Tracking of Human Body Parts using the Multiocular Contracting Curve Density Algorithm

Markus Hahn, Lars Krüger, Christian Wöhler
DaimlerChrysler Group Research
Environment Perception
P. O. Box 2360, D-89013 Ulm, Germany
{Markus.Hahn | Lars.Krueger | Christian.Woehler}@dcx.com

Horst-Michael Gross
Technical University of Ilmenau
Department of Neuroinformatics and Cognitive Robotics
P. O. Box 100565, D-98684 Ilmenau, Germany
Horst-Michael.Gross@tu-ilmenau.de

Abstract

In this contribution we introduce the Multiocular Contracting Curve Density algorithm (MOCCD), a novel method for fitting a 3D parametric curve. The MOCCD is integrated into a tracking system and its suitability for tracking human body parts in 3D in front of cluttered background is examined. The developed system can be applied to a variety of body parts, as the object model is replaceable in a simple manner. Based on the example of tracking the human hand-forearm limb it is shown that the use of three MOCCD algorithms with three different kinematic models within the system leads to an accurate and temporally stable tracking. All necessary information is obtained from the images, only a coarse initialisation of the model parameters is required. The investigations are performed on 14 real-world test sequences. These contain movements of different hand-forearm configurations in front of a complex cluttered background. We find that the use of three cameras is essential for an accurate and temporally stable system performance since otherwise the pose estimation and tracking results are strongly affected by the aperture problem. Our best method achieves 95% recognition rate, compared to about 30% for the reference methods of 3D Active Contours and a curve model tracked by a Particle Filter. Hence only 5% of the estimated model points exceed a distance of 12 cm with respect to the ground truth, using the proposed method.

1 Introduction

Today, industrial production processes in car manufacturing worldwide are characterised by either fully automatic production sequences carried out solely by industrial robots or fully manual assembly steps where only humans work together on the same task. Up to now, close collaboration between human and machine is very limited and usually not possible due to safety concerns. Industrial production processes can increase efficiency by establishing a close collaboration of humans and machines exploiting their unique capabilities. The recognition of interactions between humans and industrial robots requires vision methods for 3D pose estimation and tracking of the motion of both human body parts and robot parts based on 3D scene analysis.

This paper addresses the problem of marker-less pose estimation and tracking of the motion of human body parts in front of a cluttered background. A new 3D pose estimation method, the Multiocular Contracting Curve Density algorithm (MOCCD), is introduced and applied to track the 3D pose of the hand-forearm limb with a traditional Kalman Filter framework.

2 Related Work

Due to safety concerns in our application we require precise 3D pose estimation and reliable tracking of human body parts. The Contracting Curve Density Algorithm (CCD) [8] fits a parametric curve model to an image by refining an initial parameter set. Due to its short processing time the CCD algorithm is suitable for real-time systems, and it is robust with respect to changing backgrounds while achieving a high accuracy. Krüger and Ellenrieder [10] introduce an approximate variant of the CCD algorithm for an extension to multiple calibrated cameras, such that 3D pose estimation becomes possible. In this study we utilise the newly developed Multiocular Contracting Curve Density algorithm (MOCCD) for obtaining 3D pose measurements.

Moeslund et al. [12] give a detailed introduction to and overview about the large field of human motion capture. Similar to [13, 14] we apply a multi-view 3D pose estimation algorithm which is based on silhouette information.

Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)
0-7695-2939-4/07 $25.00 © 2007 IEEE

Plänkers and Fua [13] use 3D data generated by a stereo camera to obtain a pose estimation and tracking of the human upper body. The upper body is modelled with implicit surfaces, and silhouettes are used in addition to the depth data to fit the surfaces. Rosenhahn et al. [14] track a 21 DOF 3D upper body model using a four-camera setup. The pose estimation is based on silhouettes which are extracted using level set functions. Tracking is performed by using the pose in the last frame as initial pose in the current frame. Under laboratory conditions with no cluttered background they achieve a high metric accuracy, which is demonstrated by comparison to a commercial marker-based tracking system with eight cameras.

A common limitation of both approaches is the use of a single pose which is updated at every timestep. To achieve a more robust tracking, other approaches [3, 7, 15] use a Particle Filter. This probabilistic framework employs the Monte Carlo technique of factored sampling to propagate a set of samples (particles) through state space in an efficient manner. A problem in its application to tracking body parts is the high number of DOF, since the required number of particles rises exponentially with the dimension of the state space. Deutscher et al. [7] extend the Particle Filtering scheme. To avoid local minima of the state probability they use additional resampling steps in a manner similar to simulated annealing. With this modification, they are able to track full body motion with 100 particles. They use edge detection and background subtraction to weight their particles. Schmidt et al. [15] employ a Kernel Particle Filter for 3D body tracking. They use 150 particles to track a 14 DOF 3D model of the upper body with a single uncalibrated camera. The particles are weighted by the use of colour cues that are combined with ridge and edge cues. In Stenger et al. [16] a detailed 3D hand model composed of quadrics is used to segment and track the human hand. The matching is done by minimising the geometric error between the model projection and the edges of the human hand in the image. It is assumed that the hand posture does not change. An Unscented Kalman Filter is used to track a 6 DOF hand model.

