Human Body Pose Estimation Using Silhouette Shape Analysis

Abstract

We describe a system for human body pose estimation from multiple views that is fast and not dependent on a rigid 3D model. We make use of recent work in decomposition of a silhouette into 2D parts. These 2D part primitives are matched across views to build assemblies in 3D. In order to search for the best assembly, we use a likelihood function that integrates information available from multiple views about body part locations. Occlusion is modeled into the likelihood function so that the algorithm is able to work in a crowded scene even when only part of the person is visible in each view. The algorithm has potential applications in surveillance and promising results have been obtained.

1 Introduction

Determining the pose of humans is an important problem in vision and has many applications. In this paper, we target multi-camera surveillance applications where one wants to recognize the activities of people in a scene in the presence of occlusions and partial occlusions. One cannot assume that a person is visible in isolation or in full in either one or all of the views. Nor can one assume that we have a model of the person, or that the initial body pose is known. Such a system should also be reasonably fast. However, very accurate body pose values are typically not required, and an answer close to the actual body pose might be adequate. We describe an algorithm that can form the basis of such a surveillance system.

Our system estimates the 3D pose of a human body from multiple views. We make use of recent work in decomposition of a silhouette into 2D parts. These 2D part primitives are matched across views to build primitives in 3D which are then assembled to form a human figure. In order to search for the best assembly, we use a likelihood function that integrates information available from multiple views about body part locations. Greedy search strategies are employed so as to find the best assembly fast.

1.1 Related Work

Human Body pose estimation has received considerable interest in the past few years and several approaches have been tried for different applications.

There are many methods for incremental model-based body part tracking where a model of an articulated structure (person) is specified upfront[5, 20, 1, 21, 6]. Delamarre and Faugeras [5] try to align the projection of an articulated structure with the silhouettes of a person obtained in multiple views by calculating forces that need to be applied to structure. Drummond and Cipolla [20] use Lie algebra to incrementally track articulated structures. Bregler and Malik [1] use twists and exponential maps to specify relationships between parts and to track an articulated structure incrementally. Sidenbladh[17] and Choo [4] use monte carlo particle filtering to incrementally update the posterior probabilities of pose parameters. These methods need to have both a 3D model of the human structure and a good initialization and have potential applications in motion-capture [6].

Another class of algorithms [12, 8, 18, 15] try to detect body parts in 2D using template matching and then try to find the best assembly using some criteria. Some other methods learn some models of human motion. These models can be based on optical flow [7], exemplars [14, 19], feature vectors [18], support vector machines [15], or statistical mappings (SMA) [16]. These models can then be used to detect and estimate the pose of a human in an observed image.

Our work is most closely related to the work of Kakadiaris and Metaxas [11] who try to acquire 3D body part information from silhouettes extracted in orthogonal views. They employ a deformable human model so that any size of the human can be recognized. The distinguishing feature of our work is that it is able to work in a crowded scene so that in all of the views, the person might be fully or partially occluded. This is accomplished by explicitly modeling occlusion and developing prior models for person shapes from the scene. This helps us to decouple the problems of pose estimation for multiple people so that the degrees of freedom of the problem are decreased substantially.

The paper is organized as follows. Section 2 describes the method of extraction of silhouettes in a crowded scene. Section 3 describes shape analysis of silhoettes and matching parts across views to obtain 3D part primitives. Section 4 describes the likelihood function used for assembly evaluation. Section 5 describes the algorithm used to find the best assembly. We conclude with some preliminary results in section 6.

2 Extracting Multiple Silhouettes in a Cluttered Scene

We use the method developed by Mittal and Davis [13] in their system M$_2$Tracker for extracting silhouettes of people in a cluttered scene. The method is able to segment regions belonging to different people even when they are not visually isolated. Here, we provide a brief review of the method.

M$_2$Tracker develops two types of models for each person.

