Journal of Neuroscience Methods 135 (2004) 43–54

JOURNAL OF NEUROSCIENCE METHODS

www.elsevier.com/locate/jneumeth

A posture optimization algorithm for model-based motion capture of movement sequences

Jure Zakotnik [a,*], Tom Matheson [b], Volker Dürr [a]

[a] *Department of Biological Cybernetics, University of Bielefeld, P.O. Box 10 01 31, Bielefeld 33501, Germany*
[b] *Department of Zoology, University of Cambridge, Downing Street, Cambridge CB2 3EJ, UK*

Received 21 July 2003; received in revised form 24 November 2003; accepted 28 November 2003

Abstract

We have developed and evaluated a new optical motion capture approach that is suitable for a wide range of studies in neuroethology and motor control. Based on the stochastic search algorithm of *Simulated Annealing* (SA), it utilizes a kinematic body model that includes joint angle constraints to reconstruct posture from an arbitrary number of views. Rather than tracking marker trajectories in time, the algorithm minimizes an error function that compares predicted model projections to the recorded views. Thus, each video-frame is analyzed independently from other frames, enabling the system to recover from incorrectly analyzed postures. The system works with standard computer and video equipment. Its accuracy is evaluated using videos of animated locust leg movements, recorded by two orthogonal views. The resulting joint angle RMS errors range between 0.7° and 4.9°, limited by the pixel resolution of the digital video. 3D-movement reconstruction is possible even from a single view. In a real experimental application, stick insect walking sequences are analyzed with leg joint angle deviations between 0.5° and 3.0°. This robust and accurate performance is reached in spite of marker fusions and occlusions, simply by exploiting the natural contraints imposed by a kinematic chain and a known experimental setup.

Keywords: Locomotion; Motion estimation; Posture reconstruction; Limb movement; Stick insect; Monte Carlo method; Simulated Annealing

1. Introduction

Automated 3D analysis of movements (*motion capture*) has become an important method for studies in biomechanics, motor control and neuroethology. The study of limb movements in particular requires automated and reliable aquisition of large datasets to cope with variability of movements within a broad natural action range and to study their context-dependent control. Insects such as locusts and stick insects are prominent model systems for the study of motor physiology (Burrows, 1996; Bässler, 1983), and recent experiments have emphasized the need to analyze the variability of insect limb movements in a range of situations (Dürr and Matheson, 2003).

Commercial video-based optical motion capture systems (e.g. *Vicon Motion Systems*, *Peak Performance Technologies Inc.*) use retroreflective markers placed on the analyzed body. Typically, marker positions are recorded from multiple views, which enables the system to reconstruct their 3D-trajectories. Beginning with an initial marker assignment, joint positions and angles can subsequently be calculated for each time frame from a set of identified markers. However, optical motion capture systems have to deal with a number of problems:

- Markers can disappear in a view, for example when they are occluded by the segment on which they are placed, by other body parts, or when they are rotated parallel to the light source. They can also occlude each other, which appears as a marker fusion in the video.
- Ghost markers appear at positions where the experimental setup or the body surface reflects the light, e.g. the shiny cuticle of some insects.
- Simple marker tracking algorithms depend on the time resolution of the trajectories, because they use tracking techniques to reconstruct the initial marker assignment in subsequent frames. This approach is inapplicable for fast movements when relying on common video equipment with a time resolution of 50 Hz.

* Corresponding author. Tel.: +49-521-106-5519.
E-mail address: jure.zakotnik@uni-bielefeld.de (J. Zakotnik).

0165-0270/$ – see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.jneumeth.2003.11.013

Incorrect marker identifications require time-consuming manual reassignments in individual video frames, so the robustness of the system against the afore-mentioned problems is of great importance for usability. Motion capture systems can improve their robustness by using high speed cameras and several views of the scene, but this increases cost and complexity. Here, we present an approach that achieves very good performance while using only standard laboratory equipment.

In the simplest case of optical motion capture, marker positions are tracked in 2D image space and their 3D positions are reconstructed by triangulation algorithms (Blackman and Popoli, 1999; Faugeras and Robert, 1994; Chen et al., 1994). To improve robustness, some human motion analysis systems use kinematic body models as well as temporal movement models (Aggarwal and Cai, 1999; Gavrila, 1996). *Kinematic models* impose position constraints on markers, because of constant segment lengths and joint angle limits (DiFranco et al., 2001). A human kinematic model is used by Herda et al. (2001) to verify reconstructed marker trajectories (performed by stereo triangulation) and to predict marker occlusions. Marker trajectory identification is also checked by a skeleton model in (Lopatenok and Kudrjashov, 2002) by application of rules for plausible joint positions. The latter two use the skeleton only as a validation technique for reconstructed marker trajectories. Eian and Poppele (2002) use a kinematic model and camera dilation formulas to infer joint angles. This is done even from a single view, but marker occlusions are not dealt with and it was tested on very constrained postures.

To further constrain plausible marker movements, the kinematic model can also be used as part of a more general state space model including *time dynamics* of the movement. O'Rourke and Badler (1980) describe a cyclic scheme consisting of four steps: prediction of state, synthesis, image analysis and state estimation. State space filters like the Extended Kalman Filter compare reprojections of the predicted model state to detected image features like marker positions. State variables usually include joint angles and velocities (Cerveri et al., 2003; Liu et al., 1999; Nickels and Hutchinson, 2001; Ringer and Lasenby, 2000). Hidden Markov Models model the dynamics of movement (Karaulova et al., 2000). A posteriori constraints for smoothing of angular time courses are applied by DiFranco et al. (2001). However, the prediction step of these systems is only feasable, if the movement is sampled at a sufficient frame rate. In many experimental situations, it would be preferable to use standard video and therefore a sampling rate of only 50 Hz, or even concatenated videos of independent movement sequences. Particularly, concatenation can be useful in behavioral experiments with many trial repetitions, because the video sequence does not need to be cut and no manual marker assignment is necessary for individual trials.

We present a new algorithm, which utilizes a constrained kinematic body model to allow 3D motion capture of PAL/NTSC avi-videos with only two views and single frame analysis. This is a stochastic algorithm that manipulates the posture of the model to minimize a distance measure between projections of the model and the recorded markers in every single frame. Less accurate 3D reconstruction is possible from a single 2D view, albeit with less accuracy.

In (Ohya and Kishino, 1994), a stochastic method is used to determine human posture utilizing genetic algorithms. This silhouette matching algorithm produces rather large errors in posture reconstruction. A stochastic error function minimization approach is also used by Rockwood and Winget (1997) to reconstruct 3D-models of objects from 2D photos in an engineering application, but without analysis of natural movements.

In contrast, our approach optimizes body posture by means of an error function, using markers on a kinematic chain. Thus, we use joint constraint information to limit the search space. The Simulated Annealing (SA) algorithm exploits these constraints to find efficiently the best match between the model and the camera views. The algorithm is implemented in a software package called *VideoTrack* and is shown to accurately reconstruct natural movements.

