Extracting Moving People from Internet Videos

Juan Carlos Niebles[1,2], Bohyung Han[3], Andras Ferencz[3], and Li Fei-Fei[1]

[1] Princeton University, Princeton NJ, USA
[2] Universidad del Norte, Colombia
[3] Mobileye Vision Technologies, Princeton NJ, USA

Abstract. We propose a fully automatic framework to detect and extract arbitrary human motion volumes from real-world videos collected from *YouTube*. Our system is composed of two stages. A person detector is first applied to provide crude information about the possible locations of humans. Then a constrained clustering algorithm groups the detections and rejects false positives based on the appearance similarity and spatio-temporal coherence. In the second stage, we apply a top-down pictorial structure model to complete the extraction of the humans in arbitrary motion. During this procedure, a density propagation technique based on a mixture of Gaussians is employed to propagate temporal information in a principled way. This method reduces greatly the search space for the measurement in the inference stage. We demonstrate the initial success of this framework both quantitatively and qualitatively by using a number of *YouTube* videos.

1 Introduction

Human motion analysis is notoriously difficult because human bodies are highly articulated and people tend to wear clothing with complex textures that obscure the important features needed to distinguish poses. Uneven lighting, clutter, occlusions, and camera motions cause significant variations and uncertainties. Hence it is no surprise that the most reliable person detectors are built for upright walking pedestrians seen in typically high quality images or videos.

Our goal in this work is to be able to *automatically* and *efficiently* carve out spatio-temporal volumes of human motions from arbitrary videos. In particular, we focus our attention on videos that are typically present on internet sites such as *YouTube*. These videos are representative of the kind of real-world data that is highly prevalent and important. As the problem is very challenging, we do not assume that we can find every individual. Rather, our aim is to enlarge the envelope of upright human detectors by tracking detections from typical to atypical poses. Sufficient data of this sort will allow us in the future to learn even more complex models that can reliably detect people in arbitrary poses. Two example sequences and the system output are shown in Fig. 1.

Our first objective is to find moving humans automatically. In contrast to much of the previous work in tracking and motion estimation, our framework does not rely on manual initialization or a strong *a priori* assumption on the

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 527–540, 2008.

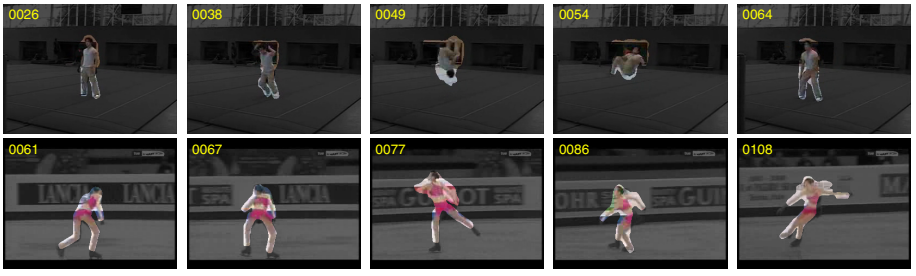


Fig. 1. Two example outputs. Our input videos are clips downloaded from *YouTube* and thus are often low resolution, captured by hand-held moving cameras, and contain a wide range of human actions. In the top sequence, notice that although the boundary extraction is somewhat less accurate in the middle of the jump, the system quickly recovers once more limbs become visible.

number of people in the scene, the appearance of the person or the background, the motion of the person or that of the camera. To achieve this, we improve a number of existing techniques for person detection and pose estimation, leveraging on temporal consistency to improve both the accuracy and speed of existing techniques. We initialize our system using a state-of-the-art upright pedestrian detection algorithm [1]. While this technique works well on average, it produces many false positive windows and very often fails to detect. We improve this situation by building an appearance model and applying a two-pass constrained clustering algorithm [2] to verify and extend the detections.

Once we have these basic detections, we build articulated models following [3,4,5] to carve out arbitrary motions of moving humans into continuous spatio-temporal volumes. The result can be viewed as a segmentation of the moving person, but we are not aiming to achieve pixel-level accuracy for the extraction. Instead, we offer a relatively efficient and accurate algorithm based on the prior knowledge of the human body configuration. Specifically, we enhance the speed and potential accuracy of [4,5] by leveraging temporal continuity to constrain the search space and applying semi-parametric density propagation to speed up evaluation.

