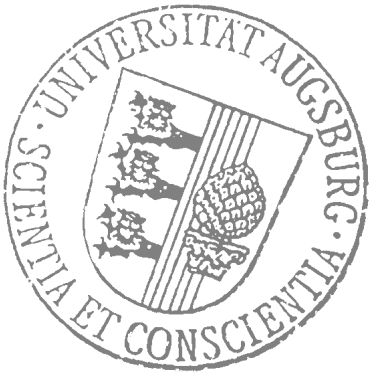
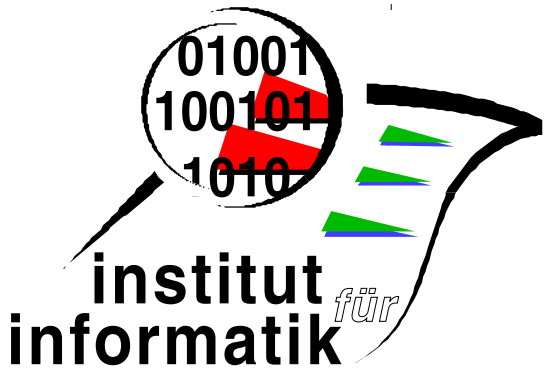
Universität Augsburg



A kinematic model for Bayesian tracking of cyclic human motion

Thomas Greif and Rainer Lienhart

Report 2009-16 Oktober 2009



Institut für Informatik
D-86135 Augsburg

A kinematic model for Bayesian tracking of cyclic human motion

Thomas Greif and Rainer Lienhart

Multimedia Computing Lab, University of Augsburg, Augsburg, Germany
{greif,lienhart}@informatik.uni-augsburg.de

ABSTRACT

We introduce a two–dimensional kinematic model for cyclic motions of humans, which is suitable for the use as temporal prior in any Bayesian tracking framework. This human motion model is solely based on simple kinematic properties: the joint accelerations. Distributions of joint accelerations subject to the cycle progress are learned from training data. We present results obtained by applying the introduced model to the cyclic motion of backstroke swimming in a Kalman filter framework that represents the posterior distribution by a Gaussian. We experimentally evaluate the sensitivity of the motion model with respect to the frequency and noise level of assumed appearance–based pose measurements by simulating various fidelities of the pose measurements using ground truth data.

Keywords: Motion model, Bayesian tracking, pose estimation, swim motion analysis

1. INTRODUCTION

Human pose estimation and pose tracking in videos are among the most challenging subjects in the domain of visual motion analysis. There exists a multiplicity of possible applications covering for example video indexing, surveillance or automatic analysis of sports videos. Generative models and Bayesian approaches to track human motions have become popular in the video domain[1] because frameworks such as Kalman[2] and particle[3,4] filters can cope with uncertainties as well as adapt quickly to changes by outputting posterior distributions over the pose state rather than a single pose hypothesis. Such approaches also encode temporal information about an underlying body representation. This information can significantly reduce the search-space in the image, can make background subtraction obsolete, and can facilitate the dealing with partial or complete occlusions. However, exploiting all these strengths requires the definition of an accurate motion model.

As opposed to the closely related works[5,6] the approach presented in this work only depends on angular accelerations and covariances extracted from training data, and our model is subject to simple kinematic properties. Hereby we can compute the temporal prior in an efficient way without the need to reduce the dimensionality where we could possibly lose important information. We can also skip the expensive projection and back projection (compared to Ref. 5) or Fourier transform (compared to Ref. 6). This makes our procedure less computationally expensive while still providing predictions accurate enough to recover the state distributions by exploiting the full dimension. Our decision to model only cyclic motion is motivated by the fact that many human activities (like walking, swimming, running) already follow a periodic pattern[7] and therefore many actions can very well be modeled with our cyclic motion model. We hence focus on the problem of developing an accurate and computationally efficient two–dimensional motion model for cyclic motions that can (1) provide the appearance model with a suitable temporal prior distribution, (2) be used in combination with popular Bayesian tracking frameworks, and (3) be easily extended to three dimensions.