2. Motion capture as an optimization task

In a typical application in neuroethology or physiology of animal locomotion, retroreflective markers are used to label locations on an articulated body. Movement of the body is then video-recorded, and films are digitized and deinterlaced. To detect 2D marker coordinates d in each camera view, a number of image processing steps are applied to the resulting AVI-file (Fig. 1): first, the image can be filtered with standard image processing filters for threshold, erosion

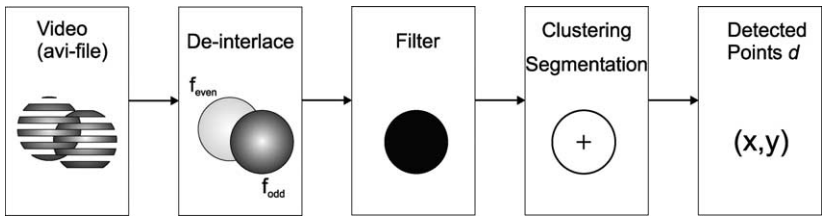

Fig. 1. Image processing steps to detect 2D marker positions with a sample marker image at each step. Video frames are de-interlaced into even and odd half-frames (f_{even}, f_{odd}). Every de-interlaced video frame is filtered (e.g. thresholded) into a binary image. From this, marker pixels are clustered and separated from the background into marker regions. Centroids of the regions determine a set of marker positions d.

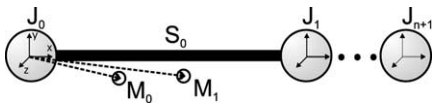

Fig. 2. A kinematic chain with joints J with their rotational axes, segments S and markers M. Each joint J contains a local coordinate system (indicated in the circles). Subsequent joints are connected by segments. Joint J_{n+1} defines the end-effector of the chain. The location of markers on the chain is defined by cartesian coordinates within the corresponding coordinate system. Note that markers are typically located on the surface of a limb, whereas segments denote the main axis connecting two joints in the model.

and color as described by Gonzalez and Wintz (1991). Then the filtered image is segmented, so marker positions are determined by the centers of the resulting regions.

2.1. Kinematic model

The articulated body is described as a set of kinematic chains (Fig. 2). Every chain consists of rigid segments S, joints J and markers M. Segments are defined by a constant length (translation T) and a default rotation R^s with respect to the originating joint to which they are connected. An arbitrary number of markers can be positioned on each segment with a 3D translational vector.

Every joint coordinate system is defined by three rotational degrees of freedom (DOF) expressed in a rotational matrix R, a constant default orientation R^{d} of the joint on the originating segment and a translation T. R^{d} simplifies the definition of joint constraints, because the resting orientation of a joint can be reproduced in the model. Direct kinematics for a given kinematic chain are obtained through multiplication of homogenous transformation matrices (Eq. (1)), starting from a root joint J_0, followed by segments S_n and joints J_n and finally terminated by an end-effector J_{n+1}.

$$J_0 = R_0 \cdot R_0^{\mathrm{d}}; \quad J_{n+1} = J_n \cdot S_n \cdot R_{n+1} \cdot R_{n+1}^{\mathrm{d}} \cdot T_{n+1} \qquad (1)$$

$$S_n = T_n^s \cdot R_n^s$$

where $R = R_y \cdot R_x \cdot R_z$ is the homogenous rotation matrix defining the rotation of the joint coordinate system by rotation according to Euler angles; and T is the homogenous translation matrix.

In the simplest case, the origin of the root (J_0) is constant, which means that its position is not changed by manipulation of the posture. Alternatively, it is marked by a root marker, which is tracked by a next-neighbor algorithm in all viewplanes and determines an offset-position for each frame.

For direct kinematics of each one of d marker positions, the matrix M_d, which describes the transformation of the nth marker M_n^s on segment s into the root coordinate system, is calculated according to Eq. (2).

$$M_d = J_s \cdot S_s \cdot M_n^s \qquad (2)$$

Projection onto the camera viewplane is determined by multiplication with a projection matrix of the appropriate view. For example, an orthogonal top projection of a 3D homogenous marker translation m onto a 2D homogenous vector $p = (x_p, y_p, 1)^{\mathrm{T}}$ is given in Eq. (3) with scale factor s.

$$\begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix} = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot m \qquad (3)$$

The exact projection matrix values depend on the camera parameters and are derived from camera calibration methods as proposed by Zhang (1999) and implemented in the Matlab camera calibration toolbox.

2.2. Joint angle constraints

Although all joints of the kinematic model have three DOF and can rotate arbitrarily, additional constraints need to be specified in most cases. They can be categorized into two constraint types. *Physiological constraints* determine the range of angles that can be actively controlled by a specific body. For example, many insect leg joints are typically modelled as hinge joints (Cruse and Bartling, 1995), which means that two rotational axes are locked. *Movement constraints* describe the range of angles that the articulated body actually uses for a particular type of movement. For example, walking behavior of an insect typically consists of a sequence of swing and stance movements with limited angular ranges. In general, neither type of constraint can be determined exactly. The constraint mechanism should either be able to update the angle constraints based on the analyzed movements or allow manual control by an expert.

In our implementation, joints are characterized by discrete angle probability distributions H for each rotation axis i in 360 bins. They are initialized with an uniform distribution across the physiologically plausible angular range, and zero values in the remaining bins. For each measured joint angle ξ, the value of a distribution bin x can be updated and normalized as in Eq. (4), where N is the number of trials for the updated distribution.

$$H_{i,N}(x) = \frac{(N-1)H_{i,N-1}(x) + \Delta}{N}; \qquad \Delta = \begin{cases} 1, & \xi \in x \\ 0, & \text{else} \end{cases} \qquad (4)$$

2.3. Optimization by means of an error function

For each model posture, an error E is calculated as a similarity measure between model and recorded markers. Therefore E is the sum of all Euclidean distances between projected model markers p and their nearest detected points d in $v_{\max}$ views (Eq. (5), see Fig. 3 for scheme).

$$E = \sum_{k=1}^{v_{\max}} \sum_{i=1}^{m_{\max}} \min_c (\| p_i^k - d_c^k \|) \qquad (5)$$

where $0 \le c \le C$ and C is the number of detected points. C does not necessarily equal $m_{\max}$ (i.e. in the case of ghost

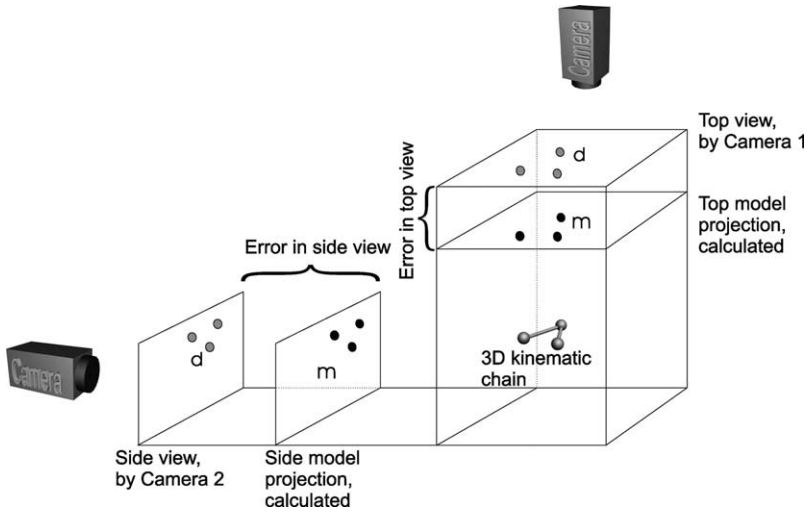

Fig. 3. Schematic view of error calculation. A 3D kinematic chain is projected onto a side and a top view (points m) and compared to the recorded marker positions d in the video. In the experiments, a mirror was used instead of a second camera.

markers). Wrong matching of model markers and detected points is discussed in Section 4.

