Efficient and Robust Shape Matching for Model Based Human Motion Capture

Gerard Pons-Moll, Laura Leal-Taixé, Tri Truong, and Bodo Rosenhahn

Leibniz University, Hannover, Germany

Abstract. In this paper we present a robust and efficient shape matching approach for Marker-less Motion Capture. Extracted features such as contour, gradient orientations and the turning function of the shape are embedded in a 1-D string. We formulate shape matching as a Linear Assignment Problem and propose to use Dynamic Time Warping on the string representation of shapes to discard unlikely correspondences and thereby to reduce ambiguities and spurious local minima. Furthermore, the proposed cost matrix pruning results in robustness to scaling, rotation and topological changes and allows to greatly reduce the computational cost. We show that our approach can track fast human motions where standard articulated Iterative Closest Point algorithms fail.

1 Introduction

Markerless motion capture is an active field of research [2, 3, 9, 10, 23]. The high dimensionality of the state space and the inherent depth ambiguities make estimating 3D motion from 2D images a difficult and interesting problem. The integration of priors learned from training data is now a very popular approach to mantain robustness in difficult conditions [13, 19–22, 26]. Although human pose estimation benefits from learned priors, many applications require a general solution without imposing strong assumptions on the dynamics of the activity to be captured. The majority of algorithms are generative and model based, in which a surface mesh of the subject is matched with 2D image observations. Generative approaches aim at modelling the likelihood with a cost function that measures how well the model explains the image observations. Local optimization (LO) methods estimate the pose by iteratively linearizing the cost function to find a descent direction. Here, recovery from false local minima is a major issue. To overcome such limitations particle based global optimization algorithms have been proposed [9, 10]. However, this last group of approaches, while robust, are computationally very expensive and do not provide a smooth and temporaly consistent motion like local approaches. It has been reported that (LO) can easily get trapped in local minima during fast motions. One reason for that is that correspondences between model and image observations are typicaly obtained with variations of the well known *Iterative Closest Point* (ICP) algorithm [4, 7, 8, 19, 25]. We argue that this practice has led LO algorithms to the inferior performance for capturing highly dynamic activities. In fact, provided with the correct correspondences, LO converges to the correct solution in almost all the cases, even for large displacements.

In this paper, we show how the performance of standard LO is greatly improved by employing a robust and efficient model image association algorithm based on bipartite

R. Mester and M. Felsberg (Eds.): DAGM 2011, LNCS 6835, pp. 416–425, 2011.

graph matching. In order to increase the robustness and decrease the computational cost of the matching problem, extracted shape features such as the turning function of contours and gradients are used to reduce the space of possible associations. By representing the contours as strings, upper and lower bounds for the matches are found using the Longest Common Subsequence (LCS) algorithm. This enables robust tracking even for highly dynamic activities as we will show in the experiments.

2 Related Work

Shape matching is a rich sub-field of computer vision research in itself (see for example [5, 27]). ICP is one of the most popular algorithms for finding correspondences, mainly due to its simplicity. Nonetheless, ICP gets trapped in local minima for sequences with fast motions due to the fact that only the localy closest points are considered. Richer shape descriptors such as Shape Context [15, 29] or Chamfer distance [9] reduce ambiguities in the matching costs. The advantage of Shape Context as used in [15, 29] is that it uses a global optimization algorithm, namely the Hungarian matching algorithm, which provides a globally optimal assignment. This property is particularly useful for fast motions, but unfortunatelly global matching is computationaly very expensive. Contour-based matching exploits the order of the points to improve data association [1]. As presented in [6, 14], shape contours can be expressed as strings, and therefore, the shape matching problem can be solved using string matching methods, which are fast and can be efficiently implemented using dynamic programming. In the context of human pose estimation, sophisticated shape descriptors and matching algorithms have been used in discriminative approaches where the goal is to learn direct mappings from shape or image features to the pose space [2, 3, 11, 15, 21].

