

# Loughborough University Institutional Repository

---

## *Dynamic segment-based sparse feature-point matching in articulate motion*

This item was submitted to Loughborough University's Institutional Repository by the/an author.

**Citation:** LI, B. and HOLSTEIN, H., 2002. Dynamic segment-based sparse feature-point matching in articulate motion. IN: IEEE International Conference on Systems, Man and Cybernetics: Conference Proceedings, Yasmine Hammamet - Tunisia, 6th-9th October, Vol. 4, pp.672-677

### **Additional Information:**

- This is the accepted manuscript version of the paper. © 2002 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

**Metadata Record:** <https://dspace.lboro.ac.uk/2134/20265>

**Version:** Accepted for publication

**Publisher:** © IEEE

**Rights:** This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: <https://creativecommons.org/licenses/by-nc-nd/4.0/>

Please cite the published version.

# Dynamic Segment-Based Sparse Feature-Point Matching in Articulate Motion

Baihua Li and Horst Holstein

University of Wales, Aberystwyth  
Department of Computer Science  
Aberystwyth, Ceredigion, SY23 3DB, UK  
bb100@aber.ac.uk, hoh@aber.ac.uk

*Abstract*—

We propose an algorithm for identifying articulated motion. The motion is represented by a sequence of 3D sparse feature-point data. The algorithm emphasizes a self-initializing identification phase for each uninterrupted data sequence, typically at the beginning or on resumption of tracking. We combine a dynamic segment-based hierarchical identification with a inter-frame tracking strategy for efficiency and robustness. We have tested the algorithm successfully using human motion data obtained from a marker-based optical motion capture (MoCap) system.

*Keywords*— articulated motion, articulated point pattern matching, non-rigid pose estimation, motion tracking and object recognition.

## I. INTRODUCTION

Tracking the position and identifying the pose of articulated objects has received growing attention in computer vision research in the past decade. The motivation of directing motion analysis of rigid objects towards non-rigid articulated objects, especially human motion, is driven by many potential applications, such as human-machine interfaces, biomedical motion analysis, the entertainment industry, video compression and indexing.

Numerous algorithms have been investigated for this task in the usual domain of color or intensity images [1]. Among these, some use only sparse feature point data [3]. For such point-feature trackers, tracking algorithms have been investigated extensively, based on assumptions such as smooth or small inter-frame motion or high-level knowledge related to a specific motion [4][5][8]. However, feature point identification, to know which point in an observed data corresponds to which point in its prototype model, still remains an open problem, especially at the start or recommencement of tracking. Currently, most tracking approaches deal with incremental pose estimation relying on manual initialization, or on an assumption of initial pose similarity to the model. Commercial marker-based optical motion capture (MoCap) systems [7], currently used in clinical gait analysis, sports studies, computer games and animation etc., provide an auto-labeling facility, but may fail in the task of identification at times.

The articulated motion we are considering describes segment-based jointed motion, such as vertebrate biological motion. The motion of each segment can be

considered as rigid or nearly rigid, but the motion of the whole object is high-dimensionally non-rigid. The number of feature points employed on an articulated object is normally sufficient to indicate its segment-based skeletal structure. In this case, the spatio-temporal information of an articulated motion is reduced to only a sequence of moving point data over time. As evidenced by Johansson's Moving Light Displays (MLDs) [6], humans can perceive high-dimensional non-rigid structure and motion solely on the sequence of unordered moving lights. Unfortunately, identifying these points and reconstructing their structure in real-world movements is inherently difficult for machine vision. Much of the extensive research on point pattern matching and pose estimation has been restricted to rigid, affine and projective point correspondence.

In this paper, we propose a general dynamic segment-based hierarchical point-matching (DSHPM) algorithm for the task of identification in 3D point-feature tracking systems. Therefore, our algorithm assumes the availability of 3D feature-point data obtained by stereo-vision techniques. We emphasize that the proposed algorithm can work as a self-initializing point-feature tracker. The algorithm considers the following aspects in the real world situation: 1) articulated structure requiring globally high-dimensional non-rigid but locally near-affine mappings, 2) limited distortion in each segment, 3) possible missing data (due to occlusion) and noise data, 4) independence of human initialization for identification and tracking, 5) unexpected and arbitrary poses in freeform movements.

