Motion as Shape: A Novel Method for the Recognition and Prediction of Biological Motion

Abstract

We introduce a method for the recognition and prediction of motion, based on the idea that different motions trace out different shapes in some state space. In the recognition step we use a multidimensional generalization of the shape context [2] to find the closest prototype motion to the observed data. When tested against motion capture data, our model yields excellent (99%) recognition of gait and good (83%) recognition of identity. In addition to recognition, this process also allows us to find an aligning transform T_{DP} that maps the observed data D onto the prototype P. Given this transform, and its inverse T_{PD}, we use a Bayesian approach to make optimal predictions about the data in the prototype space and then map these predictions back into data space. This approach gives accurate predictions over several gait cycles despite the fact that there is often a significant difference between the observed data and the prototype manifold.

1 Introduction

Consider a movie of a person performing a particular motion. Each frame of the movie can be thought of as a point in an N-dimensional pixel space, where N is the number of pixels. As each picture is a point in pixel space, the movie defines a set of such points and traces out a trajectory within this space. More generally we can consider any state space $\mathscr{S}$ and think of a motion as tracing out a shape in that space. If $\mathscr{S}$ is 'well chosen' (e.g. to represent salient features of the motion - such as thigh and knee angles for gait recognition), then we can expect to find that similar motions trace out similar shapes and dissimilar motions trace out dissimilar shapes in $\mathscr{S}$. Therefore, given $\mathscr{S}$ we can approach the problem of motion recognition from the perspective of shape recognition. In particular, if we define a set of prototype motions, then we can recognize a motion simply by finding the 'most similar' prototype shape to the measured data.

In this paper we use a state space defined by the left and right thigh and knee angles (illustrated in the middle panel of figure 1) of various human subjects. Within this space we introduce a multidimensional generalization of the shape context [2] to compute an aligning transform T_{DP_M} between the observed angle data D (in data-space $\mathscr{D}$) and each prototype P_M for motion M (in prototype space $\mathscr{P}$). By looking at the details of these transformations (in particular the affine part of each transformation) we can use supervised linear discriminant analysis [8] to achieve excellent (99%) recognition of gait and fairly good (83%) recognition of individuals.

Given that we have recognized the motion, we then tackle the problem of motion prediction by making the simplifying assumption that all future data will continue to be

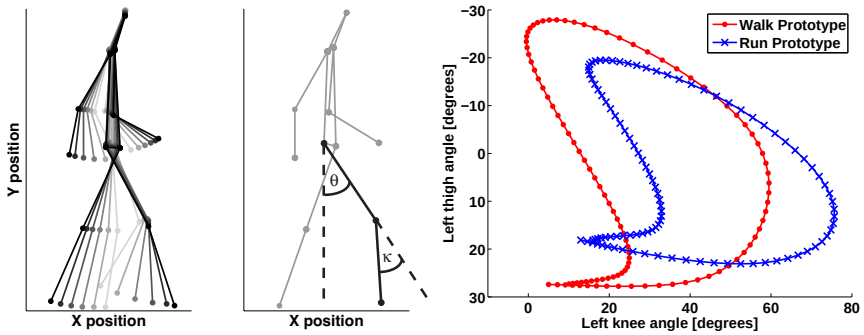

Figure 1: *Left*: Walking figure illustrating the positions of the markers over time. Darker lines correspond to more recent time points. *Middle*: Definitions of the thigh, θ, and knee, κ, angles as used in this paper. *Right*: Two dimensional projections of the prototype motion manifolds used in this paper.

mapped reliably onto P_M via T_{DP_M}. Hence to predict future values of D we simply project D onto P_M via T_{DP_M}; solve the prediction problem in $\mathscr{P}$ and then project the predictions back into $\mathscr{D}$ using $T_{P_M D}$.

This approach allows us to make accurate predictions of future thigh and knee angles over several gait cycles, despite often quite large differences between the data and the prototype manifolds.

1.1 Related work

There have been many varied approaches to the problem of action recognition and prediction, see [1] for a review. Our recognition approach is perhaps most similar to that of Elgammal and Lee [3]. In their paper they use a nonlinear manifold learning algorithm to embed silhouette data into a low dimensional subspace. They managed to align the low dimensional projections from several different subjects by using a thin-plate spline transformation to achieve separation of style (the person's identity) from content (the activity). The main problem with their approach is that there is no straightforward way to apply this algorithm quickly to a new person.