The paper is organized as follows. After reviewing previous work in the area of human motion analysis in Section 1.1, we describe the overall system architecture in Section 2. Two main parts of our system, person detection/clustering and extraction of moving human boundaries, are presented in Sections 3 and 4, respectively. Finally, implementation details and experimental results are described in Section 5.

1.1 Related Work

Body Tracking. The most straightforward method to track humans is to consider them as blobs and use generic object tracking methods such as [6,7]. More complex methods attempt to model the articulation of the body

[8,9,10,11,12,13,14,15]. Most of these methods rely on a manual initialization, strong priors to encode the expected motion, a controlled or very simple environment with good foreground/background separation, and/or seeing the motion from multiple cameras.

Pedestrian Detection and Pose Estimation. Several fairly reliable pedestrian detection algorithms have been developed recently [1,16,17,18,19,20]. However, these methods typically deal with upright persons only, and the detection accuracy is significantly reduced by even moderate pose variations. Furthermore, these algorithms offer little segmentation of the human, providing only a bounding box of the body.

To model body configurations, tree shaped graphical models have shown promising results [3,4,5]. These generative models are often able to find an accurate pose of the body and limbs. However, they are less adept at making a discriminative decision: is there a person or not? They are typically also very expensive computationally in both the measurement and inference steps.

We build on these models and address the discrimination problem by initializing detections with an upright person detector. To improve computational efficiency, our algorithm exploits temporal information and uses more efficient semi-parametric (Gaussian mixture) representations of the distributions.

Based on similar intuitions, [21] uses temporal information to reduce the search space progressively in applying pictorial structures to videos. Ren et al. [22] takes another approach to human pose estimation in videos by casting the figure tracking task into a foreground/background segmentation problem using multiple cues, though the algorithm seems to rely on objects having a high contrast with the background.

2 System Architecture

Our system consists of two main components. The first component generates object-level hypotheses by coupling a human detector with a clustering algorithm. In this part, the state of each person, including location, scale and trajectory, is obtained and used to initialize the body configuration and appearance models for limb-level analysis. Note that in this step two separate problems – detection and data association – are handled simultaneously, based on the spatio-temporal coherence and appearance similarity.

The second component extracts detailed human motion volumes from the video. In this stage, we further analyze each person's appearance and spatio-temporal body configuration, resulting in a probability map for each body part. We have found that we can improve both the robustness and efficiency of the algorithm by limiting the search space of the measurement and inference around the modes of the distribution. To do this, we model the density function as a mixture of Gaussians in a sequential Bayesian filtering framework [23,24,25].

The entire system architecture is illustrated in Fig. 2. More details about each step are described in the following two sections.

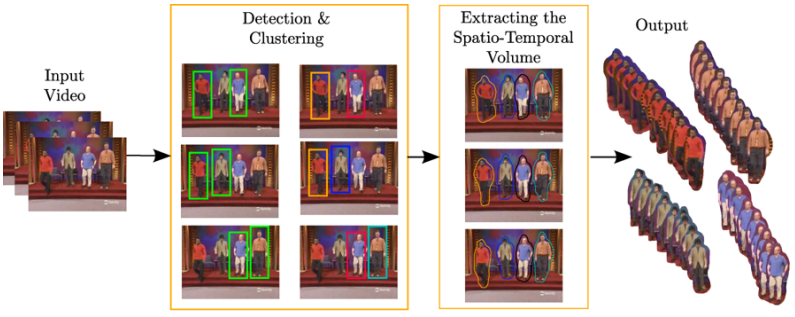


Fig. 2. Overall system

The focus of our work is to extract arbitrarily complex human motions from *YouTube* videos that involve a large degree of variability. We face several difficult challenges, including:

1. Compression artifacts and low quality of videos
2. Multiple shots in a video
3. Unknown number of people in each shot or sequence
4. Unknown human motion and poses
5. Unknown camera parameters and motion
6. Background clutter, motion and occlusions

We will refer back to these points in the rest of the paper as we describe how the components try to overcome them.

