Drift-free Tracking of Rigid and Articulated Objects

Juergen Gall, Bodo Rosenhahn, and Hans-Peter Seidel
Max-Planck-Institute for Computer Science, Campus E1 4, 66123 Saarbrücken, Germany
{jgall, rosenhahn, hpseidel}@mpi-inf.mpg.de

Abstract

Model-based 3D tracker estimate the position, rotation, and joint angles of a given model from video data of one or multiple cameras. They often rely on image features that are tracked over time but the accumulation of small errors results in a drift away from the target object. In this work, we address the drift problem for the challenging task of human motion capture and tracking in the presence of multiple moving objects where the error accumulation becomes even more problematic due to occlusions. To this end, we propose an analysis-by-synthesis framework for articulated models. It combines the complementary concepts of patch-based and region-based matching to track both structured and homogeneous body parts. The performance of our method is demonstrated for rigid bodies, body parts, and full human bodies where the sequences contain fast movements, self-occlusions, multiple moving objects, and clutter. We also provide a quantitative error analysis and comparison with other model-based approaches.

1. Introduction

3D tracking of rigid bodies or humans is essential for many applications in different areas ranging from motion analysis in sports and medical diagnostics to entertainment. While commercial marker-based systems are widely-used, vision-based motion capture is still a challenging task. In the last decade, model-based approaches like [11] have become popular where the position, rotation and joint angles of a known 3D model are estimated from video data of one or multiple calibrated cameras. In this work, we address the multi-camera case as it is common for marker-based systems but we rely only on natural features present in the images instead of attached markers.

Image features can be tracked over time by flow-based methods [17, 19] or by patch-based 2D tracker like KLT [22] or interest point matching [18]. Under the assumption that the pose is well estimated for the current frame, the 2D correspondences between the current frame and the next frame can be used to estimate the 3D pose for the next frame as illustrated in Figure 1 for a rigid object. The main drawback of these approaches is the error accumulation over time resulting in a drift away from the object. To overcome this limitation, the combination of multiple cues was proposed, *e.g.* optic flow and edges for face tracking [9] or optic flow and contour for rigid objects [4]. In [15] an iterative analysis-by-synthesis approach was suggested for face tracking.

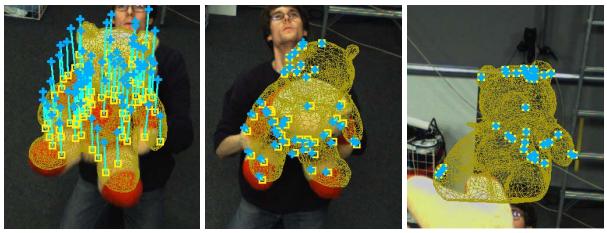

Figure 1. **Motivation.** When the pose of the target object *(projected mesh)* is known at the current frame, 2D correspondences between the current frame *(square)* and the next frame *(cross)* are often used to estimate the pose for the next frame. While this is sufficient at the beginning when the pose is well estimated *(left)*, small errors that accumulate over time result in a drift away from the target *(center)*. In the worst case, the object is completely lost *(right)*. The drift is even more problematic when occlusions occur, *e.g.* occlusions by other objects (Figure 5) or self-occlusions in the case of humans (Figure 8).

We go beyond the tracking of rigid objects and faces and propose a framework that combines the ideas of multi-cue integration and analysis-by-synthesis for the challenging task of human motion capture and tracking in the presence of multiple objects where drift becomes even more problematic due to occlusions as shown in Figures 5 and 8. To recover from errors and to detect occlusions, we propose the use of a synthesized image, which is generated with the predicted pose of the object and a static texture, as a reference image for each frame. For both prediction and correction by synthesis, patch-based matching is performed as outlined in Figure 2. While the illumination properties between two successive frames are similar and therefore a large number of matches can be provided in the predic-

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

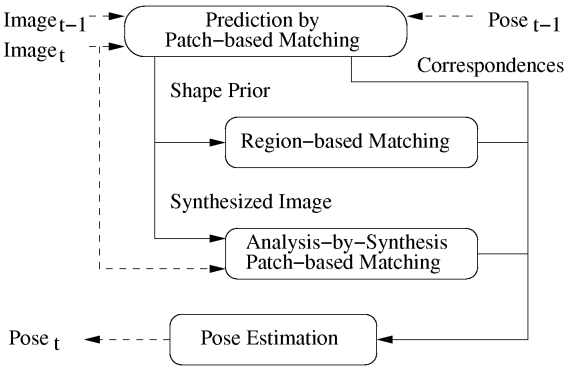

Figure 2. Having estimated the pose for time $t-1$, the pose for the next frame is predicted by matching patches between the images of frames $t-1$ and t. The predicted pose provides a shape prior for the region-based matching and defines the pose of the model for synthesis. The final pose for frame t is estimated from weighted correspondences emerging from the prediction, region-based matching, and analysis-by-synthesis, see also Figure 4.

tion step (see Figure 4), a static texture in the synthesis step provides correspondences that are not affected by error accumulation during tracking. Since the surfaces of human body parts are not always covered by patterns, which can be tracked well by patch-based matching, correspondences for homogeneous body parts are obtained by region-matching where the segmentation is improved by a shape prior from the predicted pose.