Probabilisitic models are not new to tracking (e.g. [11]). Urtasun et al. [13, 12] have developed models based on motion priors to aid tracking of golf swings and walking from video data. While similar in some ways to our prediction step, they are limited by the fact that their predictions are necessarily less accurate for motion 'far away from' the training set. Thanks to the aligning transform, we suffer this problem less.

2 Data Acquisition

Gait data was obtained using the ReActor motion capture system. Infra-red emitters were placed at 13 body positions (head, shoulders, elbows, wrists, hips, knees, ankles) on 6 human subjects (3 male). 3D spatial positions of markers were acquired at 33 frames/s. Subjects walked or ran at different set speeds on a treadmill. From this position data we

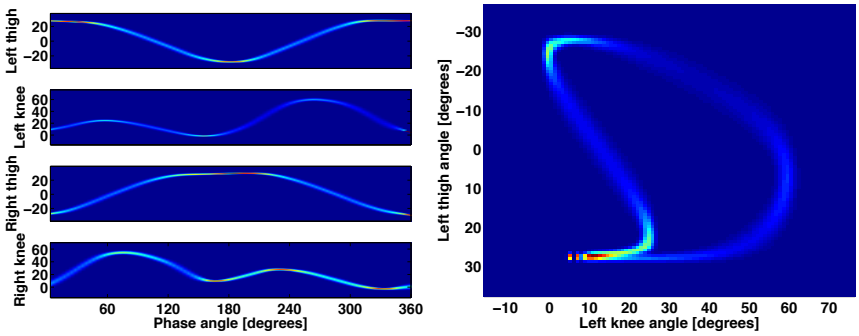

Figure 2: *Left*: $p(x_i|\phi,M)$, the prior probability, learned from the data, that each low level variable i (e.g. left thigh angle) takes on the value x_i given that the phase angle is ϕ and the motion is M. *Right*: $p(x_1,x_2|M)$, the joint distribution over the left thigh and knee angles for the walking prototype.

calculate the thigh (θ) and knee (κ) angles as shown in the middle panel of figure 1.

3 Defining prior probabilities and prototype manifolds

The first step in building the model is to get a good set of prototypes from the data. To do this we calculate a probability distribution over possible manifold shapes for a particular person performing the motion of interest (e.g. running or walking) and then take the 'most likely' manifold as our prototype.

We begin by labeling each point in the data set with a phase angle ϕ. This is done very simply by defining the start and end of each gait cycle by the extrema of the low-level variables and then interpolating ϕ as a linear function of time between $\phi = 0$ at the start of the gait cycle and $\phi = 2\pi$ at the end of each gate cycle.

Next we estimate $p(x_i|\phi,M)$, the probability that the low level variable i takes on the value x_i given that the motion is M and that the phase angle is ϕ. To do this we first bin each of the data points into one of 100 equally spaced bins based on their assigned phase. Then we make the assumption that the distribution over x_i within each bin is a Gaussian and calculate the mean and standard deviation of the distribution. The results of this process for the left and right thigh and knee angles for the walking prototype are shown in the left hand panel of figure 2.

Assuming that all the low level variables are independent given ϕ, the prototype, P_M, for motion M is then the set of N dimensional vectors $\{\mathbf{q}_j\}$ defined by the means $\{\mu_i(\phi_j)\}$ of the $p(x_i|\phi_j,M)$ distributions, i.e. $P_M = \{\mathbf{q}_j\}$; $\mathbf{q}_j = [\mu_1(\phi_j),\mu_2(\phi_j),...,\mu_N(\phi_j)]$, where ϕ_j is the phase angle at the center of the jth bin and the number of dimensions, $N = 4$. Two dimensional projections of the prototypes for run and walk are shown on the right of figure 1; such a projection is also known as a cyclogram [6, 5].