3 People Detection and Clustering

As Fig. 2 shows, our system begins with a step to estimate location, scale, and trajectories of moving persons. This step is composed of the following two parts.

3.1 Initial Hypothesis by Detection

We first employ an human detection algorithm [1] to generate a large number of hypotheses for persons in a video. This method, which trains a classifier cascade using boosting of HOG features to detect upright standing or walking people, has serious limitations. It only detects upright persons and cannot handle arbitrary poses (challenge 4). The performance is degraded in the presence of compression artifacts (challenge 1). Moreover, since it does not use any temporal information, the detection is often inconsistent and noisy, especially in scale. It is, therefore, difficult to reject false positives and recover miss-detections effectively. The complexity increases dramatically when multiple people are involved (challenge 3). This step, therefore, serves only as an initial hypotheses proposal stage. Additional efforts are required to handle various exceptions.

3.2 People Clustering

The output of the person detector is a set of independent bounding boxes; there are no links for the same individual between detections. The detections also have significant noise, false alarms and miss-detections especially due to the low quality of the video (challenge 1). In order to recover from these problems, we incorporate a clustering algorithm based on the temporal and appearance coherence of each person. The goal of clustering in our system is to organize all correct detections into groups, where each corresponds to a single person in the sequence (challenge 3), while throwing away false alarms. To achieve this, we apply a constrained clustering paradigm [2] in two hierarchical stages, adding both positive (should link) edges and negative (can not link) constraints between the detections. See Fig. 3 for an example.

Stage 1. In the first stage, we focus on exploiting the temporal-coherence cue by associating detections from multiple frames with the help of a low-level tracking algorithm [7]. When the first detection is observed, a low-level tracker is initialized with the detected bounding box. A new detection in a consequent frame is assigned to an existing track if it coherently overlaps with the tracker predictions. In this case, we reinitialize the tracker with the associated detection bounding box. When no existing track can explain the new detection, a new track is created. Due to the complexity of the articulated human body, a low-level tracker is susceptible to drift from the person. We thus limit the temporal life of the tracker by counting the number of frames after the last detection and terminating the track at the last detection if the maximum gap (e.g. 100 frames) is surpassed. Very small clusters with few detections are discarded. The clusters produced in this first stage are almost always correct but over-segmented tracks (see Fig. 3 (b)). This is because the person detector often fails to detect a person in the video for many frames in a row – especially when the person performs some action that deviates from an upright pose.

Stage 2. The stage 2 agglomerative constrained clustering views the stage 1 clusters as atomic elements, and produces constraints between them with positive weights determined by appearance similarity and negative constraints determined by temporal/positional incompatibility.

For the appearance similarity term, we select multiple high-scoring detection windows for each stage 1 cluster, and generate probability maps for the head and torso locations using a simple two-part pictorial structure [4]. We use these results to (1) remove false detections by rejecting clusters that have unreliable head/torso estimation results (e.g., high uncertainty in the estimated head and torso locations), and (2) generate a weighted mask for computing color histogram descriptors for both the head and the torso. The appearance of the person in each cluster is then modeled with the color distributions of head and torso.

After the second pass of our hierarchical clustering, we obtain one cluster per person in the sequence. Fig. 3 (c) illustrates the final clustering result, which shows that three different persons and their trajectories are detected correctly, despite the fact that the appearance of these individuals are very similar (Fig. 3 (d)).