However, few works (*e.g.* [24, 29]) have focused on integrating robust shape matching constrains in a generative model based pose estimation algorithm. The main reason is the high complexity of optimal assignment algorithms. As we will show in this work, rich shape descriptors can be used not only to resolve ambiguities in the matching process but also to reduce the computational complexity. This enables us to use a global shape matching method to feed a model based tracker with robust correspondences. Hence, the resulting tracker has the desirable properties of global optimization algorithms such as recovery from tracking failures with reasonable computational complexity.

3 Tracking System

We model human motion by a skeletal kinematic chain containing $N = 22$ joints that are connected by rigid bones. The global position and orientation of the kinematic chain are parameterized by a twist $\xi_0 \in \mathbb{R}^6$ [16]. Together with the joint angles $\Theta := (\theta_1 \ldots \theta_N)$, the configuration of the kinematic chain is fully defined by a $D = 6 + N$-dimensional vector of pose parameters $\mathrm{X} = (\xi_0, \Theta)$. We assume here for simplicity that all joints are modelled by concatenating 1 DoF revolute joints, for a description of the parameterization using free axes of rotation to model *ball joints* we refer the reader to [18]. Let $\mathcal{J}_i \subseteq \{1, \ldots, n\}$ be the ordered set of parent joint indices of the i-th bone. The absolute rigid motion $\mathbf{G}_i^{TB}$ of the bone is given by concatenating the global transformation matrix $\mathbf{G}_0 = \exp(\hat{\xi}_0)$ and the relative rigid motions matrices $\mathbf{G}_i$ along the chain by

$$\mathbf{G}_i^{TB} = \mathbf{G}_0 \prod_{j \in \mathcal{J}_i} \mathbf{G}_i = \mathbf{G}_0 \prod_{j \in \mathcal{J}_i} \exp(\theta_j \hat{\xi}_j). \tag{1}$$

where $\exp(\theta_j \hat{\xi}_j)$ is the exponential map of the j-th joint and ξ_j is the constant twist of the j-th joint in the chain. A surface mesh of the actor is attached to the kinematic chain by assigning every vertex of the mesh to one of the bones. Let $\bar{\mathbf{p}}$ be the homogeneous coordinate of a mesh vertex $\mathbf{p}$ in the zero pose associated to the i-th bone. For a given pose X, the vertex in a rest position $\bar{\mathbf{p}}$ is transformed using $\bar{\mathbf{p}}(X) = \mathbf{G}_i^{TB} \bar{\mathbf{p}}$.

In order to find correspondences between model points and image features we project the mesh points belonging to the occluding contour $\mathbf{p}_i \in \mathcal{O}$ obtaining a set of 2D projections $\hat{\mathbf{r}}_i \in \mathcal{M}$. Then we match the model projections $\hat{\mathbf{r}}_i \in \mathcal{M}$ to the image contour points $\mathbf{r}_j \in \mathcal{I}$ using the algorithm explained in Sect. 4. Given a set of 2D-2D correspondences we minimize the sum of squared distances between the 3D counter part of the model projections $\mathbf{p}_i$ and the projection rays L_i casted by the 2D image contour points $\mathbf{r}_i$. Let $L_i = (\mathbf{n}_i, \mathbf{m}_i)$ be the Plücker coordinates of the line corresponding to the image point $\mathbf{r}_i$. Then, the cost function for N correspondences can be written as

$$e(\mathrm{X}_t) = \sum_i^N \|\mathbf{e}_i(\mathrm{X}_t)\|^2 = \sum_i^N \|\mathbf{p}_i(\mathrm{X}_t) \times \mathbf{n}_i - \mathbf{m}_i\|^2 \tag{2}$$

where the scalar $e(\mathrm{X}_t) \in \mathbb{R}$ is the total error and $\mathbf{e}_i(\mathrm{X}_t) \in \mathbb{R}^3$ is the individual error associated with the i-th correspondence. To minimize Eq. (2) we use the Levendberg Marquadt algorithm. Let $\mathbf{e} = (\mathbf{e}_1, \ldots \mathbf{e}_N) \in \mathbb{R}^{3N}$ dennote the vector valued error function containing the individual correspondence errors. Then at each iteration the descent step $\Delta\mathrm{X}$ is found as