It is also interesting to calculate the prior distribution over the low level variables for motion M, $p(x_1,x_2,...,x_N|M)$. This is easily done if we make the assumption that $p(\phi|M)$ is uniform (which is ensured by our choice of ϕ as a linearly increasing function of time) and the result for the walk prototype is shown in figure 2.

Finally, for the prediction step we will need $p(\phi|x_i,M)$. Since $p(\phi|M)$ is a constant, $p(\phi|x_i,M)$ is readily obtained by renormalizing $p(x_i|\phi,M)$ with respect to ϕ.

4 Recognition with deformable templates

Recognition is achieved in three steps. First we use a multidimensional generalization of the shape context to solve the correspondence problem between the observed data D and the various prototype manifolds P_M. Next we estimate a set of aligning transformations T_{DP_M} to align the data to the prototype. Finally we use the details of the transformations T_{DP_M} to achieve recognition.

4.1 Two dimensional shape context

The two dimensional shape context [2] was introduced by Belongie et al. as a rich descriptor of shape that was used to solve the correspondence problem between two related, but not necessarily identical shapes.

Given a set of points $\{\mathbf{p}_i\}$ lying on shape S_1, we can describe the shape completely (up to a translation) simply by calculating the set of differences $\mathbf{d}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ for some point j on the shape. The two dimensional shape context for point j is then calculated by expressing the $\mathbf{d}_{ij}$ in polar coordinates $\mathbf{d}_{ij} = (r_{ij}, \theta_{ij})$ and computing the histogram h_j over r and θ.

The correspondence between two shapes is then calculated by using the Hungarian method [9, 7] to find the set of correspondences C_{ij} between point i on shape 1 and point j on shape 2 that minimizes the sum of the 'differences' Δ_{ij} between the shape contexts of corresponding points. In the current work, as in [2], Δ_{ij} is defined by the χ^2 test statistic.

In order to deal with outliers and potential differences in the numbers of points on shape 1 and shape 2, the algorithm allows for the possibility of mapping points onto dummy nodes whenever the cost of the best assignment is above some threshold ε_d. As we shall see this is very important in our case as it allows us to calculate correspondences between the densely sampled prototypes and the more sparsely sampled data.

4.2 Multidimensional shape context

The shape context is readily generalized to N dimensions. As before, for each point j on the shape we calculate the set of differences $\mathbf{d}_{ij} = \mathbf{p}_i - \mathbf{p}_j$, but this time the $\mathbf{d}_{ij}$ are N dimensional vectors. In analogy with the two dimensional case, we express $\mathbf{d}_{ij}$ in hyperspherical coordinates and calculate the shape context as the histogram over the radius, azimuthal angle and $N-2$ polar angles. We note that the case $N=3$ was used in [4] Once we have calculated the set of shape contexts for both shapes, then the correspondence problem is solved as before.

The left hand panel of figure 3 shows two dimensional projections of the four dimensional shapes for a sample of walking data (shown in blue) and the walking prototype (shown in red). Correspondences calculated using the 4-dimensional shape context are shown with black dashed lines. Note that despite significant differences in the paths of the two manifolds and in the number of points on each manifold, the algorithm still manages to find good correspondences.

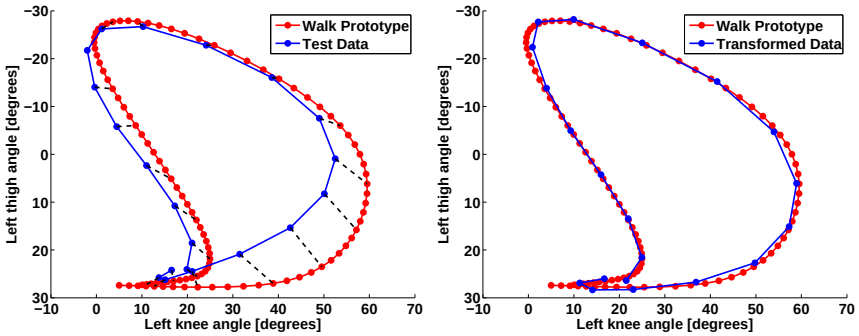

Figure 3: *Left*: Correspondences (black dashed lines) calculated with the four dimensional shape context between the data (shown in blue) and the walk prototype manifold (shown in red). *Right*: After the aligning transformation is applied to the data.

