining the correspondence between image sequences by the repeated calculation of usual correspondence between stationary image pairs. Unfortunately, this simple strategy is not adequate. The reason for this is illustrated in Figure 1. Assume the prototypes are two arm movements of a person that have the same trajectory in space, but which are executed with different timing. Assume further, that the two patterns are combined with equal linear weights so that the feature positions of the linear combination at the time  $t_0$  are exactly halfway between the original feature positions in the images of the prototypical sequences. The figure shows that in this situation a strong deformation of the shape of the arm arises. The combined pattern does not look very natural and can thus not be used to encode a natural movement sequence.

A more satisfying solution would be to time-warp the two sequences into a single pattern with a normalized timing. In this way, the image frames in the two sequences that are most similar would be brought into correspondence. The linear combination could then be defined by lin-

early combining the associated temporal shifts resulting in new arm movements with the same spatial trajectory as the original patterns, but with an intermediate timing that varies continuously between the timings of the prototypical movements. This method implies that we must admit not only *spatial shifts*, but also *temporal shifts* during the correspondence process.

Mathematically, this *spatio-temporal correspondence problem* can be formulated in the following way: Given are two image sequences that are characterized by two potentially high-dimensional trajectories<sup>2</sup> of feature positions  $\mathbf{x}_1(t)$  and  $\mathbf{x}_2(t)$ . Bringing the image sequences into correspondence means to determine sets of spatial shifts  $\boldsymbol{\xi}(t)$  and temporal shifts  $\tau(t)$  that map the two trajectories onto each other. The image frame in the sequence  $\mathbf{x}_1$  that matches best the frame at time  $t$  in the second sequence  $\mathbf{x}_2$  may be shifted in time, occurring at  $t' = t + \tau(t)$ . The temporal and spatial displacement fields must, therefore, obey the following two equations:

$$\mathbf{x}_2(t) = \mathbf{x}_1(t') + \boldsymbol{\xi}(t) \quad (2)$$

$$t' = t + \tau(t) \quad (3)$$

Assume that a set of prototypical image sequences is given. Relative to a reference sequence  $\mathbf{x}_0(t)$  the prototypical sequences are temporally and spatially shifted with the shifts  $\boldsymbol{\xi}_p(t)$  and  $\tau_p(t)$ . A satisfying definition of the *linear combination of motion patterns*, that permits also the interpolation between sequences that are time warps of each other, is obtained by linearly combining the spatial and temporal shifts using the same linear weights  $c_p$ . The spatial and temporal shifts of the superpositioned image sequence is then given by:

$$\begin{aligned} \boldsymbol{\xi}(t) &= \sum_{p=1}^P c_p \boldsymbol{\xi}_p(t) \\ \tau(t) &= \sum_{p=1}^P c_p \tau_p(t) \end{aligned} \quad (4)$$

To obtain the feature positions of the linearly combined sequence, equations (2) and (3) can be used, yielding  $\mathbf{x}(t) = \mathbf{x}_0(t + \tau(t)) + \boldsymbol{\xi}(t)$ .

---

<sup>2</sup> For simplicity, we discuss our method with a continuous time parameter.

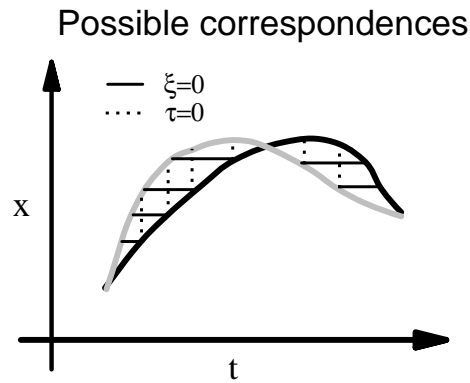


Figure 2. The correspondence problem is ill-posed: The shifts  $\xi(t)$  and  $\tau(t)$  are not unique.

#### 4. Correspondence Algorithm

Having defined the meaning of correspondence between image sequences, we have to specify how such correspondences can be calculated. Before devising a correspondence algorithm, it is important to recognize that the underlying problem is *ill-posed*. This is even true when there is no ambiguity with respect to the features that have to be brought into correspondence, as illustrated in Figure 2. Assume for the moment the very simplified case that there is only a single feature that moves along a single direction. An "image sequence" is then given by a simple spatio-temporal trajectory. The figure shows two such trajectories that are time-warps of each other. Obviously, there are infinitely many possibilities to bring these two trajectories into correspondence. One may try to account for all differences between the trajectories by admitting only spatial shifts ( $\tau(t) \equiv 0$ ). This leads to the correspondences indicated by the dashed line in figure 2. Another possibility is to compensate, where ever possible, the differences between the trajectories by temporal shifts. This leads to zero spatial shifts ( $\xi(t) \equiv 0$ ), as illustrated by the solid lines in figure 2. There are many other possibilities to combine spatial and temporal shifts that lead to smooth spatio-temporal shift functions. To make the solution of the correspondence problem unique we must specify the *relative trade-off between spatial and temporal shifts*. This trade-off determines the regimes over which the obtained representation interpolates in space and time between the prototypical image sequences.

The idea of an external specification of the trade-off between space and time leads to a correspondence algorithm in a relatively straight forward way. We obtain the correspondences by minimizing an error

function that is a weighted sum of the spatial and temporal shifts:

$$E_c[\boldsymbol{\xi}, \tau] = \int [|\boldsymbol{\xi}(t)|^2 + \lambda \tau(t)^2] dt \quad (5)$$

$\lambda$  is a positive constant that defines the relative weight of spatial and temporal errors. The value of this parameter is chosen in a way that ensures good interpolation between prototypical sequences that differ with respect to their spatial and temporal structure. Unfortunately, the minimization of the error functional  $E_c$  is complicated by additional restrictions for the temporal shift function  $\tau(t)$ . It must be ensured that the mapping between the time  $t$  and the corresponding time  $t'$  is continuous and one-to-one. Otherwise the same frame of the sequence  $\mathbf{x}_2$  would be associated with multiple frames, or no frame in the image sequence  $\mathbf{x}_1$ . This would violate the uniqueness of the solution of the correspondence problem. A unique solution can be enforced by requiring the mapping of  $t$  onto  $t'$  to be strictly monotonic. Additionally, it makes often sense to map the begin of the first image sequence onto the begin of the second, and the last frames of the first sequence onto the last frame of the second. This yields the following constraints for the continuous temporal shift function  $\tau(t)$ :

$$d\tau/dt > -1 \quad (6)$$

$$\tau(0) = \tau(t_{\max}) = 0 \quad (7)$$

It turns out that *Dynamic Programming* provides a method that permits a relatively efficient solution of this constrained optimization problem. In fact, a very similar optimization problem has to be solved for *Dynamic Time Warping*, which is a standard method in speech recognition for adjusting the timing of recorded speech patterns to match them with time-normalized reference patterns (Rabiner and Juang, 1993). In speech recognition, the dimensionality of the relevant feature spaces (Fourier energy components) is relatively small, and the features can be easily obtained by filtering operations with high temporal sampling rates. In image processing, the dimensionality of the feature space can be very high (e.g. the number of pixels in an image). Even if only a very reduced feature set is used (this is the approach that we adopted for the work presented in this paper), the identification of these features is usually computational expensive, and requires tracking or marking of components of the moving object. In order to reduce this computational effort, we developed an algorithm that is based on a sparse sampling of the features in time. This algorithm is described in more detail in the Appendix and in (Giese and Poggio, 1999).

