

Combining View-based and Model-based Tracking of Articulated Human Movements

Cristóbal Curio*

Martin A. Giese

Institute for Neuroinformatics
Theoretical Biology
Ruhr University Bochum
44801 Bochum, Germany

Laboratory for Action Representation and Learning
Dept. of Cognitive Neurology
University Clinic Tübingen
72072 Tübingen, Germany

Abstract

Many existing systems for human body tracking are based on dynamic model-based tracking that is driven by local image features. Alternatively, within a view-based approach, tracking of humans can be accomplished by the learning-based recognition of characteristic body postures which define the spatial positions of interesting points on the human body. Recognition of body postures can be based on simple image descriptors, like the moments of body silhouettes. We present a system that combines these two approaches within a common closed-loop architecture. Central characteristics of our system are: (1) Mapping of image features into a posture space with reduced dimensionality by learning one-to-many mappings from training data by a set of parallel SVM regressions. (2) Selection of the relevant regression hypotheses by a competitive particle filter that is defined over a low-dimensional hidden state space. (3) The recognized postures are used as priors to initialize and support classical model-based tracking using a flexible articulated 2D model that is driven by local image features using a vector field approach. We present pose tracking and reconstruction results based on a combination of view-based and model-based tracking. Increased robustness and improved generalization properties are achieved even for small amounts of training data.

1. Introduction

Action recognition in the brain is likely combining the recognition of body configurations with some form of feature tracking over time [12]. In computer vision these mechanisms have usually been addressed as two separate problems. Proposed solutions have been either based on the recognition of learned body configurations [3, 18, 13], or

on high-level model-based stochastic tracking. Models can be either predefined as 2D or 3D parameterized shape models [15, 17, 24, 6, 19], or shape models that are learned from training images [14, 5, 13, 25, 16].

Model-based tracking has the advantage that it exploits local visual cues since it is based on local image features. This makes it possible to realize relatively accurate tracking (e.g. [24, 7]). The disadvantage of this approach is that it usually requires a manual initialization of the model. Also if tracking is lost the model often cannot recover automatically, resulting in large tracking errors. Because of the relatively high dimensionality of the parameter spaces causing many local minima for the required parameter optimization a robust estimation of the relevant model parameters is often difficult.

View-based tracking methods have the advantage that they do not require an initialization process, and that they recover automatically after tracking gets lost. One disadvantage of this approach is that it typically exploits relatively unspecific global image features, like moments. Such features do not extract the precise local information from individual image frames. This limits precision of the tracking. A second problem is that this approach requires a sufficient number of training examples to guarantee the robust recognition of all relevant body postures. For a robust view-based recognition of moving humans, variations of the body kinematics and a sufficient number of views have to be covered by the available training examples. This leads to substantial storage requirements, and makes the recognition process computationally expensive. To solve this problem different solutions have been proposed. One possibility is to cluster the training examples in order to find a minimum set of critical postures [22, 10, 9]. Another approach is to use hierarchical classifiers for posture recognition that reduce the complexity of the search process [21]. We propose here a third solution. By exploiting model-based tracking that is based on local image features our system can reduce errors that result from suboptimal posture recognition. This makes

*Present address: Max Planck Institute for Biological Cybernetics, Spemannstrasse 38, 72076 Tübingen, Germany, Email: cristobal.curio@tuebingen.mpg.de

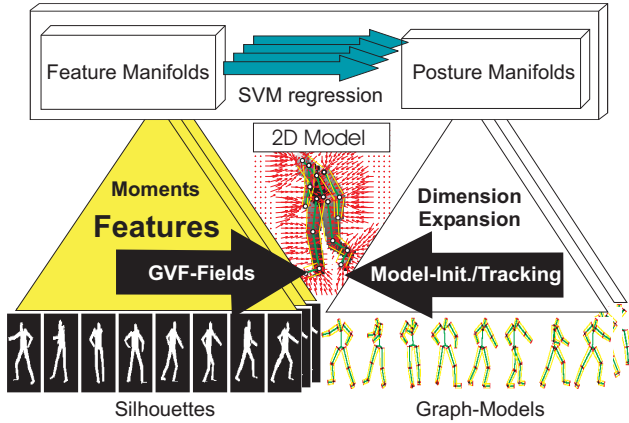


Figure 1: Overview of our system for articulated tracking that combines view- and model-based tracking.

it possible to reduce the number of stored training examples and increases the generalization capability of posture recognition to different body geometries.