Note also that the ability to deal with different numbers of points on the data and prototype manifolds is very important as it allows us to have a more densely sampled prototype. This is useful as it allows the algorithm to find better correspondences than would be possible if the prototype was as sparsely sampled as the data. We also note that empirically we have observed the algorithm to find good correspondences even with very few (i.e. less that half a gait cycle) data points.

4.3 Computing the aligning transform

Once we have computed the correspondences between the data and the prototype manifolds we then seek to find an aligning transformation $T_{DP_M}(\mathbf{x})$ that will map the data points $\mathbf{p}_i$ in 'data'-space $\mathscr{D}$ on or close to its corresponding point $\mathbf{q}_j$ in 'prototype' space, $\mathscr{P}$ on the prototype. To do this we restrict ourselves to the thin-plate spline (TPS) transformations [14]. Using the algorithm developed in [2]

$$T_{DP_M}(\mathbf{x}) = [f_{x_1}(\mathbf{x}), f_{x_2}(\mathbf{x}), ..., f_{x_N}(\mathbf{x})] \tag{1}$$

$$f_{x_i}(\mathbf{x}) = a_{x_i} + \mathbf{A}_{x_i} \cdot \mathbf{x} + \sum_{j=1}^{n} \omega_j^{(x_i)} U\left(\left\|\mathbf{x}_j - \mathbf{x}\right\|\right) \tag{2}$$

where $U(r) = r^2 \log(r^2)$. a_{x_i} corresponds to a translation in the dimension defined by variable i (e.g. left thigh angle); $\mathbf{A}_{x_i}$ is a four dimensional vector corresponding to the part of the affine transformation that affects dimension i; the constants $\omega_j^{(x_i)}$ determine the amount of non-linear transformation or 'bending' of the prototype. All of the constants: a_{x_i}, $\mathbf{A}_{x_i}$ and $\omega_j^{(x_i)}$, are determined by minimizing the bending energy required to align the two shapes subject to the constraints detailed in [2]. Note that at this stage we also calculate the inverse transfrom i.e. $T_{P_M D}$ that maps points in $\mathscr{P}$ onto points in $\mathscr{D}$, via the same algorithm. This will be useful in the prediction step.

Results of applying the aligning transform to the observed data in the left of figure 3 are shown in the right hand panel of 3.

4.4 Recognition

We recognize the motion by using the affine parts of all the calculated transformations, i.e. $\{a_{x_i}\}$ and $\{\mathbf{A}_{x_i}\}$. These components of the transformation contain most of the information about the gait. More importantly, unlike the $\omega_j^{(x_i)}$, they do not directly depend on which points on the prototype are found as correspondences to the data.

For our simple demonstration with just two prototypes we thus describe each motion with a 40 dimensional vector $\mathbf{V} = (\{a_{x_i}\}_{walk}, \{\mathbf{A}_{x_i}\}_{walk}, \{a_{x_i}\}_{run}, \{\mathbf{A}_{x_i}\}_{run})$. We then achieve recognition by using supervised linear discriminant analysis within this space [8], training on half the data set and testing on the other half.

Note that we have taken a slightly different approach to recognition from [2] in which the bending energy of the transformation was used as a distance metric. This is because the bending energy does not take into account the affine part of the transformation, which for us is where most of the information lies.

4.4.1 Experiments

We tested the algorithm on two different tasks. In the first we asked it to classify the gait of 860 samples of 6 different people running or walking at speeds ranging from 3 to 5 mph for walking and 5 to 8 mph for running. Each sample consisted of just 20 data points, equivalent to 303ms of data; this is equivalent to about one gait cycle for the motions performed at 5 mph and is significantly less than one gait cycle for the slower motions. On this simple task we achieved a 99% success rate. Note that the speed of the joint angles is *not* included as a parameter in the recognition process and that recognition is achieved only by looking at the *shape* of the data.

In the second experiment we used the same data set to test whether the algorithm could identify both the gait and the identity of the actor. Training, as before, on half the data set and testing on the other half we achieved 83% recognition of identity and gait.