Note that E is non-linear, because marker projections p_i are determined by forward kinematics. Therefore gradient-descent algorithms would often converge into local minima. Also in general the global minimum of E is non-zero due to inaccuracies in the camera setup, image processing and segment length measurements.

A suitable optimization algorithm is the *Simulated Annealing* approach, developed by Kirkpatrick et al. (1983). It is a Monte Carlo method that iteratively traverses the parameter space in a stochastic way. In every iteration n, a randomized vector v is added to the parameter vector j, which contains the angles for all rotational axes for all joints. Vector v is calculated from equally distributed random numbers $r_i \in [-1; 1]$ and the standard deviation of the appropriate joint axis distribution $s(H_i)$. Additionally, v is scaled with a value l, representing the search step length, which is equal for all joints (see Eq. (6)).

$$j_{n+1} = j_n + l \cdot v \quad \text{with} \quad v_i = s(H_i) \cdot r_i \tag{6}$$

The standard deviation $s(H)$ is larger for more variable joint axes and therefore generates larger search steps. It can also change over time, because the probability distributions are adapted during the analysis.

Additionally to Eq. (6), v_i is computed again, if $H_i(j_{n+1}) < T_h$, where T_h is a threshold describing plausible joint angles. The decision whether j_{n+1} is accepted as the new parameter vector is determined by the Metropolis criterion (Kirkpatrick et al., 1983). It is based on the difference between errors E_{n+1}, E_n and a parameter τ called temperature. If j_{n+1} is rejected, it is reset to j_n. The algorithm terminates after N_{end} iterations or if $E < \varepsilon$, where ε is an error residual set by the experimenter. For a review about the SA-algorithm and its properties, see (Aarts and Korst, 1989).

The temperature τ_i at iteration i is multiplied by $c_\tau \in [0; 1]$ each N_τ iterations and therefore decreases exponentially. The temperature annealing value c_τ must be chosen carefully, because it controls the probability of escaping local minima. Similarly, the search step length l_i is multiplied by $c_l \in [0; 1]$ each N_l iterations. It determines, how quickly the search space is reduced.

3. Evaluation results

The presented algorithm was implemented as a software application on a standard PC (Section 3.1). Its accuracy and robustness were evaluated both on artificially generated videos with known parameters (Section 3.2) and in an experimental situation with real insects (Section 3.3).

3.1. Implementation

The presented algorithm has been implemented in *Visual C++* for *MS Windows* and was tested on a PC equipped with an *Intel Pentium 4*, 1.8 GHz processor. The program *VideoTrack* features a graphic interface and loads AVI-files in DV-video format (720 × 576 pixels), with one or two views of the scene in one video. These can be obtained by using a video splitter or a mirror. A rectangular region of interest can be set, as well as the combination of filter modules for image processing. Filter parameters can be adjusted to allow extraction of as many markers as possible while suppressing ghost marker regions. Fig. 4 shows a screenshot of the program.

An appropriate kinematic chain and corresponding joint constraints are loaded from XML-files and visualized in an OpenGL-view using a scenegraph library (http://www.openscenegraph.org). The kinematic chain can be manipulated in a dialog with immediate display of its

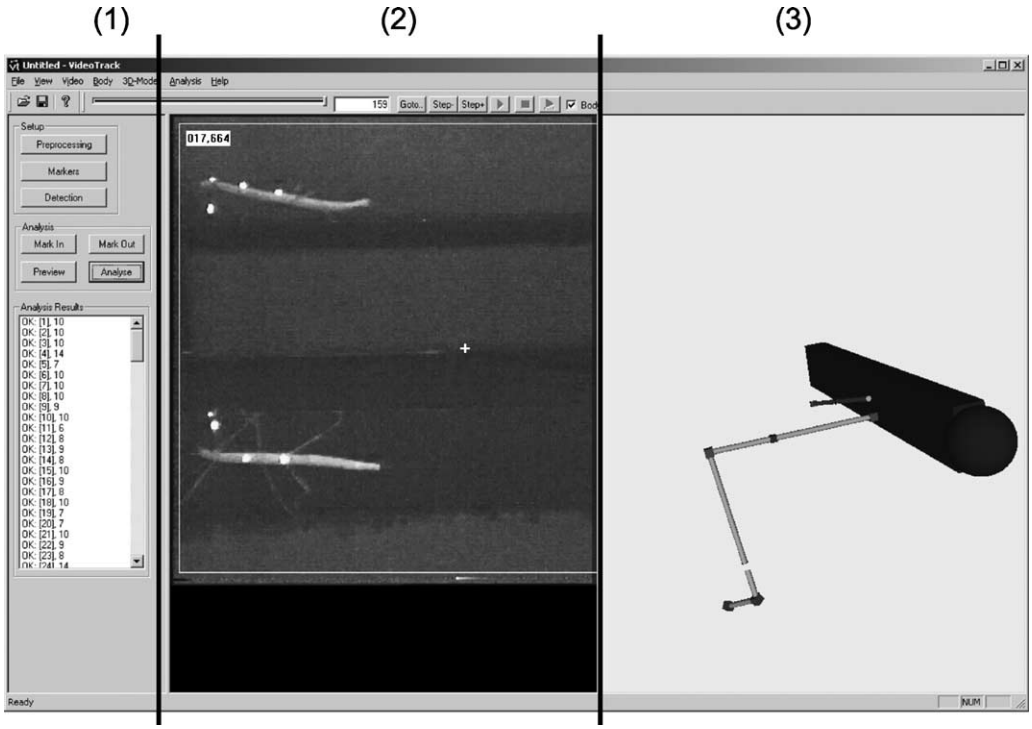

Fig. 4. Screenshot of the software *VideoTrack*. The main window is divided into three parts: The left part (1) contains a control panel and a table with error function values for each frame. In the middle (2) is a video view, which shows the current video file and marker projections. The top portion illustrates the side view of a stick insect marked with reflective discs. The lower portion shows the corresponding top view. The analyzed posture for the selected frame is shown in an OpenGL-visualization on the right side (3).

projection on the video-image. The starting posture in the first frame is calculated from mean values of joint angle distributions.

Parameters for the SA were set manually according to Table 1. As the maximum number of iterations determines the speed of the algorithm, this parameter was limited to a value that permitted sufficient convergence and reasonable rate of progress. The cooling schedule, which determines how temperature values are decreased, was set according to general rules given by Sait and Youssef (1999, pp. 66–73).