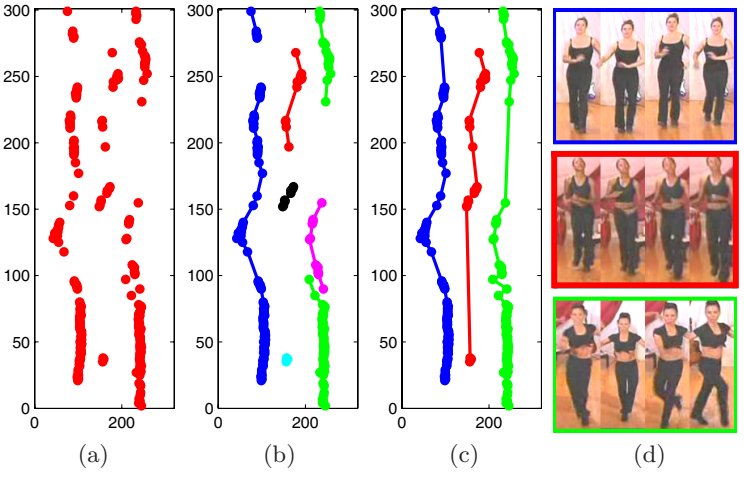


Fig. 3. Human detection and clustering result. From noisy detections, three tracks of people are identified successfully by filling gaps and removing outliers. (In this figure, the horizontal and vertical axis are the x locations and frame numbers, respectively.) (a) Original detection (b) Initial clusters after step 1 (c) Final clusters (d) Example images of three similar people that correctly clustered into different groups.

4 Extracting Spatio-temporal Human Motion Volume

We now have a cluster for each person, with a detection bounding box giving the location, scale, and appearance in some subset of the frames. Our goal is to find the body configuration for all the frames of the cluster (challenge 4), both where we have detections and where we do not. In this section, we discuss how to extract human body pose efficiently in every frame.

The existing algorithms for human motion analysis based on belief propagation such as [3,5] typically require exhaustive search of the input image because minimal (or no) temporal information is employed for the inference. Our idea is to propagate the current posterior to the next frame for the future measurement.

4.1 Overview

We summarize here the basic theory for the belief propagation and inference in [3,4]. Suppose that each body part p_i is represented with a 4D vector of $(x_i, y_i, s_i, \theta_i)$ – location, scale and orientation. The entire human body B is composed of m parts, i.e. $B = \{p_1, p_2, \ldots, p_m\}$. Then, the log-likelihood given the measurement from the current image I is

$$L(B|I) \propto \sum_{(i,j)\in E} \Psi(p_i - p_j) + \sum_i \Phi(p_i) \tag{1}$$

where $\Psi(p_i - p_j)$ is the relationship between two body parts p_i and p_j, and $\Phi(p_i)$ is the observation for body part p_i. E is a set of edges between directly connected

body parts. Based on the given objective function, the inference procedure by message passing is characterized by

$$M_i(p_j) \propto \sum_{p_j} \Psi(p_i - p_j) O(p_i) \tag{2}$$

$$O(p_i) \propto \Phi(p_i) \prod_{k \in C_i} M_k(p_i) \tag{3}$$

where $M_i(p_j)$ is the message from part p_i to p_j, $O(p_i)$ is the measurement of part p_i, and C_i is a set of children of part p_i. The top-down message from part p_j to p_i for the inference is defined by

$$P(p_i|I) \propto \Phi(p_i) \sum_{p_j} \Psi(p_i - p_j) P(p_j|I), \tag{4}$$

which generates the probability map of each body part in the 4D state.

Based on this framework, we propose a method to propagate the density function in the temporal domain in order to reduce search space and temporally consistent results. The rest of the section describes the details of our algorithm.

4.2 Initialization

The first step for human body extraction is to estimate an initial body configuration and create a reliable appearance model. The initial location of the human is given by the method presented in Section 3. Note that the bounding box produced by the detection algorithm does not need to be very accurate since most of the background area will be removed by further processing. Once a potential human region is found, we apply a pose estimation technique [4] based on the same pictorial structure and obtain the probability map of the configuration of each body part through the measurement and inference step. In other words, the output of this algorithm is the probability map $P_p(u, v, s, \theta)$ for each body part p, where (u, v) is location, s is scale and θ is orientation. A sample probability map is presented in Fig. 4 (b)-(d). Although this method creates accurate probability maps for each human body part, it is too computationally expensive to be used in video processing. Thus, we adopt this algorithm only for initialization.

4.3 Representation of Probability Map

The original probability map P_p is represented by a discrete distribution in 4D space for each body part. There are several drawbacks of the discrete density function. First of all, it requires a significant amount of memory space, which is proportional to the image size and granularity of the orientations and scales, even if most of the pixels in the image have negligible probabilities. Second, the propagation of a smooth distribution is more desirable for the measurement in the next step since a spiky discrete density function may lose a significant number of potentially good candidates by sampling.