2.1 Appearance Models

2.1.1 *Color Models*

A probabilistic model for the color distribution at different heights of the person is developed using the method of non-parametric Gaussian kernel estimation.

2.1.2 *"Presence" Probabilities*

The other attribute modeled is the "Presence" Probability (denoted by $L(h, w)$), defined as the probability that a person is present(i.e. occupies space) at height h and distance w from the vertical line passing through the person's center.

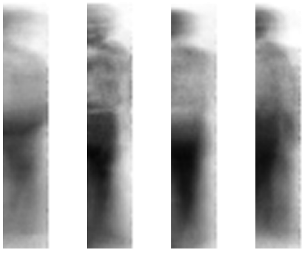


Figure 1: Sample Presence Probabilities of people.

These models are developed automatically from the scene and are used to segment images in the following way.

2.2 Pixel Classification

Bayesian Classification is used to classify each pixel as belonging to a particular person, or the background. The *a posteriori* probability that an observation $I(x)$ at pixel x originated from person j (or the background) is

$$P_{posterior}(j/I(x)) \propto P^{x}_{prior}(j) P(I(x)/j) \qquad (1)$$

The pixel is then classified as

$$\text{Most likely class} = arg\max_{j}(P_{posterior}(j/I(x)) \qquad (2)$$

$P(I(x)/j)$ is given by the color model of the person at height h. For the background, a background model of the scene is used.

The priors include occlusion information and determined using the following method. For each pixel x, a ray is projected in space passing through the optical center of the camera. Minimum distances w_j of this ray are calculated from the vertical lines passing through the currently estimated centers of the people. Also calculated are the heights h_j of the shortest line segments connecting these lines. Then, the prior probability that a pixel x is the image of person j is set as

$$P^{x}_{prior}(j) = L_j(h_j, w_j) \prod_{k\ occludes\ j} (1 - L_k(h_k, w_k))$$

$$P^{x}_{prior}(bckgrnd) = \prod_{all\ j} (1 - L_j(h_j, w_j)) \qquad (3)$$

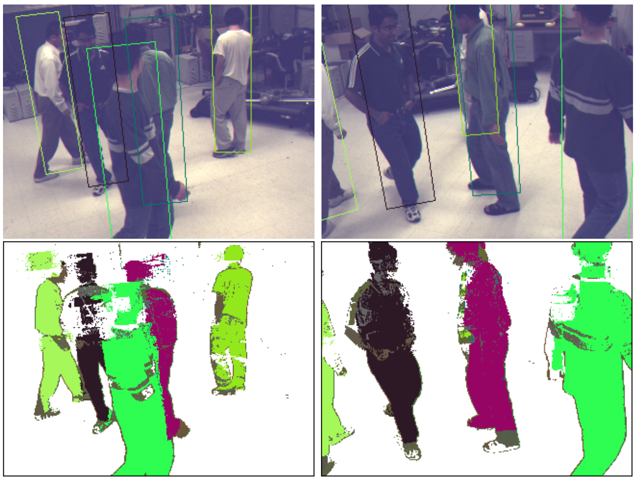


Figure 2: Some results from M_2Tracker. The first two images show detection and tracking results and the last two show segmentation results.

where $L_j(h_j, w_j)$ is the “presence” probability described earlier. A person “k occludes j” if the distance of k to the optical center of the camera is less than the distance of j to the center. The classification procedure helps to incorporate both the color profile of the people, and the occlusion information available.

The segmentation algorithm assumes knowledge of approximate person locations. These locations are obtained using a region-based stereo algorithm.

2.3 Obtaining Multiple Segmentations

There are several parameters in the segmentation algorithm. Accurate extraction of different parts of the person requires different parameters. Therefore, it is essential to vary the parameters so as to obtain multiple segmentations. The parameters that we vary are
(1) the relative weight given to the background model,
(2) the relative weight given to different foreground objects so that different objects are highlighted, and
(3) the threshold for detemining whether a pixel is unclassified pixels.
The silhouettes thus obtained are segmented using the method described in the next section.