3.2. Accuracy: analysis of virtual locust leg movement

An estimation of the posture reconstruction accuracy in real experiments has two requirements: First, all experimental parameters (such as joint angles, segment lengths and camera projection) must be known. Second, the analyzed movement should originate from empirical data, so that the algorithm can utilize natural joint characteristics during analysis. To determine the accuracy with which an experimental situation in insect motor physiology could be analyzed ideally, we implemented a 3D-model of a locust leg and rendered an artificial video. To satisfy the second requirement, 10 manually analyzed grooming movement sequences were concatenated to one video (668 frames, 13.4 s total) using kinematic data from Dürr and Matheson (2003).

The kinematic model that was used to generate the video also served for the analysis (Fig. 5): The body–coxa joint J_0 was described with three DOF, while the trochanter–femur

Table 1
SA parameters for evaluation experiments

Parameter	Value
N_{end}	35000
N_l	3500
N_τ	1500
τ_0	200
l	1
c_l	0.75
c_τ	0.75
ε	1

If the error for a frame was larger than an arbitrary value (here: 10 pixels), the frame was analyzed again, at most five times.

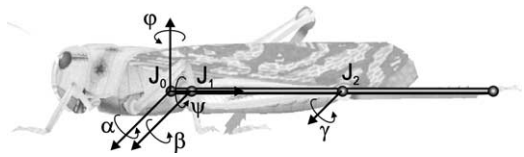

Fig. 5. Kinematic chain of a locust hind leg used for the accuracy evaluation. Body–coxa joint J_0 is the root-joint with three rotational axes and angular limits in brackets: $\varphi[120°; 220°]$, $\Psi[-40°; 40°]$ and $\alpha[-10°; 10°]$. It is connected to the trochanter–femur joint J_1 by the coxa, and the femur–tibia-joint J_2 by the femur. J_1 and J_2 are hinge joints with axes $\beta[-30°; 60°]$ and $\gamma[20°; 160°]$ respectively. Angles α, β and γ rotate the hind leg within a plane, i.e. the leg plane. Angles φ and ψ determine the orientation of this leg plane.

joint J_1 and the femur–tibia joint J_2 were considered as hinge joints. Hinge joints were locked in two of the three DOFs by setting a single angle bin of the joint histogram to 1. Markers were positioned at the joint centers in this experiment with a diameter of 4 pixels (1.4 mm).

Two orthographic views (side and top) were generated in one video, resulting in a leg length of 181 pixels (coxa: 13, femur: 91, tibia: 77) for each view. The analyzed leg uses a quarter of the PAL resolution and therefore resembles the resolution in the real experiment. The algorithm was evaluated in fully automatic mode with no manual corrections of the detected posture.

The accuracy of the image processing was determined by comparison of the calculated marker projections and observed marker centroids in every video image. Image processing was performed at a speed of 8.4 frames per second (fps). Two errors were made during the image processing: first, the mapping of continuous marker positions to discrete coordinates led to a quantization error during the generation of the video. Second, the clustering algorithm calculated a center pixel for each marker, which may not be the exact center of the marker. In a typical 5 s sequence, this caused an image processing RMS error of 0.43 pixels. Therefore, image processing algorithms introduce only a small cartesian error to the subsequent posture optimization.

The error made during optimization can be partitioned into two classes: First, the remaining error (*view error*) of the optimization function (error E in Eq. (5)), which measures the distance between the model's projection and the recorded view. Second, the angular error (*posture error*) between the analyzed model posture and the real posture in all joints. Although both errors depend on each other, optimization of one of them does not necessarily lead to a small value of the other one. For example, occluded markers can produce a large view error, even when the object posture can be estimated well. On the other hand, if only a few markers are available, they can be approximated well in all views with multiple possible postures, increasing the posture error.

Having applied SA parameters of Table 1, the speed of fitting the body to the detected points was 0.4 fps. The average view error of 2.9 pixels (S.D.: 1.1 pixels) total was added up for eight detected markers (four markers in each of two views).

A sensitivity analysis was performed to ensure that chosen SA parameters were sufficient for the task. N_{end} could be reduced to 20,000 (with linear scaling of N_l and N_τ), increasing the average view error by 5% of the previous value. The average result did not improve by further increase of the iteration number. Variation of c_l and c_τ in the range of [0.41; 0.79] kept the view error within 5% of the old value.

Resulting angular RMS errors were low and ranged from 0.7° to 4.9°, depending on the joint. Original and analyzed angle time courses for a typical sequence are plotted in Fig. 6. α-angles and β-angles systematically deviated from the real angles (RMSE: 4.7°, 4.9°). This effect was due to the short coxa segment between the parallel α- and β-axes. Numerical inaccuracies in the image processing, causing a deviation of a single pixel in detection of the marker on the coxa-femur joint J_1, changed the measured angle by $\text{atan}(1/13) = 4.4°$. Nevertheless the small overall error in $\alpha + \beta$ implied that the error in the α-joint was cancelled out by the error in the β-joint. Thus, calculation of the angles of the next distal joint remained accurate (γ-angle in Fig. 6).

In a few frames, the algorithm converged to a posture with a large view error, indicating that it converged to a local minimum. Because SA is not a deterministic algorithm, this problem was overcome by repeated analysis of affected frames with the same SA parameters, until an error threshold of 10 pixels was reached. This threshold was chosen, because it was clearly larger than the expected analysis result.

An example of a large posture error due to ambiguous postures despite small view error is shown in Fig. 6 for two frames (see arrows). Angles φ and ψ differ from the real angles, even though the view error is not increased. Therefore, there exist two postures, each of which minimize the error function with different φ- and ψ-angles. If these frames are analyzed repeatedly, the algorithm also converges to the correct posture. Possible improvements to eliminate ambiguous postures are discussed in Section 4.

For comparison, the algorithm was also tested using only a single view, same model and SA parameter set. Generally, accuracy is lower for rotations orthogonal to the view plane, e.g. φ and ψ in side view and α, β, γ in top view. That is because angles are implicitly inferred from single segment length projections (see Table 2 for detailed angle errors), which is less accurate than if several projections are measured. In case of α and β, however, the RMSEs are even lower in the single side view, because the less accurate top view (for these angles) does not have any influence on their reconstruction.

3.3. Robustness: leg movement analysis of freely walking stick insects

Having determined the accuracy of the algorithm in an ideal video, we were interested in its robustness and applicability to real experimental situations. We therefore tested the system on walking behavior of another model animal in motor physiology, the stick insect (*Carausius morosus*). A typical experimental situation causes many difficulties that a motion capture algorithm must deal with. For example, the

Table 2
RMSE for all analyzed joints for two views and single views for an artificial video (668 frames)

Rotation axis	φ (°)	ψ (°)	α (°)	β (°)	$\alpha + \beta$ (°)	γ (°)
Both views	1.7	1.5	4.7	4.9	0.7	0.7
Side view	20.8	17.4	3.9	3.6	3.1	3.5
Top view	14.7	10.4	23.7	29.6	23.4	52.4

Analysis accuracy is highest for two views or, if the rotational axes are orthogonal to the viewplane, sometimes for a single view.