Instead of using the non-parametric and discrete probability map, we employ a parametric density function. However, finding a good parametric density function is not straightforward, especially when the density function is highly multi-modal as in human body. In our problem, we observe that the probability map for each orientation is mostly uni-modal and close to a Gaussian distribution[1]. We employ a mixture of N Gaussians for the initialization of human body configuration, where N is the number of different orientations.

Denote by $\mathbf{x}_i^{(k)}$ and $\omega_i^{(k)}$ $(i = 1, \ldots, n)$ the location and weight of each point in the k-th orientation probability map. Let $\theta^{(k)}$ be the orientation corresponding the k-th orientation map. The mean $(\mathbf{m}^{(k)})$, covariance $(\mathbf{P}^{(k)})$ and weight $(\kappa^{(k)})$ of the Gaussian distribution for the k-th orientation map is then given by

$$\mathbf{m}^{(k)} = \begin{pmatrix} \mathbf{x}^{(k)} \\ \theta^{(k)} \end{pmatrix} = \begin{pmatrix} \sum_i \omega_i^{(k)} \mathbf{x}_i^{(k)} \\ \theta^{(k)} \end{pmatrix} \tag{5}$$

$$\mathbf{P}^{(k)} = \begin{pmatrix} \mathbf{V_X} & \mathbf{0} \\ \mathbf{0}^\top & V_\theta \end{pmatrix} = \begin{pmatrix} \sum_i \omega_i^{(k)} (\mathbf{x}_i^{(k)} - \mathbf{m}^{(k)})(\mathbf{x}_i^{(k)} - \mathbf{m}^{(k)})^\top & \mathbf{0} \\ \mathbf{0}^\top & V_\theta \end{pmatrix} \tag{6}$$

$$\kappa^{(k)} = \sum_i \mathbf{x}_i^{(k)} / \sum_k \sum_i \mathbf{x}_i^{(k)} \tag{7}$$

where $\mathbf{V_X}$ and V_θ are (co)variance matrices in spatial and angular domain, respectively. The representation of the combined density function based on the entire orientation maps is given by

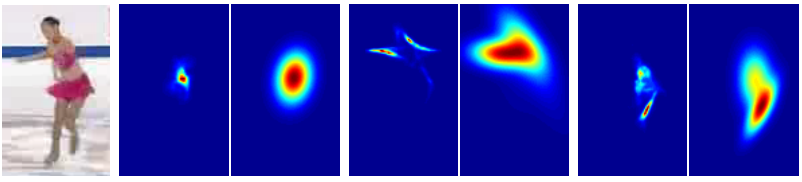
$$\hat{f}(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}} \sum_{i=1}^{N} \frac{\kappa^{(k)}}{\mid \mathbf{P}^{(k)} \mid^{1/2}} \exp\left(-\frac{1}{2} D^2 \left(\mathbf{x}, \mathbf{x}^{(k)}, \mathbf{P}^{(k)} \right) \right) \tag{8}$$

where $D^2\left(\mathbf{x}, \mathbf{x}^{(k)}, \mathbf{P}^{(k)}\right)$ is the Mahalanobis distance from $\mathbf{x}$ to $\mathbf{x}^{(k)}$ with covariance $\mathbf{P}^{(k)}$.

Although we simplify the density functions for each orientation as a Gaussian, it is still difficult to manage them in an efficient way especially because the number of components will increase exponentially when we propagate the density to the next time step. We therefore adopt Kernel Density Approximation (KDA) [26] to further simplify the density function with little sacrifice in accuracy. KDA is a density approximation technique for a Gaussian mixture. The algorithm finds the mode locations of the underlying density function by an iterative procedure, such that a compact mixture of Gaussians based on the detected mode locations is found.

Fig. 4 presents the original probability map and our approximation using a mixture of Gaussians for each body part after the pose estimation. Note that the approximated density function is very close to the original one and that the multi-modality of the original density function is well preserved.