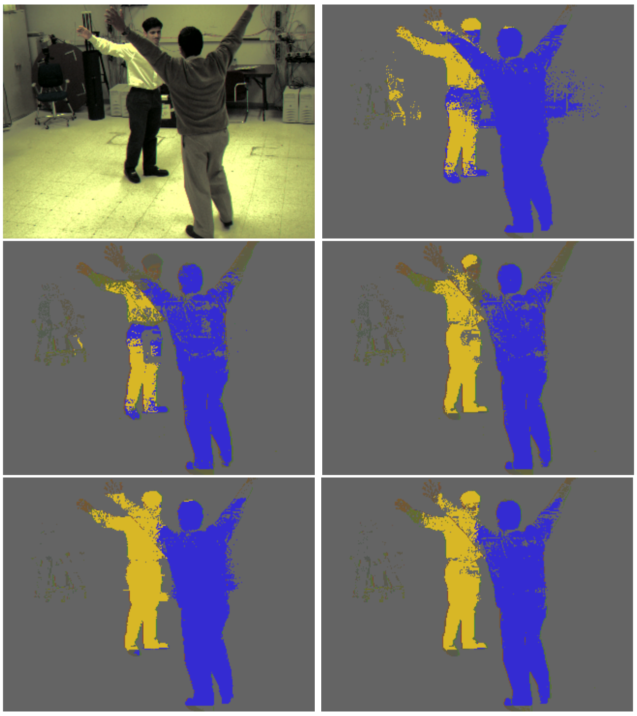
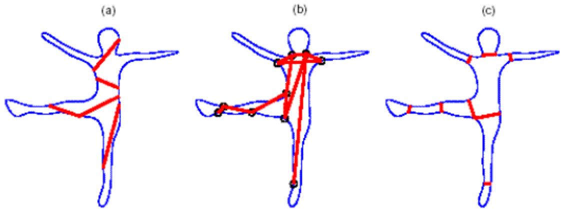


Figure 3: Multiple Segmentations Obtained for the image shown in the first image

3 Computing Body-part Primitives

3.1 2D Silhouette Shape Analysis

In order to recover the pose of a person, we break the silhouette of the person into parts. According to human intuition about parts, a segmentation into parts occurs at *negative minima of curvature* so that the decomposed parts are convex regions. Singh *et al.* noted that when boundary points can be joined in more than one way to decompose a silhouette, human vision prefers the partitioning scheme which uses the shortest cuts (A cut is the boundary between a part and the rest of the silhouette). They further restrict a cut to cross a symmetry axis in order to avoid short but undesirable cuts. However, most symmetry axes are very sensitive to noise and are expensive to compute. In contrast, we use the constraint on the salience of a part to avoid short but undesirable cuts. According to Hoffman and Singh’s [10] study there are three factors that affect the salience of a part: the size of the part relative to the whole object, the degree to which the part protrudes, and the strength of its boundaries. Among these three factors, the computation of a part’s protrusion (the ra-



Random cuts **Using negative curvature minima points** **Using shortest cuts**

Figure 4: Silhouette Decomposition

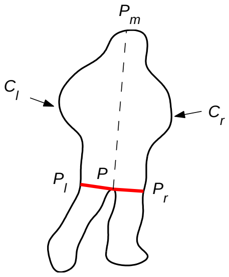


Figure 5: Computing the cuts passing through point P

tio of the perimeter of the part (excluding the cut) to the length of the cut) is more efficient and robust to noise and partial occlusion of the object. Thus, we employ the protrusion of a part to evaluate its salience; the salience of a part increases as its protrusion increases.