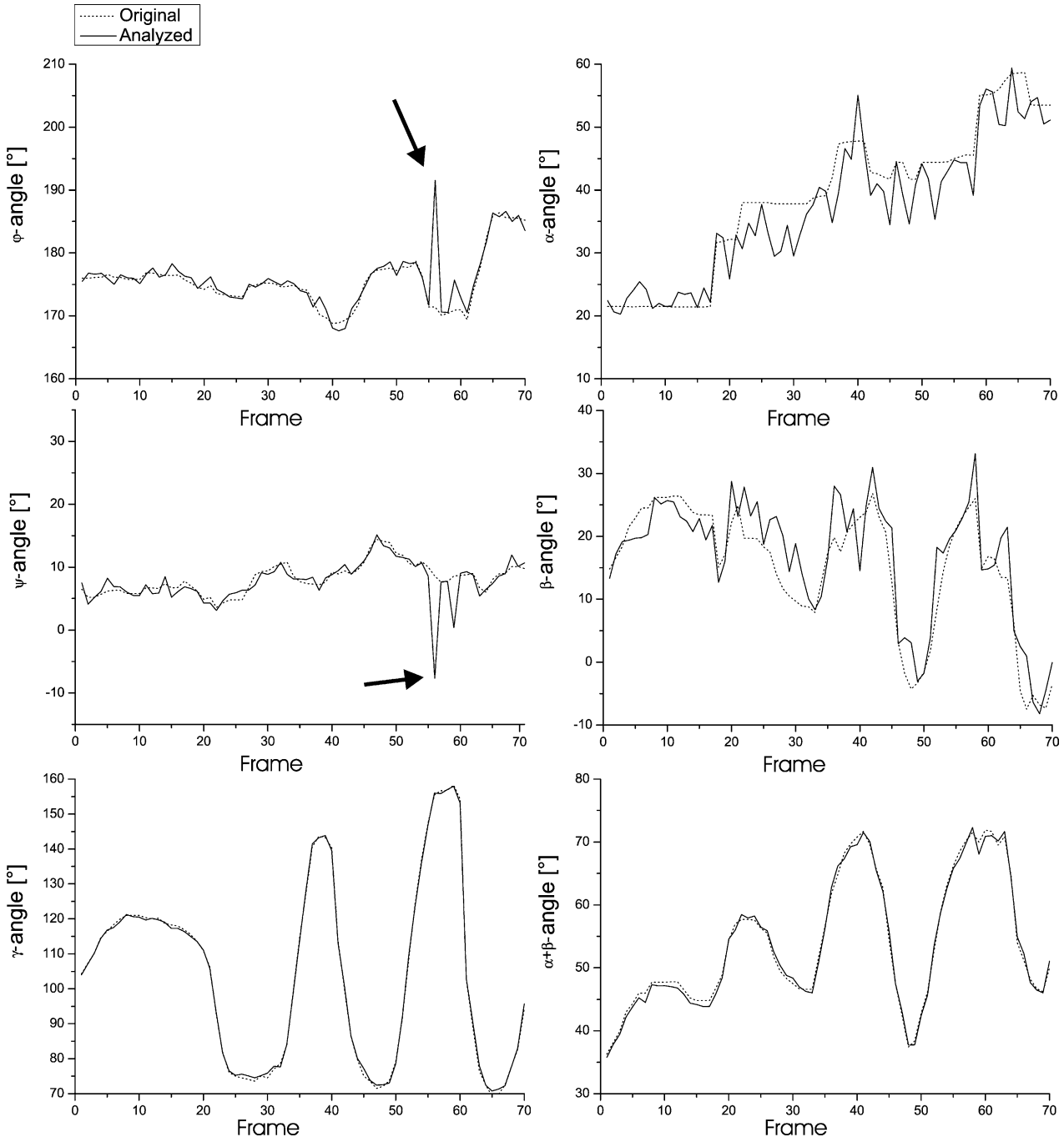

Fig. 6. Analyzed joint angles for a representative locust leg movement lasting 1.4 s (71 frames). The video was generated artificially to allow exact measurement of the algorithm's accuracy. Note the different scale for γ-angle. RMSE of the rotational axes are: φ-axis: 2.6°, ψ-axis: 2.3°, α-axis: 4.0°, β-axis: 4.5°, γ-axis: 0.7°. The analyzed α- and β-angles systematically differ from the original values due to a small segment length and the effect of image processing inaccuracies. Accordingly, their sum $\alpha + \beta$ has a RMSE value of 0.7°. Ambiguous postures (see arrows) are found for the φ- and ψ-angles in frames 56 and 58, where two possible postures minimize the error function.

segments of each leg are nearly orthogonal to one another during walking, resulting in marker occlusions and marker fusions in one or both views. In addition, inaccuracies such as inexact segment length measurements and kinematic simplifications limit the optimization of the posture. Finally, technical inaccuracies such as camera parameters influence the optimization process. Real videos of a complex movement therefore provide the best test for the robustness against inaccuracies of real experiments.

3.3.1. Setup

Animals were marked with reflective tape markers (3 M ScotchLite, diameter: 1.2 mm). To analyze the movement of the body axis and the right front leg, two markers were

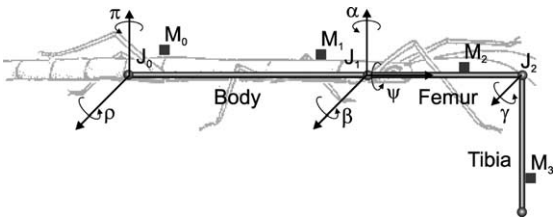

Fig. 7. Kinematic model of the right front leg of a stick insect, as used for the robustness measurements. Four markers M_0–M_3 were used and recorded by two camera views. The root of the kinematic chain is joint J_0 which determines the body axis orientation in space. Its rotation axes are $\pi[-35°; 35°]$ and $\rho[0°; 20°]$. It is connected to a thorax–coxa joint J_1 with three DOF $\alpha[150°; 240°]$, $\psi[-30°; 50°]$ and $\beta[-45°; 75°]$. The femur–tibia joint J_2 is modelled as a hinge joint with a single rotational axis $\gamma[10°; 140°]$. The segment between the thorax and the femur (the coxa) is short, allowing us to model the combined thorax–coxa and coxa–trochanter joints. The trochanter–femur joint is fused in stick insects, so it can be ignored.

placed on the body, one on the femur and one on the tibia. Marker positions and segment lengths were measured using a caliper gauge to obtain a body model for each animal. Fig. 7 shows the kinematic chain that models the stick insect front leg and body. It consists of rotational axes and joint angle constraints that were estimated manually and initialized with an equal distribution.