Due to the limited resolution of our trinocular grey scale camera setup it is infeasible in our system to model each finger of the hand as it is done in [16]. Furthermore, a cylindrical model of the forearm like [15] is too coarse due to the variability of human appearance, e.g. clothes. Our approach is therefore based on a 3D hand-forearm model which represents the 3D contour by an Akima spline [1] using control points defined by a parameter vector. The MOCCD algorithm is computational too expensive to use it in a Particle Filter framework. Hence we integrate the MOCCD algorithm in a traditional Kalman Filter based tracking framework which estimates more than one pose hypothesis at a single timestep.

3 Theoretical Background

3.1 The CCD Algorithm

The real-time CCD algorithm [8] fits a parametric curve $c(\alpha, \mathbf{\Phi})$ to an image $\mathbf{I}$. The parameter $\alpha \in [0, 1]$ increases monotonically along the curve, and $\mathbf{\Phi}$ denotes a vector containing the curve parameters to be optimised. The principle of the CCD algorithm is depicted in Fig. 1. The input values of the CCD are an image $\mathbf{I}$ and the Gaussian a priori distribution $p(\mathbf{\Phi}) = p(\mathbf{\Phi}|\widehat{\mathbf{m}}_{\mathbf{\Phi}}, \widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}})$ of the model parameters $\mathbf{\Phi}$, defined by the mean $\widehat{\mathbf{m}}_{\mathbf{\Phi}}$ and the covariance $\widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}}$. The CCD algorithm estimates a model pose by computing the maximum of the a posteriori probability (MAP) according to

$$p(\mathbf{\Phi}|\mathbf{I}) = p(\mathbf{I}|\mathbf{\Phi}) \cdot p(\mathbf{\Phi}) \; . \qquad (1)$$

In Eq. (1) $p(\mathbf{I}|\mathbf{\Phi})$ is approximated by $p(\mathbf{I}|S(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}}))$, with $S(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}})$ representing the pixel value statistics close to the curve. The maximisation of Eq. (1) is performed by iterating the following two steps until the changes of $\mathbf{m}_{\mathbf{\Phi}}$ and $\mathbf{\Sigma}_{\mathbf{\Phi}}$ fall below a threshold or a fixed number of iterations is completed. The procedure starts from the user supplied initial density parameters $(\widehat{\mathbf{m}}_{\mathbf{\Phi}}, \widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}})$.

1. Compute the pixel value statistics $S(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}})$ on both sides of the curve. For grey scale images this procedure amounts to computing a mean and a standard deviation of the pixel grey values on either side of the curve.

2. Refine the curve density parameters $(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}})$ towards the maximum of Eq. (1) by performing one step of a Newton-Raphson optimisation procedure. This step moves the segmentation boundary such that the image content (grey values) conforms better with the pixel statistics, i.e. towards an edge.

A numerically favourable form of Eq. (1) is obtained by computing the log-likelihood

$$X = -2 \ln \left[p\left(\mathbf{I}|S(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}})\right) \cdot p(\mathbf{\Phi}|\widehat{\mathbf{m}}_{\mathbf{\Phi}}, \widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}}) \right] \; . \qquad (2)$$

In terms of image processing this procedure can be seen as follows: The sum of Gaussian probability densities $p\left(\mathbf{I}|S(\mathbf{m}_{\mathbf{\Phi}}, \mathbf{\Sigma}_{\mathbf{\Phi}})\right)$ is an edge detector along the curve normal, i.e. if the curve is at the edge, the function value is maximal. In contrast to classical edge detectors (e.g. Sobel, Prewitt) the kernel size is adaptive and the function is spatially differentiable. These properties are the main reasons for the robustness and accuracy of the CCD algorithm.

3.2 The MOCCD Algorithm

We extend the CCD algorithm to multiple calibrated cameras by projecting the boundary of a 3D contour model

IEEE COMPUTER SOCIETY

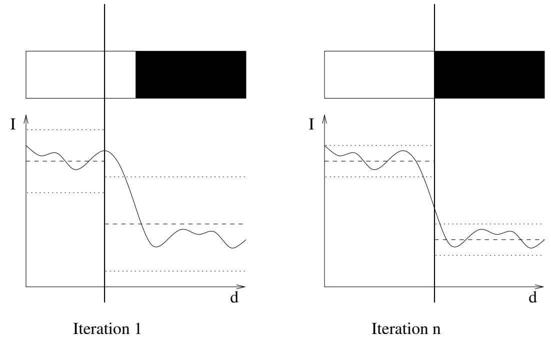

Figure 1. The principle of the CCD algorithm. Fitting the segmentation boundary (bold line) to the image grey values (solid line) by estimating the mean (dashed line) and standard deviation (dotted line) on either side of the assumed boundary. The boundary is moved such that the image grey values have the highest probability according to the means and standard deviations.