In summary, we combine the short-cut rule and the salience requirement to constrain the other end of a cut. For example in Figure 3.1, let S be a silhouette, C be the boundary of S, P be a point on C with negative minima of curvature, and P_m be a point on C so that P and P_m divide the boundary C into two curves C_l, C_r of equal arc length. Then two cuts are formed passing through point P: $\overline{PP_l}$, $\overline{PP_r}$ such that points P_l and P_r lies on C_l and C_r, respectively. The ends P_l and P_r of the two cuts are located as follows:

$$\begin{aligned} & P_l = arg \min_{P'} |\overline{PP'}| \\ & \text{s.t.} \quad \frac{|\overset{\frown}{PP'}|}{|\overline{PP'}|} > T_p, P' \in C_l, \overline{PP'} \in S \end{aligned} \tag{4}$$

$$\begin{aligned} & P_r = arg \min_{P'} |\overline{PP'}| \\ & \text{s.t.} \quad \frac{|\overset{\frown}{PP'}|}{|\overline{PP'}|} > T_p, P' \in C_r, \overline{PP'} \in S \end{aligned} \tag{5}$$

where $\overset{\frown}{PP'}$ is the smaller part of boundary C between P and P', $|\overset{\frown}{PP'}|$ is the arc length of $\overset{\frown}{PP'}$, and $\frac{|\overset{\frown}{PP'}|}{|\overline{PP'}|}$ is the salience of the part bounded by curve $\overset{\frown}{PP_l}$ and cut $\overline{PP_l}$.

Eq. (4) means that point P_l is located so that the cut $\overline{PP_l}$ is the shortest one among all cuts sharing the same end P, lying within the silhouette with the other end lying on contour C_l, and resulting in a significant part whose salience is above a threshold T_p. The other point P_r is located in the same way using Eq. (5).

Since negative minima of curvature are obtained by local computation, their computation is not robust in real digital images. We take several computationally efficient strategies to reduce the effects of noise. First, a B-spline approximation is used to moderately smooth the boundary of a silhouette, since B-spline representation is stable and easy to manipulate locally without affecting the rest part of the silhouette. Second, the negative minima of curvature with small magnitude of curvature are removed to avoid parts due to noise or small local deformations. However, curvature is not scale invariant (e.g. its value doubles if the silhouette shrinks by half). One way to transform curvature into a scale-invariant quantity is to first find the chord joining the two closest inflections which bound the point, then multiply the curvature at the point by the length of this chord. The resulting normalized curvature does not change with scale — if the silhouette shrinks to half size, the curvature doubles but the chord halves, so their product is constant.

This analysis yields 2D body parts for a person in a single view. The torso is not found directly by this method as the body part segmentations can only find protruded parts reliably. Since these protruded parts can overlap, there are a large number of torsos that can be formed from the remaining part of the silhoette. Therefore, we do not attempt to find the torsos directly and simply infer it from the other body parts.

Zhao [22] has used a similar method to develop a system for body part identification from a single view. However, body part identification from a single view is very difficult and labelings are often incorrect,