## II. FRAMEWORK OF THE DSHPM ALGORITHM

The dynamic segment-based hierarchical point matching (DSHPM) algorithm aims to find the precise one-to-one matches between the points in an observed movement sequence and the points in the model. We present the framework of the algorithm in Fig.1. The algorithm includes several main modules: i) pre-tracking, ii) candidate-tables (CTs) based iterative segment matching, iii) motion-based verification, iv) dynamic identification and v) backward error correction.

### A. Segment-based skeleton model

To make the problem tractable, model-based approaches are commonly used to recover complex and unconstrained movement in a wide variety of circum-

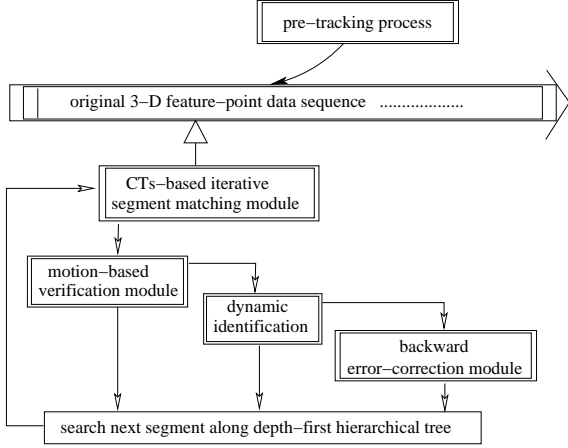


Fig. 1. Framework of the DSHPM algorithm.

stances [5][8]. When an articulated motion is described by a set of feature points, the geometrical structure of the articulated subject can be modeled by a segment-based skeleton. In our algorithm, the skeleton model is generated manually off-line using one frame of static-pose data that includes all feature points. This model consists of a set of identified feature points described by their 3D coordinates  $p_{s,i} \in \mathbb{R}^3$ , their labels  $L_{s,i}$ , and their grouping into a set of  $S$  segments:  $P = \{P_s | P_s = \{(p_{s,i}, L_{s,i}), i = 1, \dots, M_s\}, s = 1, \dots, S\}$ , each segment having  $M_s$  model points. Articulated segments share at least one model point at their joints.

To keep the articulated model general, we allow each segment of the model to undergo an independent affine transformation within the constraints of a jointed skeleton, and furthermore allow limited segmental distortion during the subject's freeform movements. We do not impose motion constraints such as feasible biological motion of a specific subject type.

### B. Observed motion data and point pre-tracking

The data for an observed movement of the subject are taken as a set of 3D-coordinates of unordered points frame by frame. When sample rate  $f_s$  satisfies the Nyquist sampling theorem, inter-frame translation of each point is considered small and smooth. That allows us to apply point pre-tracking by second-order extrapolation prediction before identification, as shown in Fig.1. In this way, the original unordered data may be presented by  $n = 1, \dots, N$  unidentified point-trajectories  $Q_n$  over time  $t = 1, \dots, T$ :

$$Q = \{Q_n | Q_n = \{q_n^t\}, \left\{ \begin{array}{l} q_n^t \in \mathbb{R}^3, t \in [t'_n, t''_n] \\ q_n^t \in [], \text{ otherwise} \end{array} \right\} \}.$$

Whenever a new point  $q_n$  appears at time  $t'_n$ , it is tracked and a new trajectory  $Q_n$  is generated. This trajectory may be broken at time  $t''_n$  due to occlusion or other source of data loss.

The point pre-tracking process allows the identification algorithm to begin at one key-frame  $t = T_K$ , in which ideally the number of non-null points is over 90% of the model points. After feature points identifica-

tion in the key-frame, the labeling will spread along pre-tracked trajectories. Integrating identification with tracking avoids an exhaustive frame-by-frame matching. Additionally, structure-from-motion (SFM) information becomes exploitable throughout the whole dynamic algorithm.