into each image. The intrinsic and extrinsic parameters of the camera model [4] are obtained by multiocular camera calibration [11]. An arbitrary number of images N_c can be used for this projection. We maximise the joint probability

$$p(\mathbf{\Phi}|\mathbf{I}_1,\dots,\mathbf{I}_{N_\mathrm{c}}) = \left[\prod_{c=1}^{N_\mathrm{c}} p(\mathbf{I}_c|S_c(\mathbf{m_\Phi},\mathbf{\Sigma_\Phi}))\right] \cdot p(\mathbf{\Phi}|\widehat{\mathbf{m}}_\mathbf{\Phi},\widehat{\mathbf{\Sigma}}_\mathbf{\Phi}) \quad (3)$$

with $S_c(\mathbf{m_\Phi},\mathbf{\Sigma_\Phi})$ representing the grey value statistics close to the projected curve in image $\mathbf{I}_c$. The underlying assumption is that images are independent random variables. Like in the original CCD framework, there is a numerically favourable form of Eq. (3) obtained by computing the log-likelihood. The MOCCD performs an implicit triangulation and is summarised in Algorithm 1.

Algorithm 1 Pseudo code of the **MOCCD algorithm**

```
Input: images $\mathbf{I}_1,\dots,\mathbf{I}_{N_\mathrm{c}}$, a priori density $(\widehat{\mathbf{m}}_\mathbf{\Phi},\widehat{\mathbf{\Sigma}}_\mathbf{\Phi})$
Output: refined model parameters $(\mathbf{m_\Phi},\mathbf{\Sigma_\Phi})$
  for $iter = 1$ to $maxIter$ do
    for $c = 1$ to $N_\mathrm{c}$ do
      Project the 3D contour to its 2D representation.
      Compute the statistics $S_c(\mathbf{m_\Phi},\mathbf{\Sigma_\Phi})$ in image $\mathbf{I}_c$.
    end for
    Refine the parameters $(\mathbf{m_\Phi},\mathbf{\Sigma_\Phi})$ towards the maximum of Eq. (3) by one Newton-Raphson step.
  end for
```

This evaluation scheme is an improvement over [10], where the 3D contour model is projected to N_c individual 2D curves which are matched individually in 2D. At a later stage the individual curves are integrated to an update of the 3D model by means of a 2D-3D pose estimation. The advantage of the improvement presented here is that spatial constraints are handled directly in the optimisation procedure instead of the 2D-3D pose estimation. Hence the model shape is optimised to fit all images equally well instead of allowing arbitrary 2D deformations as in [10].

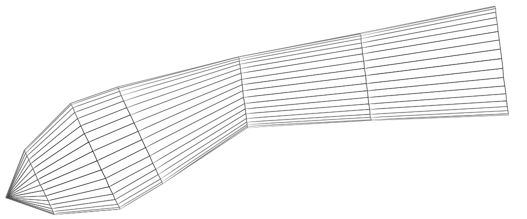

Figure 2. 3D hand-forearm model.

4 3D Tracking of the Human Hand-Forearm Limb

4.1 Modelling the Hand-Forearm Limb

We use an analytical 3D model of the human hand-forearm limb, a kinematic chain connecting the two rigid elements forearm and hand. The model consists of five truncated cones and one complete cone (Fig. 2). Fig. 3 (left) depicts the definition of the cones by the following nine parameters:

$$\mathbf{\Phi} = [p_{1x}, p_{1y}, p_{1z}, \alpha_1, \beta_1, \alpha_2, \beta_2, r_1, r_4]^T \ . \quad (4)$$

The 3D point $\mathbf{p}_1$ defines the beginning of the forearm and is part of the parameter vector $\mathbf{\Phi}$. The wrist ($\mathbf{p}_2$) and fingertip ($\mathbf{p}_3$) positions are computed by

$$\begin{aligned} \mathbf{p}_2 &= \mathbf{p}_1 + \mathbf{R_Z}(\beta_1) \cdot \mathbf{R_Y}(\alpha_1) \cdot l_\text{forearm} \cdot [1,0,0]^T \quad (5) \\ \mathbf{p}_3 &= \mathbf{p}_2 + \mathbf{R_Z}(\beta_2) \cdot \mathbf{R_Y}(\alpha_2) \cdot l_\text{hand} \cdot [1,0,0]^T , \quad (6) \end{aligned}$$

where l_forearm and l_hand are the defined lengths of the human hand-forearm limb. The matrix $\mathbf{R_Y}(\alpha)$ represents the rotation around the Y axis by the angle α and $\mathbf{R_Z}(\beta)$ the corresponding rotation around the Z axis. The radii of the cones are computed by

$$\begin{aligned} r_2 &= 0.8 \cdot r_1 \\ r_3 &= 0.65 \cdot r_1 \\ r_5 &= 1.1 \cdot r_4 \\ r_6 &= 0.026\,\mathrm{m} \quad \text{constant.} \quad (7) \end{aligned}$$