The paper is structured as follows. Section 2 describes the system architecture and provides details about the individual computational steps. Section 3 illustrates the performance of our system and demonstrates the advantage of the integrated approach over purely view-based pose estimation.

2 System architecture

An overview of the architecture of our system is given in Figure 1. Posture recognition is based on image moments that are computed from the silhouette of the moving figure. In the space of image moments the articulated movement is represented by curves in a relatively low-dimensional *feature manifold*. Model-based tracking in our system is based on a flexible 2D patch model that is moving in a gradient vector field (cf. [26]) that is computed from contour features in the video sequence. The dimensionality of the posture space of the 2D model is reduced by applying a PCA on a set of training data. In the resulting reduced posture space movements are represented as curves in a *posture manifold* with low dimensionality. In order to support model-based tracking by posture recognition we learn the mapping between the feature manifold and the posture manifold using a support vector regression approach [23]. A key problem is that this mapping is not one-to-one since very different postures can be associated with similar silhouettes. For example the silhouettes for different view angles and different phases of the gait cycle can appear quite similar [8]. In addition, the silhouettes of side views of gaits with 180 deg phase difference are roughly identical because front and back limbs are difficult to distinguish by feature descriptions of the silhouettes. These ambiguities define a *one-to-*

many mapping between feature and posture manifold. For the selection of the most likely branches from this one-to-many mapping our system implements a *competitive* particle filtering approach that also ensures the temporal continuity of the sequence of proposed postures for the model-based tracking.

The system architecture consists of the following components described in the following sections (see capital letters in Figure 2):

- A : Transformation of image silhouettes into points in feature space (Section 2.1)
- B : Learning of a posture space with reduced dimensionality for the flexible 2D model by applying PCA to a set of training postures (Section 2.2)
- C : One-to-many SVM regression that maps curves in the feature space to curves in posture space (Section 2.3)
- D : Competitive particle filter dynamics over a hidden low-dimensional state space that selects appropriate branches from the one-to-many mapping, and enforces continuity of the proposed postures over time (Section 2.4)
- E : Computation of a likelihood by evaluating the overlap of the silhouette that corresponds to predicted model configurations and the silhouette in the input image

Model-based tracking is driven by force fields that are derived from contour features (Section 2.5). The tracking process is initialized with two different initial conditions:

- F1 : One initial condition is the proposed posture that corresponds to the most likely hypothesis from the dynamic view-based posture recognition
- F2 : The second initial condition is given by the best fitting model-configuration in the previous time step.

From the two tracking results obtained with the different initial conditions the one with maximum consistency with the present image silhouette is adopted as final estimate (Section 2.5.3).

2.1 Features for posture recognition

The silhouette of the articulating figure in each image frame is extracted by background subtraction and converted into a binary image. The silhouettes are parameterized by higher order image moments (cf. also [18]) resulting in a low-dimensional feature vector. Testing different types of moments we found that Alt moments [2] were most efficient for the characterization of posture in our data. From these moments we formed a low-dimensional feature vector of the form $\mathbf{z} = [z_1 \dots z_5]^T$ denoting the lower-order image moments. The higher-order moments turned out to be too sensitive to noise. The feature vectors define a low-dimensional feature space that is signified by \mathcal{R}^q in Figure 2 A.