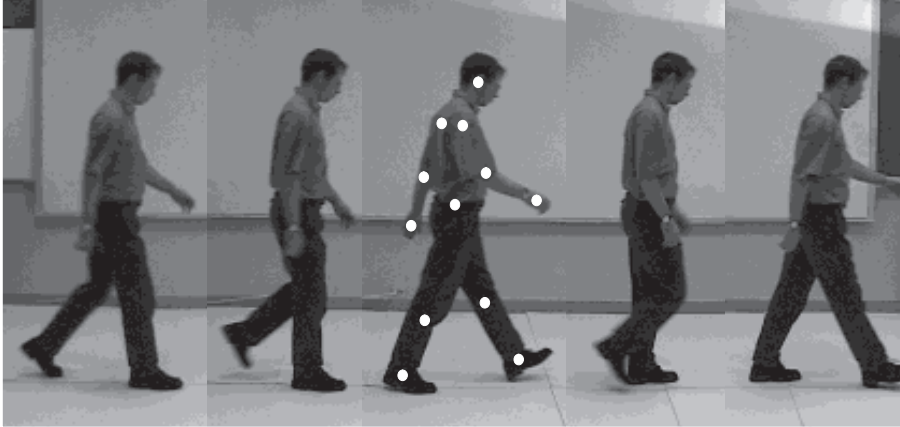


Figure 3. Real video sequence for "walking" and rendered feature points (*white dots*)

## 5. Results

We evaluated our new approach by testing its applicability for the synthesis and analysis of locomotion patterns. We used real video sequences, and simulated data sets for which the parameters of the motion could be exactly controlled. In the following, we will describe the experimental results separately for synthesis and analysis.

### 5.1. DATA SETS

**Simulated data:** The simulated data was obtained by using a three-dimensional model of a stick figure that performed three different styles of locomotion: "walking", "running", and "limping". We specified the periodic movement of the joint angles by a second order Fourier series for each individual joint, using 21 discrete time steps. The coefficients of the series were adjusted by hand such that a relatively natural appearance of the movement was achieved. "Limping" was generated from "walking" by time-warping. By reparameterizing the time axis using a monotonic distortion function, the first part of the walking movement was slowed down, and the second part was accelerated, such that the cycle time remained constant. The three-dimensional stick figure was then projected onto a two dimensional images using parallel projection. Different camera view angles could be specified for this projection. In the simulations, we used rotations of the camera axis to the side between +25 deg and -25 deg, and rotations up and down within the same angle regime. In total, over 150 such patterns were generated to evaluate the performance of our method.

**Real video sequences:** The video sequences showed a person performing different types of locomotion (see Figure 3). In addition to the three locomotion styles mentioned above, five different types of "marching" were recorded. These locomotion patterns differed with respect to the amplitude and style of the movement. They were supposed to cover a continuum with different degrees of "marching-like" appearance: starting with usual "walking", and ending with a kind of Russian "goose-step" at the other end of the continuous scale. For each movement type, we recorded multiple repeats in order to estimate the reliability of our method. In the video sequences, we "tracked" twelve feature positions by hand-marking specific points on the body (indicated by the white dots in Figure 3). Occluded feature points were interpolated, so that the resulting feature trajectories were smooth. To reduce the noise level of the feature trajectories, we fitted second order Fourier series to the individual trajectories. Additionally, the translatory movement was removed from the trajectories by identifying the hip position with the center of the coordinate system. Variations in scaling that resulted from the variable distance of the subject from the camera were compensated by normalizing the distance between hip and head to one. For this purpose, a linear function in time was fitted to the distance between hip and head, which then specified a normalization factor for adjusting the scaling. The cycle time of the patterns was normalized to a constant value, and the pattern was resampled with 21 discrete time steps per movement cycle. The number of recorded real video sequences was much smaller than the number of simulated patterns because the hand-tracking of the features was very time-consuming, and since simple automatic tracking methods were not applicable.

## 5.2. EFFICIENCY OF THE CORRESPONDENCE ALGORITHM

Before we present the results for the synthesis and analysis of complex movement patterns, we demonstrate in this section that our correspondence algorithm in fact yields spatial and temporal shifts that are consistent with the definition equations (2) and (3). Figure 4 shows the trajectory for a single feature coordinate for "walking" (dashed-dotted line) and "running" (solid line). The dashed line shows additionally the trajectory that arises if the walking trajectory is spatio-temporally warped using the estimated correspondence shifts (dashed line). The deviation between the original and the warped trajectory is small, showing that the correspondence algorithm finds a solution that fulfills the definition equations with high accuracy. A more detailed quantitative analysis shows that the deviation between the trajectories of the prototypes and their approximation by the warped reference

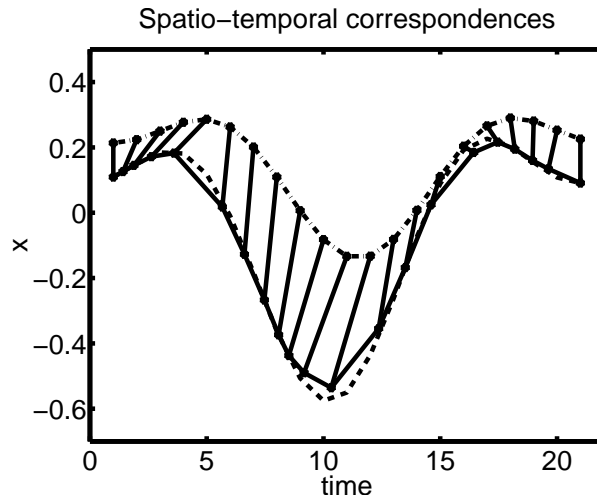


Figure 4. Feature trajectories for two prototypical patterns ("walking" and "running", *dashed-dotted and solid line*) and the established spatio temporal correspondences for a single feature trajectory. The *dashed line* indicates the trajectory that is obtained by spatio-temporal rewarping the walking pattern into the running pattern using the calculated correspondence vector fields.

trajectory is between 1 and 3% of the signal amplitude (measured in 2-norm and  $\infty$ -norm). The only exception was the correspondence between "walking" and strongly military marching ("goose step") where the deviations exceeded 5%. For the chosen value of the parameter  $\lambda = 10^{-4}$  the established correspondences specify substantial spatial as well as temporal shifts. This value for  $\lambda$  ensures good spatio-temporal interpolation for all prototypes and smooth temporal and spatial shift functions. The results presented in the following were not critically dependent of the exact choice of the parameter  $\lambda$ .