The dependencies of the radii are derived from human anatomy, see Fig. 3 (right). Only r_1 and r_4 are part of the

Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)
0-7695-2939-4/07 $25.00 © 2007 IEEE

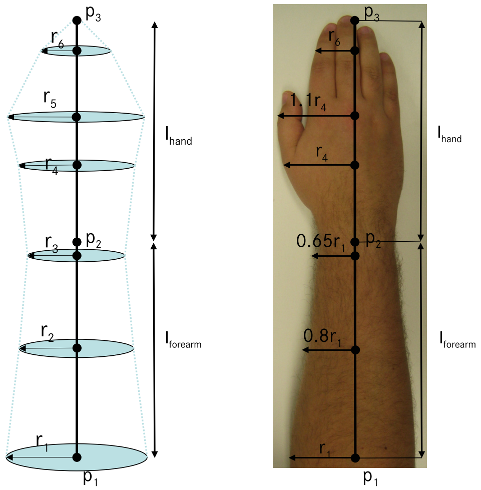

Figure 3. Left: Definition of the cones. Right: Dependencies of the radii derived from human anatomy.

parameter vector $\mathbf{\Phi}$. As the MOCCD algorithm adapts a model curve to the image, the silhouette of the 3D model in each camera coordinate system has to be extracted. Thus a vector from the origin of each camera coordinate system to the point in the wrist is computed, e.g. $\mathbf{p}_2^{\mathbf{C}_1}$ for camera 1. This vector and the direction vector of the forearm $\overline{\mathbf{p}_1\mathbf{p}_2}$ span a plane which is intersected with the 3D model to yield the 3D outline. The extracted 3D contour model for the given camera — consisting of 13 points — is projected in the pixel coordinate system of the camera. The 2D contour model is computed by an Akima interpolation [1] along the curve distance with the 13 projected points as control points. Fig. 4 depicts the extraction and projection of the 3D contour model for camera 1.

4.2 System Overview

We have developed a generic recognition system (Fig. 5) consisting of three components: N instances of the MOCCD algorithm, a subsystem to find the best measurement of the different MOCCD instances (Winner-Takes-All) and N Kalman Filters associated with the MOCCD instances. The input images are obtained from a grey scale Digiclops camera with VGA resolution manufactured by Point Grey. The following algorithm outlines the interaction of the different components:

1. Initialisation.

2. For all MOCCDs: compute the measurement in t.

3. Find the best of these measurements in t.

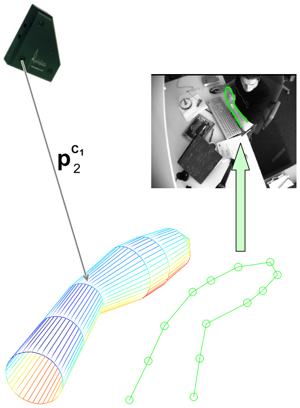

Figure 4. Extraction and projection of the 3D contour model.

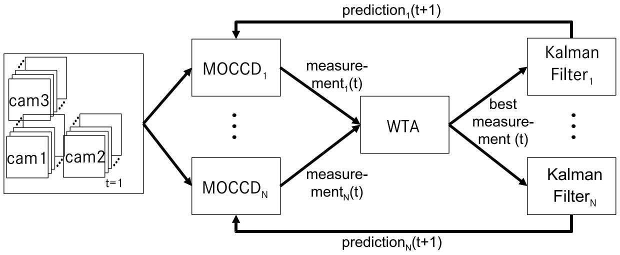

Figure 5. Generic recognition system.