5 Prediction

After the recognition stage, we have the 'closest' prototype manifold P_M to the data D and a pair of transformations T_{DP_M} and T_{P_MD} that take us from data space $\mathscr{D}$ to prototype space $\mathscr{P}$. To make clear the distinction between the data represented in $\mathscr{P}$ and $\mathscr{D}$, we use y_i to denote the low level variables in $\mathscr{D}$ and x_i to denote their projections into $\mathscr{P}$; i.e. $\mathbf{x} = T_{DP_M}(\mathbf{y}); \quad \mathbf{y} = T_{P_MD}(\mathbf{x})$. Thus, since we can easily go back and forth between $\mathscr{D}$ and $\mathscr{P}$, the prediction problem becomes one of making predictions about the data in $\mathscr{P}$.

To make these predictions we make use of the phase angle ϕ. Since the low level variables $\{x_i\}$ are essentially determined by ϕ then if we can determine the dynamics of ϕ, we can fairly easily make predictions about the future by using a simple Taylor expansion for $\phi(t)$. Since we purposefully defined ϕ to be a *linearly* increasing function of time for the 'typical' prototype motion, this allows us to make good predictions with a simple linear model.

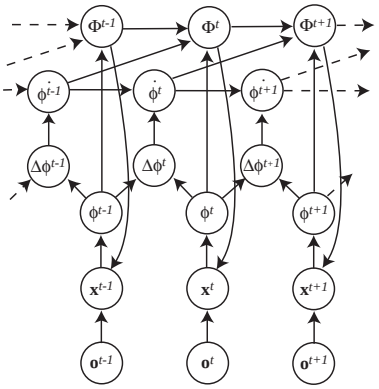

Figure 4: Schematic of the Bayesian model. See text for detailed description.

5.1 Model

Figure 4 shows an outline of the Bayesian model. It is *not* a belief network diagram in the sense of [10] in that the arrows are strictly one way. To help explain the model we break it down into four steps: bottom-up propagation of information; estimating the distribution over $\dot{\phi}$; using past information to gain an optimal estimate for the phase angle; and finally making top-down predictions.

5.1.1 Bottom-up propagation of information

At each time point t, low level data in the form of a set of observations o_i^t of the thigh and knee angles comes into the system as a set of probability distributions over x_i^t, i.e. $p(x_i^t|o_i^t) = \mathscr{G}(o_i^t, s_i)$, where $\mathscr{G}(a,b)$ is a Gaussian with mean a and standard deviation b. The s_i are set by the error in the measurement of the thigh and knee angles.

These are combined using the previously defined $p(\phi|x_i)$ to obtain an estimate for the distribution over ϕ^t, i.e.

$$p(\phi^t|\{o_i^t\},M) = \prod_i \int p(\phi|x_i,M)p(x_i^t|o_i^t)dx_i \tag{3}$$

However, this distribution does not take into account any prior knowledge about past values of ϕ or $\dot{\phi}$. To deal with this we must first estimate the phase velocity.

5.1.2 Estimating $\dot{\phi}$

We model each measurement of ϕ as independent, i.e. $p(\phi^t|\phi^{t-1}) = constant$, and estimate a distribution over $\Delta\phi^t = \phi^t - \phi^{t-1}$. This is given by the convolution

$$p(\Delta\phi^t|\{o_i^t\},\{o_i^{t-1}\}) \propto p(\phi^t|\{o_i^t\}) * p(\phi^{t-1}|\{o_i^{t-1}\}) \tag{4}$$

Clearly $\Delta\phi$ is related to $\dot{\phi}$ and this is used to update the distribution over $\dot{\phi}$ via

$$p(\dot{\phi}^t|O^t) = p(\Delta\phi^t|\{o_i^t\},\{o_i^{t-1}\})p(\Delta\phi^t|\Delta\phi^{t-1})\left[p(\dot{\phi}^t|\dot{\phi}^{t-1}) * p(\dot{\phi}^{t-1}|O^{t-1})\right] \tag{5}$$