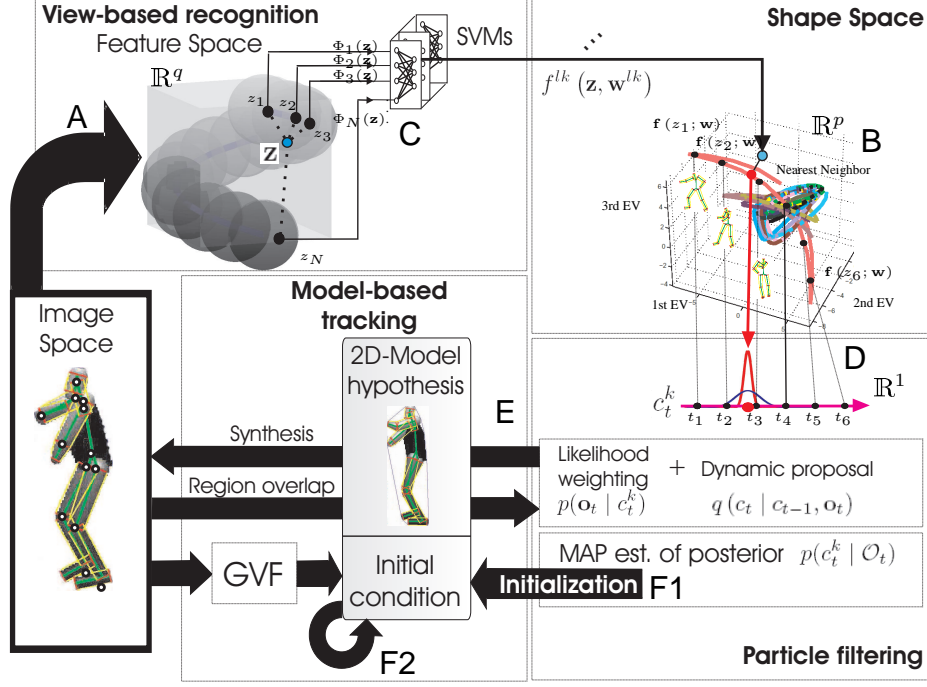


Figure 2: Detailed overview of the system architecture with the following components: **A**: From the image low-dimensional image moments are extracted. Movements correspond to curves in this feature space. **B**: Low-dimensional posture space derived from the 2D shape model by PCA. Movements correspond to curves in this space that are approximated by spline interpolation between training data points. **C**: One-to-many SVM regression to map curves in feature manifold to curves in posture manifold. **D**: Particle filtering based on importance sampling over hidden low-dimensional state space. Different values of the state variable c_t^k correspond to points on the posture manifold. **E**: Likelihood computation by synthesis of silhouettes that correspond to the present state c_t^k , and by determining their region overlap with the present image silhouette. **F1**: First initial condition for model-based tracking derived from most likely recognized posture based on steps **B-E**. **F2**: Second initial condition for model-based tracking derived from the best fitting model-configuration in the previous time step. Model-based tracking is driven by a gradient vector field (GVF). From the estimates generated from the two initial conditions (F1 and F2) the one with maximum region overlap with the present image silhouette is selected for further tracking.

2.2 Reduced posture space

Our 2D model consists of 12 trapezoidal connected 2D patches that are parameterized by the endpoints of the lines that define their main symmetry axes, and by the widths of the patches at the short edges (Figure 2). The model has 68 degrees of freedom. In order to reduce the high dimensionality of the configuration space of the model we applied PCA over a set of training data that was generated by animating an avatar with motion capture data (s.b.). Retaining only 12 principle components we could account for 98% of the variance of the corner points of the patches. These principle components define a reduced posture space (signified by \mathcal{R}^p in Figure 2 B). The dimensions of this space were rescaled by the corresponding eigen values in order to obtain dimensions with comparable variance.

2.3 One-to-many Support Vector Regression

The mapping between silhouettes and postures is non-unique since one silhouette can correspond to multiple possible body postures. However, this mapping is smooth and can be modeled by a set of smooth functions that map one point in image feature space onto multiple, e.g. K , points in reduced posture space. This mapping has multiple branches, each of which can be modeled by a smooth function. We learned these branches of functions using support vector regression (SVR) with Gaussian kernels [23] and similar to the approach employed in [1].

Let \mathbf{z} define one point in image feature space and \mathbf{y}_k , $1 \leq k \leq K$, the k -th corresponding point in reduced posture space. The mapping for the l -th component of \mathbf{y}_k is defined by

$$f^{lk}(\mathbf{z}, \mathbf{w}^{lk}) = \sum_{i=1}^M w_i^{lk} \Phi_i(\mathbf{z}) + b^{lk} \quad (1)$$

with \mathbf{w}^{lk} denoting the trained weight parameters and b^{lk} constant offset parameters. For all SVMs we use an ϵ -insensitive L_1 -loss function. $\Phi_i(\mathbf{z}) = \exp(-\|\mathbf{z} - \mathbf{z}_i\|/\sigma^2)$ is the Gaussian kernel function centered at data point \mathbf{z}_i .

The training patterns for the individual support vector regressions were derived by simulating a full walking cycle using our 2D model animated with motion capture data and computing the corresponding sequence of silhouettes for one actor. To generate training data points the movement was sampled equidistantly over time. In posture space one walking cycle corresponds to a closed curve. In image feature space a walking cycle can correspond to a curve that is run through multiple times, or that crosses itself. Each of these runs defines a separate set of training data that can be used to learn one branch of the one-to-many mapping. In this way, we define for each component function f_k one training data set $\{\mathbf{z}_i, \mathbf{y}_i^k\}_{1 \leq i \leq M_k}$. It turned out that two branches of the one-to-many mapping were sufficient for an accurate approximation of our data in most cases, so that we chose $K = 2$. This result is corroborated by results on visual manifold learning from image silhouettes [8]. Our approach is related to the function clustering approach in [18]. Contrasting with this approach we exploit the fact that movements are represented by smooth trajectories on the posture manifold. In this sense, our approach specifically also exploits the temporal order of different postures.