Walking animals were recorded on PAL videotape using a CCD-camera (COHU 4910) located above the setup. A side view was obtained from a coated mirror set at 45° and 150 mm from the animal. The setup was illuminated by an LED flash light consisting of a circle of 36 IR-LEDs (880 nm), mounted on a ring around the camera lens to achieve maximum brightness of the retro-reflective markers. The video synchronization signal was extracted electronically to trigger one flash per half-frame. The LEDs were flashed to maximize their power consumption capacity and therefore their brightness.

The camera was positioned 1300 mm above the animal, with a focal length of 3900 mm. No significant lens distortions were measured, so orthographic projection matrices were used (see Eq. (3), top: $s = 3.0$, side: $s = 2.86$).

Videos were digitized (miroVideo DC30, *Pinnacle Systems*), de-interlaced and processed by a threshold filter.

The starting offset of the root marker was determined manually for the top and side view in the first frame of each walking sequence. Parameters of the SA algorithm were set as shown in Table 1. We analyzed four videoclips (526 frames, 10.5 s total length) in 100 runs of the software. Repetitive analysis produced slightly different results in each trial, due to the stochastic nature of the algortihm. The mean deviation gives a measure for the precision of the algorithm.

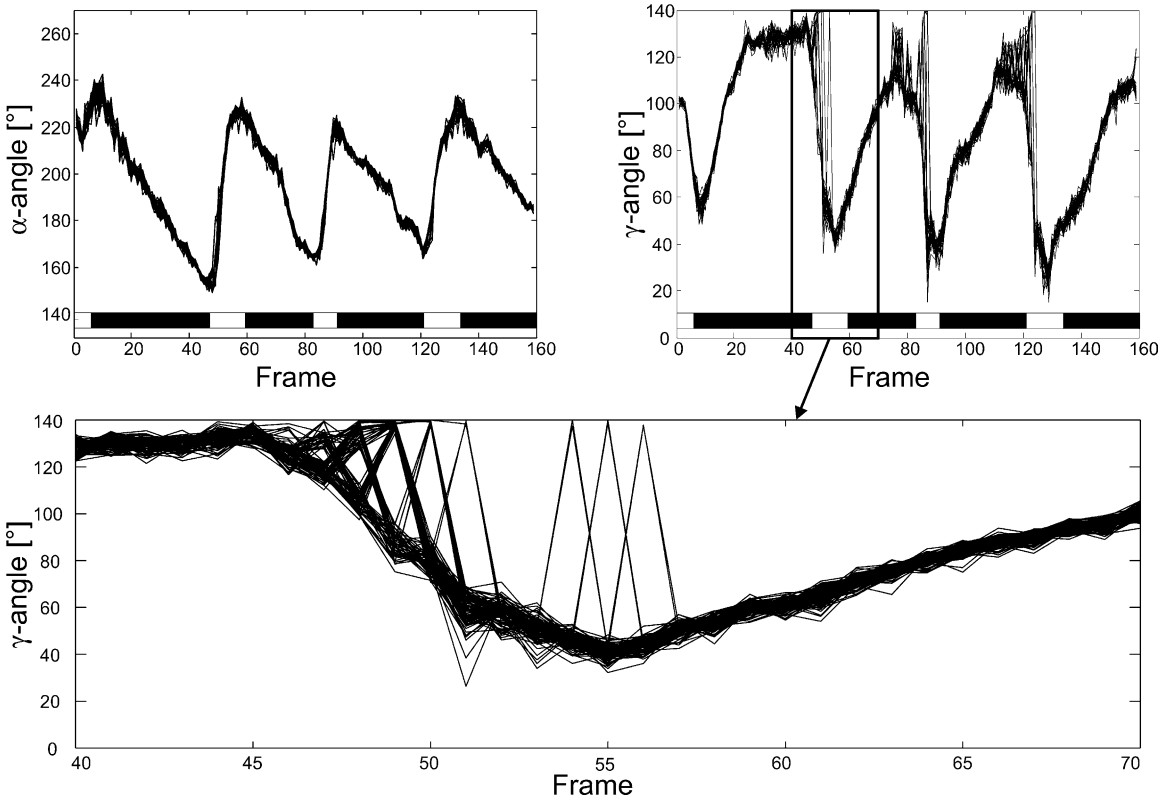

Fig. 8. Analyzed α and γ-angles for 50 analysis iterations of a single video (159 frames). Block bars indicate stance movements, open bars indicate swing phase. Results deviate because of multiple equally good local minima of the error function (mean error value of each analysis: 8.2 pixels). Although the α-angle can be reconstructed unambiguously (low deviation of 1.48° between runs), the γ-angle is more difficult to determine. The time inset shows that the analyzed angles for a particular frame do not deviate stochastically but rather contain few distinct values. These correspond to ambiguous postures as explained in Fig. 9.

Table 3
Median standard deviations for all joints after 100 analyses of four videos (526 frames total)

Rotation axis	Median of S.D. (°)	Joint angle range (°)	S.D./angle range (%)
π	0.50	70	0.72
ρ	0.91	20	4.54
α	1.71	90	1.90
ψ	3.04	80	3.80
β	2.54	120	2.11
γ	2.83	130	2.18

Because different joints are constrained by different angle ranges, we also indicate the percentage of their deviation with respect to their range.

3.3.2. Results

In contrast to the ideal setup used in Section 3.2, a number of uncertainties inevitably affected our analysis results. Of particular importance were those that caused a mismatch of the kinematic model with the real morphology, precluding an error function value of zero. Because model projections cannot match the video exactly, different postures may minimize the error function equally well. Although the minimum error did not become zero, convergence of the algorithm was optimal in the sense that it succesfully minimized the error function. The average view error was 8.1 pixels (S.D.: 3.4, $n = 52600$), accumulated for eight markers, indicating that the view error per marker is in the range of a single pixel. In this experiment, image processing speed was 6.6 fps and model fitting was performed at 0.4 fps.

Analyzed angles for all rotational axes reveal typical walking behavior, with stance and swing movements. Time courses for the α- and γ-angles for one video are displayed in Fig. 8, showing four stance and swing movements of a leg. The precision of the system is described by the median values of the S.D. for all rotational axes (see Table 3). Deviations range from 0.50° to 3.04° (0.7% to 4.5% joint angle range), while some sections of the movement are analyzed more precisely and others contain ambiguous postures. An example of a common ambiguous posture is shown in Fig. 9. Two distinct γ-angles both minimize the error function, so the algorithm stochastically converges into either one of them. Improvements to the algorithm to eliminate such ambiguities are discussed in Section 4.

4. Discussion

We demonstrate the power of a novel video-based motion capture algorithm that minimizes an error function by means of Simulated Annealing. The algorithm detects the current posture of an articulated body by matching projections of a kinematic model to a detected set of marker points. We have designed a system that minimizes time-consuming manual corrections of the data and provides sufficient accuracy for posture reconstruction, as tested for two model systems in motor physiology. It achieves high robustness even when using only two standard CCD camera views and 50 Hz frame rate.

