Coarse-to-Fine Tracking of Articulated Objects Using a Hierarchical Spring System*

Nicole Artner[1], Adrian Ion[2], and Walter Kropatsch[2]

[1] Austrian Research Centers GmbH - ARC, Smart Systems Division, Vienna, Austria
nicole.artner@arcs.ac.at
[2] PRIP, Vienna University of Technology, Austria
{ion,krw}@prip.tuwien.ac.at

Abstract. Tracking of articulated objects is a challenging task in Computer Vision. A highly target specific model can improve the robustness of the tracking by eliminating or reducing the ambiguities in the association task. This paper presents a flexible framework, which allows to build target specific, part-based models for arbitrary articulated objects. The rigid parts are described by hierarchical spring systems in form of attributed graph pyramids and connected via articulation points, which transfer position information between the adjacent parts.

1 Introduction

Tracking the parts of articulated objects in video sequences is still a challenging task with a lot of open problems. Promising approaches dealing with this task employ part-based models and match this models into the image with the help of statistics to maximize a probability function.

A possibility to build part-based models and describe the spatial relationships between the parts of the target object in a tolerant, deformable way are spring systems. Spring systems can be represented by graphs, where each part of a target object is a vertex and the edges encode their spatial relationships. Object recognition or tracking can be done by minimizing the energy in the spring system to find the most likely configuration of the object parts in an image. Spring systems have already been proposed in 1973 by Fischler et al. [1]. Felzenszwalb et al. employed this idea in [2] to do part-based object recognition for faces and articulated objects (humans). Their approach is a statistical framework minimizing the energy of the spring system learned from training examples using maximum likelihood estimation. Ramanan et al. apply in [3] the ideas from [2] in tracking people. In [4], Mauthner et al. present an approach using a two-level hierarchy of particle filters for tracking objects described by spatially related parts in a mass spring system.

In this paper we also employ spring systems to encode the relationships in a part-based model, but in comparison to the related work we try to stress solutions

* Partially supported by the Austrian Science Fund under grants P18716-N13 and S9103-N13.

that emerge from the underlying structure, instead of using structure to verify statistical hypothesis. The aim is to supply a flexible framework that allows to build part-based models for arbitrary objects with varying number of rigid parts and articulation points. Each rigid part is robustly tracked with the help of a hierarchical spring system encoding the spatial relationships of coarse and fine features. The articulation points in the model act as agents of the information transfer between the parts of the object. They transfer position information from reliable parts to ambiguous parts. The approach presented here refines and extends our previous work in [5]. Possible applications are action recognition, human computer interfaces, motion based diagnosis and identification, etc.

There is a vast amount of work in the field of tracking articulated objects and motion analysis [6–8]. It would go beyond the scope of this paper mentioning all of this work. In comparison to many related approaches our approach does not need any training and we do not employ motion models. The presented approach relies on the spatial relationships of object parts and their features, and hence resulting distance constraints.

The paper is organized as follows: Sec. 3 explains how the hierarchical spring systems are built. In Sec. 4 the task of articulation points is described. Sec. 5 sums up the presented concepts and describes their combination in tracking. Sec. 6 presents experiments to prove and qualitatively evaluate the concept of our approach and in Sec. 7 we draw conclusions.

2 The building blocks of the spring hierarchy

Articulated objects are made out of rigid parts connected through articulation points. On each rigid part, multiple features are tracked through a mixture of many independent trackers, one for each feature, and a spring system, one for each part. The final position of each feature is decided based on the offset vectors from the tracker and the spring system. This section recalls Mean shift, the method used for the independent trackers, and the spring system.

2.1 Mean shift algorithm

The Mean shift algorithm [9] is employed to associate the features of the object parts between consecutive frames. It does this by efficiently finding local maxima in a probability distribution, and generating an offset vector pointing to the corresponding position. The distribution encodes the probability that a given feature from the previous frame is in a certain position in the current frame. To compute the probability that a certain feature, *the target*, matches the feature at a certain position, the following similarity measure is used (see Eq. 3).

Region covariance was introduced by Porikli and Tuzel as a feature for detection, classification and tracking in [10, 11]. It is invariant to scaling and rotation up to a certain degree (depends on the feature selection) and allows the combination of multiple features in an elegant way. Furthermore, compared to other

region descriptors, region covariance is low-dimensional and can be efficiently calculated using integral images.