In our experiments, we show that our framework solves the drift problem better than approaches relying only on multi-cue integration and we present tracking results for rigid bodies, body parts, and full human bodies in various sequences that contain fast movements, self-occlusions, multiple moving objects, and clutter. We also provide a quantitative error analysis and compare our method with a statistical appearance model [1].

1.1. Related Work

Optical flow is very popular for model-based tracking, see *e.g.* [3]. It was also used for constructing a 3D flow field [25] to refine the pose, however, this approach requires accurate silhouettes and a relatively large number of cameras to get a stable 3D flow field. More recently, the segmentation and pose estimation process was coupled in terms of a shape prior for the level-set [20] or graph-cut segmentation [2]. Since the performance depends on the accuracy of the shape prior, optic flow was again used to predict the shape of rigid objects [4].

Leptit *et al.* [14] handled the drift problem by regarding tracking as a detection problem. They stored patches of a textured model from different viewpoints in a preprocessing step and matched each frame to one of the keyframes. Although there exists a real-time implementation [13], it is not suitable for articulated objects since the large number of degrees of freedom requires a large number of keyframes. In addition, small body parts like hands cannot be estimated by features alone. To handle illumination differences between synthesized images and original images, illumination templates were proposed for head tracking [7]. In [1] a statistical model was used for human motion tracking that relies on a template from the previous frame and a stable template that changes only slowly over time.

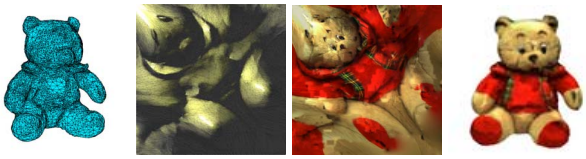

Figure 3. **From left to right:** *a)* Triangulation. *b)* Parameterization. *c)* Texture map. *d)* Textured model.

2. Synthesis

For synthesizing images, the texture of the model needs to be acquired. As it is common for model-based tracking approaches, we assume that a triangulated 3D model is available as shown in Figure 3 a), which might be obtained by any 3D acquisition or modeling technique. Since the image domain is only 2D, a parametrization of the 3D surface is necessary. For this purpose, the mesh is manually cut and mapped to a square where unavoidable distortions of the triangles are reduced by a quasi-harmonic map [26], see Figure 3 b). The images for the texture can be either acquired directly from the tracking sequence or in a preprocessing step by capturing the object from different viewpoints with a calibrated camera. Having images of the object from different views, the silhouettes are extracted by background subtraction and the pose of the model is estimated from the silhouettes [10]. The object is then mapped onto the squared texture map for each camera and the visible parts are fused by multiresolution splines [6] in order to remove seams between triangles from different views. Invisible triangles are filled up by linear interpolation. A resulting texture map is shown in Figure 3 c), which can be used to render the model in any pose. The texture acquisition for articulated models is the same as for rigid models.

3. Cues

3.1. Region-based Matching

Region-based matching minimizes the difference between the projected surface of the model and the object region extracted in the image, see Figure 4 b). For this pur-

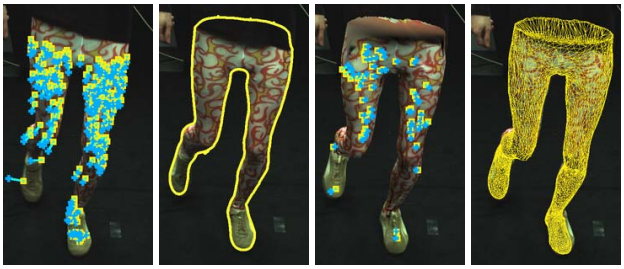

Figure 4. **From left to right:** *a)* Correspondences between current frame *(yellow square)* and next frame *(blue cross)*. *b)* The extracted contour for region-based matching also provides correspondences for homogeneous body parts like the right foot. *c)* Correspondences between the synthesized image and the original image. *d)* Projection of estimated pose.

pose, 2D-2D correspondences between the contour of the projected model and the segmented contour are established by a closest point algorithm [27]. Since the projected points on the contour relate to 3D vertices of the mesh as shown in Figure 4 d), 3D-2D correspondences between the model and the image can be derived.