2.4 Dynamic propagation of information in posture space and hypothesis selection

In order to select the most likely posture over time, and over different views we apply a *competitive particle filter* algorithm (similar to CONDENSATION [14]). We define for each curve in posture space a corresponding one-dimensional continuous hidden *configuration (state) variable* c^k that is defined on the interval $[0, 1]$ (Figure 2, D). Each point of the curve in reduced posture space corresponds to a specific value of this configuration variable, e.g. the start of the curve to 0 and the end point of the curve to 1. Different views and branches k are represented by separate configuration variables.

Our probabilistic algorithm infers probability distributions over the set of configuration variables. For this purpose, we distribute a common set of particles over the space of all configuration variables. By joint normalization of the particle distribution a competition between the different hypotheses that are defined over different views and by the different branches of the one-to-many mapping is induced.

Probabilistic dynamics: By applying Bayes theorem the dynamic posterior over all configuration variables c^k at time t reads

$$p(c_t^k | \mathcal{O}_t) \propto p(\mathbf{o}_t | c_t^k) \int p(c_t^k | c_{t-1}^k) p(c_{t-1}^k | \mathcal{O}_{t-1}) dc_{t-1}^k$$

based on previous observations \mathcal{O}_{t-1} and the current observation \mathbf{o}_t . The transition dynamics that determines $p(c_t^k | c_{t-1}^k)$ is linear first-order, and its propagation velocity is sequentially adapted.

The evolution of this distribution is approximated by a particle filter with importance sampling. The importance weights of the particles are given by

$$w_t(c_t^{(i)}) \propto p(\mathbf{o}_t | c_t^{(i)}) \frac{p(c_t^{(i)})}{q(c_t^{(i)} | \mathbf{o}_t)},$$

where the particles $c_t^{(i)}$ are drawn from the proposal distribution q (see below) and weights $w_t(c_t^{(i)})$ normalized.

To propagate the particles $c_t^{(i)}$ and $w_t(c_t^{(i)})$ through time we use a standard recursive weight update scheme that is given by

$$w_t^i \propto \frac{p(\mathbf{o}_t | c_t^i) p(c_t^i | c_{t-1}^i)}{q(c_t^i | c_{t-1}^i, \mathbf{o}_t)}.$$

The proposal density was designed as the mixture of two distributions as follows

$$q(c_t | c_{t-1}, \mathbf{o}_t) = r q_{SV}(c_t | c_{t-1}, \mathbf{o}_t) + (1 - r) p(c_t | c_{t-1}).$$

The first distribution $q_{SV}(c_t | c_{t-1}, \mathbf{o}_t)$ is defined by a Mixture of Gaussians in hidden state space. The training data for the support vector regression defines a system of curves in reduced posture space. The centers of the Gaussians are given by the values c_t^k which correspond to the points of this curve system that are closest to the actual output of the SVM regression network (branch k) for the present image silhouette. The variances of these Gaussians were chosen to guarantee a sufficient coverage of the state space¹. The second distribution $p(c_t | c_{t-1})$ is simply the gaussian transition density from the previous state c_{t-1} . The mixture coefficient r determines how fast the particles tend to migrate to the location of the new measurements instead of remaining close to the previous state. We chose the value for $r = 0.8$ for our implementation.

Likelihood function: To define the likelihood function $p(\mathbf{o}_t | c_t^{(i)})$ for the different postures we determine the overlap between predicted posture and the real image silhouette. For a given state of each configuration variable we determine the corresponding posture in shape eigenspace, and using the patch model its backprojection into 2D image space. The value of the likelihood is given by the non-overlapping area weighted by a Gaussian function.