1.1 Related work

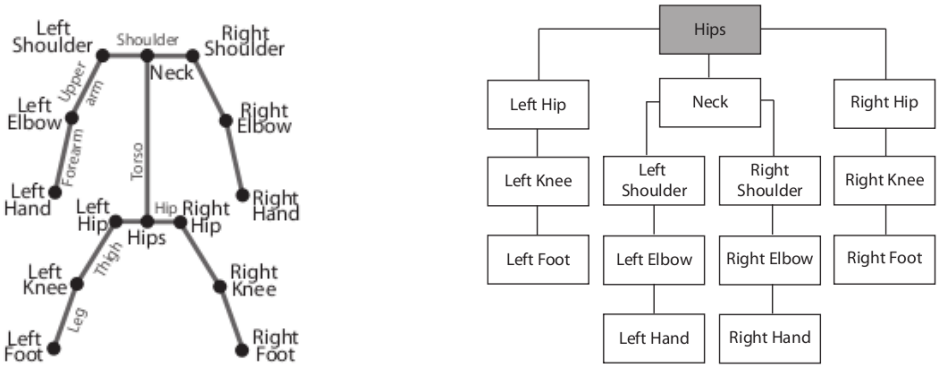
Although there exist several non–Bayesian approaches to track periodic motion[7,8] or action specific motion[9,10] we restrain ourselves to Bayesian techniques because of the advantages mentioned above. Out of the many Bayesian approaches especially particle filtering has become very popular since no assumptions are made concerning the posterior distribution.[11] The accuracy of the estimated posterior distribution rises with the number of samples used; however propagating them becomes computationally very expensive. Much work addresses this problem. Deutscher et al.[12] proposed a modified particle filter that uses deterministic annealing to reduce the number

of samples required. Rius et al.[13] used a mean cycle of the training data to avoid particle wastage. Other approaches reduce the dimensionality of the model itself. Agarwal and Triggs[14] learned and applied a linear model in a lower–dimensional subspace. For cyclic motions Sidenbladh[5] and Ormoneit et al.[6] introduced similar models. The latter is also concerned with extracting and aligning cycles from training data automatically, and their approach also accounts for missing information within the training data by making use of the Fourier transform.

2. MOTION MODEL

2.1 Body pose representation

We use a standard stick figure model to describe the high–dimensional and non–linear human motions. This model is composed of eleven rigid body parts and fourteen joints as depicted in figure 1a. The pose of a person is defined by the angles between each joint and its corresponding parent joint (relative to the image coordinate system translated into the parent), where the hierarchy shown in figure 1b is imposed on the joints with the center of the hip as root.



(a) (b)

Figure 1: Representation of a human body. (a) Stick figure model; joints are colored in black, body parts in gray. (b) Hierarchy imposed on the joints; angles of each joint are calculated relative to the coordinate system of the parent.

2.2 Model overview

We model the cyclic motion with respect to a *reference cycle* of defined length l (specified in number of frames) and corresponding *reference velocity* $v_{ref} = 1/l$. The state of a person within a cycle at time t is defined by the *state vector* $x_t = \left[\phi_t^0, v_{\phi,t}^0, \ldots, \phi_t^{12}, v_{\phi,t}^{12}, c_t, v_t\right]^T$ which contains the angles ϕ and respective angular velocities v_ϕ of the joints. c_t specifies the position within the reference cycle and v_t the velocity with which the person moves through each cycle. Additional parameters are used for tracking: the position of the root joint within a video frame and the segment lengths of the stick figure model. These parameters, however, can be determined independently and are not part of the pose state, because they do not constitute the motion model itself.

In a Bayesian tracking framework the motion model provides the temporal prior distribution $p(x_t|x_{t-1})$ that encodes how the state changes over time.[11] The essential component of the motion model is the *state transition function* that describes how this distribution is obtained from the previous state distribution. One can either generate samples according to this function (when e.g. using a particle filter) or apply the function directly (when e.g. using a Kalman filter).[11] In this work we propose a simple state transition function that is solely based on kinematics with additive noise:

$$x_t = \mathbf{A}_t x_{t-1} + \mathbf{a}_t + \epsilon_t, \tag{1}$$

where $x_t \in \mathbb{R}^m$ is the m–dimensional state at time t, $\mathbf{A}_t$ the $m \times m$ dimensional *state transition matrix*, $\mathbf{a}_t$ the m–dimensional *acceleration vector* encoding the effects of accelerations and ϵ_t a Gaussian noise vector with zero

mean. Unlike other approaches we do not try to learn the state transition function from training data. Instead we let simple kinematics govern this function. For each angle ϕ_t^j and angular velocity $v_{\phi,t}^j$ we know the following holds:

$$\phi_t^j = \phi_{t-1}^j + v_{\phi,t-1}^j \Delta t + \frac{1}{2} a_{\phi,c_{t-1}}^j (\Delta t)^2 \quad (2)$$

$$v_{\phi,t}^j = v_{\phi,t-1}^j + a_{\phi,c_{t-1}}^j \Delta t \quad (3)$$

where $a_{\phi,c_{t-1}}^j$ denotes the *angular* acceleration of joint j around angle ϕ at the relative temporal position c_{t-1} in the reference cycle. Δt is the difference in time between the former and the new state (here we always assume $\Delta t = 1$ frame). The same kinematics applies to c_t and v_t:

$$c_t = c_{t-1} + v_{t-1} \Delta t + \frac{1}{2} a_t (\Delta t)^2 \quad (4)$$

$$v_t = v_{t-1} + a_t \Delta t. \quad (5)$$

a_t is called the *temporal* acceleration which simply states whether nor not a person accelerates or decelerates the overall cyclic motion. This has to be included so that the model can adapt itself to different speeds. The above dependencies can also be expressed in matrix notation (to keep the notation simple, we only write it down for a single joint j):