Standard methods identify and track markers to determine their 3D positions and subsequently infer the body posture by kinematics calculations exploiting known marker identity, position and velocity (e.g. Allard et al. (1995)). Such methods become unreliable when analzsing fast limb movements, because the trajectories are sparsely sampled. In addition, marker occlusions, fusions and ghost markers hinder frame-by-frame identification of individual markers. For example, a marker fusion in one single frame may lead to incorrectly switched labels and therefore erroneous data in all subsequent frames. Although predictive algorithms like the Extended Kalman Filter adapt to the quality of their prediction, their history dependence makes them error prone in cases where the frequency of the observed movement is at a similar range as the sample frequency. Then, joint angle accelerations become too erratic to be predicted. In the experimental situations tested in the present study, 50 Hz recordings of fast locust kicks and stick insect swing movements represent such difficult situations, because extension/flexion cycles can alternate during only a few frames.

In contrast, our approach analyzes each frame independently from previous ones. Apart from optional tracking of a root-joint that defines the overall movement and the frame of reference, no explicit marker-labelling or tracking is performed. The camera time-resolution does not influence the quality of analysis. One consequence is that even concatenated videos of non-continuous movements can be analyzed, as is useful in cases in which several movement sequences

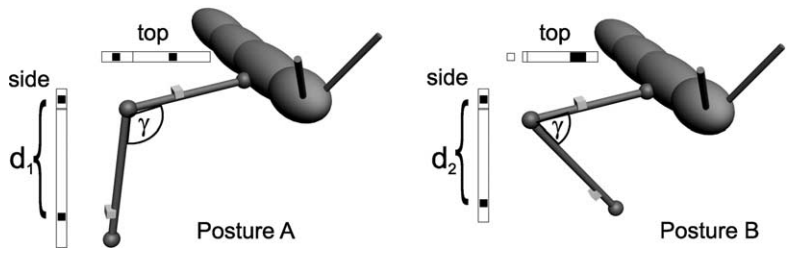

Fig. 9. Example of two ambiguous postures (postures A and B shown in the 3D model) from stick insect walking, in which the algorithm detects two minima of the error function. Both possible femur–tibia γ angles generate very similar side projections, i.e. marker distances d_1 and d_2 differ by a very small value, and therefore contribute little information to the posture reconstruction. The top projection can be interpreted in two ways: Either as the real posture (A) which matches all observed markers or as an erroneous posture B which fuses two markers and assumes that the remaining is a ghost marker (white square).

were recorded on one video tape at different times. Furthermore, no initial marker assignment is necessary, because the system optimizes the posture in the first frame. It also recovers automatically from badly or inaccurately analyzed postures.

On the other hand, the posture has to be determined unambiguously for each frame. For example, videos of stick insect movements often contained marker fusions, which result in several possible postures as shown in Fig. 9. To minimize the number of possible postures in each frame, the search space is constrained to valid postures by the use of joint angle histograms. In the terminology of Gleicher and Ferrier (2002), joint angle histograms are typical *character constraints*, because they describe limits for joint angles and, in addition, the probability of a specific movement type. In our approach, histograms are adapted during the analysis and saved in an XML data format, allowing standardized exchange of constraint archives and continuous improvement of subsequent experiments. The SA algorithm uses the histograms to limit the search space to likely values. Other data-mining methods such as principal components analysis could also be utilized to simplify the searched posture space.

Disambiguation of detected postures can be achieved by further constraining the model predictions to a maximum velocity or acceleration. For example, violation of such constraints can force large increments on the error function or trigger an automatic re-analysis of the frame. In this solution, the algorithm would not rely on particular joint angles from previous frames, nevertheless exploit a plausible motion continuity property. Another disambiguation heuristic would increase the error function, if not all markers in a videoframe are next-neighbors to a model projection. In both cases, SA avoids these situations automatically.

Designing the motion capture process as an optimization task reveals three interesting properties. First, analyses of artificial videos showed that SA is capable of solving the minimization problem, because the remaining error averages out to only a few pixels equalling the range of image processing errors. Second, using modern PCs, it is possible to analyze experimental data over a timescale where the experimenter can watch the progress. Moreover, the speed is easily adjustable by manipulations of the maximum number of iterations and/or the terminating error threshold. Although the system has not been designed for real-time usage, a real-time application seems possible by reduction of the maximum number of iterations with more inaccurate results. Using a kinematic model with a larger number of joints (e.g. all six legs of an insect, or a human body) and more markers is possible, but more complex calculations of forward kinematics would slow the algorithm. Third, the remaining error after optimization provides an explicit confidence rating for posture reconstruction, i.e. it allows the experimenter to immediately detect frames that need manual corrections or re-analysis. This is especially important, when the user uses the software in semi-automatic mode, in which he verifies the results of the analysis for each experiment.

Ghost markers generally do not affect the performance of the algorithm, because they do not change the minimum of the error function. They do not match constant segment lengths or angle constraints and therefore are not closest points to model marker positions. The latter property is particularly valuable in situations where the experimental setup does not permit control of the contrast between markers and background. This has been a significant problem in analyzing the movements of insects, which often have highly reflective surfaces (e.g. wings).

As an option for more accurate reconstruction, our algorithm easily scales to a variable number of camera views. Addition of additional cameras is simple as it requires only the measurement of their projection matrices and the incrementation of the parameter $v_{\max}$ in the error function(Eq. (5)). The method does not perform geometric reconstruction of 3D-positions of identified markers, like the widely used DLT-algorithm described by Chen et al. (1994). Rather, our approach yields posture data from manipulation of a forward model and does not require solution of the inverse kinematics problem. As a result, singularities are avoided, irrespective of the number of DOF and markers.

To generalize the algorithm to markerless motion estimation, the error function could compare the model to the video image on pixel-level. However, this would require a more accurate model of the analyzed body and good knowledge of the illumination of the experimental setup. Another possibility would be the extraction of suitable features in the video, which would require a higher level image processing step. For example, in MacIver and Nelson (2000), a 3D-mesh model of a knifefish without markers was fitted to match the video image of the animal.

Here, we evaluated the accuracy of the system with artificial videos in both single and double views. Overall angular RMSE was smallest, if both views were used, but postures could also be determined even from a single view. As publications on the accuracy of commercial systems use different setups and evaluation criteria, the quantitative comparison of achieved angular accuracy is limited. In Richards (1999), a typical angle RMS error of 3° was reported for the angle between three markers that were mounted on a rotating disc and filmed by six cameras. McQuade et al. (2000) evaluated the *Peak Performance System*, reporting an angular mean S.D. of 1.47° using three cameras. As listed in Table 2, joint angle RMS errors of the presented approach range from 0.7° to 4.9° using only two views. Therefore our algorithm performs extremely well while using a simpler and cheaper setup than other state of the art motion capture systems used in neuroethology.

Having measured the precision of the system by repeated analyses of natural stick insect movements, median standard deviation was well in the range of commercial systems (1.72° on average, see Table 3). For example, Selfe (1998) reported a mean S.D. of 5.72° for the Peak 5 system in real knee joint angle measurements, considering position uncertainties for the marker placement.