where O^t is the set of all observations up to and including time t and $p(\dot{\phi}^t|\dot{\phi}^{t-1})$ is the transition probability that tells us how much we can rely on $\dot{\phi}^{t-1}$ to tell us anything about $\dot{\phi}^t$. $p(\Delta\phi^t|\Delta\phi^{t-1})$ is important as it takes into account the correlations between $\Delta\phi^t$ and $\Delta\phi^{t-1}$ which occur because $\Delta\phi$ is defined as the difference between two independent random variables (i.e. ϕ^t and ϕ^{t-1}). This is readily calculated as

$$p(\Delta\phi^t|\Delta\phi^{t-1}) = \int d\phi^{t-1} p\left(\phi^{t-1}|\{o_i^{t-1}\}\right) p\left(\phi^t = \phi^{t-1} + \Delta\phi^t|\{o_i^t\}\right) \times p\left(\phi^{t-2} = \phi^{t-1} - \Delta\phi^t|\{o_i^{t-2}\}\right) \quad (6)$$

5.1.3 Optimal estimate of the phase angle

Once we have calculated $p(\dot{\phi}^{t-1}|O^{t-1})$, it is then relatively straighforward to make predictions for the phase angle at the next time step. For clarity we introduce Φ^t to represent the phase angle infered from all possible data. We do this to ensure that there is no confusion when computing the distribution over $\Delta\phi$ and that we do not count information twice by using a predicted value of ϕ^t to artificially increase our belief in our estimate of $\dot{\phi}^t$.

We make our prediction for Φ^t as

$$p(\Phi^t|O^{t-1}) = p(\Phi^{t-1}|O^{t-1}) * p(\dot{\phi}^{t-1}|O^{t-1}) \quad (7)$$

where $*$ indicates a convolution. Then $p(\Phi^t|O^t)$ can be written as

$$p(\Phi^t|O^t) = p(\Phi^t|O^{t-1}) p(\phi^t|\{o_i^t\}, M) \quad (8)$$

5.1.4 Top-down propagation of information

Finally we update our estimate for the low level variables by making the top down predictions

$$p(x_i|O^t) = p(x_i|o_i^t) \int d\phi^t p(x_i|\Phi^t, M) p(\Phi^t|O^t \backslash o_i^t) \quad (9)$$

where $p(x_i|\Phi^t, M) = p(x_i|\phi, M)$, i.e. the prior defined in section 3 and

$$p(\Phi^t|O^t \backslash o_i^t) = \frac{p(\Phi^t|O^t)}{\int p(\phi|x_i, M) p(x_i^t|o_i^t) dx_i} \quad (10)$$

is the probability over ϕ given all the information O^t except o_i^t. Using this stops us counting information twice.

5.2 Results

The results of the prediction experiments are shown in figure 5. The top two and bottom left panels show the results of trying to predict the motion of one particular person. The aligning transform is calculated on the basis of the first 20 data points, after which (at $t =$ 0.3s) the input is removed (denoted by the black dotted lines). After this point the system can only make predictions based on its estimates for Φ and $\dot{\phi}$. In the top two panels we show the estimated thigh and knee angles from the Bayesian model (in red) compared with the actual observed angles(in blue). In the bottom left panel we shown the predicted phase angle (red) compared with the actual phase angle (blue).