$$\underbrace{\begin{pmatrix} \phi_t^j \\ v_{\phi,t}^j \\ c_t \\ v_t \end{pmatrix}}_{x_t} = \underbrace{\begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}_t} \cdot \underbrace{\begin{pmatrix} \phi_{t-1}^j \\ v_{\phi,t-1}^j \\ c_{t-1} \\ v_{t-1} \end{pmatrix}}_{x_{t-1}} + \underbrace{\begin{pmatrix} \frac{1}{2} a_{\phi,c_{t-1}}^j (\Delta t)^2 \\ a_{\phi,c_{t-1}}^j \Delta t \\ \frac{1}{2} a_{t-1} (\Delta t)^2 \\ a_{t-1} \Delta t \end{pmatrix}}_{\mathbf{a}_t} + \epsilon_t. \quad (6)$$

Given c_{t-1}, the pose at time t is completely defined by means of the acceleration vector $\mathbf{a}_t$ and corresponding noise ϵ_t. The new state distribution thus depends only on these vectors and they are coevally the only information we learn from training data.

2.3 Incorporating training data

Sample cycles specifying the configuration of the stick figure model in every frame are used to create the training data set by normalizing the duration (in frames) of each cycle using cubic spline interpolation. The angular accelerations are computed by differentiating them twice with respect to time using a common derivative filter,[15] and we base our estimation of the true accelerations on the mean vector $\mu_{a_{\phi,c_t}}$ and the covariance matrix $\Sigma_{a_{\phi,c_t}}$ of these accelerations at each c_t. Both quantities, however, apply only to the reference velocity v_{ref}, i.e., only if the modeled person moves through the cycle with the same velocity. When being applied to different velocities we have to alter them[16] based on the ratio $k = v_t / v_{ref}$ by the factor k^2.

We will combine the accelerations from the training data at time c_t with the Gaussian noise vector ϵ_t so that the added noise is subject to the variation within the training data, and hence the learned motion or action. In other words, the mean of the Gaussian noise will correspond to the mean accelerations extracted from training data, and the covariance is defined by the variation within these accelerations. Beginning with the angular components of the resulting noise vector we assume the accelerations extracted from the training data at c_t to be normally distributed according to

$$\left[a_{\phi,c_t}^0, a_{\phi,c_t}^1, \ldots, a_{\phi,c_t}^{12}\right]^T \sim \mathcal{N}(\mu_{a_{\phi,c_t}}, \Sigma_{a_{\phi,c_t}}) \quad (7)$$

and write the angular components of the acceleration vector $\mathbf{a}_t$ as a linear combination of the matrix K and the

training data accelerations:

$$\begin{pmatrix} \frac{1}{2}a^0_{\phi,c_t}(\Delta t)^2 \\ a^0_{\phi,c_t}\Delta t \\ \vdots \\ \frac{1}{2}a^{12}_{\phi,c_t}(\Delta t)^2 \\ a^{12}_{\phi,c_t}\Delta t \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{1}{2}(\Delta t)^2 & \dots & 0 \\ \Delta t & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \frac{1}{2}(\Delta t)^2 \\ 0 & \dots & \Delta t \end{pmatrix}}_{=:K} \cdot \begin{pmatrix} a^0_{\phi,c_t} \\ \vdots \\ a^{12}_{\phi,c_t} \end{pmatrix}. \tag{8}$$

By making use of the Gaussian property that states that if a random variable X is normally distributed, $X \sim \mathcal{N}(\mu, \Sigma)$, and another random variable Y is a linear combination of X, $Y = CX + b$, then Y is also normally distributed according to $Y \sim \mathcal{N}(C\mu+b, C\Sigma C^T)$, we know that the angular components of the acceleration vector are distributed according to

$$\left[\frac{1}{2}a^0_{\phi,c_t}(\Delta t)^2, a^0_{\phi,c_t}\Delta t, \dots, \frac{1}{2}a^{12}_{\phi,c_t}(\Delta t)^2, a^{12}_{\phi,c_t}\Delta t\right]^T \sim \mathcal{N}(K\mu_{a_{\phi,c_t}}, K\Sigma_{a_{\phi,c_t}}K^T). \tag{9}$$