¹In addition, contributions from terms in Eqn. 1 with weak activation of the corresponding kernel functions were eliminated. Weak activation indicates that the actual data is very distant from the training data resulting in non-reliable predicted postures

2.5 Model-based tracking

Our model-based tracking is based on a connected 2D patch-model that embeds symmetry constraints. Its movement is similar to the gradient-based optimization approach in [4]. Our approach combines two force fields that are derived from edge features in the image sequence. Edge feature flow fields allow to achieve a relatively accurate registration of model and image contours. A second flow field that is based on symmetry features introduces an additional constraint for the registration of the patches. Symmetry feature-based image forces have the advantage that they produce less spurious correspondences, in particular over large image distances. A symmetry-based extension of a cost function has been also used for a stochastic search approach [20]. In our approach the two force fields act on the outer edges, respectively on the symmetry axes of the patch model. To avoid instabilities, the patch model has to be stabilized with additional internal forces that are determined by an elastic mechanical model (Figure 3). Like for an active contour, the final deformation of the patch model is determined by the image feature force fields and by the internal energy that depends on the mechanical properties of the model.

2.5.1 Flow fields

Edge maps are derived from the energy of simple gradient filter detectors. From these maps we derive flow fields by applying nonlinear diffusion using the PDE approach adopted from [26]. The length of the individual flow vectors was normalized after convergence. An additional symmetry vector flow field is easily obtained by inverting the sign of the forces inside the figure. The inside part of the figure is known since we restrict our application to segmented figures.

2.5.2 Motion equations for coupled graph model

Model-based tracking is realized by iteration of the equations that describe the movement of the patch model in the force fields. The patch model reaches its optimal position when the whole mechanical model is in equilibrium implying that the sum of the forces and the sum of the moments is zero. Regarding a single edge a with length l_a of the graph in Figure 3, the total force \underline{F} that is caused by the locally distributed forces \underline{f}_i can be subsumed in net forces \underline{E}_{a0} and \underline{E}'_{a0} at the endpoints of the edge:

$$\underline{F} = \sum_{i=1}^N \underline{f}_i = \underline{E}_{a0} + \underline{E}'_{a0}.$$

Here, forces of occluded positions along the edge have to be omitted in order to realize an adequate occlusion handling.

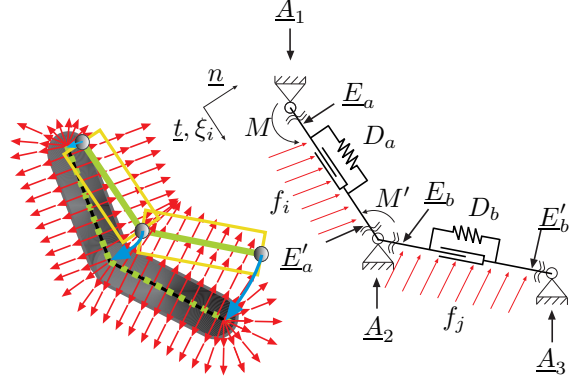


Figure 3: Left: Force field (red) derived from symmetry features. Arrows indicate the (negative) forces that act on the skeleton of the model, resulting in an alignment of the model with the image features. Right: Mechanical model to stabilize the patch model. Length of the segments is stabilized by springs. Overall, the effect of the force fields can be subsumed in net forces $A_1 \dots A_3$ by computing force and momentum equilibrium for the whole mechanical model.

For an equilibrium state also the moments around the start and endpoints of the edge have to vanish, i.e.:

$$M = \sum_{i=1}^N \xi_i \left(\underline{f}_i \underline{n} \right) = l_a \underline{E}_{a0} \underline{n} \quad \text{and}$$

$$M' = \sum_{i=1}^N (l_a - \xi_i) \left(\underline{f}_i \underline{n} \right) = l_a \underline{E}'_{a0} \underline{n}$$

where \underline{n} is a normal vector along the edge, and where ξ_i signifies the position coordinate along the edge.

From the last equations the equivalent forces E_i of an edge can be derived by solving a linear equation system, where the moments M and M' and the net force \underline{F} is derived by the vector field considering only non-occluded parts of the edge:

$$\begin{pmatrix} \mathbf{I}_{2 \times 2} & \mathbf{I}_{2 \times 2} \\ \underline{n}^T & \mathbf{0}^T \\ \mathbf{0}^T & \underline{n}^T \end{pmatrix} \begin{pmatrix} \underline{E}_{a0} \\ \underline{E}'_{a0} \end{pmatrix} = \begin{pmatrix} \underline{F} \\ M/l_a \\ M/l_a \end{pmatrix}$$