4. For all Kalman Filters: compute their prediction using the best measurement.

In the initialisation step the parameter vector $\mathbf{\Phi}(t = 1)$, which describes the position of the hand-forearm limb at the first timestep $t = 1$, is defined. After initialisation the developed system is able to track the human hand-forearm limb in 3D. The tracking is realised by steps 2 through 4. In the second step the measurements of all N MOCCDs in timestep t are computed. The measurement of MOCCD i for the parameter vector $\mathbf{\Phi}$ consists of $(\mathbf{m}_{\mathbf{\Phi}}^{(i)}(t), \mathbf{\Sigma}_{\mathbf{\Phi}}^{(i)}(t))$. The measurement of a MOCCD is based on the Gaussian a priori distribution $p(\mathbf{\Phi}) = p(\mathbf{\Phi}|\widehat{\mathbf{m}}_{\mathbf{\Phi}}, \widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}})$ of the model parameters $\mathbf{\Phi}$, which are refined according to Sec. 3.2. At timestep $t = 1$ the Gaussian a priori distribution is user defined and in all other timesteps it is computed by the prediction of the tracker associated with the MOCCD. To select starting parameters for Algorithm 1 the mean $\widehat{\mathbf{m}}_{\mathbf{\Phi}}^{(i)}(t)$ of MOCCD i at timestep t is obtained from the predicted parameter vector $\widehat{\mathbf{\Phi}}^{(i)}(t)$. The covariance $\widehat{\mathbf{\Sigma}}_{\mathbf{\Phi}}^{(i)}(t)$ is assumed to be a constant matrix.

After all MOCCD measurements are computed, the best measurement $\mathbf{\Phi}(t)$ of the parameter vector $\mathbf{\Phi}$ at timestep t is extracted based on a Winner-Takes-All approach with re-

Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)
0-7695-2939-4/07 $25.00 © 2007 IEEE

spect to $(\mathbf{m}_{\mathbf{\Phi}}^{(1)}(t), \mathbf{\Sigma}_{\mathbf{\Phi}}^{(1)}(t)), \ldots, (\mathbf{m}_{\mathbf{\Phi}}^{(N)}(t), \mathbf{\Sigma}_{\mathbf{\Phi}}^{(N)}(t))$.

As criteria for the measurement quality, we utilize (i) the confirmation measurement of the MOCCD, (ii) the quality of the prediction, and (iii) the difference of the grey value statistics along the model curve.

The confirmation measurement is introduced by Hanek [9] and is an indicator of the convergence of the MOCCD. The second criterion describes how similar the prediction of the tracker and the measurement of the MOCCD are. With the third criterion it is ensured that the MOCCD separates grey value statistics along the projected curve. A measurement that is better than any other in at least two criteria is deemed the winner.

In the fourth step of the algorithm the best measurement $\mathbf{\Phi}(t)$ is used in each Kalman Filter to produce the prediction of the parameter vector $\mathbf{\Phi}$. These N predictions $\widehat{\mathbf{\Phi}}^{(1)}(t+1), \ldots, \widehat{\mathbf{\Phi}}^{(N)}(t+1)$ are used to produce the measurements in $t+1$.

4.3 Kinematic Models

We investigate three different kinematic models [2] to find the best compromise between computational effort and tracking capabilities: (i) a single Kalman Filter with a constant-velocity model, (ii) two Kalman Filters with constant-acceleration and constant-velocity models, and (iii) three Kalman Filters with constant-acceleration, constant-velocity, and constant-position models.

Each of the Kalman Filters implements a different kinematic model, assuming a different object motion. Thus if three Kalman Filters are used, three different motion models are evaluated simultaneously. The Winner-Takes-All component will then select the best-fitting model in the next timestep. The idea behind this kinematic modelling is to provide a sufficient amount of flexibility for changing hand-forearm motion. It is required for correctly tracking reversing motion, e.g. occurring during tightening of a screw.

5 Experimental Investigations

The system is evaluated on 14 real-world test sequences. These sequences contain movements of different hand-forearm configurations of four different test persons in front of complex cluttered background. Each sequence contains at least 100 image triples and the mean distance of the test persons to the camera system varies from 0.85 m to 1.75 m.

5.1 Ground Truth

The ground truth consists of three points in the world coordinate system (Fig. 6). These three points correspond to the points $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ of the 3D model of the human hand-forearm limb (Sec. 4.1). To compute the ground truth three markers were fixed to the tracked limb. These markers were manually labelled in the three images of the camera system and the 3D coordinates were computed by minimisation of the Euclidean backprojection error [17].

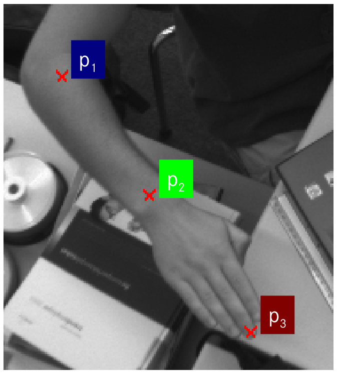

Figure 6. Determination of the ground truth.

5.2 Results

The results of a system variant are depicted as a histogram (Fig. 7). We computed the Euclidean distance to the ground truth at every timestep for the estimated points $\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$ of the 3D model. The mean Euclidean distance of the three points to the ground truth for a single sequence is computed, and the results of all sequences are arranged as a histogram. The abscissa of the histogram is divided from the left to the right into five intervals:

- **very good**: $[\mathbf{0},\ \mathbf{0.03}[$ m
- **good**: $[\mathbf{0.03},\ \mathbf{0.06}[$ m
- **satisfactory**: $[\mathbf{0.06},\ \mathbf{0.09}[$ m
- **sufficient**: $[\mathbf{0.09},\ \mathbf{0.12}[$ m
- **insufficient**: $[\mathbf{0.12},\ \mathbf{\infty}[$ m.