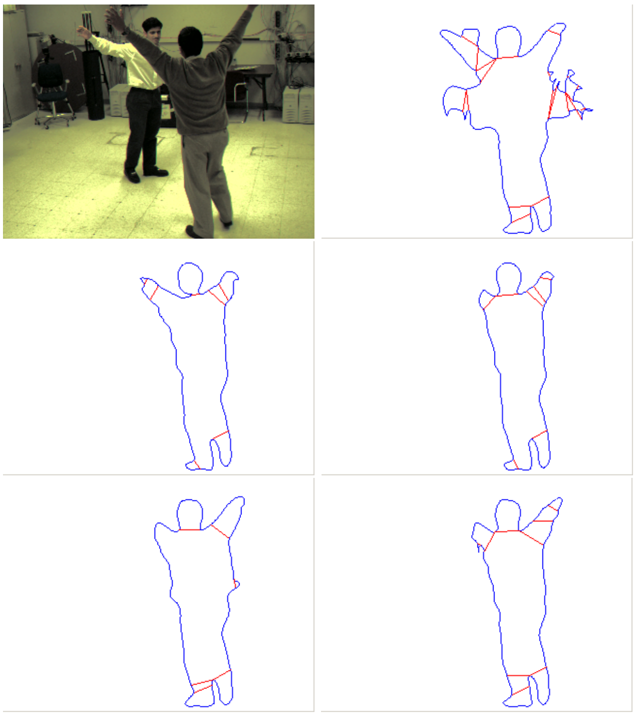


Figure 6: Multiple Body parts obtained using the segmentations shown in Fig. 3

especially in the case of partial-occlusions and self-occlusions where some body parts are not visible. The problem is also underconstrained since depth information is not available. Another difficulty is that their system requires extraction of good silhouettes which are not easy to obtain in a dense scene. As opposed to Zhao's work, we use multiple cameras and identify body pose in 3D using a global analysis.

3.2 Computing Body-part Primitives in 3D

2D part primitives obtained using Silhouette Analysis are used to obtain part primitives in 3D. First, parts that are relatively close to each other are combined with each other. Second, the decomposed parts are matched across views using epipolar geometry to yield 3D body parts. The two endpoints of a part in one view are matched to the corresponding endpoints in the other view. The matching is based on simply lying on the corresponding epipolar line. An additional constraint that can be used is the color profile of the body parts. The disadvantage is that if the viewpoints are substantially different, the color profiles can vary significantly. Also, the color profiles for different body parts can be very similar (for e.g. the two legs can have very similar color profiles.)

Once matching is done, a certain number of body parts are selected based on their matching score and their end points are projected in space to yield 3D body parts.

4 Assembly Evaluation using the Observation Likelihood

Labelings are assigned to these 3D parts by building an assembly that has the maximum likelihood according to an appropriate likelihood function. From the set of 3D body parts, we form sets of possible heads, hands and legs based on size constraints. Additional knowledge, if available, can be used. Such information might consist of the knowledge that the legs are close to the floor or that the person is standing (constraint on head and hand positions). Then, the problem reduces to finding a head, two hands (or a single or no hands, if not found) and two legs (or 0 or 1 legs), such that the assembly has the highest likelihood. The likelihood function we use is described in the next section.

4.1 Observation Likelihood

In order to evaluate a particular assembly $\mathcal{A}$, we determine the observation likelihood $Pr(I_1, I_2, ..., I_n/\mathcal{A})$, which is the likelihood of observing images $I_1, I_2, ..., I_n$ given the particular assembly $\mathcal{A}$. Assuming that assemblies have equal priors, the assembly having the highest likelihood is also the assembly with the highest posterior. Since we do not know the body pose of other people in the scene, the observation likelihood cannot be determined unless the problems of body pose determination of different people are coupled with one another. This leads to an exponential increase in the complexity of the algorithm.

We can decouple the problem, however, if make some simplifying assumptions. Specifically, we can use the method developed in M$_2$Tracker[13] to determine priors using presence probabilities. Then, the general formula for the observation probability at a particular pixel x can be written as:

$$p(I(x)) = \sum_j P_{prior}(j) Pr(I(x)/j) \qquad (6)$$

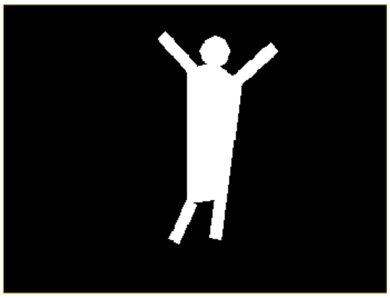


Figure 7: Determining the Projection of an Assembly

where the summation is done over all persons j and the background, and $I(x)$ is the observation at pixel x. If the location of the assembly is given, the function $L_j(x)$ (Presence Probability defined in section 2.1.2) for the person under consideration changes from a probabilistic to a fixed function so that:

$$L_j(x) = \begin{cases} 1 & \textit{if assembly projects to pixel x} \\ 0 & \textit{if it does not project to pixel x} \end{cases} \tag{7}$$