The training data obviously cannot provide information about the temporal accelerations, because every training cycle has the same length l. Since we assume steady cyclic motions without persons rapidly accelerating or decelerating within the period of a single frame, we simply model these temporal accelerations by a Gaussian with zero mean, $a_t \sim \mathcal{N}(0, \sigma^2_{a_t})$, and applying the same properties as we did before, we know that the temporal components of the acceleration vector are distributed according to

$$\left[\frac{1}{2}a_t(\Delta t)^2, a_t\Delta t\right]^T \sim \mathcal{N}([0,0]^T, L\sigma^2_{a_t}L^T) \tag{10}$$

with $L = \left[\frac{1}{2}(\Delta t)^2, \Delta t\right]^T$. Under the assumption that angular and temporal accelerations are probabilistically independent, we can combine the above results to the new noise vector γ_t that is completely governed by the training data accelerations, and hence the modeled action, and rewrite the original state transition function as

$$x_t = \mathbf{A}_t x_{t-1} + \gamma_t \tag{11}$$

where $\gamma_t \sim \mathcal{N}(\mu_{c_t}, \Sigma_{c_t})$ and

$$\mu_{c_t} = \left[K\mu_{a_{\phi,c_t}}, 0, 0\right]^T, \Sigma_{c_t} = \begin{pmatrix} K\Sigma_{a_{\phi,c_t}}K^T & \mathbf{0} \\ \mathbf{0} & L\sigma^2_{a_t}L^T \end{pmatrix}. \tag{12}$$

The introduced model is a very simple but coevally a very effective one (see section 3). It covers the dynamics involved for a certain action and the training data already specifies how much variance there is to expect, making an additional empirical analysis obsolete. Only mean accelerations and corresponding covariances of the reference cycle are stored in memory, and the evaluation of the state transition function is of low complexity. This makes the model very computationally efficient without disregarding any dimensions of the state.

3. EXPERIMENTAL RESULTS

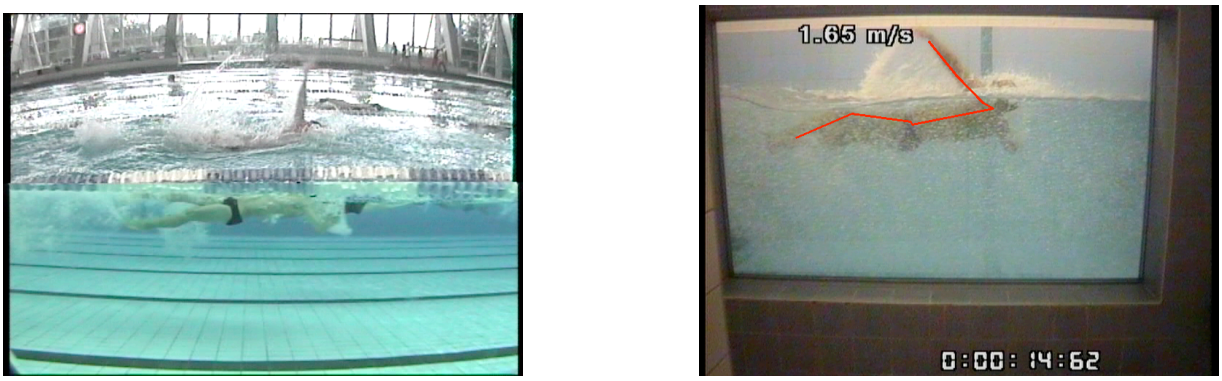
We evaluated the performance of the introduced model by using it within a Kalman filter and a particle filter framework to track the cyclic motion of backstroke swimmers. The measurement in each frame was a noisy observation of the true pose, i.e. the true angles at time t. Since Bayesian filters require the measurement z_t to be a conditional probability rather than a simple scalar value , i.e., $p(z_t|x_t) \sim \mathcal{N}(\mu, \sigma)$, we use the true angles as mean μ and the introduced noise σ as the standard deviation, defining roughly how far (in radians) the angles measured by the appearance model may deviate from the true ones of the pose. The measurement only provides information about the predicted angles, and not about the velocities or the progress, meaning that only these quantities of the state vector are updated by the measurement update. As stated, our main goal is to provide

a decent motion model for cyclic motion that can be used with any Bayesian tracking framework and is thus independent from the used appearance model. In order to systematically evaluate this model we have to control when and with what quality (in terms of measurement noise) a measurement update occurs. This gives us the ability to evaluate the model according to different criteria like how frequent measurements should occur, how accurate the measurement should be, and how long phases without any update can be handled. It then lets us state the requirements that should be met by an appearance model using our motion model. The evaluation in this section thus focuses completely on the motion model: how well is the motion itself predicted, i.e. how well are the angles approximated?

3.1 Datasets

The training data set consists of ten sample cycles of swimmers swimming backstroke. They were recorded from the side either with one camera (swimming canal; capturing the swimmer above and below the water surface) or two cameras (open-air pool; one camera recording the swimmer above and one below the water surface). In each frame the two–dimensional locations of the joints were manually labeled. We use female and male swimmers as well as two different locations to introduce enough variations into the data set: Six cycles (three by female/male swimmers) were recorded in a swimming canal, four cycles (two by female/male swimmers) in an open-air pool. The reference cycle length was set to $l = 100$ frames.