[1] Arms occasionally have significant outliers due to their flexibility. A uni-modal Gaussian fitting may result in more error here.



(a) (b) torso (c) upper-left arm (d) upper-left leg

Fig. 4. Comparison between the true probability map for the pose estimation (left in each sub-figure) and its Gaussian mixture approximation (right) for each body part. The approximated density functions are propagated for the measurement in the next time step. Note that our approximation results look much wider since different scales in the color palette are applied for better visualization.

4.4 Measurement, Inference and Density Propagation

Fast and accurate measurement and inference are critical in our algorithm. As shown in Eq. (2) and (3), the bottom-up message is based on all the information up to the current node as well as the relative configuration with the parent node. Exhaustive search is good for generating the measurement information at all possible locations. However, it is very slow and, more importantly, the performance for the inference may be affected by spurious observations; noisy measurement incurred by an object close to or moderately far from the real person may corrupt the inference process. A desirable reduction of search space not only decreases computation time, but also improves the accuracy. The search space for measurement and inference is determined by a probability density function characterizing potential state of human body, where a mixture of Gaussians are propagated in sequential Bayesian filtering framework [23,24,25].

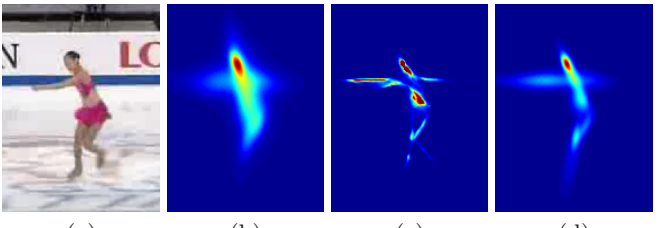
In our method, we perform local search based on the spatio-temporal information. We first diffuse the posterior density function from the previous frame, which is done analytically thanks to the Gaussian mixture representation. Based on the diffused density, locally dense samples are drawn to make measurements and a discrete density function is constructed. Note that inference is performed using the discrete density function. But a parametric representation of density function is propagated to the next time step for the measurement. After the inference, the pose estimation density function is converted to a mixture of Gaussians by the method described in Section 4.3. The posterior is given by the product of the diffused density and the pose estimation density function in the current frame. This step is conceptually similar to the integration of the measurement and inference history (temporal smoothing). We denote by $\mathbf{X}$ and $\mathbf{Z}$ the state and observation variable in the sequential Bayesian filtering framework, respectively. The posterior at the time step t of the state is given by the product of two Gaussian mixture as follows:

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \propto p(\mathbf{Z}_t|\mathbf{X}_t)p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) \tag{9}$$

$$= \left(\sum_{i=1}^{N_1} \mathcal{N}(\kappa_i, \mathbf{x}_i, \mathbf{P}_i)\right)\left(\sum_{j=1}^{N_2} \mathcal{N}(\tau_j, \mathbf{y}_j, \mathbf{Q}_j)\right), \tag{10}$$

Algorithm 1. Moving human body extraction

1: Apply human detection algorithm to a sequence
2: Apply clustering algorithm based on the detection. Create the initial body configuration and appearance at the first detection. Also, obtain the number of people in the video.
3: Construct pose estimation density function for each body part based on a mixture of Gaussians in the first frame, where it is also used as the posterior.
4: **while** not the end of sequence **do**
5: Go to the next frame
6: Diffuse the posterior of the previous frame
7: Perform the measurement and inference with the locally dense samples
8: Create a Gaussian mixture with the discrete pose estimation distribution
9: Compute the posterior by multiplying diffusion and pose estimation density
10: **if** there exists the detection of the same person **then**
11: Reinitialize the appearance and body configuration of the person (optional)
12: **end if**
13: **end while**



(a) (b) (c) (d)

Fig. 5. Density functions in one step of the human motion extraction. (a) Original frame (cropped for visualization) (b) Diffused density function (c) Measurement and inference results (d) Posterior (Note that the probability maps for all orientations are shown in a single image by projection.)

where $\mathcal{N}(\cdot)$ represents a Gaussian distribution with parameters of weight, mean, and covariance. The first and second terms in the right hand side represent diffusion and pose estimation density function, respectively. Note that the product of two Gaussian mixtures is still a Gaussian mixture, but it causes the exponential increase of the number of components. So KDA is required again to maintain a compact representation of the density function.