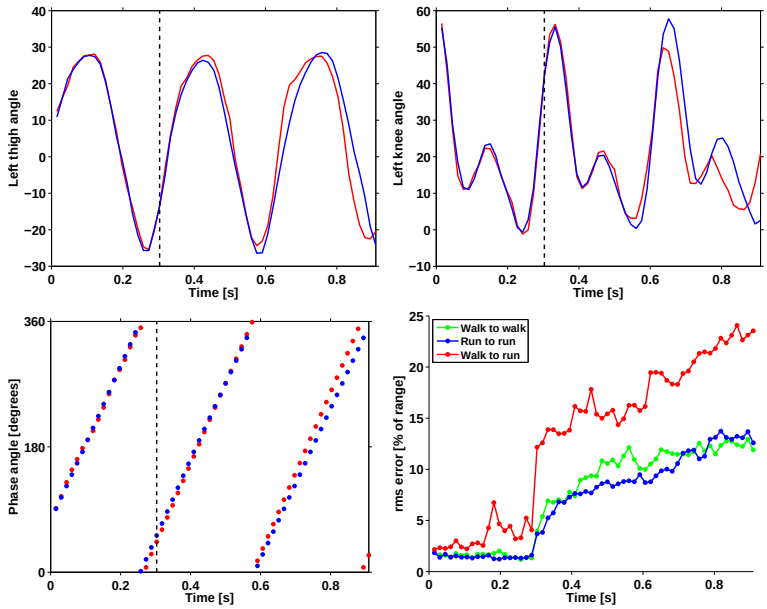

Figure 5: Prediction experiments. In all panels, the data were presented for 0.3s at which point (indicated by the black dotted lines) the input is removed and the system continues to make predictions on its own. *Top two panels*: Comparison between the observed (blue) and the predicted values (red) of the left thigh and knee angles. *Bottom left*: Comparison between the inferred (red) and actual (blue) phase angle as a function of time for the same motion. *Bottom right*: Root mean square prediction error averaged over 50 different example motions of the same person and expressed as a percentage of the range of each variable as a function of time.

In the bottom right panel we plot the mean square prediction error for 50 different motion samples of the same person as a percentage of the range of the low level variables. Expressing the error in this way allows us to average across variables (such as thigh and knee angles) with different ranges more fairly. For each motion, the aligning transform is calculated from the first 20 data points (303ms of data). We contrast three different cases, the green line shows the error when the walk manifold is used to predict walking data, the blue line when the run manifold is used to predict running data and the red line when the walking manifold is used to predict running data. Clearly this final case leads to much greater errors. This is an important result as without it one could imagine that, thanks to the aligning transform, *any* closed loop in $\mathscr{P}$ would suffice as a prototype for prediction. This result shows that the closer the prototype is to the actual motion the better the predictions will be.

6 Conclusions

We have presented a method for the recognition and prediction of motion capture data based on the idea that we can approach the problem of motion recognition from the per-

spective of shape recognition. In particular we introduced a proof of concept method using the shape context to compute an aligning thin-plate spline transformation between the data and a set of prototype motion shapes. Recognition was achieved by looking at the details of the affine parts of these transformations. Our model achieved excellent (99%) recognition of gait and reasonable (83%) recognition of individuals on our trial dataset.

Prediction was achieved by solving the prediction problem in prototype space and using the precalculated transformations to map the predictions back into the data space. Our model gives good predictions of the running and walking data over several gait cycles for the modest set of people included in our data set.

Acknowledgements

We thank Hanna Wallach for useful conversations about belief networks and the Center for Human Modeling and Simulation for use of the ReActor system. This research was supported by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the Office of Naval Research under grant N00014-01-1-0625.

References

[1] J.K. Aggarwal and Q. Cai. Human motion analysis: a review. *Computer Vision and Image Understanding*, 73 (3):428–440, 1999.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:509, 2002.

[3] A. Elgammal and C.-S. Lee. Separating style and context on a nonlinear manifold. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-04)*, Washington D.C., 2004.

[4] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik. Recognizing objects in range data using regional point descriptors. *ECCV*, 2004.

[5] A. Goswami. A new gait parameterization technique by means of cyclogram moments: Application to human slope walking. *Gait Posture*, 8(1):15–36, 1998.

[6] D.W. Grieve. The assessment of gait. *Physiotherapy*, 55(11):452–460, 1969.

[7] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38:325–340, 1987.

[8] W. J. Krzanowski. *Principles of multivariate analysis*. Oxford University Press, 1988.

[9] C. Papadimitriou and K. Stieglitz. *Combinatorial optimization: algorithms and complexity*. Prentice Hall, 1982.

[10] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. .* Morgan Kaufmann, San Mateo, CA, 1988.

[11] H. Sidenbladh, M. J. Black, and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking. *Proc. ECCV*, pages 784–800, 2002.

[12] R. Urtasun, D. J.Fleet, and P. Fua. Monocular 3d tracking of the golf swing. *In Conference on Computer Vision and Pattern Recognition (CVPR) San Diego, CA*, 2005.

[13] R. Urtasun, D. J.Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. *In International Conference in Computer Vision (ICCV) Beijing, China*, 2005.

[14] G. Wahba. Spline models for observational data. *Soc. Industrial and Applied Math.*, 1990.