Figure 5 shows additionally an example from synthetic data where two prototypes were generated by time-warping using a nonlinear warping function<sup>3</sup>. The one prototype was perceived as "walking" and the time-warped pattern as "limping". The solid line shows an estimate of the time warping function that is determined by the temporal correspondences according to the second equation in (2). The true time-warping function is indicated by the dashed line in the figure. For the chosen value  $\lambda = 0.5 \cdot 10^{-5}$  the correspondence algorithm recovers almost exactly the applied temporal warping function. For larger values of  $\lambda$  a trade-off between spatial and temporal shifts occurs.

<sup>3</sup> The applied function was given by  $t = t' + 4 \sin((t' - 1)\pi/20) + 2 \sin((t' - 1)\pi/10)$ .

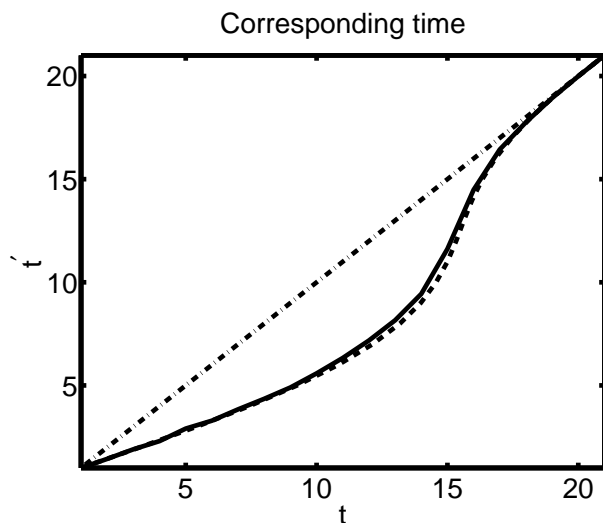


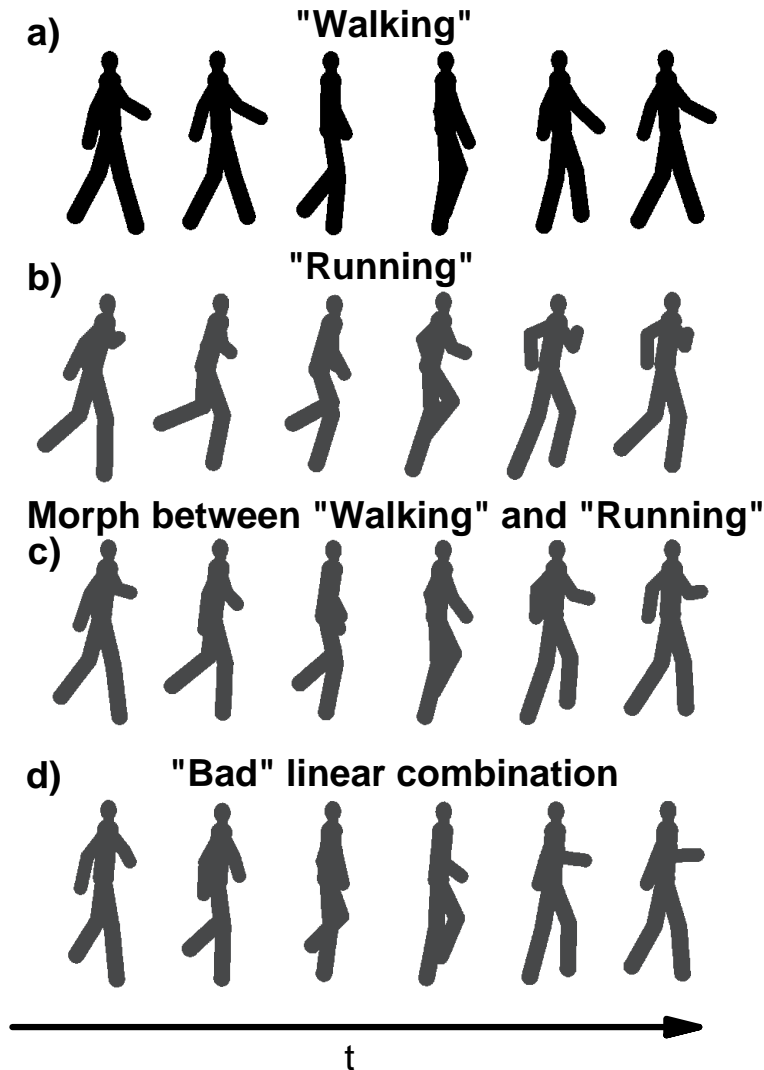
Figure 5. Recovering of the time-warping function when two prototypes are brought into correspondence that differ only with respect to their timing. The dashed line shows the true nonlinear function that was used to time-warp the pattern. The solid curve is the estimate that is obtained from the temporal shifts  $\tau(t)$  using equation (3).

### 5.3. SYNTHESIS OF NEW MOTION PATTERNS

For the synthesis of new motion patterns, a set of prototypical sequences was stored, and the correspondences between these sequences and a reference pattern were calculated<sup>4</sup>. Then the obtained spatial and temporal shifts were linearly combined using equations (4). For most experiments, we chose weight vectors  $\mathbf{c}$  with non-negative elements that were normalized by requiring  $\sum_p c_p = 1$ . The feature trajectories  $\mathbf{x}(t)$  of the obtained superpositioned pattern were used to animate a stick figure model that permitted to evaluate in how far the superposition looked like natural biological motion.

We found that the linear combination of different locomotion patterns, like "walking" and "running", or "walking" and "marching", results in mixed patterns that look amazingly natural. Such movements are perceived as intermediate forms of locomotion, like "fast walking", "walking in a military way", or "slight limping". Observers could typically not distinguish whether such patterns were based on real tracking data, or if they were generated synthetically by our method. The il-

<sup>4</sup> The reference sequence was the "centroid" of all prototypes. For the calculation of the centroid, first one prototype was chosen as reference pattern. The average of all correspondence fields was then used for readjusting all calculated temporal and spatial shifts for the centered reference pattern.