To test the performance we used two different test sets. The first test set consists of eight cycles from a known environment, but from different swimmers: two cycles each from female/male swimmers in the swimming canal and the open-air pool. The second test set contains four sample cycles of a female and a male swimmer in a swimming hall and thus was taken from an unknown environment for the motion model. The test set consisted of an overall number of 816 frames. Figure 2 depicts two example frames of the videos used.



(a) (b)

Figure 2: Two examples of the videos used: (a) Frame obtained from two synchronized cameras, (b) frame obtained from a single camera with manually labeled segments (only one body half is drawn for convenience, source: IAT Leipzig[17]).

3.2 Kalman filter results

The Kalman filter returns for every frame the state distribution represented by its moments, i.e. the mean and the covariance, rather than a single pose. We simply took the mean of this distribution as the estimated pose (which we denote by Φ_t) in each frame. The model was initialized from ground truth and training data, i.e. the angles were taken from the initializing frame in the ground truth data, and the velocities were estimated by the mean of the training data velocities at this frame. The quality of the estimated poses was measured by the absolute error between the predicted joint angles and the true ones, averaged over all joints and all frames in the test set.

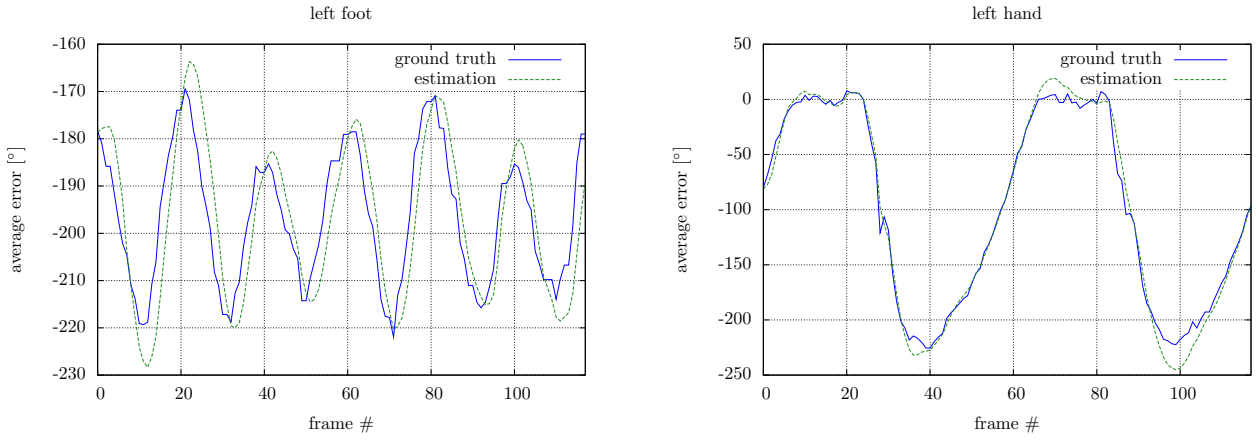
Table 1 shows the errors on both test sets for a varying measurement noise σ. Note that for simplification all angles are displayed in degrees rather than radians. As expected, the error rises with increased noise, i.e. with an increased uncertainty about the value of the measurement. We achieve a very low average error of

11.12° and 11.17° respectively for the highest measurement noise ($\sigma = 115°$) on both test sets, and even smaller ones (8.09°/8.21°) for a more realistic value of about 29°. Moreover, the choice of the test set, does not seem to influence the resulting error negatively, meaning that our model has no difficulties in adapting to unknown environments, and in the following we will therefore no longer distinguish between both sets.

	$\sigma = 1°$	$\sigma = 3°$	$\sigma = 6°$	$\sigma = 29°$	$\sigma = 57°$	$\sigma = 115°$
test set 1	2.8110	4.3950	5.3210	8.0852	9.5177	11.1174
test set 2	2.8865	4.6235	5.5994	8.2082	9.5934	11.1659

Table 1: Average absolute difference (in degrees) over all angles for varying measurement noise σ broken down by test sets.

Figure 3 shows an example of the true and estimated trajectories of the left foot as well as the left hand for two complete cycles. The angles estimated by our model follow the true ones closely, and our model even predicted a rather smooth motion when compared to the noisy ground truth data.



(a) (b)