For comparison with manual analysis, Dürr and Matheson (2003) report manual digitizing accuracy of five pixels, using an experimental situation equivalent to the setup used in Section 3.2. This is more than ten times the view error of our accuracy analysis (Section 3.2) and approximately five times the view error of our robustness analysis (Section 3.3). Accordingly, the presented system produces considerably smaller errors than expected from a manual analysis, while the time effort is greatly reduced.

In general, calculated joint angles deviate from the real angles for two different reasons. First, inexact specification of the model, due to non-rigid segments or wobbly masses, together with inaccuracies of marker detection lead to a manifold of different postures, all of which approximate but do not match exactly the detected marker positions. They can be detected by a large remaining *view error* after optimization. Second, multiple postures are possible without any difference in the value of the error function (*posture error*). These do not deviate stochastically, but are distinct solutions in the search space as illustrated by the discontinuities in Fig. 8 and an example in Fig. 9. This type of error is strongly dependent on the number of camera views. Thus, it can certainly be improved by additional views, but possibly also by using a different configuration of markers. For example, the described ambiguity in the γ-angle could be removed by a front view of the insect. Optimal marker positioning will differ according to the observed movement type and could reduce marker occlusions and improve accuracy of the reconstruction.

As a last resort, the presented *VideoTrack* software also supports manual corrections of the analyzed postures in each frame via a graphical interface.

In conclusion, the presented motion capture system works with a low-cost and simple setup using standard equipment like a PC and CCD camera. It was succesfully applied for motion analysis of two model systems in motor physiology and neuroethology. The versatility of the method makes it suitable for analysis of a wide range of animal movements, including those of humans. Its accuracy rivals that of commercial systems that require considerably more experimental and financial effort.

Acknowledgements

Supported by the Graduiertenkolleg Strukturbildungsprozesse, University of Bielefeld. Part of this work was supported by research grants from the Isaac Newton Trust (Cambridge, UK) and BBSRC (UK) to Tom Matheson.

References

Aarts E, Korst J. Simulated Annealing and Boltzmann Machines: a stochastic approach to combinatorial optimization and neural computing. Chichester: John Wiley and Sons; 1989. p. 272.

Aggarwal JK, Cai Q. Human motion analysis: a review. Computer Vision and Image Understanding 1999;73(3):428–40.

Allard P, Stokes IA, Blanchi JP (Eds). Three-dimensional analysis of human movement. Champaign: Human Kinetics; 1995. p. 371.

Bässler U. Neural basis of elementary behavior in stick insects, Heidelberg: Springer; 1983. p. 169.

Blackman S, Popoli R. Design and Analysis of modern tracking systems. Norwood: Artech House; 1999. p. 1232.

Burrows M. The Neurobiology of an insect brain. Oxford: Oxford University Press; 1996. p. 682.

Cerveri P, Pedotti A, Ferrigno G. Robust recovery of human motion from video using kalman filters and virtual humans. Hum Mov Sci 2003;22(3):377–404.

Chen L, Armstrong CW, Raftopoulos DD. An investigation on the accuracy of three-dimensional space reconstruction using the direct linear transformation technique. J Biomech 1994;27(4):493–500.

Cruse H, Bartling C. Movement of joint angles in the legs of a walking insect, *Carausius morosus*. J Ins Physiol 1995;41:761–71.

DiFranco DE, Cham TJ, Rehg JM. Reconstruction of 3-d figure motion from 2-d correspondences. In: Computer vision and pattern recognition (CVPR01), vol. 1. Los Alamitos: IEEE Computer Society Press; 2001. p. 307–15.

Dürr V, Matheson T. Graded limb targeting in an insect is caused by the shift of a single movement pattern. J Neurophys 2003;90(3):1754–65.

Eian J, Poppele R. A single-camera method for three-dimensional video imaging. J Neurosci Methods 2002;120(1):65–83.

Faugeras O, Robert L. What can two images tell us about a third one? Int J Comput Vision 1994;18:5–19.

Gavrila DM. Vision-based 3-d tracking of humans in action. Ph.D. thesis, University of Maryland; 1996.

Gleicher M, Ferrier N. Evaluating video-based motion capture. In: Computer animation 2002 (CA02). Los Alamitos: IEEE Computer Society Press; 2002. p. 75–81.

Gonzalez RC, Wintz P. Digital image processing. 2nd ed. Reading: Addison-Wesley; 1991. p. 503.

Herda L, Fua P, Plaenkers R, Boulic R, Thalmann D. Using skeleton-based tracking to increase the reliability of optical motion capture. Hum Mov Sci 2001;20:313–41.

Karaulova IA, Hall P, Marshall A. A hierarchical model of dynamics for tracking people with a single video camera. In: Mirmehdi M, Thomas B, editors. Proceedings of the Eleventh British Machine Vision Conference (BMVC2000). Bristol: ILES Press; 2000. p. 352–61.

Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science 1983;220(4598):671–80.

Liu X, Zhuang Y, Pan Y. Video based human animation technique. In: Proceedings of the 7th ACM International Multimedia Conference. Orlando: ACM; 1999. p. 353–62.

Lopatenok T, Kudrjashov O. The model-based approach of markers identification and visualisation in motion capturing systems. In: Proceedings of Simulation und Visualisierung 2002, Magdeburg; 2002. p. 99–110.

MacIver M, Nelson M. Body modeling and model-based tracking for neuroethology. J Neurosci Methods 2000;95:133–43.

McQuade KJ, Parker J, Rodgers M. Comparison of electromagnetic tracking and video motion analysis using dynamic pendulum motion. In: Proceedings of the 24th Annual Meeting of the American Society of Biomechanics, Chicago; 2000, p. 52.

Nickels K, Hutchinson S. Model-based tracking of complex articulated objects. IEEE Trans Robot Automat 2001;17(1):28–36.

Ohya J, Kishino F. Human posture estimation from multiple images using genetic algorithm. In: Proceedings of the 12th International Conference on Pattern Recognition (ICPR94), vol. 1. Los Alamitos: IEEE Computer Society Press; 1994. p. 750–53.

O'Rourke J, Badler NI. Model-based image analysis of human motion using constraint propagation. IEEE Trans Pattern Recognit Machine Intelligence 1980;2(6):522–36.

Richards J. The measurement of human motion: a comparison of commercially available systems. Hum Mov Sci 1999;18:589–602.

Ringer M, Lasenby J. Modelling and tracking articulated motion from multiple camera views. In: Mirmehdi M, Thomas B, editors. Proceedings of the Eleventh British Machine Vision Conference (BMVC2000). Bristol: ILES Press; 2000. p. 172–81.

Rockwood A, Winget J. Three-dimensional object reconstruction from two-dimensional images. Computer-Aided Design 1997;29(4):279–85.

Sait SM, Youssef H. Iterative computer algorithms with applications in engineering. Los Alamitos: IEEE Computer Society Press; 1999. p. 410.

Selfe J. Validity and reliability of measurements taken by the Peak 5 motion analysis system. J Med Eng Technol 1998;22(5):220–5.

Zhang Z. Flexible camera calibration by viewing a plane from unknown orientations. In: Proceedings of the International Conference on Computer Vision, vol. 1. Los Alamitos: IEEE Computer Society Press; 1999. p. 666–73.