The silhouette of the object is extracted by a level-set segmentation that divides the image into fore- and background where the contour is given by the zero-line of a level-set function Φ. As proposed in [20], the level-set function Φ is the minimum of the energy functional

$$E(\Phi, \widehat{x}) = -\int_{\Omega} H(\Phi) \ln p_1 + (1 - H(\Phi)) \ln p_2 \, dx + \nu \int_{\Omega} |\nabla H(\Phi)| \, dx + \lambda \int_{\Omega} (\Phi - \Phi_0(\widehat{x}))^2 \, dx, \quad (1)$$

where H is a regularized version of the step function, p_1 and p_2 are the densities of the fore- and background modeled by local Gaussian densities. While the first term maximizes the likelihood, the second term regulates the smoothness of the contour by parameter $\nu = 2$. The last term penalizes deviations from the projected surface of the predicted pose $\Phi_0(\widehat{x})$ with $\lambda = 0.06$.

3.2. Patch-based Matching

Patch-based matching extracts correspondences between two successive frames for prediction and between the current image and a synthesized image for avoiding drift as outlined in Figure 2. The synthetic image is obtained by projecting the predicted textured model onto the current image as shown in Figure 4 c). For reducing the computation effort of the keypoint extraction [16], a region of interest is selected by determining the bounding box around the projection and adding fixed safety margins that compensate for the movement. To cope with the illumination differences between the synthetic and the current image, we apply PCA-SIFT [12] as local descriptor that is trained for the object by building the patch eigenspace from the object texture. 2D-2D correspondences are then established by nearest neighbor distance ratio matching [18] where the search is reduced to keypoints inside a local neighborhood, *e.g.* 100×100 pixels, to deal with repeating patterns on the surface. Since each 2D keypoint x of the projected model is inside or on the border of a triangle with vertices v_1, v_2, and v_3, the 3D counterpart is approximated by $X = \sum_i \alpha_i V_i$ using barycentric coordinates $(\alpha_1, \alpha_2, \alpha_3)$. The corresponding triangle for a 2D point can be efficiently determined by a look-up table containing the color index and vertices for each triangle.

The patch matching produces also outliers that need to be eliminated. In a first coarse filtering step, mismatches of the torso and each limb are removed by discarding 2D-2D correspondences with an Euclidean distance that exceed the average of the torso or limb by a multiple. After deriving the 3D-2D correspondences, the pose is estimated and the new 3D correspondences are projected back. By measuring the distance between the 2D correspondences and their reprojected counterparts, the remaining outliers are detected.

For the patch-based matching between two successive frames, only keypoints on the projected surface of the model are kept and the filtering thresholds for the limbs are given by predicting the largest 2D translation of the points of each limb. For this purpose, the joint configuration is predicted by an autoregression

$$\widehat{x}_t = a_1 x_{t-1} + a_2 x_{t-2} + a_3 x_{t-3}, \quad (2)$$

where the coefficients a_i are computed from a training sequence. By approximating each limb with a cuboid, the maximal translation can be efficiently calculated for each view. Since the projected surface depends on the previous estimated pose, parts of the correspondences might belong to the background as demonstrated in Figure 1. Hence, corresponding features of the torso or of a limb with the same location are deleted if the average is above a threshold. An inaccurate pose can also yield a wrong limb association of a keypoint when self-occlusions occur. The confidence of a correspondence is therefore significantly reduced if neighboring pixels of the keypoint belong to two unconnected limbs. Filtered 2D-2D correspondences are shown in Figures 4 a) and c).

4. Pose Estimation

For estimating the pose, we seek for the transformation that minimizes the error of given 3D-2D correspondences denoted by pairs (X_i, x_i) of homogeneous coordinates. A suitable representation for articulated models are twists $\theta\hat{\xi}$ that relate to a 3D rigid motion by $M = \exp(\theta\hat{\xi})$ [3]. A joint j is modeled as zero-pitch screw about a given axis,

i.e., the joint motion depends only on the rotation angle θ_j. Hence, a transformation of a point X_i on the limb k_i influenced by n_{k_i} joints is given by

$$X_i' = M(\theta\hat{\xi})M(\theta_{\iota_{k_i}(1)})\dots M(\theta_{\iota_{k_i}(n_{k_i})})X_i, \qquad (3)$$

where the mapping ι_{k_i} represents the order of the joints in the kinematic chain.

Since each 2D point x_i defines a projection ray that can be represented as Plücker line $L_i = (n_i, m_i)$ [24], the error of a pair (X_i, x_i) is given by the norm of the perpendicular vector between the line L_i and the point X_i

$$\|\Pi(X_i) \times n_i - m_i\|_2, \qquad (4)$$

where Π denotes the projection from homogeneous coordinates to non-homogeneous coordinates. Using the Taylor approximation $\exp(\theta\hat{\xi}) \approx I + \hat{\xi}$ where I denotes the identity matrix, Equation (3) can be linearized. Hence, the sought transformation is obtained by solving the linear least squares problem

$$\frac{1}{2}\sum_i \left\| \Pi\left(I + \theta\hat{\xi} + \sum_j^{n_i} \theta_{\iota_{k_i}(j)}\hat{\xi}_j \right) \times n_i - m_i \right\|_2^2, \qquad (5)$$

i.e. by solving a system of linear equations.