$$\Delta\mathrm{X} = -(\mathbf{J}^T \mathbf{J} + \mu \mathbf{I})^{-1} \mathbf{J}^T \mathbf{e} \tag{3}$$

where $\mathbf{J} \in \mathbb{R}^{3N \times D}$ is the analytical Jacobian matrix of the vector valued error function w.r.t the pose parameters $\mathbf{J} = \frac{\Delta \mathbf{e}}{\Delta \mathrm{X}}$ and μ is the adaptive damping parameter of the Levendberg Marquadt (LM) algorithm. As with any local method LM can get trapped in local minima. Fortunately, provided with the correct correspondences and adequate adaptive damping parameter μ it converges to the correct solution even for fast motions as we will show in the experiments.

4 Motion Capture with String Matching

To minimize Eq. (2), we must find correspondences between the set of projected mesh points $\hat{\mathbf{r}}_i \in \mathcal{M}$ and the set of contour points $\mathbf{r}_i \in \mathcal{I}$ of the image. We formulate the shape matching as a Linear Assignment Problem and propose to use Dynamic Time Warping on the string representation of the contours to discard semantically dissimilar matchings, thereby greately reducing computational time and increasing robustness to scaling, rotation, holes and topological changes.

4.1 Linear Assignment Problem

Let us define a weighted bipartite graph $G = (V, E)$, where its vertexes (or nodes) are partitioned into two distinct sets: the projected points of the occluding contour of the mesh $\mathcal{M}$ and the contour image points $\mathcal{I}$. All the edges $(i, j) \in E$ of the graph connect a vertex from one of the vertexes sets to the other ($E \subseteq \mathcal{M} \times \mathcal{I}$). Each edge (i, j) has a weight or cost $C(i, j)$, computed using the Euclidean distance. The shape matching problem is then reduced to finding the maximum weighted bipartite matching, which can be formulated as a Linear Assignment Problem (LAP) by defining a set of flags $x_{i,j}$, which take the value 1 when nodes i and j are matched, and 0 otherwise. In this setting, the LAP is formulated as the minimization of the objective function:

$$\min \sum_{i,j} C(i,j)x_{i,j} \quad \text{subject to:} \quad \sum_{i} x_{i,j} = 1 \text{ for } i \in \mathcal{M} \quad \sum_{j} x_{i,j} = 1 \text{ for } j \in \mathcal{I}.$$

The algorithms used to solve the LAP can be classified into three categories, depending if they are based on: *maximum flow*, e.g. Hungarian algorithm; *Linear Programming*, e.g. Simplex algorithm; and the methods based on *shortest paths*, like the LAPJV presented by Jonker and Volgenant [12]. In this paper, we use the LAPJV algorithm since its notably faster than the commonly used Hungarian algorithm. Next, we present how we use DTW on 1D shape representations to prune the graph.

4.2 String Representation of Shapes

To map a contour $\mathcal{A}$ onto a 1D string we first parametrize it by the arclength s. Thereby, a contour $\mathcal{A}$ is represented by a set of n ordered points along the curve $\mathcal{A}(s) = (x(s), y(s))$, $s \in \{1 \dots n\}$ forming a polygon. As a 1D descriptor, we use the cumulative angle function, or turning function $\Theta_{\mathcal{A}}(s)$ of a polygon that measures the angle between the counterclockwise tangent and the x-axis as a function of the arclength s. To leverage the influence of noise and the number sampled points in the contour, we compute the turning function as the cumulative angle differences between robust contour gradients $\nabla I(s)$. Hence, the 1D string representation of the contour is:

$$\Theta_{\mathcal{A}}(n) = \sum_{s=0}^{n} \arccos\left(\frac{\nabla I(s) \cdot \nabla I(s+1)}{\|\nabla I(s)\| \cdot \|\nabla I(s+1)\|}\right)\left(\hat{\mathbf{z}} \cdot \frac{\nabla I(s) \times \nabla I(s+1)}{\|\nabla I(s)\| \cdot \|\nabla I(s+1)\|}\right),$$