The choice of the intervals is justified by the system concept. A distance of 12 cm is the upper limit, as this range can be covered by close range sensors on the robot. On the ordinate the number of sequences — whose mean Euclidean distance to the ground truth for the respective point falls into the defined bucket — is depicted.

The histogram of the system variant using a single Kalman Filter with a constant-velocity model is depicted in Fig. 7. Many sequences fall into the "insufficient" bucket. This behaviour is due to false predictions of the Kalman Filter during motion reversal, e.g. when the test person tightens a screw. The false predictions result from the inadequate motion model, leading to poor convergence of the MOCCD algorithm during the subsequent timesteps. If the misprediction exceeds the convergence radius of the MOCCD, the

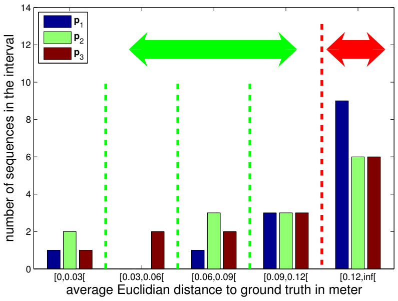

Figure 7. Results of a single Kalman Filter with a constant-velocity model.

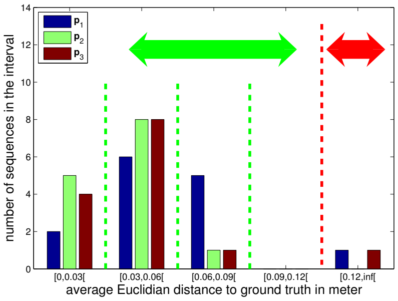

Figure 9. Results of three Kalman Filters representing constant-acceleration, constant-velocity, and constant-position models.

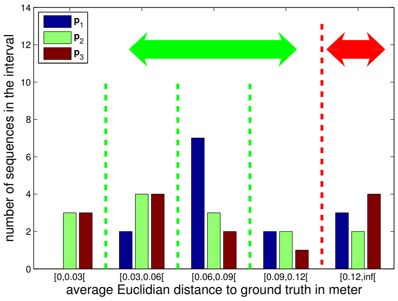

Figure 8. Results of two Kalman Filters with constant-acceleration and constant-velocity models.

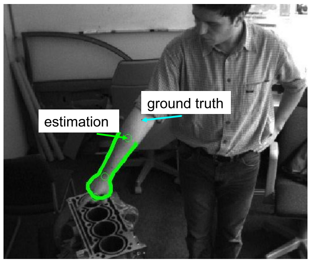

Figure 10. Stable, but somewhat inaccurately tracked sequence.

MOCCD can not compensate the too large distance between the depicted object and its assumed position.

The use of two Kalman Filters with constant-acceleration and constant-velocity kinematic models improves the system performance (Fig. 8), as only a fifth of all sequences falls into the “insufficient” interval. The measurement selection component is able to recognise false predictions and therefore the tracking is more stable than with only one Kalman Filter.

The use of three Kalman Filters with the different kinematic models constant-acceleration, constant-velocity, and constant-position leads to a further improvement (Fig. 9) and is the best system variant obtained in this investigation. Only two sequences fall into the “insufficient” interval and most of the sequences fall into the “good” bucket. This yields an error rate of about 5%. The remaining 95% of the sequence have a maximal error of less than 9 cm. This is one fourth better than required in our system concept and will lead to a better stability of the overall system as the close range sensors will rarely have to stop the robot. With our Matlab implementation, the time required for this system variant is 0.2 frames per second on a 3 GHz P4. Fig. 13 shows some short scenes taken from the sequences and the corresponding result of the best system variant.

In the result histogram of the best system variant (Fig. 9) one can see that two sequences still fall into the “insufficient” bucket. The reason is the coarse symmetric model of the hand-forearm limb. The arising silhouette of the pointing hand can not be represented by the present hand-forearm model. The system estimates the elbow point ($\mathbf{p}_1$) too far towards the hand (Fig. 10) as the overall scaling of the arm depends on the scaling of the hand. The estimated model likelihood is nearly constant when shifting the outline of the forearm model along the depicted forearm. Thus it does not stop the MOCCD from shrinking the model. However, even in presence of such input images violating the assumed geometry of the model the tracking does not fail for the whole sequence.

With another experiment we examined the importance of using three cameras. Although the system is still capable of triangulation, the performance decreases because about 50% of the sequences fall into the “insufficient” bucket. The reason is that the MOCCD is strongly affected by the aperture problem in one direction due to the elongated shape of the hand-forearm limb. With the use of the third camera in Fig. 9 the ambiguities are avoided, resulting in good triangulation in both directions.