4.1. Tracking

After the prediction, the final pose is estimated from correspondences that are extracted by patch-based and region-based matching as outlined in Figure 2. Since the number of correspondences from the contour varies according to scale, shape and triangulation of the object, we weight the summands in Equation (5) such that the influence between patches and silhouette is independent of the model.

We denote the set of correspondences from the original images, the synthetic image, and the contour by C_o, C_s, and C_c, respectively. The invariance is obtained by setting the weights for the equations for C_o and C_s in relation to C_c:

$$w_o = \alpha\frac{|C_c|}{|C_o|}, \quad w_c = 1, \quad w_s = \beta\, w_o. \qquad (6)$$

While the influence of the image-based patches and the contour is controlled by α independent of the number of correspondences, the weight w_s reflects the confidence in the matched patches between the synthesized and original image that increases with the number of matches $|C_s|$ relative to $|C_o|$. Since illumination differences between the two images entail that $|C_s|$ is less than $|C_o|$, β compensates for the difference, *c.f.* Figures 4 a) and c). For the experiments, we set $\alpha = 0.2$ and $\beta = 10.0$.

To avoid that the system of linear equations (5) becomes under-determined for small and homogeneous body parts, we add a low weighted regularization term that penalizes the deviation of a joint angle θ_j from the predicted pose (2). Self-intersections are prevented by learning the physical constraints of the human skeleton from training data similar to [5] where the probability of a pose p_{pose} is estimated by a Parzen-Rosenblatt estimator with Gaussian kernels over a small set of skeleton configurations. Since the dependency between the joints of the head, the upper and the lower body is low, the sample size is reduced by splitting p_{pose} up into three independent probabilities p_{pose}^{head}, p_{pose}^{upper} and p_{pose}^{lower}, respectively. Indeed, we use only 200 samples of the CMU motion database [8].

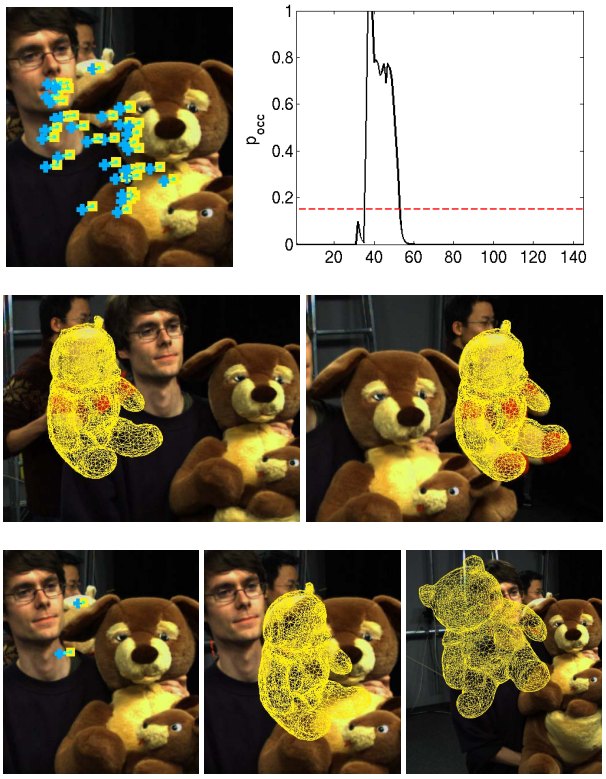

Figure 5. **From left to right. Row 1:** *a)* Most features belong to the occluding object (frame 44). The bear (target) is moving from left to right and the kangaroo from right to left. *b)* Probability of an occlusion in the shown view for a sequence with 150 frames. **Row 2:** *c)* The occlusion is correctly detected (frame 36). *d)* After the occlusion, the probability drops below the threshold 0.15 and the object is still correctly tracked (frame 52). **Row 3:** Frame 44. *e)* Almost all wrong matches are removed. *f)* Estimate after removing wrong matches. *g)* Estimate without occlusion handling.