Using this definition, one can redetermine the priors for all people using equation (3) and calculate the observation probability using equation (6). This would be the conditional probability $Pr(I(x)/\mathcal{A})$. Assuming that observations at different pixels are independent, the overall observation probability is then simply the product of the observation probabilities at each pixel in each view.

$$Pr(I_1, I_2, ..., I_n/\mathcal{A}) = \prod_{i=1}^{n} \prod_{\text{all pixels x}} Pr(I(x)/\mathcal{A}) \tag{8}$$

In order to determine the projection of the assembly on an image, we model the hands and legs as cylinders with approximate widths and the head as a sphere and determine their projections onto a view (Figure 7). The torso is built by filling in the polygon formed by taking the joint locations of the (five) parts as the vertices. More accurate projection can be formed by building a 3D structure based on the joint locations and finding its projection onto the views. That will, however, add to the running time of the algorithm.

M_2Tracker determines the probability $Pr(I(x)/j)$ used in equation (6) using color models at different height slices. This puts only occupancy constraints on the likelihood. However, apart from the hypothesis that the given assembly projects to a particular pixel, we also have information as to which part of the assembly projects to the pixel. Using this information, we can improve results by including in the likelihood function information available from the views about possible body part locations. For e.g., we might be able to find the head using a face detector. If we have a skin detector, we might want to exclude the torso from the set of body parts that can give rise to it. In the present work, we include an additional term in the likelihood $Pr(I(x)/j)$.

First, we determine the probability that a particular body part has a particular aspect ratio $Pr_{bp}(ar)$. This probability is modeled as a 1D Gaussian, its mean and standard deviation learnt using training data. Now, we consider body parts detected from the silhouettes extracted for the person and find their aspect ratios. Finding the value of the function $Pr_{bp}(ar)$, we assign this value to all pixels belonging to the part in the silhouette. Since the torso is not observed directly, we cannot determine this probability for pixels belonging to it and hence they are assigned a constant value. Since we have multiple silhouettes and hence multiple probability estimates for the aspect ratio at a given pixel, we average them to yield a single result. For pixels lying outside any silhouette, the probability is zero. This will yield the function $Pr(ar/bp)$ for each pixel x and each body part bp. During evaluation of an assembly, we can compute the value of this function since we know the projections of the body parts onto the image. This probability value can be multiplied with the color likelihood to yield the likelihood function $Pr(I(x)/j)$ used in equation (6).

5 Searching for the Optimal Assembly

We believe that the best assembly can only be found by an exhaustive search in $O(n^5)$ time (where $n \sim O(10)$) is the number of possible primitives for each part). However, in practive, we have found that the same result can be obtained in $O(n)$ time if we have a good initial estimate of the body part positions , and in $O(n^2)$ time during the initialization phase. We first describe the incremental scheme.

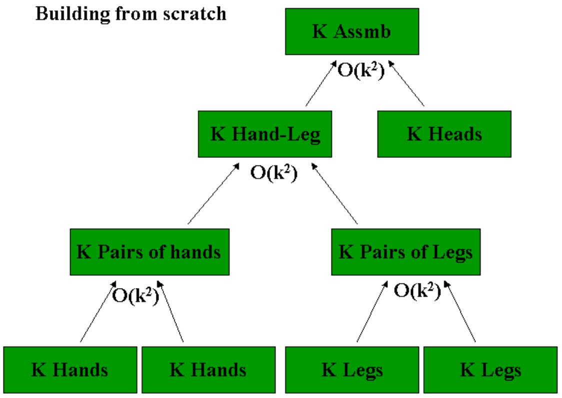


Figure 8: Schematic for the Initialization procedure

5.1 *Incremental Algorithm*

If we have a sufficiently good estimate of the current body part locations, we use a greedy approach. The idea is to first try to replace each part with candidate parts. If the assembly with the original part has a higher likelihood than the ones with any of the new primitives, we keep the original one. This is repeated for different parts. We have found that, apart from being very fast, this method yields the best results (better than initialization method) since it is often the case that some body parts have no good candidates at a particular time step, in which case we can keep the old estimate.