where the gradient $\nabla I(s) = \nabla I(x(s), y(s))$ is computed at $x(s), y(s)$ on the silhouette image using Gaussian derivative filters. This results in a more robust and smoother version of the turning function. The second term, where $\hat{\mathbf{z}} = [\,0\;0\;1\,]^T$ is the unit vector in the z direction, simply keeps track of the turning direction taking the value +1 for right hand turns and -1 for left hand turns. Note that this representation is already translation invariant. Now, matching two contours $\mathcal{M}$ and $\mathcal{I}$ simplifies to a comparison between the corresponding strings which have in general different lengths $\Theta_{\mathcal{M}}(s)$, with $s \in \{1, 2 \dots n\}$, $\Theta_{\mathcal{I}}(t)$ with $t \in \{1, 2 \dots m\}$. To obtain a scale invariant solution we use the Dynamic Time Warping (DTW) which finds the optimal alignment between strings allowing non-linear deformations along the arclength dimension s as we will explain in Sect. 4.3. This further allows us to be robust to topology changes such as

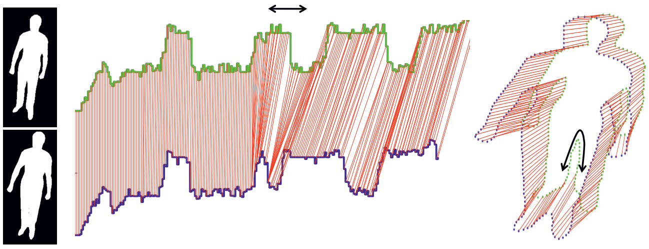

Fig. 1. Matching the turning function with the Longest Common Subsequence (LCS). The LCS allows the string to be compressed and stretched, which allows us to correctly match silhouettes with holes or disappering contours (like the contour between the legs marked with a black arrow).

self occlusions that often occur during tracking. To bring both strings into a common starting salient point, we find the region in both shapes with highest consequent similarity. The problem, which is known as Longest Common Consecutive Substring (LCCS) match, consists in finding the longest strings which are substrings of both $\Theta_{\mathcal{M}}(s)$ and $\Theta_{\mathcal{I}}(s)$ and can be efficiently solved using Dynamic Programming. The substring found is used to bring both sequences to a common starting point. As a result, the turning function is more robust against rotations (see Fig. 2(a)).

4.3 The Wagner-Fischer Algorithm

The most common problem we face when matching human silhouettes are topology changes such as disappearing contours, as shown in an example in Fig. 1. In the first frame, the legs are separated enough so that they can be distinguished in the silhouette. In the next frame though, the legs are too close to each other, and the contour that separates them suddenly dissapears. Intuitively, this means that one contour needs to be warped in a non-linear fashion to match another contour. Dynamic Time Warping (DTW) is a well-known technique to find an optimal alignment between two given sequences, allowing the sequences to be stretched or compressed in order to be better matched. When the sequences consist of discrete symbols, i.e., strings, the objective is to find the Longest Common Subsequence (LCS). The LCS algorithm used in this paper was proposed by R. Wagner and M. Fisher [28], and is based on the *edit distance*, also called the *Levenshtein distance*. Let $A = A_1, A_2 \dots A_m$ and $B = B_1, B_2 \dots B_n$ be two strings. The *Levenshtein distance* between A and B, $D(A, B)$ is computed in $O(m, n)$ time in a dynamic programming fashion:

$$D(i,j) = \begin{cases} D(i-1, j-1), & \text{if } A_i = B_j \\ \min \begin{cases} D(i-1, j-1) + 1, \\ D(i, j-1) + 1, \\ D(i-1, j) + 1, \end{cases} & \text{if } A_i \neq B_j \end{cases}$$

where $i = 1 \dots m$ and $j = 1 \dots n$. Once we have the edit distance matrix $D(A, B)$, a recursive function is used to find the assignments. Note that the assignments are not