## III. THE DSHPM ALGORITHM

The flow chart of the DSHPM algorithm is shown in Fig.2.

### A. Pre-segmentation and the SFM matrix

Identification begins at a key-frame, whose time is denoted by  $* = T_K$ . Due to pre-tracking, trajectories are available in an interval around the key-frame. The first step in the DSHPM algorithm is to determine those point pairs that maintain relatively constant distances as possible intra-segmental points. This is done by calculating the Euclidean distance  $D_{(i,j)}^t = d(q_i^t, q_j^t)$  for observed point pairs  $(q_i^t, q_j^t)$  at frames  $t = * + k\Delta, k = 0, \pm 1, \dots, 5, \Delta = f_s/6$ . If the criteria:

$$D_{(i,j)}^* < 1.2 \times \max_s \max_{i,j} d(p_{s,i}, p_{s,j})$$

$$\frac{\max_k (D_{(i,j)}^{*+k\Delta}) - \min_k (D_{(i,j)}^{*+k\Delta})}{D_{(i,j)}^{*+k\Delta}} < 2 \times \max_s (\varepsilon_s) \quad (1)$$

are both satisfied, where  $\varepsilon_s$  is the maximum tolerable distortion rate on segment  $P_s$ , then we set  $SFM_{(i,j)} = 1$ , indicating that points  $q_i$  and  $q_j$  may belong to the same segment; otherwise we set  $SFM_{(i,j)} = 0$ .

### B. Candidate-tables (CTs) based iterative segment matching

Matching with the model at segment level is preferable to brute point-to-point searching, because articulated motion maintains geometric invariance in the segment. We perform local segment matching in two steps: CTs generation and CTs-based iterative matching[9][10].

#### B.1 Candidate-tables (CTs) generation

We first collect plausible points in candidate-tables, to reduce the search space.

**Step1:** Create a pivotal point sequence for segment  $P_s$ :

If no point has been identified in segment  $P_s$ , arbitrarily choose a pivot point  $p_{s,l} \in P_s$  and order the remaining  $p_{s,i} \in P_s$  by non-decreasing distance from the pivot, to define an ordered pivotal sequence for this segment.

**Step2:** Find candidates and generate CT:

Choose a match candidate  $q_k^*$  from the unidentified points  $\bar{Q}_M^*$  for the pivot  $p_{s,l}$ , and generate a column of match candidates for each non-pivotal element of the pivotal sequence. The resulting array is a candidate table for segment  $P_s$ . The candidate  $q_j^* \in \bar{Q}_M^*$  in the CT is chosen on account of a relaxed inter-point distance criterion and pre-segmentation information from the SFM matrix, as described in Eq.2.