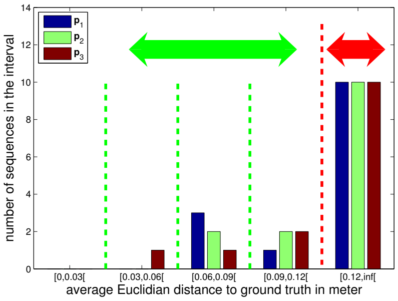

Figure 11. Results of the Particle Filter based curve model approach [3].

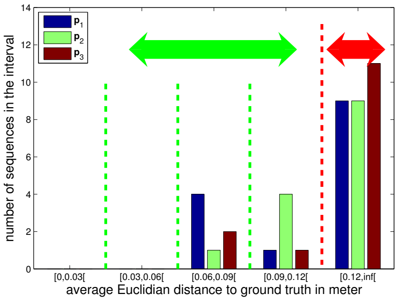

Figure 12. Results of the 3D Active Contour experiment [6].

To compare our tracking system we applied two available approaches — a curve model tracked by a Particle Filter [3] and 3D Active Contours [6] — which yields good results in former applications. The Particle Filter based approach in [3] was extended to use 3D measurements by projecting a 3D model curve to the three images. Furthermore we extended the 3D Active Contours in [6] by a tracking system using the same three Kalman Filter framework as for the MOCCD.

Processed with the Particle Filter based approach (Fig. 11), about 70% of the sequences fall into the "insufficient" bucket. Although we used 10000 particles at 9 DOF, the algorithm gets stuck at edges due to the cluttered background and shading-related edges. This result illustates the higher robustness of the MOCCD due to the fact that it adapts to the spatial scale of an edge.

Analysing the sequences using 3D Active Contours (Fig. 12) yields a similar error rate of about 70% of the sequences in the "insufficient" bucket. The algorithm gets stuck in background and shading-related edges that have a sufficient strength. Again the adaptive behaviour of the MOCCD proves to be superior compared to edge extraction methods on fixed scales, regardless of the subsequent recognition method.

6 Summary and Conclusion

Our experiments with the 14 real-world test sequences have shown that the use of three MOCCD algorithms with three different kinematic models within the tracking system leads to a reasonably accurate and temporally stable system.

Only coarse information about the test person is necessary: the lengths of forearm and hand as well as the initial position. The test sequences include the following hand-forearm configurations: an outstretched hand, a fist, a hand holding a cup, a hand holding a wrench, and a pointing hand. It is possible to track the motion of different hand-forearm configurations: reversing motion, e.g. tightening of a screw, movement of the hand in depth with a constant forearm position, lateral movement of the complete hand-forearm limb, and pointing gestures. The system is able to track them in a temporally stable manner.

A possible improvement is the use of colour cameras. The MOCCD can be easily applied to colour images, which may overcome problems with grey value statistics. A possible extension of our system may include the modelling of the whole upper body with complex kinematic chains and a hierarchical tracking like in [5] .

References

[1] H. Akima. A new method of interpolation and smooth curve fitting based on local procedures. *Journal of the Association for Computing Machinery*, 17(4):589–602, 1970.

[2] Y. Bar-Shalom and X. R. Li. *Estimation and Tracking: Principles, Techniques, and Software*. 1993.

[3] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[4] J.-Y. Bouguet. Camera calibration toolbox for matlab, 1997.

[5] G. K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time part ii: Applications to human modeling and markerless motion tracking. *International Journal of Computer Vision*, 63(3):225–245, 2005.

[6] P. d'Angelo, C. Wöhler, and L. Krüger. Model based multi-view active contours for quality inspection. In *Proc. of International Conference on Computer Vision and Graphics*, Warszaw, Poland, 2004.

[7] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2126–2133, 2000.

[8] R. Hanek. The contracting curve density algorithm and its application to model-based image segmentation. In *Proc. of CVPR*, pages 797–804, 2001.

[9] R. Hanek. *Fitting Parametric Curve Models to Images Using Local Self-adapting Separation Criteria*. PhD thesis, Technische Universität München, München, 2004.

[10] L. Krüger and M. M. Ellenrieder. Pose estimation using the multiocular contracting curve density algorithm. In *Proc. of VMV 2005*, Erlangen, 2005.