The general solution for the endpoint forces is given by

$$\underline{E}_a = \underline{E}_{a0} + \alpha_a \underline{t} \quad \text{and} \quad \underline{E}'_a = \underline{E}'_{a0} - \alpha_a \underline{t},$$

since symmetric tangential forces, $\alpha_a \underline{t}$, at the endpoints do not affect the force and momentum equilibrium. In order to stabilize the length of the individual patches of the model we assume that the tangential forces are given by a linear spring characteristics of the form

$$\alpha_a = D_a \left(l_a - \tilde{l}_a \right),$$

where $l_a = \|\underline{\xi} - \underline{\xi}'\|$ is the actual length during optimization and \tilde{l}_a the equilibrium length of the edge. From the segment's forces the net forces at the nodes of the graph can be computed. In this case we find:

$$\begin{aligned} \underline{A}_1 &= \underline{E}_{a0} + \alpha_a \underline{t}_a \\ \underline{A}_2 &= \underline{E}'_{a0} + \underline{E}'_{b0} - \alpha_a \underline{t}_a + \alpha_b \underline{t}_b \\ \underline{A}_3 &= \underline{E}'_{b0} - \alpha_b \underline{t}_b. \end{aligned}$$

The node positions \underline{x}_i are updated by displacements along these net forces in the form:

$$\Delta \underline{x}_i = \mu \frac{1}{\max_{j \in \mathcal{N}} \|\underline{A}_j\|} \underline{A}_i$$

normalizing by the maximum of the forces over the set \mathcal{N} of all nodes. The step width μ is a positive constant.

2.5.3 Automatic model (re)-initialization

The main step of our system is the integration of view-based and model-based estimates to automatically initialize gradient-based model tracking, and to control the re-initialization of that model. For this purpose, our approach initializes the model-based tracking with two different initial conditions for each time step. One initial condition is given by the model configuration corresponding to the MAP estimate from the view-based recognition of body configurations. The second initial condition is given by the patch-model configuration in the previous time step (Figure 2, F1 and F2). For each initial condition our algorithm generates new configuration estimates with model-based tracking. As a final estimate we use the configuration that has maximum overlap with the silhouette image in the present time step. The banners of the panels in Figure 5 indicate frames in which our final graph estimate is based on the initial condition from the view-based MAP estimate (*new initialization*, red graph), and the previous time step (*tracking helps*, yellow graph).

3 Results

We tested our system with a variety of movements from martial arts and gait patterns of normal people and patients. In order to provide ground truth data for the evaluation of the accuracy of the estimates we used movies that were generated by animating an avatar model with motion capture data, that we obtained with a VICON 612 motion capture system with 6 respectively 11 cameras. Tests of the method with real video data are in progress. Figure 5 monitors example frames from a gait pattern of four different neurological patients. The system was trained with one actor (T),

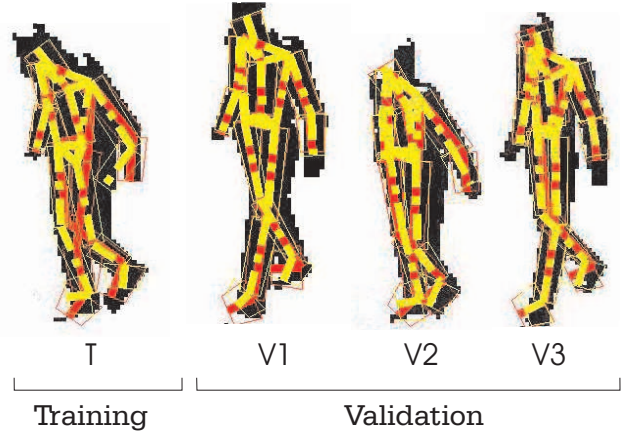


Figure 4: Results from combined two model-based tracking conditions (new initialization condition: yellow, tracked condition from previous time step: red) for different subjects, same view point. The system was trained on one actor (T) and validated with three other actors (V1-V3) (same view-point).