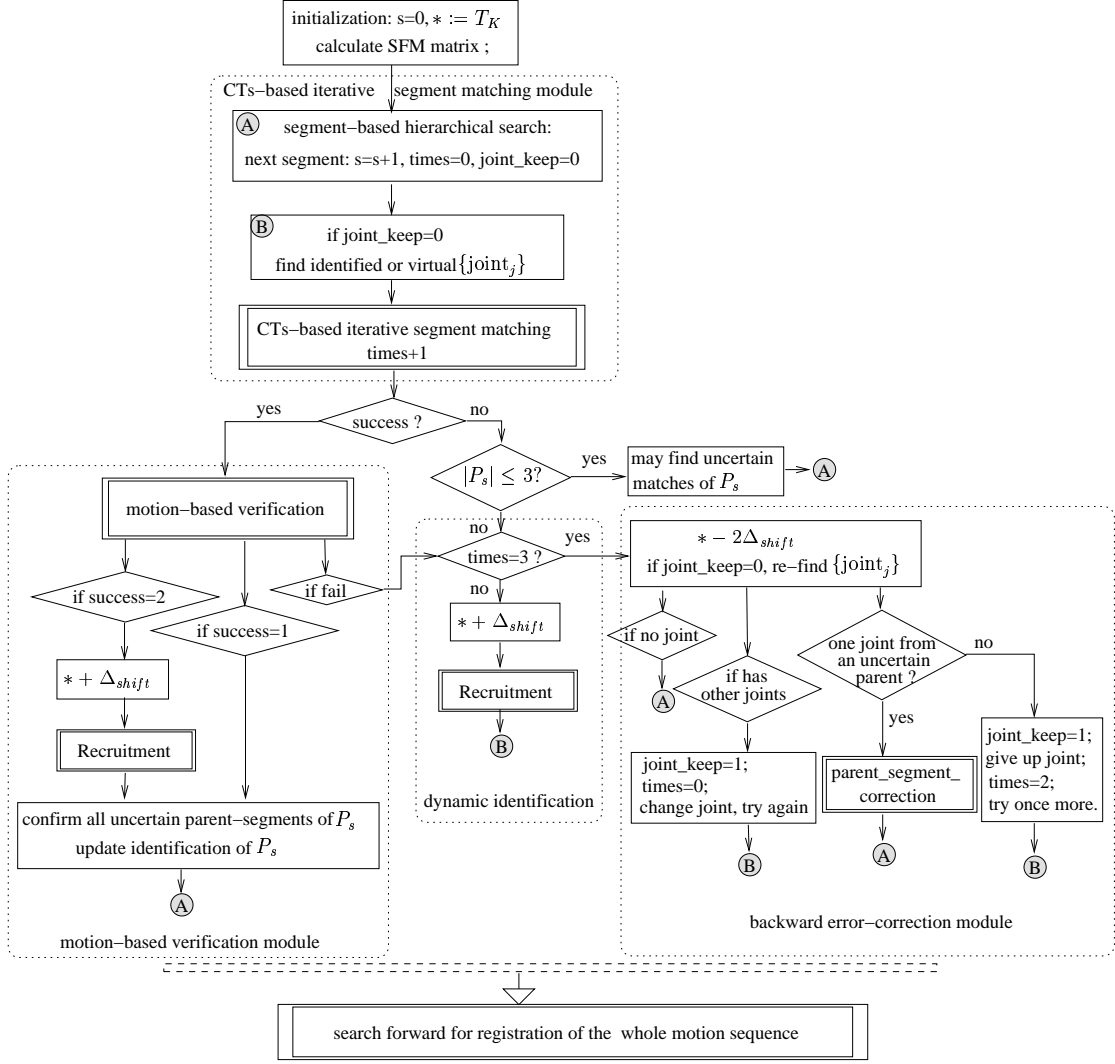


Fig. 2. Flow chart of the DSHPM algorithm.

$$\frac{|d(p_{s,l}, p_{s,i}) - d(q_k^*, q_j^*)|}{d(p_{s,l}, p_{s,i})} \Bigg|_{q_j^*: SFM_{(k,j)}=1} < 2\varepsilon_s \quad (2)$$

### Step3 : CTs optimization:

Each unmatched point  $q_k^* \in \bar{Q}_M^*$  is, in turn, a match candidate for the pivot  $p_{s,l}$ , and obtains a CT, but at most one of these tables includes a correct match of segment  $P_s$ . Heuristically, the CT with the correct match contains more candidates than other CTs. Therefore, to speed up the search procedure, we 1) discard those CTs with low numbers of candidates; 2) arrange priority of the rest based on the number of candidates included; 3) for each remaining CT, order the column of non-null candidates by increasing value of the distance ratio in Eq.2, thereby placing the best candidates towards the column head.

If a point, acting as a joint  $\{joint_j\}$  in the segment  $P_s$ , has already been identified during its parent-segment

identification, reasonably such a point is used as pivot. In this case, only one CT is generated for the segment  $P_s$ . This implies a large reduction of the search space.