4.2. Occlusion

Since patch-based matching between two successive frames is prone to occlusions, it requires the removal of correspondences not belonging to the target, see Figure 5 a). In

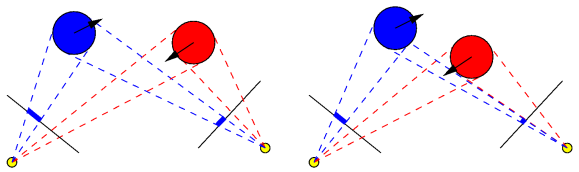

Figure 6. Occlusions are detected by recognizing changes of the projected surfaces. **From left to right.** *a)* The scene contains two moving objects captured by two cameras. The target object *(blue)* is so far not occluded by the unknown object *(red)*. *b)* Since the target is now occluded in the right camera view, the visible area of the target is much smaller than in the previous frame. The ratio of the covered areas between the left and the right image plane has also changed.

our analysis-by-synthesis framework, occluded patches are detected by comparing the original image with the synthesized image. For this purpose, the patches are mapped into the CIELab color space that mimics the human perception of color differences. To calculate the cross-correlation of color images [21], we represent each pixel with Lab color values as quaternion by $Li + aj + bk$. A correspondence is then labeled as occluded if the difference between the mean of the patch on the original image $P^o = \{p_1^o, \ldots, p_n^o\}$ and the patch on the synthesized image $P^s = \{p_1^s, \ldots, p_n^s\}$ is large or if the normalized cross correlation

$$NCC = \frac{\left|\sum_{i=1}^{n} \tilde{p}_i^o \overline{\tilde{p}_i^s}\right|}{\sqrt{\sum_{i=1}^{n} \tilde{p}_i^o \overline{\tilde{p}_i^o}} \sqrt{\sum_{i=1}^{n} \tilde{p}_i^s \overline{\tilde{p}_i^s}}} \tag{7}$$

is below a given threshold, where $\tilde{p}_i^o = p_i^o - \frac{1}{n}\sum_k p_k^o$ and $\overline{\tilde{p}_i^o}$ denotes the conjugate. An example for eliminating occluded patches is shown in Figures 5 a) and e).

To make the removal of patches more efficient, it is only performed when occlusions are detected for a camera view which is illustrated in Figure 6. When the target becomes occluded, the visible area of the projected surface gets smaller. By observing changes of the covered area for one view and the ratio between all views, occlusions can be detected. Since the visible areas of the projections cannot be measured, we use the number of matches as indicator, *i.e.* the difference of the absolute and relative number of matches between two successive frames for each view v:

$$\Delta_{abs}^{v,t} = |C_o^{v,t}| - |C_o^{v,t-1}| \tag{8}$$

$$\Delta_{rel}^{v,t} = \frac{|C_o^{v,t}|}{\sum_u |C_o^{u,t}|} - \frac{|C_o^{v,t-1}|}{\sum_u |C_o^{u,t-1}|}. \tag{9}$$

While these numbers indicate the beginning of an occlusion, the occluded area is measured by the number of occluded patches $|C_{occ}^{v,t}|$ relative to all matches $|C_o^{v,t}|$. Based on these observations, we propose a recursive model for the probability of an occlusion at time t:

$$p_{occ}^{v,t} = \frac{|C_{occ}^{v,t}|}{|C_o^{v,t}|} - f(\Delta_{abs}^{v,t}) - f(\Delta_{rel}^{v,t}) + \frac{2}{5} p_{occ}^{v,t-1} - \frac{1}{2}, \tag{10}$$

where $p_{occ}^{v,t}$ is truncated to the interval $[0, 1]$ and $f(x) = x$ if $x < -0.2$ else zero. The function f ensures that only significant changes of at least 20% are taken into account. Using this model, the detection and removal of occluded patches is only performed when the probability is higher than 0.15, marked as the dashed line in Figure 5 b).

4.3. Initialization

Unsupervised initialization is important for applications since an initial pose is typically not given. Since neither a predicted pose nor a shape prior is available, we estimate the pose from the textured model assuming that the object is observable. To this end, some initial views are preliminarily rendered by rotating the textured object with a fixed joint configuration, *e.g.* the same as used for the texture acquisition, extract the features, and store them together with the mean values of the patches $\bar{P}^s$ and the corresponding pose parameters. For initialization, the extracted keypoints for the first frame are matched with the database and the best initial view is selected for estimating the pose. The obtained correspondences with mean values $\bar{P}_i^f$ and $\bar{P}_i^s$ are weighted by $|\bar{P}_i^f - \bar{P}_i^s|^{-2}$ for stabilizing the estimation.

5. Results

For evaluating the performance of our approach, we captured several scenes with different objects by 3–5 synchronized and calibrated cameras with 25 frames per second and resolution of 1004×1004 pixels. The 3D models were acquired by a 3D scan and the images for the texture acquisition were taken from a sequence where lighting conditions and camera positions differed from the test sequences.

Row 1 of Figure 8 shows some results for a sequence with a stuffed bear tracked using a rigid model. The bear with non-trivial shape is tossed by a human – the second moving object – and rotates in the air by more than 180 degrees. The scene contains background clutter and is captured by 3 cameras. The occlusion detection is demonstrated in Figure 5. The stuffed kangaroo moves from right to left and occludes the stuffed bear moving from left to right such that the occluded target and the occluding object are moving at the same time. The occlusion between the frames 36 and 52 is correctly recognized and the pose is accurately estimated during the entire sequence whereas the tracking fails without an occlusion handling as shown in row 3 of Figure 5. For more results with occlusions, we refer to the supplementary video.