*Figure 6.* Motion morphing between prototypical image sequences: The first two panels a) and b) show one half-cycle of the two prototypical sequences for "walking" and "running". The figure tries to illustrate the movement by showing a series of snapshots from the image sequences that were equally spaced in time. Panel c) shows a half-cycle the linear combination of the prototypes for walking and running with equal weights 0.5. This sequence is typically perceived as a "fast walking" or a "slow running", and observers are not able to tell if this movement is natural or artificially generated. Panel d) illustrates an example for a linear combination that does not look like a natural form of locomotion. Observers see the figure walking obliquely in the direction of the camera. At the same time, the upper body of the figure performs a strange-looking rotational movement to the left side. (Only a half-cycle of the periodic movement is presented.)

illustration of the dynamical properties of biological motion patterns within a paper is difficult. We try to give an impression by Figure 6 which shows snapshots from the image sequences that were taken with equidistant temporal sampling<sup>5</sup>. The figure shows only one half-cycle of the periodic movements. Panels a) and b) show the prototypical sequences for "walking" and "running". Panel c) shows the image sequence that results from the linear combination of these two prototypes with equal linear weights. This movement pattern is perceived as a natural form of locomotion ("fast walking"). All linear combinations of "walking" and "running" with positive weights and  $0 \leq c_p \leq 1$  and  $\sum_p c_p = 1$  looked very natural. Our method permits thus continuous morphing between different styles of locomotion. This was true for both data sets, using trajectories from simulated and real data as prototypes. The same result was obtained when we morphed between "walking" and "limping" and between "walking" and "marching". Choosing weights  $c_p$  that exceeded one permitted to create movement patterns that exaggerate characteristic properties of individual locomotion styles ("very dynamic running" with extensive movement amplitudes of the arms and legs). This can be interpreted as a form of extrapolation in the pattern space of movement patterns.

Naturally looking linear combinations were, however, not always obtained. Panel d) illustrates a synthetic locomotion pattern that arises when "walking" orthogonal to the camera view axis is linearly combined with "running" in a direction that forms an angle of 45 deg with the camera axis. The illustrated patterns is perceived as unnatural by most observers. The figure seems to walk in a direction that is oblique with respect to the camera axis. At the same time the whole upper body of the walker seems to perform a strange rotational movement around the vertical body axis. The unnatural appearance of the pattern can not be accounted for by errors that are introduced by the correspondence algorithm. We tested that the correspondences between walking orthogonal to the camera axis and the running patterns in oblique directions all fulfill the definition equation (2) with sufficient accuracy (see above). Also, the effect is not dependent on the choice of the parameter  $\lambda$ .

The results of a more detailed analysis with simulated sequences using 15 prototypes (five different view angles for "walking", "running", and "limping") are subsumed in Table I. When patterns showing the same locomotion pattern (e.g. "walking") and different view angles are linearly combined, the resulting patterns look very natural. This result was expected because stationary images showing the same object

---

<sup>5</sup> MPEG demonstrations of the synthesized patterns can be found on the Internet on the page: <http://www.ai.mit.edu/projects/cbcl/people/giese/projects1.html>.

Table I. Results for synthesis: ++ very natural, + natural, and – unnatural natural appearance of the linearly combined movement patterns

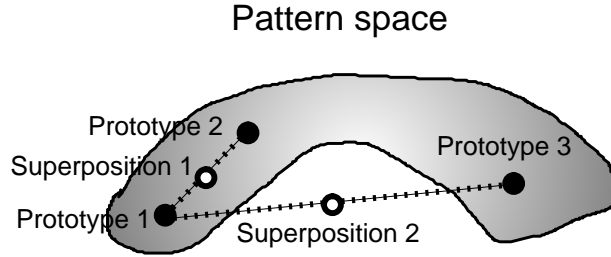
Locomotion Pattern		
View Angle	same	different
same	×	+
different	++	–

with different view angles lead also to reasonably looking interpolated patterns (e.g. Beymer and Poggio, 1996; Vetter and Poggio, 1997), predicting good results for a combination of sequences of such image pairs.

The fact that linear combinations of different locomotion styles with similar view angles lead also to naturally looking patterns shows that the investigated examples for biological motion all seem to lie within a continuous pattern space, permitting continuous *morphing* between them. The possibility to morph continuously between the the simulated patterns for "walking" and "limping", that were time-warps of each other, shows that our method, in fact, permits smooth interpolation between patterns with strongly different timing, but similar spatial trajectories of the movement.

Consistent with the example that was illustrated in the last panel of Figure 6, the more systematic analysis shows that linear combinations of different forms of locomotion with very different view angles of the camera lead to synthetic movement patterns that do not look natural. Linearly combining, for instance, "limping" and "running" with very different view angles (camera being 25 deg rotated to the side for one pattern, and 25 deg rotated upwards for the other) leads to combined patterns with geometrical distortions, like periodic contractions of individual limbs.

This result has an important theoretical implication: The fact that not even all convex linear combinations of the prototypical patterns are perceived as naturally-looking movement means that the space of biological motion patterns *does not have the topology of an Euclidian linear space* (at least not in the parameterization that is provided by our method). The fact that we can however smoothly interpolate between patterns that differ only with respect to the walking style, or the view angle shows that the patterns space is continuous, and seems locally to have a structure that can be approximated by a linear manifold. This is illustrated in Figure 7: Assuming that the pattern space can



*Figure 7.* When the pattern space is a nonlinear manifold linear superpositions are not always part of the manifold when the superposed prototypes are not close.

be approximated by a curved manifold, the superposition of patterns that are close on this manifold leads patterns that still lie within the manifold ("good looking movement"). Superposition of patterns that are far apart can result in superpositions that are outside the manifold (corresponding to "unnatural-looking" movement).

#### 5.4. RECOGNITION OF MOTION PATTERNS

Another application of the linear superposition approach is pattern recognition. For stationary images, it was shown that the linear combination of prototypes permits not only the synthesis of new views of an object, but also the recognition, e.g. of the face of a certain individual. Additionally, continuous parameters, like the pose angles of the face, can be estimated. For this reason, we tested if our approach permits the classification of motion patterns and the estimation of continuous parameters that characterize these patterns. Again we used simulated and real data sets for evaluation.

**Parameter estimation:** Assume that a new pattern is given by the feature trajectories  $\mathbf{x}(t)$ . First, the correspondence fields between this pattern and the reference sequence is calculated. Using equation (4), this correspondence field is approximated by a linear superposition of the correspondence fields of the prototypes ( $\xi_p(t)$  and  $\tau_p(t)$ ). The weights  $c_p$  of the linear superposition must be estimated. In simple cases, this is possible by minimizing the quadratic error function

$$E_a(\mathbf{c}) = \int \left[ \left| \xi(t) - \sum_{p=1}^P c_p \xi_p(t) \right|^2 + \lambda_a \left( \tau(t) - \sum_{p=1}^P c_p \tau_p(t) \right)^2 \right] dt \quad (8)$$

$\lambda_a > 0$  specifies the trade-off between temporal and spatial errors for the linear approximation. The solution of this minimization problem can be found by solving a simple linear equation system (for the details see Giese and Poggio, 1999). In cases where there are only few proto-

typical patterns this usual least squares estimation approach works well.