### B.2 CTs-based iterative matching

In order to detect a correct segment match from among the generated CTs, we choose a CT from the prioritized order and take its most reasonable candidates to form an assumed correspondence with the pivotal sequence,  $\{(q_{s,i}^*, p_{s,i})\}'_M$  say. The case of fewer than 3 non-null matches is discussed in section III-C.2. If the number of non-null matches  $|\{(q_{s,i}^*, p_{s,i})\}'_M|$  is at least 3, then calculate the affine transformation  $[R'_s, T'_s]$  by the SVD-based motion estimate algorithm [2]. Satisfaction of the matching quality criterion in Eq.3,

$$\bar{\varepsilon}_s = \frac{\sum_{i=1}^{M_s} r_{s,i} \times \|q_{s,i}^* - R'_s p_{s,i} - T'_s\|}{\bar{l}_s \times \sum_{i=1}^{M_s} r_{s,i}} < \varepsilon_s \quad (3)$$

regards  $\{(q_{s,i}^*, p_{s,i})\}'_M$  as a partial match of  $P_s$ , otherwise the assumed match includes spurious matches, and must be updated from the CT iteratively until a partial match is found. In Eq.3,  $\bar{l}_s = \sum_{i,j} d(p_{s,i}, p_{s,j})$  denotes the average segmental length. The norm  $\|\cdot\|$  denotes the Euclidean distance. We set  $r_{s,i} = 1$  if the point  $p_{s,i}$  has a non-null match  $q_{s,i}^*$ , otherwise set it to 0.

In order to achieve a best match satisfying Eq.3 as well as a matching size criterion Eq.4,

$$|\{(q_{s,i}^*, p_{s,i})\}'_M| = \sum_{i=1}^{M_s} r_{s,i} > \beta M_s \quad (4)$$

where  $\beta$  indicates percentage of matching quantity (we choose  $\beta \approx 0.8$ ). We apply the obtained motion estimate  $[R'_s, T'_s]$  to all the model points  $p_{s,i}$ , and reassign a correspondence by finding their closest match neighbors  $(q_{s,i}^*, p_{s,i})$  with a wider threshold in Eq. 5, to allow for matches excluded in the CTs or dropped during iteration.

$$\min_{q_j^* \in \bar{Q}_M^*} \{d(R'_s \times p_{s,i} + T'_s, q_j^*)\} < 2\bar{l}_s \varepsilon_s \quad (5)$$

This iterative procedure is applied to prioritized CTs until a segment best match  $\{(q_{s,i}^*, p_{s,i})\}_M$ , satisfying criteria Eq. 3 and Eq. 4, is found or assessed not to exist.

### C. Dynamic segment-based matching

A single frame may provide insufficient spatial data for determining the correct correspondence. A dynamic approach is employed to make improvements over single frame matching in this case. As shown in Fig.1 and Fig.2, the dynamic approach includes three main stages: motion-based verification, dynamic identification and backward error correction.

#### C.1 Motion-based verification module

Motion-based verification (see Fig.3) is used to confirm a segment identification  $\{(q_{s,i}^*, p_{s,i})\}_M$  at key-frame \*. Confirmation requires that, after a slight shift  $\Delta_{shift} = f_s/2$ , the correspondence  $\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M$  is maintained for the “rigid” segment  $P_s$ .

If at the shifted frame  $*+\Delta_{shift}$  reveals more matches of  $P_s$ , we update the key-frame to  $T_K = * + \Delta_{shift}$ . A recruitment function (see Fig.4) is then applied in the DSHPM algorithm to all identified segments in order to recruit newly appeared or previously missed matches.

#### C.2 Dynamic identification

Some frame data for a segment may be of poor quality, on account of having excessive missing data or distortion. In this case, CTs-based segment matching may fail. This will break off the segment-based hierarchical search and result in serious uncertainty for successive child-segment identification. The dynamic idea is to attempt to re-identify the segment up to three times with successive shifts  $\Delta_{shift}$  of the key-frame.

A special case of failure of the CTs-based segment identification occurs when a CT includes only two non-null candidate columns (the SVD-based motion calcu-

For a shifted matches  $\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M$  :

**Case1:**  $|\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M| \geq 3$   
calculate motion [R,T] on  $\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M$   
**if:**  $\bar{e}_s < \varepsilon_s$   
motion-based verification: success=1;  
**if:**  $P_s$  has unidentified points:  
transform all unmatched model points by [R,T];  
**if:** found closest neighbors as re-added matches,  
calculate motion [R,T] on the new matches;  
**if:** again  $\bar{e}_s < \varepsilon_s$  and more matches at  $* + \Delta_{shift}$   
motion-based verification: success=2;  
**otherwise:**  
motion-based verification: success=-1, failed!