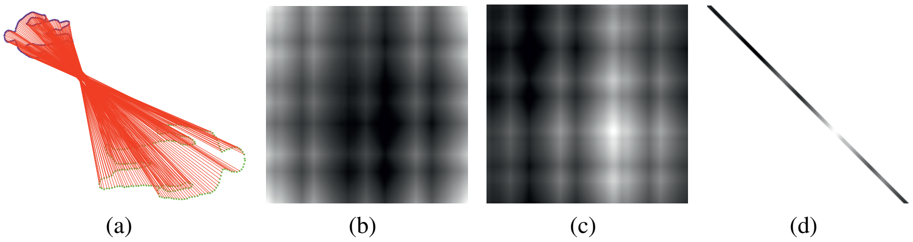

Fig. 2. (a) Matching of two rotated and scaled shapes with the proposed method. (b) Original distance matrix. (c) Distance matrix after LCCS. (d) Final distance matrix, only the values inside the LCS boundaries are computed; as we can see, most of the edges (in white) are erased, efficiently reducing the computational time and increasing the matching accuracy of the algorithm.

necessarily unique and they depend on the order how we run the LCS algorithm. Therefore, we run the LCS twice, once in each direction, and consider these assignments to be the upper and lower bound of the set of possible assignments. By erasing the elements of the cost matrix $C(i, j)$ in Eq. (4) which are outside these bounds, we reduce most of the edges, (Fig. 2(d)), therefore reducing computational time (see Section 5.1).

5 Experiments

In this section we evaluate our proposed algorithm by comparing it to a standard articulated ICP. For validation, we use the publicly available database (MPI08) [17], which contains a wide variety of human motions ranging from simple ones such as walking to really challenging ones such as lying down, throwing and non-scripted freestyle motions. The database is recorded in an indoor setup with 8 calibrated cameras. It consists of 4 subjects performing 14 different motion patterns. In total, more than 10 minutes of video footage are used for our validation study. Unless otherwise specified, we used 7 cameras for tracking and *left out* one frontal camera for validation. The overlap measure between the validation camera and the mesh silhouette projection is used as error metric. For a sequence of T frames the error measure is computed as

$$e = \frac{1}{T} \sum_{f=0}^{T} \left(1 - \frac{\mathcal{I}_{val}^{f} \cap \mathcal{I}_{templ}^{f}}{\mathcal{I}_{val}^{f} \cup \mathcal{I}_{templ}^{f}} \right) \tag{4}$$

where $\mathcal{I}_{val}^{f}$ and $\mathcal{I}_{templ}^{f}$ are the silhouette image of the validation camera, and the rendered model silhouette at frame f respectively.

5.1 Computational Time

We compare the computation time for solving the LAP using different 4 different methods, i.e. (1) Hungarian on the distance matrix, (2) Hungarian on the pruned cost matrix obtained with the method explained in Section 4, (3) LAPJV on the distance matrix

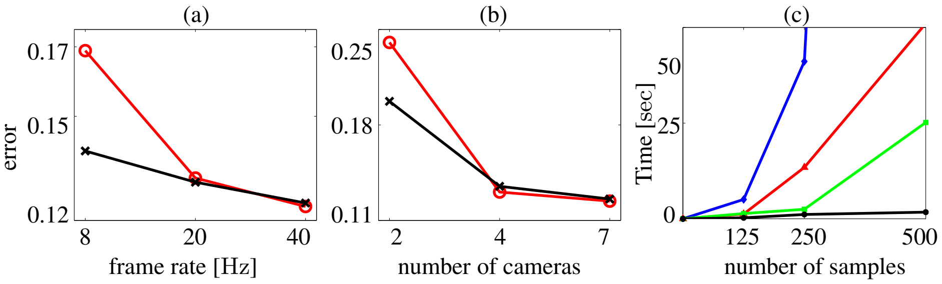

Fig. 3. (a) mean error vs frame rate of a walking sequence: ICP (red ●) vs. proposed (black ●) , **(b)** mean error vs the number of cameras: ICP (red ●) vs. proposed (black ●), **(c)** comparison of computation time for the methods: Hungarian (blue ●) , Hungarian+SM (red ●), LAPJV (green ●) and proposed (black ●)

and (4) our proposed method (LAPJV + pruned cost matrix). In Fig. 3(c) the computational time for matching is shown as a function of the number of sampled points in the contour. Our approach scales much better with the number of sampled points than the other 3 methods thanks to the graph pruning. In addition, the processing time per frame is comparable to that of a simple articulated ICP even though we use global matching.