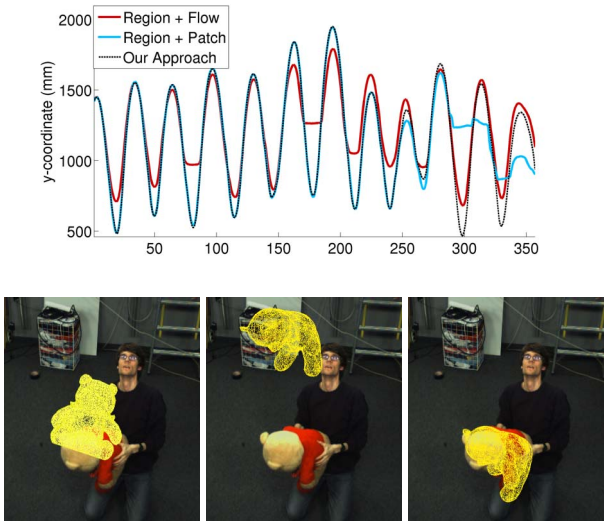

Figure 7. Comparison. **Row 1:** Estimated y-coordinate of the bear for the sequence shown in row 1 of Figure 8. The approaches that rely only on multi-cue integration cannot handle the drift. **Row 2:** Frame 298. **From left to right:** *a)* Region-based matching with optical flow [4]. *b)* Region-based matching with PCA-SIFT. *c)* Our approach estimates the pose without drift.

To compare our method with multi-cue approaches that were proposed for rigid objects, we used the sequence corresponding to row 1 of Figure 8 and plotted the estimates for the y-coordinate in Figure 7. We applied the same level-set segmentation for all methods to make a fair comparison. While the approaches that combine region-based matching with optical flow [4] or patch-based matching cannot prevent an accumulation of estimation errors over time, our method tracks the stuffed bear accurately over the entire sequence. It demonstrates that our framework solves the drift problem better than approaches that rely only on multi-cue integration.

The lower part of a human body was tracked using an articulated model with 18 DoF. As one can observe from the images with the projected meshes of the estimates in row 2 of Figure 8, the sequence recorded with 4 cameras is very challenging for a tracker. The movement is fast where the velocity and the direction change rapidly. In addition, self-occlusions occur since the legs are frequently crossed. We also tracked a full human body using an articulated model with 30 DoF. In rows 3 and 4 of Figure 8, estimates for 2 of 5 views are shown. The sequence with a human walking in a circle contains several difficulties. Self-occlusions occur since the arms are close to the body and the segmentation is hindered by clutter – particularly due to cables and metallic pipes –, shadows, and the similarity between the dark color of the sports suit and the background. Some body parts like the hands are furthermore small and homogeneous yielding only few correspondences from patch-based matching.

For a quantitative error analysis, we applied our approach to the `HumanEva-II` dataset [23] and measured the absolute 3D tracking error. The available model is not perfect since it does not contain the clothing of the subject $S4$ wearing a white T-shirt and blue jeans. The texture was acquired from the first frame and the available silhouettes were treated as an additional channel for the segmentation. Even though the surface of the object is rather homogeneous, we achieve accurate estimates as shown in Figure 9. Since the set-up and movement of the sequence, namely walking in a circle, is similar to the one used in [1], we compare the results in Table 1. Our implementation requires 7.6 seconds per image which is faster than the 90 seconds reported in [1].

	Our approach	*RoAM* body model [1]
error (mm)	36.16 ± 9.12	> 60

Table 1. Our framework performs significantly better than a statistical appearance model for human motion capture.

6. Summary

We have presented a model-based tracking framework for solving the drift-problem for rigid and articulated objects. An occlusion detection, which evaluates the probability of an occlusion, observes significant changes of the visible area of the projected surface during the sequence and initiates a recognition of occluded patches by comparing the original image with a synthesized image, if it is necessary. Since the synthesized image also provides accurate correspondences, an accumulation of estimation errors is prevented. By combining the complementary concepts of region and patch-based matching, both structured and homogeneous body parts can be tracked. A comparison with other model-based approaches for rigid objects has revealed that the proposed method handles the drift problem better. Our experiments have demonstrated that our framework is not restricted to a single rigid object but tackles the drift problem also for multiple moving objects and humans in challenging scenes containing fast movements, occlusions, and clutter. Although our framework benefits from objects with structured surfaces and accurate 3D models, a quantitative error analysis for the `HumanEva-II` dataset has shown that we still achieve accurate results when these assumptions are not completely satisfied. Indeed, the tracking error is significantly lower than the one that is obtained by a statistical appearance model for human motion capture. In general, our framework does not require stronger assumptions than any other model-based approach.

[0]The research was funded by the Max Planck Center VCC and the Cluster of Excellence on Multimodal Computing and Interaction.