**Case2:**  $|\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M| = 2$   
**if:**  $\frac{|d(q_{s,1}^{*+\Delta_{shift}}, q_{s,2}^{*+\Delta_{shift}}) - d(p_{s,1}, p_{s,2})|}{d(p_{s,1}, p_{s,2})} < \varepsilon_s$   
motion-based verification: success=1;  
**otherwise:**  
motion-based verification: success=-1, failed!

**Exit**

Fig. 3. Motion-based verification

For an identified  $P_s$  with unmatched points:

**if:**  $|\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M| \geq 3$   
calculate motion [R,T] on  $\{(q_{s,i}^{*+\Delta_{shift}}, p_{s,i})\}_M$ ;  
transform all unmatched model points in  $P_s$  by [R,T];  
**if:** found their closest-neighbors as recruited matches  
re-calculate motion [R,T] on the new matches;  
**if:** again  $\bar{e}_s < \varepsilon_s$   
apply motion-based verification;  
**if:** success, update the new matches.

Fig. 4. Recruitment algorithm

lation needs at least three matching pairs). In this situation, when only one point has been identified or recovered as a joint, we accept the other point’s match according to its prioritized candidates in the CT. Observably, such a segment may be very uncertain or non-unique. For an uncertain segment, we save all candidate information, and will further confirm or correct it in a hierarchical chain depending on whether its child-segment or even its grandchild-segment can be found.

#### C.3 Backward error-correction module

Repeated failure of the dynamic procedure to identify a segment  $P_s$  may imply a wrong or highly distorted  $\{\text{joint}_j\}$  provided by its parent-segments. In this case the algorithm attempts, after three such failures, a backward error-correction to check joints and uncertain parent-segments (see Fig.2 and Fig.5).

### D. Global hierarchical search strategy

Global articulated matching is achieved by integrating the dynamic segment-based matching with a hierarchical search strategy as follows. In the segment-based articulated model, we assume that one of its segments contains more points and has more segments linked to it

```

function parent_segment_correction(s): {
do if: segment  $P_s$  is identified using a joint
    provided by an uncertain parent-segment  $P_{s-1}$ )
    if: this joint is the last candidate in CT
        deny this joint identity;
        s=s-1;
        call parent_segment_correction(s) function.
    else:
        correct the joint to the next candidate;
        s=s-1;
        Break to A: re-identify segment  $P_s$ .
until: meet a segment with certain joint.
    skip all investigated segments with uncertain joints;
Break to A: re-identify the first call  $P_s$  without joint. }

```

Fig. 5. Recursive algorithm for parent\_segment\_correction

than most other segments. We treat such a segment as root. The global hierarchical search strategy begins at the root. After the root has been located, searching proceeds depth first to children along hierarchical chains. In this process, some points, such as joints, may have been located during its parent-segment identification, therefore they can be used as known pivots in child-segments. This linkage considerably increases the reliability and efficiency of child-segment identification. In the case of missing joints, if an identified parent-segment includes at least three points, we can utilize the obtained motion transformation to recover a virtual joint, and ensure that child-segment identification can still proceed reliably. When a search chain in the hierarchy is broken by a failed segment identification, global search will tend to identify segments on other chains and leave the remaining child-segments on that broken chain the last to be solved.

#### E. Identification with tracking for registration of the whole motion sequence

Having registered segment-by-segment around a key-frame range, the identity of each feature point can be propagated along its pre-tracked trajectory until the trajectory is broken somewhere due to occlusion. So, after key-frame range identification, the DSHPM algorithm tracks forward the whole motion sequence and identifies any reappearing points. ‘‘Recruitment’’ and ‘‘CTs-based segment matching’’ functions are combined advantageously with modules of motion-based verification and backward error-correction. The complexity of this stage is much reduced, because recruitment occurs with respect to a largely correct existing identification.

### IV. EXPERIMENTAL RESULTS

The proposed DSHPM algorithm has been implemented in Matlab. We have tested its performance on sequences of motion data. We report some experimental results on a real-world application: registration of human movements in 3D Moving Light Displays (MLDs).