The density propagation algorithm for inference is summarized in Algorithm 1, and illustrated in Fig. 5.

5 Experiments

In order to evaluate our proposed approach, we have collected a dataset of 50 sequences containing moving humans downloaded from *YouTube*. The sequences contain natural and complex human motions and various challenges mentioned

Table 1. Precision-Recall Table: Performance comparison

	Detection only			Detection & Clustering			Full model		
	Prec	Rec	F	Prec	Rec	F	Prec	Rec	F
Rate	0.89	0.31	0.46	0.89	0.30	0.45	0.83	0.73	**0.78**
	0.90	0.25	0.39	0.91	0.24	0.38	0.87	0.62	**0.72**
	0.92	0.19	0.32	0.92	0.19	0.32	0.86	0.51	**0.64**
	0.93	0.16	0.27	0.94	0.15	0.27	0.92	0.43	**0.58**
	0.94	0.13	0.24	0.94	0.13	0.23	0.88	0.32	**0.46**

in Section 2. Many videos have multiple shots (challenge 2), so we divide the original videos into several pieces based on the shot boundary detection, which is performed by global color histogram comparison with threshold [27]. We deal with each shot as a separate video. We have made this dataset public and it can be found at `http://vision.cs.princeton.edu/projects/extractingPeople.html`.

Instead of 4D state space for human body configuration, 3D state space for location and orientation is utilized and scale is determined based on the detection size. Although person detector is not so accurate in scale estimate, the extraction algorithm is robust enough to handle some variations of the scale. Also, the gaps between detections are not generally long, and it is not often the case that we observe significant change in scale between two detections.

The measurement is based on edge template and color histogram as in [4], but search space for the measurement is significantly reduced. Fig. 5 (b) illustrates the search space reduction, where low density areas are not sampled for the observations.

We evaluate the retrieval performance of our system in terms of the precision-recall measures. For each sequence, we have generated ground-truth by manually labeling every human present in each frame with a bounding box. We compare the precision-recall rates at three stages of our system: pedestrian detection only [1], people detection and clustering, and the full model. For a fixed threshold of the human detector, we obtain the three precision-recall pairs in each row of Table 1. Our full system provides the highest performance in terms of the F-measure[2]. This reflects the fact that our system achieves much higher recall rates by extracting non-upright people beyond the pedestrian detections.

We also evaluate the performance of our system in terms of the segmentation of the moving people. We create ground-truth for the spatial support of the moving people in the form of binary masks. We have labeled a random sample of 122 people from our 50 sequences. The evaluation of the pose estimation is performed at frames t_d, t_d+5 and t_d+10, where t_d is a frame containing a pedestrian detection, and no detections are available in $[t_d+1, t_d+10]$. The average accuracies are 0.68, 0.68 and 0.63 respectively. Note that the accuracy decrease in the extracted person mask is moderate, and the temporal error propagation is small.

[2] The F-measure is defined [28] as: $2 \cdot (precision \cdot recall)/(precision + recall)$.

Fig. 6. Experimental results for various sequences. Each row corresponds to a separate sequence and two failure examples are illustrated in the last two rows. Please visit `http://vision.cs.princeton.edu/projects/extractingPeople.html` for more sample videos.

The results for several *YouTube* videos are presented in Fig. 6. Various general and complex human motions are extracted with reasonable accuracy, but there are some failures that are typically caused by inaccurate measurements. In a PC with a 2.33 GHz CPU, our algorithm requires around 10-20 seconds for the measurement and inference of each person, one order of magnitude faster than the full search method of [4].

6 Conclusion and Future Work

We presented a method to *automatically* extract human motion volumes from natural videos. Our system achieves promising results although many improvements can still be made. Our future work is to make detection/tracking and pose estimation module interact more closely to create positive feedback and improve the quality of estimation. Currently, the measurement is based only on the top-down pictorial structure, but we plan to incorporate bottom-up cues for more robust and efficient processing. We also aim to build a large data set with detailed labeling for human motion, which would be very helpful resource for human motion analysis research [29,30,31].