The covariance feature is extracted out of an one dimensional intensity or a three dimensional color image I. F is a $W \times H$ dimensional feature image extracted from I, encoding a feature vector of size d at each position $F(x, y)$:

$$F(x, y) = \phi(I, x, y), \tag{1}$$

where the function ϕ can be any mapping including e.g. intensity, color, gradients and so on. A rectangular region of interest $R \subset F$ can be represented by the $d \times d$ covariance matrix

$$C_R = \frac{1}{n-1} \sum_{k=1}^{n} (z_k - \mu)(z_k - \mu)^T, \tag{2}$$

where $\{z_k\}_{k=1..n}$ are the d-dimensional feature vectors of the points in R and μ is the mean over all points. The following distance measure is used to calculate the similarity between two covariance matrices [11]:

$$\rho(C_1, C_2) = \sqrt{\sum_{i=1}^{n} \ln^2 \lambda_i(C_1, C_2)}, \tag{3}$$

where $\{\lambda_i(C_1, C_2)\}$ are the generalized eigenvalues of C_1 and C_2.

2.2 Spring system as a graph

An attributed graph (AG) is a possible data structure for a spring system. The attributes of the vertices of the graph are the features and their corresponding positions. We use covariance matrices as the features (Sec. 2.1), but other features can also be used (e.g. 3D color histogram features [5]).

Given the features, the edges of the AG are obtained by a Delaunay triangulation. A fully connected graph (connected each vertex with each other vertex in the graph) could also be used but it would increase the complexity of the optimization process.

The elastic behavior (tolerance to variations in the structure) of a spring system can be modeled by graph relaxation. As the tracked object parts are rigid, the objective of the relaxation is to maintain the tracked structure as similar as possible to the initial structure. Thus the aim is to keep the edge lengths as similar as possible to the initial length. The total energy of the spring system is 0 in the initial state and increases with the deformation of the structure.

The variation of the edge lengths in the AG and their directions are used to determine a structural offset for each vertex. This offset vector is the direction where a given vertex should move such that its edges restore their initial length and the energy of the structure is minimized. This structural offset vector $\boldsymbol{O}$ is calculated for each vertex v as follows:

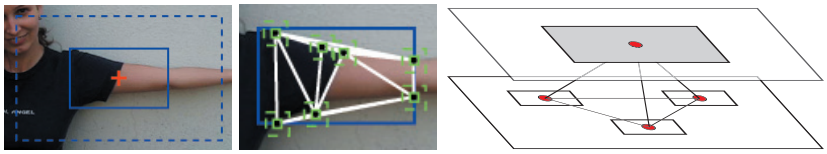
$$\boldsymbol{O}(v) = \sum_{e \in E(v)} k \cdot (|e'| - |e|)^2 \cdot (-\boldsymbol{d}(e, v)), \tag{4}$$

where $E(v)$ are all edges e incident to vertex v, k is the elasticity constant of the edges in the structure, e is the edge length in the initial state and e' at a different point in time. $\boldsymbol{d}(e, v)$ is the unitary vector in the direction of edge e that points toward v. For more details see [5].

3 Building the hierarchical spring system

Each rigid part of a target object is described and tracked in a coarse-to-fine manner. Each part is described by a two level spring system represented by an attributed graph pyramid [12].

As shown in Fig. 1(a), the top level is described by one covariance feature C_t, extracted out of a region of interest (ROI) covering the whole object part. The bottom level consists of several features, which are from the same ROI (see Fig. 1(b)). A Harris corner detector is applied on the ROI to find promising positions for the region covariance features $\{C_b\}_{i=1..n}$ of the bottom level. Around each corner point a small ROI is built to calculate C_b (e.g. 9×9 pixels).



(a) (b) (c)

Fig. 1. Extracting region covariance features. (a) Feature of top level. (b) Features of bottom level. The white edges are the edges of the AG. (c) Attributed graph pyramid.

The **AG pyramid** is built as follows (see Fig. 1(c)). In the **top level** of the pyramid: one vertex to which the coarse feature C_t of the whole rigid part is assigned. In the **bottom level** all fine features C_b. The edges in the bottom level of the pyramid are inserted with a Delaunay triangulation. The vertex in the top level is connected with every vertex (child) in the bottom level. The spring system is initialized with the state in the first frame, meaning that the total energy of the spring system is considered 0 in this configuration.

4 Articulation points: Agents of the information transfer

An *articulation point* connects several rigid parts. It allows them to move independent from each other, but forces them to always keep the same distance. From this follows that the movement of a rigid part in the image plane is constraint to the circle centered at the articulation point and spanned by the radius corresponding to its size (in 3D it is a sphere). Fig. 2 visualizes this concept. It would be possible to connect every point with the articulation point, but to

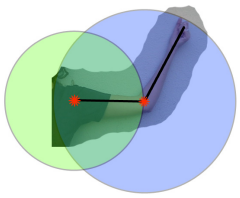


Fig. 2. Distance constraints imposed by articulation points.

reduce the complexity we only built a spring system with two reliable vertices in the bottom level and with the top vertex.