All model data and motion data were acquired from a commercial marker-based optical motion capture (Mo-Cap) system - Vicon 512. The system includes 7 high-resolution cameras capturing a control volume of

about 4m(length)  $\times$  4m(width)  $\times$  2.5m(height). It can reconstruct the 3D coordinates of an infrared reflective marker if the marker position is located in at least two cameras. The measurement accuracy of this system is to the level of a few millimeters in the capture volume.

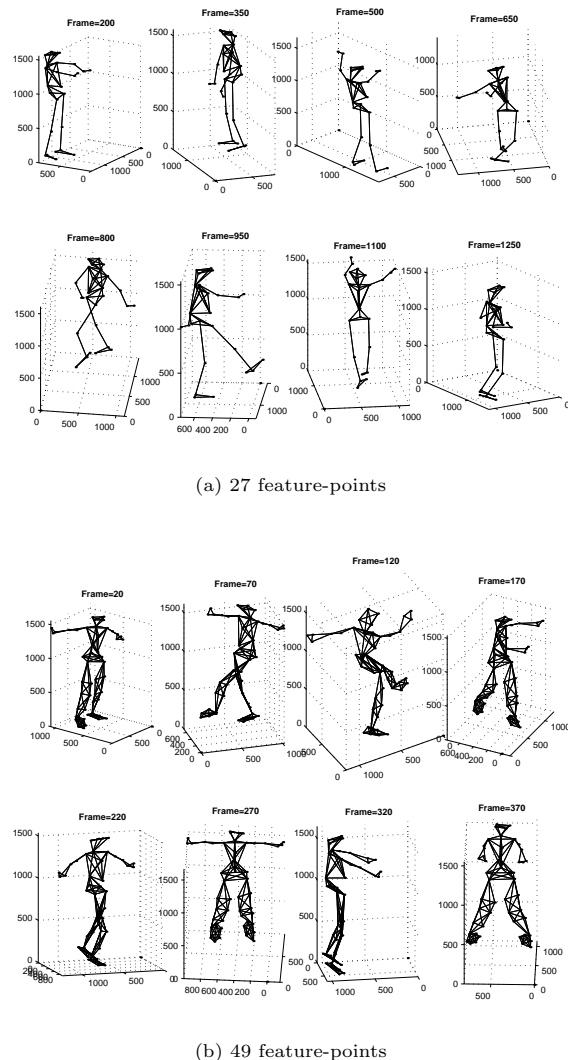


Fig. 6. Identified human freeform movements in MLDs

Human motion represents a typical articulated motion with deformable segments. In our experiments, the feature points are retro-reflective spherical markers attached to tightly clothed sites on the subjects. A number of subjects, movements and feature-point distributions were investigated. Results from several representative configurations for a human model with 15 segments are shown in Fig.6. Each subject model is manually established off-line using one frame of static pose data. The subject’s freeform movements to be identified are then captured. The data may have missing points, noise points and segment distortion. From the example results in Fig.6, in which identified points of the same

segment are shown linked, we observe that the proposed algorithm is capable of registering point-feature articulated movements from noisy data.

## V. DISCUSSION

The DSHPM was developed to achieve accurate and efficient registration in self-initializing point-feature tracking for articulated motion. The algorithm employs several techniques that are robust to data noise:

1) combination of identification and tracking to avoid exhaustive frame-by-frame identification.

2) matching with the model at the hierarchical segment level rather than by a brute point-by-point search. Segment-based searching is capable of dealing with missing and redundant data in a real-world situation.

3) CTs generation and optimization that results in a considerable search space reduction. In particular, use of CTs and SVD-based affine motion estimation allows us to locate the crucial first segment without any assumption for an initial correspondence.

4) pre-segmentation by the *SFM* matrix to indicate some possible intra-segmental points. This further improves the performance of CTs-based segment matching.

5) dynamic and motion-based identification, verification and backward error correction to enhance the robustness and feasibility of the algorithm for handling real-world data.