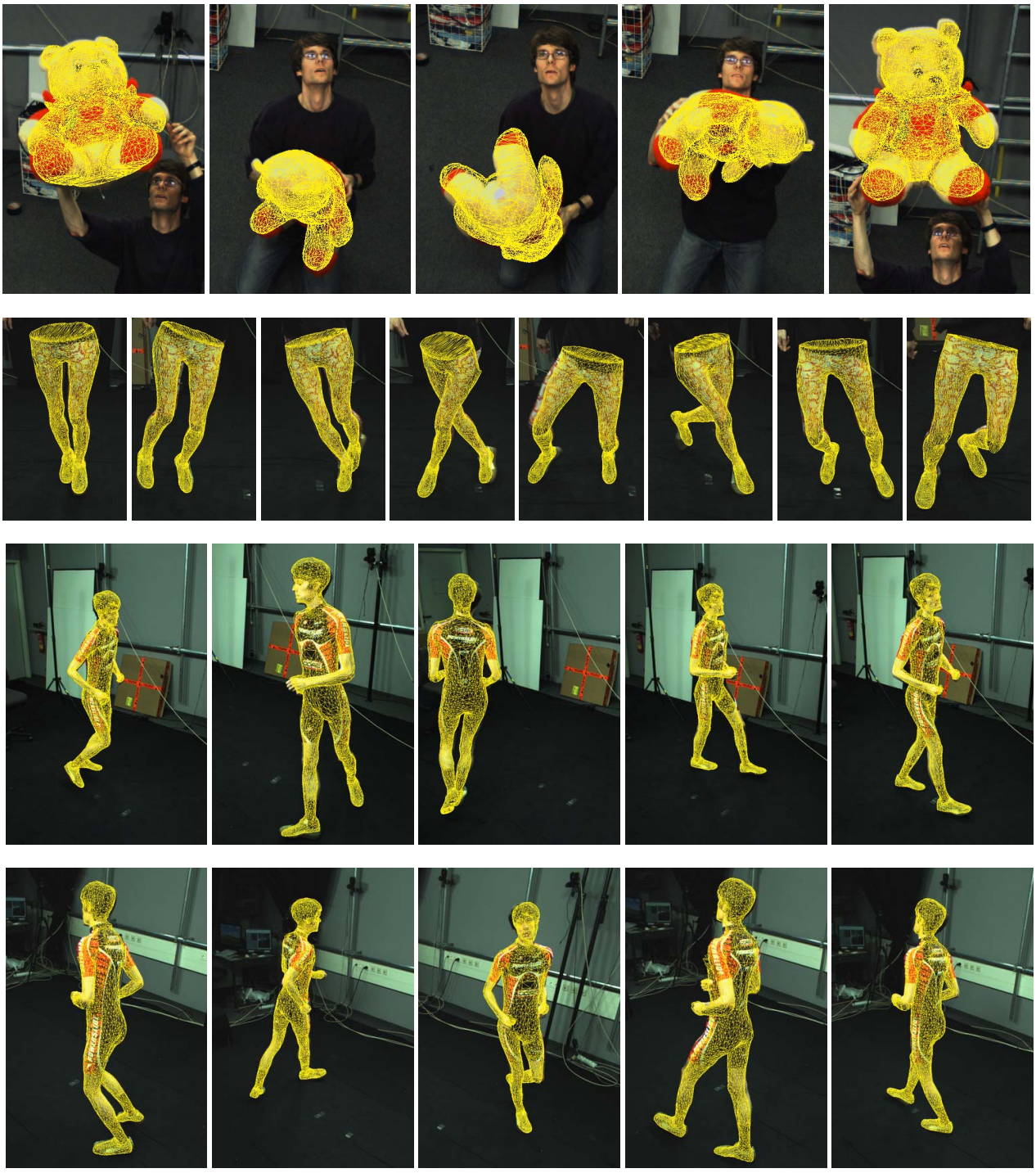

Figure 8. Estimates for three different sequences. **Row 1:** The bear is tossed and rotates (360 frames). One of three views for frames 60, 115, 180, 235, and 315. **Row 2:** Complex and fast movements of the legs including many self-occlusions (400 frames). One of four views for frames 45, 90, 135, 180, 225, 270, 315, and 360. **Row 3, 4:** Full human body walking in a circle with clutter (205 frames). Two of five views for frames 35, 70, 105, 140, and 175.