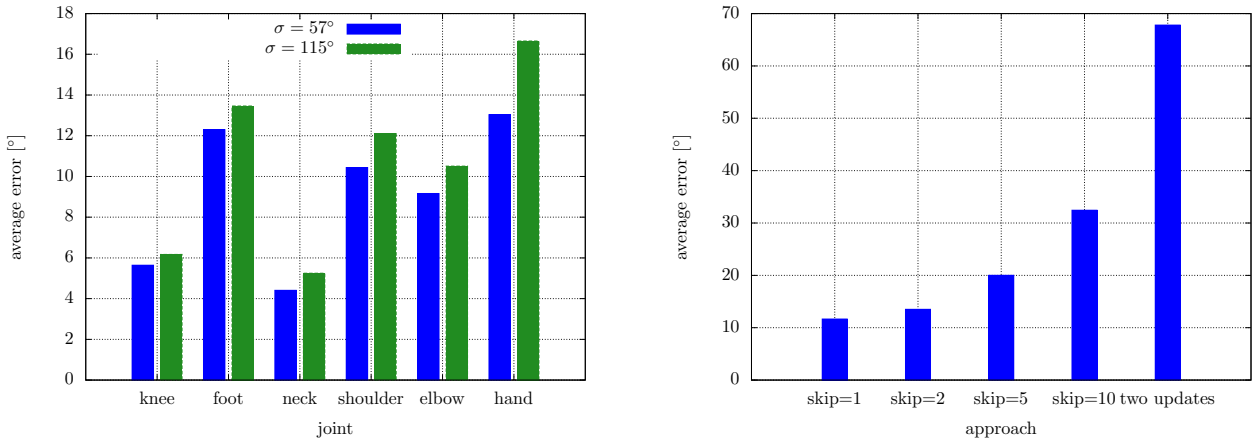
Figure 3: Ground truth and estimated trajectories of (a) the left foot and (b) the left hand with a measurement noise of $\sigma = 6°$

When we break down the error by several joints as depicted in figure 4a, we see that the motion of the legs is more easily followed than the motion of the arms. This indicates that the movement of the legs is more consistent among different swimmers than the movement of the arms and that accelerations are better estimated for steady, repeating motions within a cycle. The arms thus rely more strongly on the measurement update.

Importance of measurement frequency: In order to evaluate the quality of the motion model according to its ability to handle missing measurements, we conducted more experiments on the test set with the standard deviation of the measurement update set to $\sigma = 6°$. First, we let the model predict several consecutive frames without any update (the number of frames where we did not perform an update is denoted by *skip*), i.e. we provided the measurement only every *skip* number of frames, so that mispredictions by the model had greater impact. Second, we simulated very long phases of uncertainty, only providing two measurement updates per cycle when the arms of the swimmers pointed straight out of the water (denoted by *two updates*). The results of these experiments are shown in figure 4b.

The small error of 12.12° when skipping two frames hints that the model can cope with missing measurements, resulting from e.g. occlusion. However, as the experiments show, the error rises considerably the longer the phases get without any measurement. This stems from the fact that the training data itself is very generic, combining different swimmers, different scales and different locations, in order to cover many aspects of the learned motion.

Although this significantly increases applicability, it also demands that the model is updated regularly so that it can adapt to the specific motion of the tracked person. This is also reflected in the error when providing only two updates per cycle. Two frames clearly cannot provide the information required to infer all aspects and dynamics of the currently tracked motion.



(a) (b)

Figure 4: (a) Errors broken down by certain angles for very high measurement noise. (b) Error when skipping measurement updates for a varying number of frames. The measurement noise was set to $\sigma = 29°$

3.3 Particle filter results

We demonstrate that the model can be used with different Bayes filters by additionally using it within a particle filter framework where we conducted the same experiments as we have done with the Kalman filter. The particle filter represents the state distribution by a discrete set of samples $\mathcal{X}_t$. Each sample s is an instance of the state and is assigned a normalized weight $w_t^{[s]}$ that corresponds to the likelihood of its state. We used the expected pose of all samples to estimate the predicted pose in a frame, .i.e. $\Phi_t = \sum_s w_t^{[s]} \Phi_t^{[s]}$. However, since angles are periodic, we cannot simply add the weighted angles, because this would result in the arithmetic estimate, which is not what we want. Take for example two angles, $\phi^0 = 45°, \phi^1 = 315°$, the arithmetic average would yield $\phi'_{est} = 180°$, but the value we actually want is $\phi_{est} = 0°$. To overcome this, we calculate the Cartesian coordinates of each angle, compute the expected values in this space, and transform the coordinates back to angles. The model was initialized in the same way as the Kalman filter, i.e using ground truth angles of the initial frame and training data velocities.

Table 2 shows the error for a varying number of samples and a measurement noise of $\sigma = 29°$. For both test sets we see a clear decrease in the error with an increasing number of samples. A minimum at 4.86° and 4.77° is attained for 100,000 samples, which is the expected behavior of a particle filter: the higher the number of particles, the more accurate the approximation of the true distribution. A high number of samples will lead to more samples being in the vicinity of the correct pose and these samples will with a high probability be propagated into the next iteration, whereas less accurate samples are bit by bit eliminated. The high errors for 500 and 1000 samples indicate that the dimension of the state vector demands a higher number of samples, because the framework obviously suffered from particle deprivation. In line with the results we have seen with the Kalman filter, the particle filter also performs equally well on both test sets.