Computational cost of the DSHPM algorithm varies not only with the number and distribution of model points and segments, and the length of an observed motion sequence, but also with the quality of the data, such as the segmental distortion level, and the frequency and amount of missing and noisy data. All these disadvantageous aspects will result in identification and tracking difficulties. Moreover, the presence of interrupted trajectories also leads to an increased search burden.

Table I gives further results for the identification of human movement. We executed the DSHPM algorithm in Matlab code on a 866MHz Compaq with 256MB of RAM. The identification rate is indicated by the ratio of the number of correctly identified trajectories to the total number of trajectories encountered. From the results, we observed the identification rate is acceptable both for the sparse and denser feature-point distributions, even for large accelerative movements (such as jumping). Execution time is competitive with real motion time. This suggests it is possible to realize buffered real-time registration for an on-line tracker.

## VI. CONCLUSION

We have proposed a dynamic segment-based hierarchical point matching (DSHPM) algorithm for registration of articulated motion with 3D sparse feature points, under the condition of available 3D data. This topic has received little attention in the literature. We do not make assumptions regarding restriction to a specific motion or model pose similarity, nor do we require absence of data noise.

The algorithm contributes to the study of self-initializing identification, this being the process that

TABLE I  
EXAMPLES OF HUMAN MOTION IDENTIFICATION.

freeform movement	frames (T)	trajectories(N)	identification rate	CPU time(s)
subject 1: 27 feature-points, $f_s = 60$ Hz				
walking	296	50	100%	2.7
running	1100	64	98%	4
gymnastics	1370	69	95%	6
subject 2: 33 feature-points, $f_s = 120$ Hz				
walking	994	45	100%	2
gymnastics	2652	48	95%	6
dance	1034	74	93%	4.4
subject 3: 49 feature-points, $f_s = 60$ Hz				
walking	1170	72	98%	5
jumping	393	65	95%	3
walk, run jump, bend	860	109	92%	11.2

ultimately determines the robustness of motion tracking. It contrasts with previous approaches concentrating on incremental inter-frame tracking. Experimental results demonstrate the robustness of the algorithm in real-world situations. This algorithm has a potential usage in real-time applications.

### Acknowledgments

Model and motion data in our experiments were obtained by a Vicon 512 MoCap system, installed at the Department of Computer Science, UWA.

### REFERENCES

- [1] J. Aggarwal, Q. Cai, W. Liao, and B. Sabata. "Articulated and elastic non-rigid motion: A review". In *Proc. IEEE Workshop on Motion of Non-Rigid and Articulated Object*, pp. 2–14, Austin, TX, 1994.
- [2] K. Arun, T. Huang, and S. Blostein. "Least square fitting of two 3-D point sets". *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.
- [3] C. Cédras and M. Shah. "A survey of motion analysis from moving light displays". In *Proc. IEEE CVPR'94*, pp. 214–221, Washington, June 1994.
- [4] Z. Chen and H. Lee. "Knowledge-guided visual perception of 3-D human gait from a single image sequence". *IEEE Trans. Sys., Man, Cybern.*, 22(2):336–342, 1992.
- [5] L. Herda, P. Fua, R. Plänkers, R. Boulic, and D. Thalmann. "Skeleton-based motion capture for robust reconstruction of human motion". In *Proc. Computer Animation*, Philadelphia, USA, May 2000.
- [6] G. Johansson. "Visual motion perception". *Scientific American*, 232(6):75–80, 85–88, 1975.
- [7] J. Richards. "The measurement of human motion: a comparison of commercially available systems". *Human Movement Science*, 18(5):589–602, Oct. 1999.
- [8] M. Ringer and J. Lasenby. "Modelling and tracking articulated motion from multiple camera views". In *Proc. British Machine Vision Conf.*, pp. 172–182, Bristol UK, Sept. 2000.
- [9] P. B. V. Wamelen, Z. Li, and S. S. Iyengar. "A fast algorithm for the point pattern matching problem". *IEEE Trans. PAMI. last revised Dec.*, 1999.
- [10] Z. Zhang. "Iterative point matching for registration of free-form curves". *Computer Vision*, 13(2):119–152, March 1994.