References

[1] A. Balan and M. Black. An adaptive appearance model approach for model-based articulated object tracking. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 758–765, 2006. 2, 6

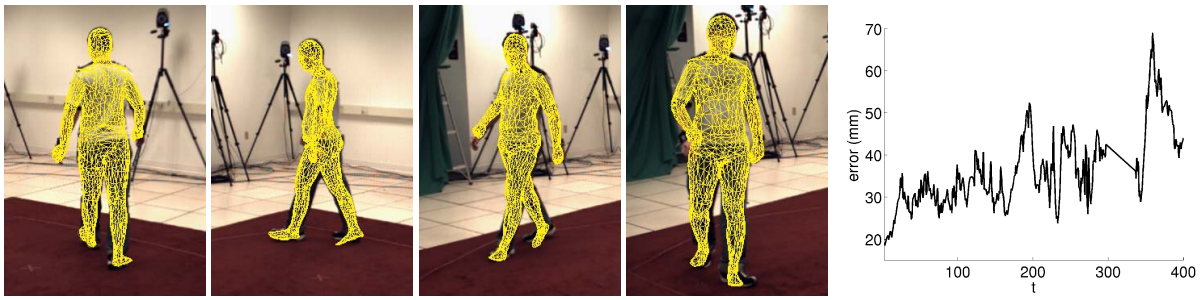

Figure 9. Quantitative error analysis for subject $S4$ of `HumanEva-II`. Estimates for frames 80, 160, 240, and 320. The frames $298-335$ are neglected for the error analysis since the ground truth is corrupted for these frames.

[2] M. Bray, P. Kohli, and P. Torr. Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *European Conf. on Comp. Vision*, pages 642–655, 2006. 2

[3] C. Bregler, J. Malik, and K. Pullen. Twist based acquisition and tracking of animal and human kinematics. *Int. J. of Computer Vision*, 56(3):179–194, 2004. 2, 3

[4] T. Brox, B. Rosenhahn, D. Cremers, and H.-P. Seidel. High accuracy optical flow serves 3-d pose tracking: Exploiting contour and flow based constraints. In *European Conf. on Comp. Vision*, pages 98–111, 2006. 1, 2, 6

[5] T. Brox, B. Rosenhahn, U. Kersting, and D. Cremers. Nonparametric density estimation for human pose tracking. In *Pattern Recognition*, volume 4174 of *LNCS*, pages 546–555. Springer, 2006. 4

[6] P. Burt and E. Adelson. A multiresolution spline with application to image mosaics. *ACM Trans. on Graphics*, 2(4):217–236, 1983. 2

[7] M. Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *IEEE Trans. on Patt. Analysis and Machine Intell.*, 22(4):322–336, 2000. 2

[8] CMU. Graphics lab motion capture database. http://mocap.cs.cmu.edu/. 4

[9] D. Decarlo and D. Metaxas. Optical flow constraints on deformable models with applications to face tracking. *Int. J. Computer Vision*, 38(2):99–127, 2000. 1

[10] J. Gall, B. Rosenhahn, and H.-P. Seidel. Clustered stochastic optimization for object recognition and pose estimation. In *Pattern Recognition*, volume 4713 of *LNCS*, pages 32–41. Springer, 2007. 2

[11] D. Gavrila and L. Davis. 3-d model-based tracking of humans in action: a multi-view approach. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 73–80, 1996. 1

[12] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 506–513, 2004. 3

[13] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 775–781, 2005. 2

[14] V. Lepetit, J. Pilet, and P. Fua. Point matching as a classification problem for fast and robust object pose estimation. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 244–250, 2004. 2

[15] H. Li, P. Roivainen, and R. Forcheimer. 3-d motion estimation in model-based facial image coding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(6), 1993. 1

[16] D. Lowe. Object recognition from local scale-invariant features. In *Int. Conf. on Computer Vision*, pages 1150–1157, 1999. 3

[17] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Int. Joint Conf. on Artificial Intelligence*, pages 674–679, 1981. 1

[18] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 257–263, 2003. 1, 3

[19] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int. J. of Computer Vision*, 67(2):141–158, 2006. 1

[20] B. Rosenhahn, T. Brox, and J. Weickert. Three-dimensional shape knowledge for joint image segmentation and pose tracking. *Int. J. of Computer Vision*, 73(3):243–262, 2007. 2, 3

[21] S. Sangwine and T. Ell. Hypercomplex auto- and cross-correlation of color images. In *IEEE Int. Conf. on Image Processing*, pages 319–322, 1999. 5

[22] J. Shi and C. Tomasi. Good features to track. In *IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 593–600, 1994. 1

[23] L. Sigal and M. Black. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, 2006. 6

[24] J. Stolfi. *Oriented Projective Geometry: A Framework for Geometric Computation*. Academic Press, Boston, 1991. 4

[25] C. Theobalt, J. Carranza, M. Magnor, and H.-P. Seidel. Combining 3d flow fields with silhouette-based human motion capture for immersive video. *Graphical Models*, 66(6):333–351, 2004. 2

[26] R. Zayer, C. Rössel, and H.-P. Seidel. Discrete tensorial quasi-harmonic maps. In *Int. Conf. on Shape Modeling and Applications*, pages 276–285, 2005. 2

[27] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int. J. of Computer Vision*, 13(2):119–152, 1994. 3