We also evaluated the particle filter performance with infrequent measurement updates. In other words, in frames where no measurement was provided we assumed a uniform likelihood, making each sample equally likely prior to resampling. The measurement noise was set to $\sigma = 29°$. Figure 5a depicts the results for 10,000 and 50,000 samples. We chose the number of samples according to the error the Kalman filter committed for the same measurement noise. The particle filter behaves similarly to the Kalman filter in that it copes with short

	500	1,000	5,000	10,000	50,000	100,000
test set 1	25.7307	23.7371	9.3002	9.6875	5.5931	4.8555
test set 2	25.1875	30.3826	11.3332	9.4183	5.5368	4.7666

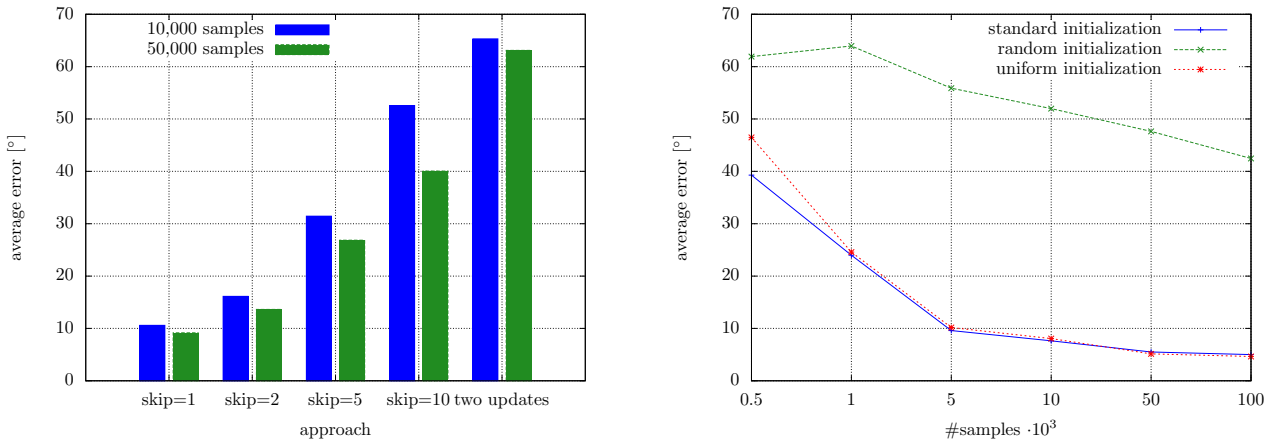
Table 2: Average absolute difference (in degrees) over all angles for a varying number of samples broken down by test sets.

periods of uncertainty, however, as these periods get longer, many samples start to drift away from the vicinity of the correct pose as the components of their states are not directly updated. This results in higher prediction errors.

Importance of model initialization: In order to test how much the model initialization influences the prediction quality of the particle filter we reran the experiments. First, we initialized the samples in a random fashion rather than using ground truth angles. We randomly generated values in $[-\pi, \pi]$ and initialized the angles and velocities accordingly. Second, we initialized each sample uniformly over the cycle position c_t. That is, we pretended to be right about the angles and velocities of the initial pose, but wrong about the position within the cycle. Our model should be able to concentrate the samples around the true pose after a few frames. This means that if we start computing the error after these frames, it should be comply with the error obtained when initializing with the true cycle position. We therefore began to compute the error after the fifth frame.

Figure 5b shows the committed error on the test set of the alternative initializations compared to the standard initialization, again, for a varying number of samples and $\sigma = 29°$. Clearly, when initializing the model randomly the framework fails to recover the poses. This was expected since a random initialization means that during the first frames the bulk of all samples will not be in the vicinity of the correct pose, and since the measurement update only selects significant samples rather than really updates their values, the true distribution cannot be approximated. However, initializing the samples uniformly over the cycle position yields almost the same errors compared to a standard initialization. This is because at least a fraction of all samples (the ones where the value of c_t is close to the index of the initializing frame) were initialized approximately correct, meaning that most of these samples get propagated.

This reveals that although an initialization with proper values is of great significance, it suffices that these values resemble plausible angles and velocities that actually occur with the modeled action.



(a) (b)

Figure 5: (a) Errors when skipping measurement updates for a varying number of frames with 10,000 and 50,000 samples. (b) Average error on the test set when using standard, random, and uniform initialization for a varying number of samples. The measurement noise was set to $\sigma = 29°$ for both experiments.

As the experiments showed, the model can easily be put into a Kalman or particle filter framework, and both implementations achieved low error rates. When using many samples (e.g. more than 50,000) the particle filter outperformed the Kalman filter, but at the cost of high computing times. It takes the Kalman filter only 0.39s to run the experiments consisting of 816 frames, which is approximately .0004s/frame, whereas the particle filter takes 129s (255s) which is approximately .16s/frame (.31s/frame) for 50,000 (100,000) samples. Execution times were measured on a 2.53GHz Intel i7 CPU with 16 parallel threads. We thus have to trade off accuracy against computational complexity: if we require real-time capable tracking, we have to put up with a loss in accuracy and should stick to a Kalman filter implementation. However, if we want to achieve better results and can handle longer execution times, we better use a particle filter with more samples.