This is not the case when larger sets of prototypical patterns are used. This is illustrated in figure 8 (left panel). The figure shows, using color coding, for 48 test example patterns the distributions of the estimated weights for 15 prototypes. The test patterns are plotted along the horizontal axis, and the prototypes along the vertical axis. Test patterns and the prototypes are ordered with respect to the associated locomotion styles (W: "walking", R: "running" and L: "limping"). (The different entries for the same walking style differ with respect to the view angles.) The figure shows a relatively uniform distribution of the weights. This means that test patterns for "walking" lead to substantial loads also on the weights for the locomotion patterns "running" and "limping". We tested that a reliable classification of motion patterns based on these weight vectors is not possible.

A more detailed analysis reveals two reasons for this unreliable estimation of the linear weights: (a) By the relatively large number of prototypes, the function system which is defined by the linear combination of the correspondence fields (according to equation (4)) is too complex, or mathematically more precisely, its *capacity* is too large. This leads to overfitting and bad generalization properties. This motivates the requirement that the capacity (complexity) of the approximating linear function system should be as small as possible. (b) Inspection of Figure 8 (left panel) shows that the estimated weights often specify contributions of different prototypes which, when linearly combined, would not specify naturally looking motion patterns. Such linear combinations would thus correspond to points outside the admissible pattern manifold according to our considerations in section 5.3. It seems rea-

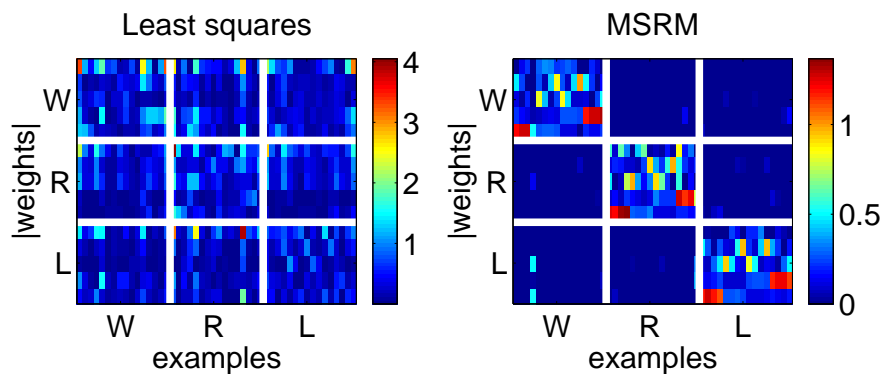


Figure 8. Absolute values of the weights estimated by (regularized) least squares and modified structural risk minimization method (MSRM).

sonable to require that only linear combinations are estimated that lie within this manifold, because only these can be meaningfully interpreted within the topology of the pattern space that is defined by our method.

An estimation method for the linear weights that takes into account the formulated requirements can be derived by modifying the Structural Risk Minimization Principle from Statistical Learning Theory (Vapnik, 1998). The underlying idea is to minimize the capacity of the approximating function system (measured by the Vapnik-Chervonenkis dimension) under the constraint that equation (4) is fulfilled within certain error bounds. Additionally, solutions which specify simultaneous contributions from incompatible prototypes, that would not lead to naturally looking motion patterns if they are linearly combined, can be punished by an additional regularization term. It is shown in Giese and Poggio (1999) that these different requirements can be embedded into a single constrained optimization problem. If  $E_a$  is the quadratic equation error of the the linear superposition equation (8) and  $\mathbf{c}$  the vector of the linear weights the function

$$E_S(\mathbf{c}) = E_a(\mathbf{c}) + |\mathbf{c}^T \mathbf{W} \mathbf{c}| \quad (9)$$

has to be minimized.  $\mathbf{W}$  is an adequately chosen positive weight matrix with non-negative elements. The function can be minimized efficiently with quadratic programming methods. With respect to further details we refer to Giese and Poggio (1999).

The right panel in Figure 8 shows that the *Modified Structural Risk Minimization Method* (MSRM) leads to sparse distributions of non-zero weights, and to a much more reliable estimation of the weight parameters. The test patterns load usually strongly only on prototypes with the same locomotion style, such that classification is now easily possible. The Modified Structural Risk Minimization Principle permits thus to embed the *a priori knowledge* about the topology of the pattern space into the estimation of the linear coefficients.

**Classification:** *Discriminating functions* are constructed from the estimated weight vectors. Let  $\mathbf{c}^p$  be the weight vector that is estimated when the prototypical patterns  $p$  is used as test pattern itself. Let  $\mathbf{c}$  be the weight vector of a new pattern. If  $I_k$  is the index set that contains all indices of prototypes for the locomotion pattern  $k$  (e.g. walking), a discrimination function for this pattern can be defined in the following way:

$$D_k(\mathbf{c}) = \frac{\sum_{r \in I_k} \exp(-|\mathbf{c} - \mathbf{c}^r|^2 / 2\sigma^2)}{\sum_{q=1}^P \exp(-|\mathbf{c} - \mathbf{c}^q|^2 / 2\sigma^2)} \quad (10)$$

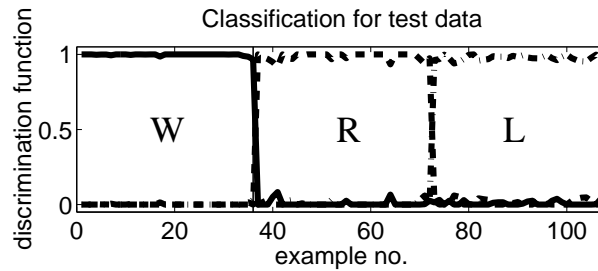


Figure 9. Discrimination between walking (W), running (R), limping (L) for 108 simulated patterns: The discrimination functions for walking (*solid line*), running (*dashed line*), and limping (*dashed-dotted line*) are plotted against the test examples ordered by locomotion types.

$\sigma$  is a positive parameter that determines the noise sensitivity of the discrimination. The quantity  $D_k$  is always between zero and one. Figure 9 shows the discrimination functions that are obtained for simulated data. The 108 test patterns are ordered along the horizontal axis according to the three different locomotion styles "walking" (W), "running" (R), and "limping" (L). The solid, dashed and dashed-dotted line show the corresponding values of the discrimination functions  $D_k$ . If a threshold of value for  $D_k$  of 0.5 has to be exceeded for a positive classification result all test patterns would be classified correctly. The classification is very robust.

Figure 10 shows the discrimination functions for the four locomotion patterns "walking" (W), "running" (R), "limping" (L), and "marching" (M) for twelve test examples of real video sequences. None of these test examples was used as prototype. Again defining a threshold value of 0.5 (dashed line in the figure), all patterns would be classified correctly. The variability of the discriminating functions is higher than in the case of the simulated data, but still the discrimination is very robust.