Articulation points can be initialized manually or automatically by observing the articulated motion of the target object [5, 13] parts.

Deriving the position of the articulation points: In the frame in which the position of the articulation point is initialized, for each adjacent part and each pair of features of the bottom level of the part, a local coordinate system is created. The coordinates of the articulation point in this coordinate systems is stored. Having the position of any two features is then enough to reconstruct the position of the articulation point, and thus at any time, each part can generate a hypothesis for the position of all adjacent articulation points. This hypothesis is produced with the local coordinate system of the two most reliable features (see Sec. 4.1) – further on named *reference vertices* – of each part.

The hypothesis of all connected parts of an articulation point are combined with a weighted sum, where the weight for each hypothesis depends on the reliability of the corresponding part (see Sec. 4.1). With this weighting, the influence of ambiguous parts (e.g. occluded parts) on the position of the articulation point is low and for reliable parts high.

After the position of the articulation point is computed, the articulation "transfers" position information from reliably to ambiguously tracked parts through its distance constraints (circles). This is done in spring systems, where the articulation point is connected to the reference vertices and the top vertices.

4.1 Computing the reliability of features and parts

The reliability of a feature b depends on the number of incident edges in the spring system I_b, the energy of the incident edges in the spring system E_b and the similarity S_b (see Eq. 3) of the covariance feature C_b to the template covariance feature from the first frame:

$$\begin{aligned} R_b &= I_b \cdot \alpha_I + E_b \cdot \alpha_E + S_b \cdot \alpha_S \\ I_b &= \frac{E(v)}{E}\,, \quad E_b = \frac{\sum_{e \in E(v)} k \cdot (|e'| - |e|)^2}{T_{E_p}}\,, \quad S_b = \frac{\rho(C_b, C_t)}{\rho_{\max}}. \end{aligned} \tag{5}$$

$E(v)$ are all edges incident to vertex v (feature), E is the number of edges in the spring system, T_{E_p} is the total energy of the spring system, and $\rho_{\max}$ is the highest similarity in the same part as feature b. In our experiments: $\alpha_I = 0.2$, $\alpha_E = 0.4$, $\alpha_S = 0.4$. The reliability of a part p is:

$$R_p = D_p \cdot \alpha_D + E_p \cdot \alpha_E + S_p \cdot \alpha_S$$
$$D_p = \frac{F_p}{F}, \quad E_p = \frac{T_{E_p}}{T_E}, \quad S_p = \frac{\sum_{b \in p} \rho(C_b, C_t)}{F_p} \tag{6}$$

is computed out of the size D_p of the part p, the energy of the part E_p, and the similarity S_p of the covariance features in comparison to their templates. F_p is the number of features of part p, F is the number of all features, and T_E is the energy of all spring systems. In our experiments: $\alpha_D = 0.2$, $\alpha_E = 0.4$, $\alpha_S = 0.4$. Intuitively the two measures model the mixture of "seeing" and "knowing".

5 Tracking as a hierarchical optimization process

Tracking is done in a coarse-to-fine manner – from top to bottom level of each part (summarized in Algorithm 1).

Algorithm 1 Algorithm for tracking articulated objects.

```
1: processFrame
2:     associate top vertex of each part with Mean shift
3:     associate bottom vertices of each part with Mean shift and structural offsets
4:     select reference vertices for each part
5:     calculate current position of articulation points
6:     transfer position information over articulation points to top and bottom levels
7: end
```

The first step is to associate the top vertices of each part using the positions from the previous frame and applying Mean shift to a probability distribution built with the similarity measure in Eq. 3.

In the next step the bottom vertices of each part are associated by combining Mean shift offsets and structural offsets. The structural offsets are generated out of the spring systems of the bottom levels and the spring systems connecting each bottom vertices with the corresponding top vertices. For each feature (vertex) depending on its reliability value R_b a mixing gain $g = 0.5 - (R_b - 0.5)$ is computed and used to combine the offsets.

Then for each part the two vertices with the highest R_f are selected to generate the hypothesis for the positions of the articulation points. The hypothesis of the parts connected to a articulation point are mixed with a weighted sum depending on the reliability value R_p of each part.

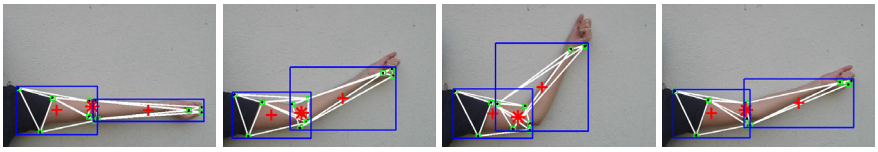
In the last step, the position information between the parts is transfered over the articulation points to the top and reference vertices which forward the

information to the vertices not directly connected. This transfer is again done in a combined iterative process with Mean shift and structure.