5.2 *Initialization*

In order to find an initial solution, or reinitialize the method if the incremental method fails, we use the following approach. First, we try to find good leg pairs. We find K best pairs (in $O(K^2)$ time) based on the likelihood function by building an assembly of just the two legs. Similarly, we find K best pairs of hands. Next, we find K best assemblies consisting of two hands and two legs using the hand and leg pairs found earlier (Figure 8). For this step, we construct the torso using the four joint locations. Finally the head is added and the best assembly is found. Although this method does not find the optimal assembly, we have found that it is extremely effective in practice and with the right choice of K, yields results very close to an exhaustive search.

If computational cost is available, we can find the result using both algorithms, taking the assembly with the higher likelihood as the answer.

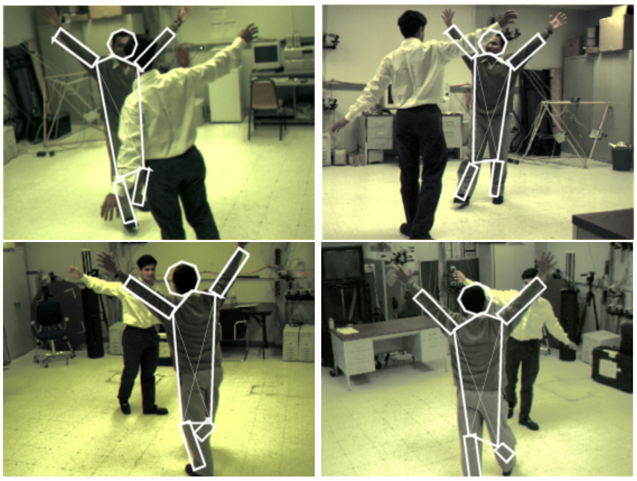


Figure 9: Results of the algorithm for a person at a particular time instant from multiple perspectives. Note how the person's body parts are correctly detected even though he is partially occluded from some views.

6 Results

We have obtained promising results for the algorithm. We tested our algorithm on a 5-perspective sequence with two people partially occluding each other in several views. We were able to correctly identify the body parts of the people when they were extended from the body. When the parts were close to the body, the algorithm labeled the part as missing and correctly identified the other parts. Figure 9 shows the result obtained at a particular time instant for the sequence. Figure 10 shows the results over time from a particular view. No initialization was done, nor any exact 3D model of the person specified. The algorithm took about 10s/frame on a Dual 933MHz Pentium III processor where most of the time was spent in evaluating different assemblies.

7 Summary and Conclusions

We have presented an algorithm for body pose estimation that does not require any initializations or models to be specified upfront and is able to work in a crowded

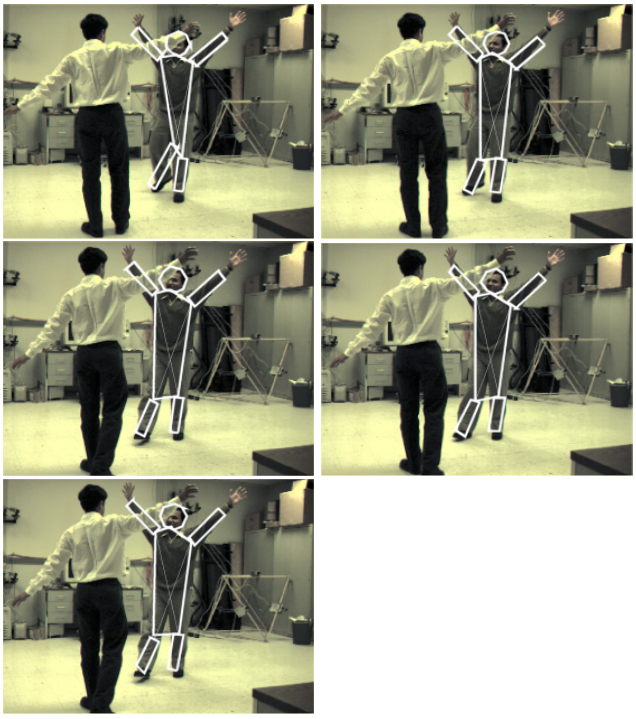


Figure 10: Results for five frames of the sequence.