**Estimation of continuous parameters:** In order to estimate continuous parameters, the mappings between the linear weight vector  $\mathbf{c}$  and different interesting parameters were learned using radial basis function networks (Girosi et al., 1995). These networks were trained using sets of examples for which the true value of the interesting parameter was known.

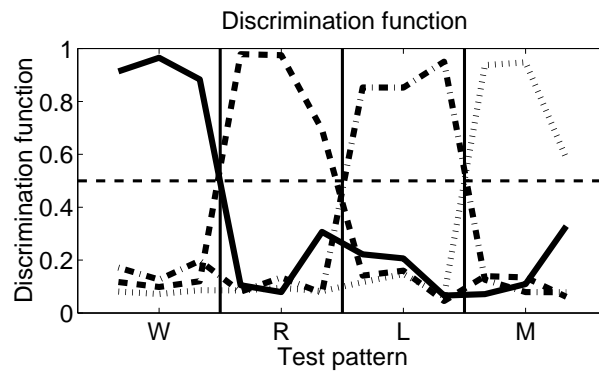


Figure 10. Discrimination between walking (W), running (R), limping (L), and marching (M): The discrimination functions for walking (*solid line*), running (*dashed line*), limping (*dashed-dotted line*), and marching (*dotted line*) are plotted against the test examples ordered by locomotion types.

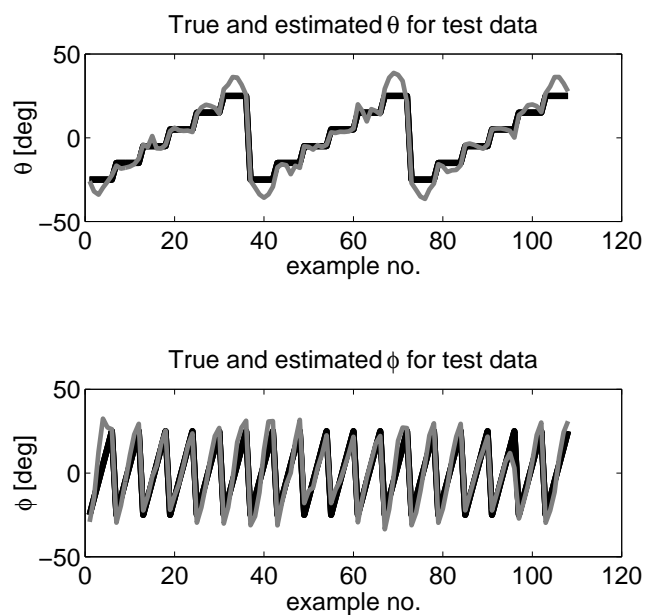


Figure 11. Estimated (*gray*) and real (*black*) view angles of the camera  $\theta$  and  $\phi$  for 108 simulated walking patterns

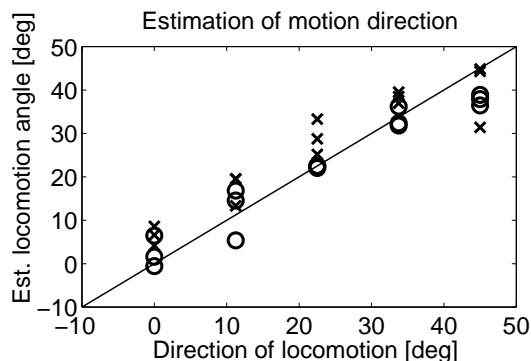


Figure 12. Estimation of locomotion direction for 30 real video sequences. (Angles are measured relative to the orthogonal of the camera axis.) *Crosses* indicate estimates obtained from walking test patterns, and *disks* estimates from running patterns.

Figure 11 shows the estimation of the two camera view angles (side rotation  $\phi$  and top / down rotation  $\theta$ ) for 108 simulated test patterns. The black curves indicate the true camera angles, and the gray curves the estimates obtained from the radial basis function networks. In this simulation, we used fifteen different prototypes with three locomotion styles (W, R, L) and five different view angles for each locomotion style.

Separate radial basis function networks were trained for the prototypes that belong to different locomotion styles. The estimates of these separate networks were then mixed by weighting them linearly with the (thresholded) discrimination functions  $D_k$  of the corresponding walking style. This ensures that the estimate is always determined by the sub-network that belongs to the most probable walking style.

Figure 12 shows the estimated motion direction for "walking" patterns (crosses) and "running" patterns (disks) from real video sequences. The variability of these estimated angles is higher than in the case of the estimation of the view angles for simulated data. However, the method seems to permit a coarse determination of the locomotion direction. The networks were trained with examples for "walking" and "running" for which the person moved with angles of 0, 22.5 and 45 deg relative a horizontal axis orthogonal to the optical axis of the camera. The estimates for the angles 11.25 and 33.75 deg belong thus to locomotion patterns that had no equivalents in the training data. These estimate reflect the generalization properties of the representation to similar patterns. All data in the figure is from test patterns that were not used to train the basis function networks.

Finally, we wanted to demonstrate that our method permits also to estimate continuous parameters that characterize more *abstract proper-*

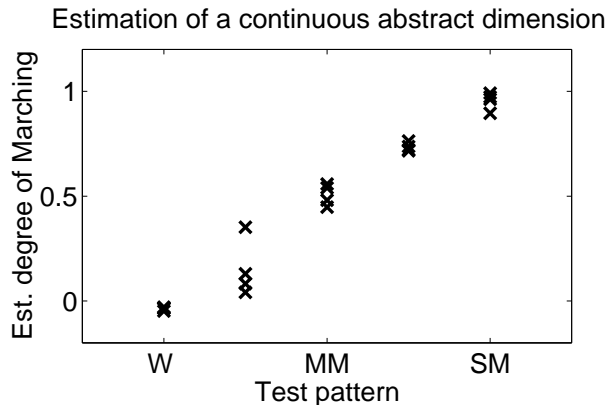


Figure 13. Estimation of an abstract parameter: "degree of marching"

*ties of movements.* An example is shown in Figure 13 where we tried to estimate the "degree of marching-like appearance" of locomotion patterns. The system was trained with three different movements: regular walking, marching with a medium strength, and an extreme degree of marching in the form of a Russian "goose-step". During training, walking (W) was paired with the parameter value  $\theta = 0$ , the second pattern (MM) with the value  $\theta = 0.5$ , and the "goose-step" (SM) with the value  $\theta = 1$ . For testing, new patterns were generated by marching with five different degrees of "strength" (from walking to "goose-step"). Figure 13 shows the data from 20 test sequences ordered by the executed "strength of marching" along the horizontal axis, and the estimated "degree of marching strength"  $\theta$  on the vertical axis. The method not only estimates correctly the values of the parameter  $\theta = 0, 0.5$  and 1 for the three locomotion patterns that were used during training. It also assigns intermediate parameter values to the intermediate marching patterns between walking (W) and medium marching (MM), and medium marching and strong marching (SM). The method permits thus the estimation of continuous abstract dimensions that characterize a movement, like the "strength" or "militaryness" of marching.