5.2 ICP vs. Proposed Method

To test the robustness of the proposed approach to fast motions we tracked one of the walking sequences of the database with reduced frame rates. In Fig. 3 **(a)** we show the mean error as a function of the frame rate for the ICP and the proposed method. The proposed method outperforms ICP for low frame rates as ICP easily gets trapped in local minima during matching. Similar results are obtained when the number of cameras are reduced, see Fig. 3(b). In Fig. 5 the tracking error of ICP vs. our proposed method is

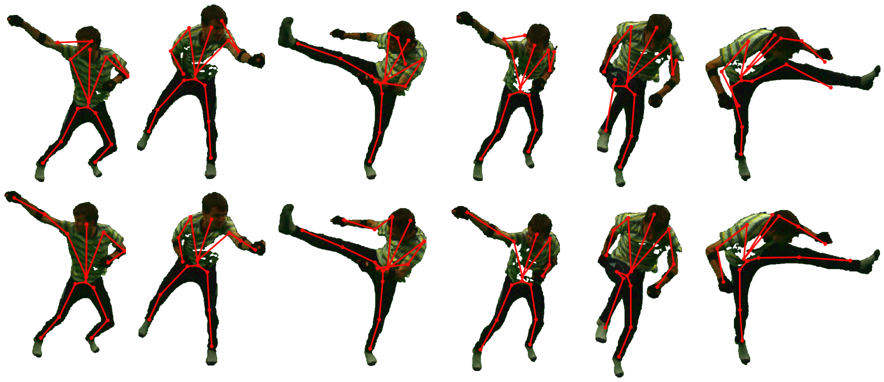

Fig. 4. Freestyle sequence: Top row (ICP) and bottom row (proposed)

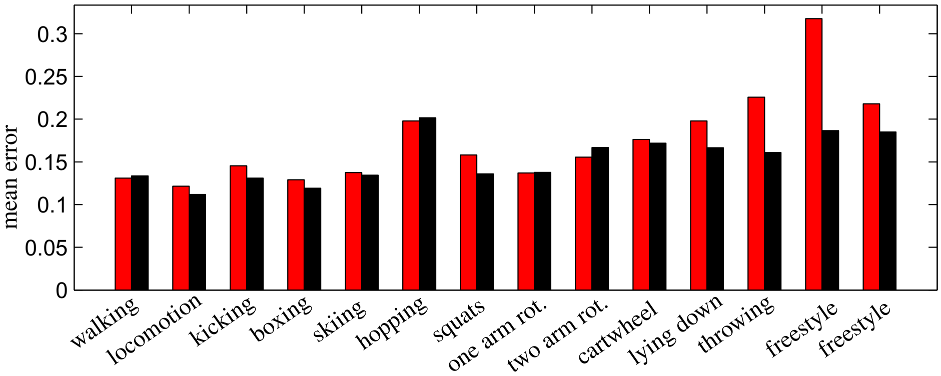

Fig. 5. Mean error for the sequences in the MPI08 database for methods (ICP) in red ● and (proposed) in black ●. Each of the 14 motion patterns was performed by 4 different subjects.

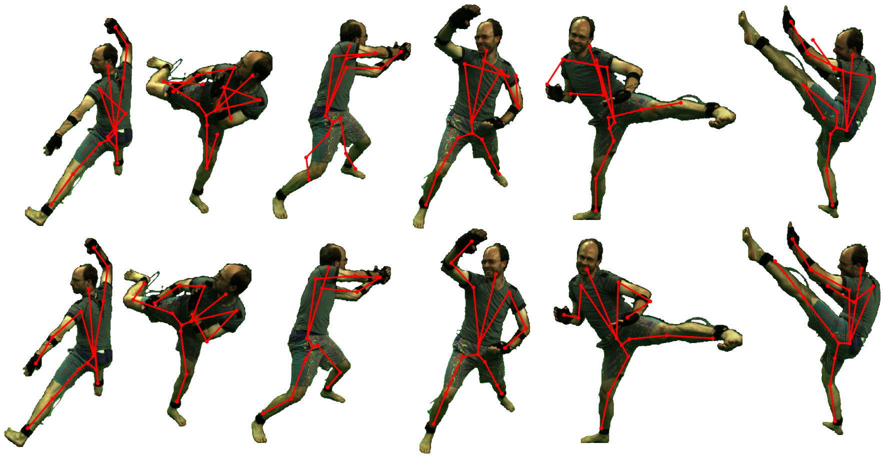

Fig. 6. Freestyle sequence for (ICP) top row and bottom row (proposed). The segmented image is shown with the reconstructed skeleton overlaid.