References

1. Laptev, I.: Improvements of object detection using boosted histograms. In: BMVC, Edinburgh, UK, vol. III, pp. 949–958 (2006)
2. Klein, D., Kamvar, S.D., Manning, C.D.: From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In: ICML (2002)
3. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. IJCV 61, 55–79 (2005)
4. Ramanan, D.: Learning to parse images of articulated objects. In: NIPS, Vancouver, Canada (2006)
5. Ramanan, D., Forsyth, D., Zisserman, A.: Tracking people by learning their appearance. PAMI 29, 65–81 (2007)
6. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. IJCAI, 674–679 (1981)
7. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: CVPR, Hilton Head, SC, vol. II, pp. 142–149 (2000)
8. Cham, T., Rehg, J.: A multiple hypothesis approach to figure tracking. In: CVPR, Fort Collins, CO, vol. II, pp. 219–239 (1999)
9. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: CVPR, Hilton Head, SC (2000)
10. Han, T.X., Ning, H., Huang, T.S.: Efficient nonparametric belief propagation with application to articulated body tracking. In: CVPR, New York, NY (2006)
11. Haritaoglu, I., Harwood, D., Davis, L.: W4: Who? When? Where? What? - A real time system for detecting and tracking people. In: Proc. of Intl. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, pp. 222–227 (1998)
12. Lee, C.S., Elgammal, A.: Modeling view and posture manifolds for tracking. In: ICCV, Rio de Janeiro, Brazil (2007)

13. Sigal, L., Bhatia, S., Roth, S., Black, M., Isard, M.: Tracking loose-limbed people. In: CVPR, Washington DC, vol. I, pp. 421–428 (2004)
14. Sminchisescu, C., Triggs, B.: Covariance scaled sampling for monocular 3D body tracking. In: CVPR, Kauai, Hawaii, vol. I, pp. 447–454 (2001)
15. Sminchisescu, C., Kanaujia, A., Li, Z., Metaxas, D.: Discriminative density propagation for 3d human motion estimation. In: CVPR, San Diego, CA, vol. I, pp. 390–397 (2005)
16. Leibe, B., Seemann, E., Schiele, B.: Pedestrian detection in crowded scenes. In: CVPR, San Diego, CA, vol. I, pp. 878–885 (2005)
17. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, San Diego, CA, vol. I, pp. 886–893 (2005)
18. Tuzel, O., Porikli, F., Meer, P.: Human detection via classification on riemannian manifolds. In: CVPR, Minneapolis, MN (2007)
19. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: ICCV, Nice, France, pp. 734–741 (2003)
20. Wu, B., Nevatia, R.: Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In: ICCV, Beijing, China, vol. I, pp. 90–97 (2005)
21. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR, Anchorage, AK (2008)
22. Ren, X., Malik, J.: Tracking as repeated figure/ground segmentation. In: CVPR, Minneapolis, MN (2007)
23. Arulampalam, S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. IEEE Trans. Signal Process. 50, 174–188 (2002)
24. Doucet, A., de Freitas, N., Gordon, N.: Sequential Monte Carlo Methods in Practice. Springer, Heidelberg (2001)
25. Han, B., Zhu, Y., Comaniciu, D., Davis, L.: Kernel-based bayesian filtering for object tracking. In: CVPR, San Diego, CA, vol. I, pp. 227–234 (2005)
26. Han, B., Comaniciu, D., Zhu, Y., Davis, L.: Sequential kernel density approximation and its application to real-time visual tracking. PAMI 30, 1186–1197 (2008)
27. Lienhart, R.: Reliable transition detection in videos: A survey and practitioner's guide. International Journal of Image and Graphics 1, 469–486 (2001)
28. Van Rijsbergen, C.J.: Information Retreival. Butterworths, London (1979)
29. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: ICCV, Beijing, China, pp. 1395–1402 (2005)
30. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: ICCV, Beijing, China, pp. 166–173 (2005)
31. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. IJCV 79, 299–318 (2008)