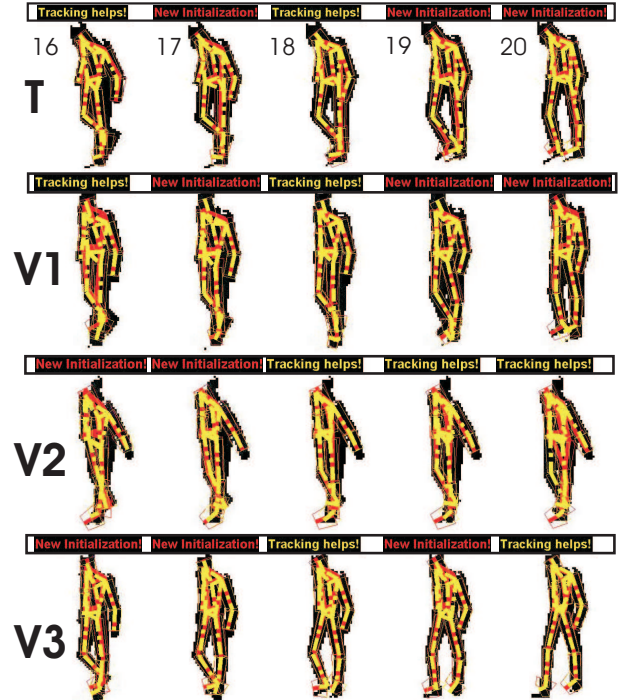


Figure 5: Sequence of tracking results that fuse estimates generated from the graph model configuration in the last frame (*tracking helps*, yellow graph), and from new view-based initialization of the model (*new initialization*, red graph).

walking at an angle of -45° towards the camera. The system automatically initializes the model and accomplishes

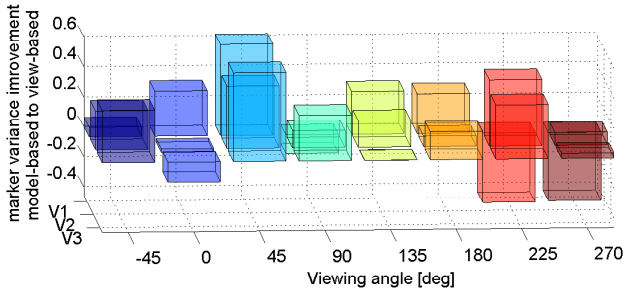


Figure 6: Index of error variance reduction: $E_{rel} = (\sigma_{View-based}^2 - \sigma_{BestFit}^2) / \sigma_{View-based}^2$. Error variances are computed for the trajectory corner points of the patch model relative to the ground truth data. $\sigma_{BestFit}^2$ describes the error of the best fit that integrates the proposals from the two initial conditions, derived from model-based and view-based tracking. $\sigma_{View-based}^2$ is the error of the trajectories estimated by view-based posture estimation without model-based tracking. If the model-based tracking would not reduce the error of the estimates E_{rel} would be zero or negative. Perfect integrated estimates ($\sigma_{BestFit} = 0$) corresponds to $E_{rel} = 1$.

accurate tracking in spite of substantial self occlusions. The red model shows the initial condition produced by the view-based posture recognition. The overlaid yellow and dashed model shows the tracked graph using the initial condition derived from model configuration in the last time step. This example shows that the proposal from the view-based posture recognition can improve the accuracy of the tracking compared to purely model-based tracking.

Furthermore, we tested the capability of our algorithm to generalize after training with one actor to actors with different body geometry (Figure 4 and 5). The system was trained with one actor (T) and tested with three different actors (V1 - V3). In all cases our algorithm accomplishes a reasonable performance, in spite of significant variations, height and limb lengths (cf. 4) and substantial variations of the arm posture (see e.g. actor V2, Figure 5). The banners (*new initialization*) in Figure 5 indicate the frames where posture recognition results in an improved tracking result. A new initialization of the graph model improves the results in particular during strong self-occlusions and when feature-based tracking diverges. To test the generalization capabilities of our algorithm, we additionally trained the system with different views of one actor (T) (0 – 360 deg sampled in 45 deg steps). The system was validated with movies from three other actors viewed from the same eight view points performing also walking.

We tried to quantify the accuracy improvement by the fusion of view-based and model-based tracking compared to purely view-based posture recognition using an index of

variance reduction E_{rel} that is displayed in Figure 6. For the majority of cases we find a substantial reduction of the tracking error, indicated by positive values of this index. For some cases the index reaches values 0.5 implying that the error variance could be reduced by factors above 2 by the integration of the two approaches.

4 Conclusion

We have presented a computer vision architecture that integrates model-based and view-based tracking of articulated figures. Compared to model-based tracking that is based on local image features our system has the advantage that it automatically initializes the model and can recover after tracking has been lost. Compared to purely view-based posture recognition our system has the advantage that it allows to reduce the number of training examples because model-based tracking can improve suboptimal posture estimates. In addition, our system improves the accuracy of the tracking compared to posture recognition because the model-based tracking step exploits detailed local information in the individual video frames. This was demonstrated by a quantitative evaluation of the system using data generated with real movement trajectories providing exact ground truth data.