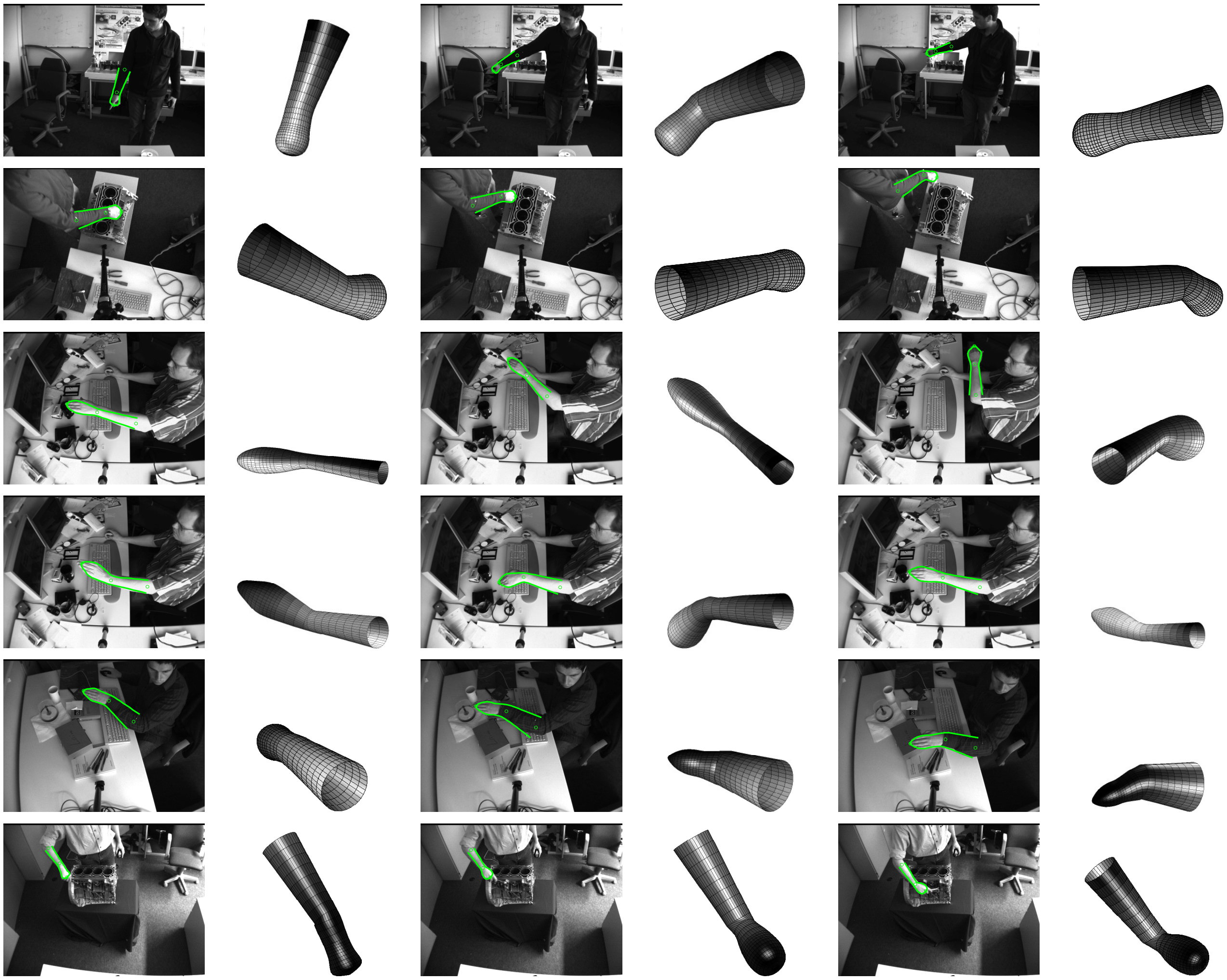

Figure 13. Results of different test sequences computed with three Kalman Filters and the different kinematic models constant-acceleration, constant-velocity, and constant-position. Columns 2,4,6: 3D reconstruction of the matching results.

[11] L. Krüger, C. Wöhler, A. Würz-Wessel, and F. Stein. Infactory calibration of multiocular camera systems. In *Photonics Europe, Automatic Target Recognition XIV*, Proc. SPIE Vol. 5457, pages 126–137, 2004.

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

[13] R. Plänkers and P. Fua. Articulated soft objects for multiview shape and motion capture. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1182–1187, 2003.

[14] B. Rosenhahn, U. Kersting, A. Smith, J. Gurney, T. Brox, and R. Klette. A system for marker-less human motion estimation. In *Pattern recognition : 27th DAGM Symposium*, volume 3663, pages 230–237, Vienna, Austria, September 2005. Springer-Verlag.

[15] J. Schmidt, J. Fritsch, and B. Kwolek. Kernel particle filter for real-time 3d body tracking in monocular color images. In *Proc. of the 7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 567–572, Washington, DC, USA, 2006. IEEE Computer Society.

[16] B. Stenger, P. R. S. Mendonça, and R. Cipolla. Model-based hand tracking using an unscented kalman filter. In *Proc. of the British Machine Vision Conference*, volume I, pages 63–72, Manchester, UK, September 2001.

[17] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000.

Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)
0-7695-2939-4/07 $25.00 © 2007 IEEE