scene so that occlusions - both full and partial - are present. These features make it especially useful for many surveillance applications. In the future, we wish to investigate more cues for body parts in an image (other than the silhouettes) like edge maps and texture regions, which might help us to reduce the number of cameras required to obtain a certain quality of results.

References

[1] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[2] Q. Cai and J.K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *Pattern Analysis and Machine Intelligence*, 21(11):1241–1247, November 1999.

[3] G.K.M. Cheung, T. Kanade, J.Y. Bouguet, and M. Holler. A real-time system for robust 3d voxel reconstruction of human motions. In *CVPR*, pages 714–720, 2000.

[4] Kiam Choo and David J. Fleet. People tracking using hybrid monte carlo filtering. In *International Conference on Computer Vision*, 2001.

[5] Quentin Delamarre and Olivier D. Faugeras. 3d articulated models and multi-view tracking with silhouettes. In *ICCV (2)*, pages 716–721, 1999.

[6] D. DiFranco, T.-J.Cham, and J.M.Rehg. Reconstruction of 3-d figure motion from 2-d correspondences. In *IEEE Computer Vision and Pattern Recognition, Kauai, Hawaii*, December 2001.

[7] R. Fablet and M.J. Black. Automatic detection and tracking of human motion with a view-based representation. In *ECCV02*, page I: 476 ff., 2002.

[8] P. Felzenszwalb and D. Huttenlocher. Efficient matching of pictorial structures. In *Computer Vision and Pattern Recognition*, pages 66–75, 2000.

[9] G. Gavrila and L. Davis. 3d model-based tracking of humans in action: a multi-view approach. In *CVPR*, pages 73–80, 1996.

[10] Donald D. Hoffman and Manish Singh. Salience of visual parts. *Cognition*, 63:29–78, 1997.

[11] Kakadiaris I.A. and D. Metaxas. Three-dimensional human body model acquisition from multiple views. *International Journal of Computer Vision*, 30(3):191–218, 1998.

[12] Sergey Ioffe and David Forsyth. Human tracking with mixtures of trees. In *International Conference on Computer Vision*, pages 690–695, 2001.

[13] A. Mittal and L.S. Davis. M_2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In *European Conference on Computer Vision*, page I: 18 ff., 2002.

[14] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *ECCV02*, page III: 666 ff., 2002.

[15] R. Ronfard, C. Schmid, and B. Triggs. Learning to parse pictures of people. In *ECCV02*, page IV: 700 ff., 2002.

[16] R. Rosales, M. Siddiqui, J. Alon, and S. Sclaroff. Estimating 3d body pose using uncalibrated cameras. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 2001.

[17] Hedvig Sidenbladh, Michael J. Black, and David J. Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV (2)*, pages 702–718, 2000.

[18] Yang Song, Xiaolin Feng, and Pietro Perona. Towards detection of human motion. In *CVPR*, pages 810–817, 2000.

[19] J. Sullivan and S. Carlsson. Recognizing and tracking human action. In *ECCV02*, page I: 629 ff., 2002.

[20] T.Drummond and Roberto Cipolla. Real-time tracking of the multiple articulated structures in multiple views. In *European Conference on Computer Vision*, 2000.

[21] M. Yamamoto, A. Sato, S. Kawada, T. Kondo, and Y. Osaki. Incremental tracking of human actions from multiple views. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 1998.

[22] Liang Zhao. *Dressed Human Modeling, Detection, and Parts Localization*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, July 2001.