## 6. Conclusions

In this paper, we presented a new learning-based approach for the representation of complex motion patterns that is based on a linear combination of prototypical image sequences. We have identified the definition of spatio-temporal correspondence between image sequences as the central theoretical problem, and we have proposed an algorithm for its solution. Based on this solution, we demonstrated that many

techniques, that have been successful for the synthesis and analysis of stationary images can be transferred to the processing of complex motion patterns. In particular, we showed that our technique permits to morph continuously between different styles of locomotion. This makes it interesting for computer graphics and animation, e.g. for building dictionaries of similar movements with realistic appearance. We also showed that the linear combination approach permits to classify motion patterns, and to estimate continuous parameters that characterize them, like locomotion or view direction, and more abstract parameters, like the "degree of marching". This makes the method also interesting for computer vision, e.g. for the automatic identification of movements and the characterizing parameters, like their "sportiness", "elegance", etc. An interesting theoretical implication of our work was that biological motion patterns seem to form abstract pattern spaces with well-defined topology that can, at least locally, be linearized. This is important to construct efficient representations of broader classes of complex motion patterns with a limited number of prototypical patterns.

To our knowledge, the concept of a linear space of complex motion patterns has not been proposed in this form before. There is related work in computer graphics on morphing between trajectories in space and time for the interactive modification of motion trajectories, and for blending over different motions smoothly for animation (e.g. Bruderlin and Williams, 1995; Lee and Shin, 1999). This work is not based on the idea of a pattern space. Dynamic Time Warping was applied to image sequences before in order to make the recognition of motion patterns more invariant against variations in timing (e.g. Takahashi et al., 1994; Darrell, Essa and Pentland, 1995). There is a number of related papers that show that the linear combination of *stationary* images can be used for computer animation (Ezzat and Poggio, 1999; Blanz and Vetter, 1999). Our theoretical considerations imply that a linear combination in space-time should lead to improvements in quality and in storage costs compared to such approaches.

## 7. Future research

There are two major directions in which the presented methods can be substantially improved. The first is the combination of the recognition system presented in this paper with automatic learning-based tracking methods. This requires a reliable tracking of components of the human body. Different solutions to this problem have been proposed (e.g. Black and Jepson, 1996; Ahmad et al., 1997; Wren et al., 1997;

Blake and Isard, 1998). The frequent occurrence of occlusions in the relevant non-rigid motion patterns makes it necessary to modify the error norm for the correspondence process in order to make it robust against occlusions. Another intriguing application of our approach seem to be methods for image-based motion morphing. Such methods could provide an alternative to the model-based methods that are presently very popular in computer graphics. Morphable models for classes of movement patterns would be learned directly from example video sequences, instead of fitting geometrical models to the image data, and to model the movement kinematics. In principle, the proposed ideas should permit even to morph between different sequences of densely sampled gray value or color images. For this purpose, our correspondence method should be combined with robust methods for the estimation of optic flow. Preliminary experiments in this direction show that a simple transfer of the optic flow algorithms that have been used for the linear combination of stationary images (Vetter and Poggio, 1997; Vetter, 1998; Ezzat and Poggio, 1999) is not possible, because biological motion creates highly discontinuous optic flow fields that lead to strong distortions in the estimated optic flow. Additionally, the superposition of the resulting flow fields should be made "robust", e.g. by keeping track of the discontinuities to inhibit them from disturbing the superpositions. Both problems might be solved by including segmentation information into the estimation of the optic flow and the calculation of the linear superpositions. The resulting method would permit motion morphing between realistic video sequences without the requirement of complex realistically-looking computer graphics models..

### Acknowledgements

We thank A. Benali, C. Nakajima, and M. Riesenhuber for help with the data acquisition. We are grateful to A. Verri for comments on the manuscript.

### Appendix: Details about the correspondence algorithm

The developed correspondence algorithm has two steps. The first step uses temporally sparsely sampled key-frames and solves a discrete optimization problem using a Dynamic Programming technique. The obtained solution is refined in the second step by calculating quasi-continuous spatial and temporal shifts. This step is based on the solution of a

continuous optimization problem that is derived by linear interpolation between the key-frames.

The input of the algorithm are two image sequences that are sampled equally-spaced in time with the sampling interval  $T$ . Let  $\mathbf{x}_1[n]$  and  $\mathbf{x}_2[n]$  indicate the discretely sampled trajectories of the feature positions. The first step of the algorithm was inspired Dynamic Time Warping methods in speech recognition (Rabiner and Juang, 1993). The error functional (5) is replaced by a time-discrete approximation. For each frame pairing of the sequences 1 and 2 with discrete times  $n$  and  $n'$ ,  $1 \leq n, n' \leq N$ , we can assign an error function value:

$$E_d(n, n') = |\mathbf{x}_1[n] - \mathbf{x}_2[n']|^2 + \lambda(n - n')^2 T^2 \quad (11)$$

The error functional (5) is then approximated by the sum

$$E_{c,d} = \sum_n E_d(n, n'[n]) \quad (12)$$

where  $n'[n]$  specifies the discrete times  $n'$  for the sequence 2 that are corresponding to the times  $n$  in the sequence 1. The function  $E_d(n, n')$  defines an error surface over the  $n$ - $n'$ -plane. The function  $n'[n]$  defines a path in this plane, and  $E_{c,d}$  can be interpreted as a cost that is associated with this path. Finding the minimum of  $E_{c,d}$  is obviously equivalent to finding the path with the minimum cost.

The dynamic programming algorithm starts with the index pair  $n = 1$  and  $n' = 1$ . Along the path  $n$  always increases by one. To implement the monotonicity constraint (6), the set of permitted path transitions for  $n'$  is restricted through the inequality:

$$n'[n - 1] \leq n'[n] \leq n'[n - 1] + 2 \quad (13)$$

To enforce the end-point constraints (7) we introduced the additional restrictions:

$$2n - N \leq n'[n] \leq N \quad (14)$$

The computational efficiency of the dynamic programming method results from the fact that the cost of an optimal path of length  $n_0$ , only depends on the costs of the optimal paths with length  $n_0 - 1$ . This makes it possible to calculate the costs of all possible permitted paths in a recursive manner. (See Rabiner and Juang, 1993 for further details.)