shown for all the 54 sequences present in MPI08 database. For every motion pattern we show the mean error of the 4 sequences corresponding to each actor. Our method performs better in almost all the sequences. A small improvement is achieved for simple motion patterns such as walking or locomotion because there ICP already performs very well. However, for complex motions such as throwing, lying down and freestyle we obtain much more accurate results. Several qualitative results showing the original segmented image with the reconstructed poses of ICP and the proposed method can be seen in Figs. 4 and 6. Finally, we show the reconstructions obtained our proposed method together with the mesh overlaid on the original images in Fig. 7.

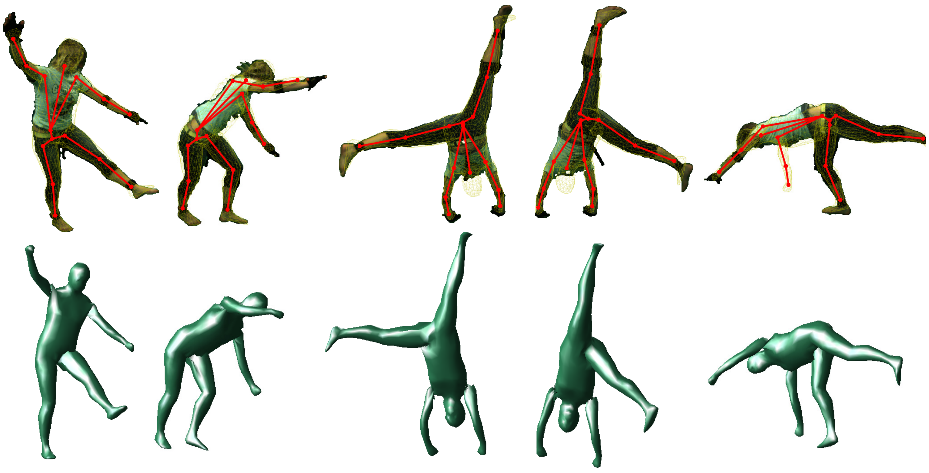

Fig. 7. Tracking results of a cartwheel sequence with our proposed method: top row original image with the model overlaid in yellow and bottom row the reconstructed pose. Even when the segmentation is not correct (see the arm missing in the last image), the pose is correctly recovered.

6 Conclusions

We have presented a robust shape matching approach based on strings for Markerless Motion Capture. Under a generative pose estimation algorithm, we define a Linear Assignment Problem to find correspondences between model projections and image features. Shape features such as contour and the turning function are used to represent the shapes as 1-D strings. Dynamic Time Warping is then used on the shape strings in order to find upper and lower bounds for the correct matches. The proposed cost matrix pruning effectively lowers the computational complexity and removes most of the non-optimalities typical of local data association algorithms. Quantitative and qualitative experiments show that our method outperforms the commonly used Iterative Closest Point (ICP) for sequences with fast motions. We also show that the proposed method allows tracking at a lowered frame rate, since it is more robust to scale and rotation and is able to deal with topological changes. In future work we will explore the use of string matching algorithms for consistent identification and localization of body parts.