4. CONCLUSION AND OUTLOOK

We have proposed an approach to model two–dimensional cyclic motion of humans solely based on simple kinematic properties. The accelerations of only a few sample cycles of the motion to be modeled serve as training data, and the means and covariances are used to propagate the state in time according to a plain state transition function. We describe the motion itself by the temporal change of angles between the segments of a standard stick figure model. The model computes the temporal prior distribution in an efficient way, so neither PCA, Fourier transform nor projection/back projection of the state vector to/from a lower dimensional space is required, while we still exploit the full dimension of the state. We thus do neither lose possibly important information nor do we have to use computationally expensive operations due to the simple function. Memory usage of the model is very low, since only one acceleration vector and one covariance matrix per frame of the reference cycle is stored. The conducted experiments within a Kalman and a particle filter framework showed that with only ten training cycles of the modeled motion a very low average error can be achieved, even when the update is applied infrequently and with high uncertainty about the value of the measurement. Furthermore we have seen that, when using a particle filter, we do not require a very accurate model initialization since the framework concentrates the samples around the true pose after few measurement updates. This relaxes the requirements for both, the appearance model and the model initialization significantly, because the model can cope with uncertainties and partial as well as complete occlusions as long as they don't occur over long periods.

The proposed two–dimensional model can easily be extended to three dimensions by describing the pose by two angles per joint and using three–dimensional motion data to train. This will be part of our future work. In addition, we will fuse the motion model with a newly developed appearance model and evaluate the results on publicly available datasets (like HumanEva[18]). This lets us compare the model to other popular approaches.

REFERENCES

[1] Moeslund, T. B., Hilton, A., and Krüger, V., "A survey of advances in vision-based human motion capture and analysis," *Computer Vision and Image Understanding* **104**(2-3), 90–126 (2006).

[2] Kalman, Rudolph, and Emil, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering* **82**(Series D), 35–45 (1960).

[3] Isard, M. and Blake, A., "CONDENSATION–Conditional density propagation for visual tracking," *International Journal of Computer Vision* **29**(1), 5–28 (1998).

[4] Gordon, N., Salmond, D., and Smith, A., "A novel approach to nonlinear/non-Gaussian bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F* **140**(2), 107–113 (1993).

[5] Sidenbladh, H., *Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences*, PhD thesis, KTH, Numerical Analysis and Computer Science (2001).

[6] Ormoneit, D., Black, M. J., Sidenbladh, H., and Hastie, T., "Learning and tracking cyclic human motion," *IN A. KENT & C.M. HALL (EDS.), ENCYCLOPEDIA OF LIBRARY AND INFORMATION SCIENCE, VOLUME 53 (SUPPLEMENT 16* , 16—27 (2001).

[7] Seitz, S. S. and Dyer, C. R., "Affine invariant detection of periodic motion," *IN PROC. IEEE CONF. COMPUTER VISION AND PATTERN RECOGNITION* , 970—975 (1994).

[8] Ran, Y., Weiss, I., Zheng, Q., and Davis, L., "Pedestrian detection via periodic motion analysis," *International Journal of Computer Vision* **71**, 143–160 (Feb. 2007).

[9] Rogez, G., Rihan, J., Ramalingam, S., Orrite, C., and Torr, P. H., "Randomized trees for human pose detection," in [*2008 IEEE Conference on Computer Vision and Pattern Recognition*], 1–8, IEEE (2008).

[10] Huazhong, N., Wei, X., Yihong, G., and Huang, T., "Discriminative learning of visual words for 3d human pose estimation," in [*2008 IEEE Conference on Computer Vision and Pattern Recognition*], 1–8, IEEE (2008).

[11] Thrun, S., Burgard, W., and Fox, D., [*Probabilistic Robotics*], MIT Press (2005).

[12] Deutscher, J., Blake, A., and Reid, I., "Articulated body motion capture by annealed particle filtering," in [*Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*], **2**, 126–133 vol.2 (2000).

[13] Rius, I., Varona, J., Gonzalez, J., and Villanueva, J. J., "Action spaces for efficient bayesian tracking of human motion," in [*Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*], **1** (2006).

[14] Agarwal, A. and Triggs, B., "Tracking articulated motion with piecewise learned dynamical models," in [*European Conf. Computer Vision*], (2004).

[15] Jähne, B., [*Practical Handbook on Image Processing for Scientific and Technical Applications*], CRC (2004).

[16] Greif, T., *A Probabilistic Motion Model for Swimmers: A Computer Vision Approach*, Master's thesis, University of Augsburg (2009).

[17] "Institut für Angewandte Trainingswissenschaft, Marschnerstr. 29, 04109 Leipzig."

[18] Sigal, L. and Black, M. J., "Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion," (2006).