6 Experiments

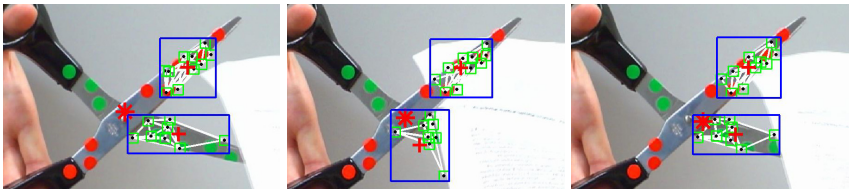
In all experiments we use prior knowledge about the structure of the object to initialize the ROIs and the articulation points. The spring constant k is set for edges in the bottom level to 0.2 and for edges connecting to the articulated point or to the top vertex to 0.5.

In experiment 1, the lower and upper arm of a human are successfully tracked through articulated motion (see Fig. 3). Experiment 2 in Fig. 4 shows frames with scissors. One part of the structure is completely occluded, but the position of the articulation point (red star) is robust and the structure relaxes when the occlusion is gone. In experiment 3 one can see the tracking of 4 parts connected with 3 articulation points (see Fig. 5).



Frame 10 Frame 70 Frame 280 Frame 305

Fig. 3. Experiment 1: Tracking a human's upper and lower arm in articulated motion.

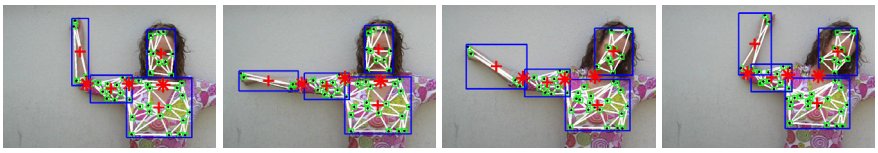


Frame 210 Frame 285 Frame 355

Fig. 4. Experiment 2: Tracking through occlusion.

7 Conclusion

This paper presented an approach for describing and tracking of articulated objects consisting of several rigid parts connected with articulation points. The object parts are described in a coarse-to-fine manner in an AG pyramid, where



Frame 20 Frame 150 Frame 440 Frame 490

Fig. 5. Experiment 3: Tracking 4 parts of a human with 3 articulation points.

the features are region covariance matrices and the spatial relationships between the features are enforced during the tracking through a hierarchical spring system. Position information is transfered between the parts over the corresponding articulation points depending on the reliability of the parts and their features. Open issues are dealing with pose changes and the corresponding changes in the structure, optimizing the information transfer in big structures and automatically initializing the structure.

References

1. Fischler, M.A., Elschlager, R.A.: The representation and matching of pictorial structures. Transactions on Computers **22** (January 1973) 67–92
2. Felzenszwalb, P.F.: Pictorial structures for object recognition. IJCV **61** (2005) 55–79
3. Ramanan, D., Forsyth, D.: Finding and tracking people from the bottom up. In: CVPR. Volume 2., IEEE (June 2003) 467–474
4. Mauthner, T., Donoser, M., Bischof, H.: Robust tracking of spatial related components. In: ICPR, IEEE (December 2008) 1–4
5. Artner, N., Ion, A., Kropatsch, W.G.: Tracking objects beyond rigid motion. In: GbR, Springer (May 2009)
6. Gavrila, D.M.: The visual analysis of human movement: A survey. CVIU **73**(1) (January 1999) 82–980
7. Moeslund, T.B., Hilton, A., Krger, V.: A survey of advances in vision-based human motion capture and analysis. CVIU **104**(2–3) (2006) 90–126
8. Aggarwal, J.K., Cai, Q.: Human motion analysis: A review. CVIU **73**(3) (March 1999) 428–440
9. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. PAMI **24**(5) (2002) 603–619
10. Porikli, F., Tuzel, O., Meer, P.: Covariance tracking using model update based on lie algebra. In: CVPR. Volume 1. (June 2006) 728–735
11. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. In: ECCV. Volume 2., Springer (May 2006) 589–600
12. Kropatsch, W.G., Haxhimusa, Y., Pizlo, Z., Langs, G.: Vision pyramids that do not grow too high. Pattern Recognition Letters **26**(3) (2005) 319–337
13. Mármol, S.B.L., Artner, N.M., Ion, A., Kropatsch, W.G., Beleznai, C.: Video object segmentation using graphs. In: 13th Iberoamerican Congress on Pattern Recognition, Springer (September 2008) 733 –740