The second step of our algorithm determines the exact spatial and temporal shifts by linearly interpolating between the discrete frames of the sequence  $\mathbf{x}_1$ . For this purpose we construct time-continuous interpolated feature trajectories by interpolating linearly between the

frames of the time discrete sequence. The interpolated (quasi) time-continuous image sequence in the time intervals  $I_1 = [(n-1)T, nT]$  and  $I_2 = [nT, (n+1)T]$  is given by the equation:

$$\mathbf{x}_1(t) = \begin{cases} \mathbf{x}_1[n] - (n - t/T) \mathbf{d}_1[n-1] & \text{for } t \in I_1 \\ \mathbf{x}_1[n] + (t/T - n) \mathbf{d}_1[n] & \text{for } t \in I_2 \end{cases} \quad (15)$$

with

$$\mathbf{d}_1[n] = \mathbf{x}_1[n+1] - \mathbf{x}_1[n]. \quad (16)$$

Introducing this approximation into the error function (5), one can analytically calculate the optimal time shifts within the two time intervals resulting in the expression

$$\tau(nT) = T \left( \frac{n|\mathbf{d}_1|^2 + \lambda T^2 n' + \mathbf{d}'_{21}[n', n] \mathbf{d}_1}{\lambda T^2 + |\mathbf{d}_1|^2} - n' \right) \quad (17)$$

where the vector  $\mathbf{d}_{21}$  is given by

$$\mathbf{d}_{21} = \mathbf{x}_2[n'] - \mathbf{x}_1[n].$$

For the interval  $I_1$  the term  $\mathbf{d}_1$  is given by  $\mathbf{d}_1 = \mathbf{d}_1[n-1]$ , whereas for the interval  $I_2$  the value  $\mathbf{d}_1 = \mathbf{d}_1[n]$  has to be chosen. In this way two candidate values for the time shift  $\tau(nT)$  are obtained, one for each interpolation interval. We selected the value that led to a smaller value of the error function  $E_c$ . Given the calculated optimal time shifts, one can use equation (15) to obtain the associated optimal spatial shifts.

## References

- Ahmad, T., C. J. Taylor, A. Lanitis, and T. F. Cootes: 1997, 'Tracking and recognizing hand gestures, using statistical shape models'. *Image and Vision Computing* **19**, (in press).
- Badler, N. I.: 1993, *Simulating Humans*. Oxford University Press, New York.
- Beymer, D. and T. Poggio: 1996, 'Image representations for visual learning'. *Science* **272**, 1905-1909.
- Beymer, D., A. Shashua, and T. Poggio: 1993, 'Example-based image analysis and synthesis'. Technical Report 1431, Massachusetts Institute of Technology, Cambridge, MA.
- Black, M. J. and A. D. Jepson: 1996, 'Eigen tracking: robust matching and tracking of articulated objects using a view-based representation'. In: *Proceedings of the European Conference on Computer Vision, Cambridge*.
- Blake, A. and M. Isard: 1998, *Active Contours*. Springer, New York.
- Blanz, V. and T. Vetter: 1999, 'Morphable model for the synthesis of 3D faces'. In: *Proceedings of SIGGRAPH 99, Los Angeles*. pp. 187-194.
- Bruderlin, A. and L. Williams: 1995, 'Motion signal processing'. In: *Proceedings of SIGGRAPH 95, Los Angeles*. pp. 97-104.

- Darrell, T. J., I. A. Essa, and A. Pentland: 1995, 'Task-specific gesture analysis in real-time using interpolated views'. Technical Report 364, Massachusetts Institute of Technology, Cambridge, MA.
- Davis, J. W. and A. F. Bobick: 1996, 'The representation and recognition of action using temporal templates'. Technical Report 402, Massachusetts Institute of Technology, Cambridge, MA.
- Essa, I. A. and A. P. Pentland: 1997, 'Coding, analysis, interpretation and recognition of facial expressions'. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**, 757-763.
- Ezzat, T. and T. Poggio: 1999, 'Visual speech synthesis by morphing visemes'. Technical Report 1658, Massachusetts Institute of Technology, Cambridge, MA.
- Gavrila, D. M.: 1999, 'The visual analysis of human movement: a survey'. *Computer Vision and Image Understanding* **73**, 82-98.
- Giese, M. A. and T. Poggio: 1999, 'Synthesis and recognition of biological motion patterns based on linear superposition of prototypical motion sequences'. In: IEEE (ed.): *Proceedings of the MVIEW 99 Symposium at CVPR, Fort Collins, CO*. pp. 73-80.
- Girosi, F., M. Jones, and T. Poggio: 1995, 'Regularization theory and neural network architectures'. *Neural Computation* **7**, 219-269.
- Jones, M. and T. Poggio: 1997, 'Model-based matching by linear combinations of prototypes'. In: *Proceedings of the DARPA Image Understanding Workshop, New Orleans, LA*. pp. 1357-1365.
- Jones, M. J.: 1997, 'Multidimensional morphable models: A framework for representing and matching object classes'. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.
- Lee, J. and S. Y. Shin: 1999, 'A hierarchical approach to interactive motion editing for human-like figures'. In: *Proceedings of SIGGRAPH 99, Los Angeles*. pp. 39-48.
- Niyogi, S. A. and E. H. Adelson: 1994, 'Analyzing and recognizing walking figures in XYT'. Technical Report 223, Massachusetts Institute of Technology, Cambridge, MA.
- O'Rourke, J. and N. I. Badler: 1982, 'Model-based analysis of human motion using constraint propagation'. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2**, 522-536.
- Rabiner, L. and B. H. Juang: 1993, *Fundamentals of Speech Recognition*. Prentice-Hall, Englewood Cliffs, NJ.
- Shelton, C. R.: 1998, 'Three-dimensional correspondence'. Master's thesis, Dept. of Computer Science, Cambridge, MA.
- Starner, T. and A. P. Pentland: 1995, 'Recognition of American Sign Language using hidden Markov models'. In: *International Workshop on Automatic Face and Gesture Recognition*.
- Takahashi, K., S. Seki, H. Kojima, and R. Oka: 1994, 'Recognition of dexterous manipulations from time-varying images'. In: *Proceedings of the Workshop on Motion of Non-Rigid and Articulated Objects*. pp. 23-28.
- Ullman, S. and R. Basri: 1991, 'recognition by linear combination of models'. *IEEE Transactions on Pattern Recognition and Machine Intelligence* **13**, 992-1006.
- Vapnik, V. N.: 1998, *Statistical Learning Theory*. Wiley, New York.
- Vetter, T.: 1998, 'Synthesis of novel views from a single face image'. *International Journal of Computer Vision* **28**(2), 103-116.

- Vetter, T. and T. Poggio: 1995, 'Linear object classes and image synthesis from a single example image'. Technical Report 1531, Massachusetts Institute of Technology, Cambridge, MA.
- Vetter, T. and T. Poggio: 1997, 'Linear object classes and image synthesis from a single example'. *IEEE Transactions on Pattern Recognition and Machine Intelligence* **19**(7), 733–742.
- Wren, C., A. Azarbayejani, T. Darrell, and A. Pantland: 1997, 'Real-time tracking of a human body'. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**, 780–785.
- Yacoob, Y. and M. J. Black: 1999, 'Parameterized modeling and recognition of activities'. *Computer Vision and Image Understanding* (**in press**).