Future work will focus on a further evaluation of the system performance with real video sequences. Further, a coupling with high-level models for complex trajectories (e.g. [11]) seems possible within the same framework. Such flexible trajectory models will allow to further constrain the space of admissible model postures, likely improving substantially the robustness of our system. Another possible extension of the proposed framework is the inclusion of 3D models, potentially capturing more adequately the invariances of articulated movements.

5 Acknowledgements

We thank W. Ilg for his help and providing the motion capture data, and W. von Seelen and H. Bühlhoff for their support. This research was supported by BMBF grant LOKI 01 IB 001 C, IST project COMIC IST-2002-32311 and the Deutsche Volkswagenstiftung. Additional support was provided by the Max Planck Institute for Biological Cybernetics, Tübingen.

References

- [1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *International Conference on Computer Vision & Pattern Recognition*, pages II 882–888, 2004.

- [2] F.L. Alt. Digital pattern recognition by moments. *Journal of the Association for Computing Machinery*, 1962.
- [3] M. Brand. Shadow puppetry. In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, December 2001.
- [4] T.-J. Cham and J. M. Rehg. A multiple hypothesis approach to figure tracking. In *Proceedings of Computer Vision and Pattern Recognition*, pages 239–245, 1999.
- [5] T. Cootes and C. Taylor. Statistical models of appearance for computer vision. Technical report, University of Manchester, Wolfson Image Analysis Unit, Imaging Science and Biomedical Engineering, Manchester M13 9PT, United Kingdom, 1999.
- [6] J. Deutscher, A. Blake, and I. Reid. Motion capture by annealed particle filtering. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 126–133, 2000.
- [7] D. DiFranco, T.-J. Cham, and J. Rehg. Reconstruction of 3-d figure motion from 2-d correspondences. In *Computer Vision and Pattern Recognition*, 2001.
- [8] A. Elgammal and C.S. Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.
- [9] D. Gavrilu, J. Giebel, and H. Neumann. Learning shape models from examples. In *Proceedings of the DAGM Symposium fr Mustererkennung*, volume 23, pages 369–376. Springer Verlag, 2001.
- [10] D. M. Gavrilu and V. Philomin. Proceedings of iee international conference on computer vision. In *Real-time Object Detection for Smart Vehicles*, pages 87–93, 1999.
- [11] M. A. Giese and T. Poggio. Morphable models for the analysis and synthesis of complex motion pattern. *International Journal of Computer Vision*, 38:59–73, 2000.
- [12] M. A. Giese and T. Poggio. Neural mechanisms for the recognition of biological movements. *Nature Review Neuroscience*, 4(3):179–192, 2003.
- [13] K. Grauman, G. Shakhnarovich, and T. Darrell. Inferring 3d structure with a statistical image-based shape model. In *Proceedings of the International Conference on Computer Vision*, 2003.
- [14] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference Computer Vision*, pages 343–356, 1996.
- [15] S. X. Ju, M. J. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated motion. In *International Conference on Automatic Face and Gesture Recognition*, pages 38–44, Killington, Vermont, 1996.
- [16] E. Ong and S. Gong. A dynamic human model using hybrid 2d-3d representations in hierarchical pca space. In *British Machine Vision Conference*, volume 1, pages 33–42, 1999.
- [17] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV95*, pages 612–617, 1995.
- [18] R. Rosales and S. Sclaroff. Learning body pose via specialized maps. In *Neural Information Processing Systems NIPS-14*, 2001.
- [19] H. Sidenbladh, M. Black, and D. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *European Conference on Computer Vision*, pages 702–718, 2000.
- [20] C. Sminchisescu. Consistency and coupling in human model likelihoods. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 27–32, 2002.
- [21] A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla. Learning a kinematic prior for tree-based filtering. In *British Machine Vision Conference*, 2003.
- [22] K. Toyama and A. Blake. Probabilistic tracking in a metric space. In *IEEE International Conference on Computer Vision*, volume II, 2001.
- [23] V. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995.
- [24] S. Wachter and H.-H. Nagel. Tracking of persons in monocular image sequences. *Computer Vision and Image Understanding*, 74:174–192, 1999.
- [25] Y. Wu, J. Lin, and T. Huang. Capturing natural hand articulation. In *Proceedings of IEEE International Conference on Computer Vision*, pages 426–432, 2001.
- [26] C. Xu and J. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, pages 359–369, 1998.