References

1. Adamek, T., O'Connor, N.: Efficient contour-based shape representation and matching, pp. 138–143. MIR (2003)
2. Agarwal, A., Triggs, B.: Recovering 3D human pose from monocular images. TPAMI 28(1), 44–58 (2006)
3. Bo, L., Sminchisescu, C.: Twin gaussian processes for structured prediction. International Journal of Computer Vision (2010)
4. Bregler, C., Malik, J., Pullen, K.: Twist based acquisition and tracking of animal and human kinematics. IJCV 56(3), 179–194 (2004)

5. Bronstein, A., Bronstein, M., Bronstein, M., Kimmel, R.: Numerical geometry of non-rigid shapes. Springer-Verlag New York Inc., Secaucus (2008)
6. Bunke, H., Buhler, U.: Applications of approximate string matching to 2D shape recognition. Pattern Recognition 26(12), 1797–1812 (1993)
7. Corazza, S., Mündermann, L., Gambaretto, E., Ferrigno, G., Andriacchi, T.: Markerless motion capture through visual hull, articulated icp and subject specific model generation. IJCV 87(1), 156–169 (2010)
8. Demirdjian, D.: Combining geometric-and view-based approaches for articulated pose estimation, pp. 183–194 (2004)
9. Deutscher, J., Reid, I.: Articulated body motion capture by stochastic search. IJCV 61(2), 185–205 (2005)
10. Gall, J., Rosenhahn, B., Brox, T., Seidel, H.P.: Optimization and filtering for human motion capture. IJCV 87, 75–92 (2010)
11. Hofmann, M., Gavrila, D.: Multi-view 3d human pose estimation combining single-frame recovery, temporal integration and model adaptation. In: CVPR, pp. 2214–2221 (2009)
12. Jonker, R., Volgenant, A.: A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing 38(4), 325–340 (1987)
13. Lee, C., Elgammal, A.: Coupled visual and kinematic manifold models for tracking. IJCV (2010)
14. Marzal, A., Palazón, V.: Dynamic time warping of cyclic strings for shape matching. In: ICAPR, pp. 644–652 (2005)
15. Mori, G., Malik, J.: Recovering 3d human body configurations using shape contexts. TPAMI, 1052–1062 (2006)
16. Murray, R., Li, Z., Sastry, S.: A Mathematical Introduction to Robotic Manipulation. CRC Press, Baton Rouge (1994)
17. Pons-Moll, G., Baak, A., Helten, T., Müller, M., Seidel, H.P., Rosenhahn, B.: Multisensor-fusion for 3D full-body human motion capture. In: CVPR, pp. 663–670 (2010)
18. Pons-Moll, G., Rosenhahn, B.: Ball joints for marker-less human motion capture. In: WACV, pp. 1–8 (2009)
19. Rosenhahn, B., Brox, T.: Scaled motion dynamics for markerless motion capture. In: CVPR (2007)
20. Salzmann, M., Urtasun, R.: Combining discriminative and generative methods for 3d deformable surface and articulated pose reconstruction. In: CVPR (June 2010)
21. Shakhnarovich, G., Viola, P., Darrell, T.: Fast pose estimation with parameter-sensitive hashing. In: ICCV, pp. 750–757 (2003)
22. Sidenbladh, H., Black, M., Fleet, D.: Stochastic tracking of 3D human figures using 2D image motion. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 702–718. Springer, Heidelberg (2000)
23. Sigal, L., Balan, L., Black, M.: Combined discriminative and generative articulated pose and non-rigid shape estimation. In: NIPS, pp. 1337–1344 (2008)
24. Sminchisescu, C.: Consistency and coupling in human model likelihoods. In: FG (2002)
25. Sminchisescu, C., Triggs, B.: Covariance scaled sampling for monocular 3D body tracking. In: CVPR, vol. 1 (2001)
26. Urtasun, R., Fleet, D.J., Fua, P.: 3D people tracking with gaussian process dynamical models. In: CVPR, vol. 1, pp. 238–245 (2006)
27. Veltkamp, R., Hagedoorn, M.: State of the art in shape matching. Principles of visual information retrieval 87 (2001)
28. Wagner, R.A., Fischer, M.J.: The String-to-String Correction Problem. J. ACM 21(1), 168–173 (1974)
29. Zhao, X., Liu, Y.: Generative estimation of 3D human pose using shape contexts matching. In: Yagi, Y., Kang, S.B., Kweon, I.S., Zha, H. (eds.) ACCV 2007, Part I. LNCS, vol. 4843, pp. 419–429. Springer, Heidelberg (2007)