International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 27, No. 7 (2013) 1355012 (22 pages)

DOI: 10.1142/S0218001413550124

World Scientific
www.worldscientific.com

A UNIFIED FRAMEWORK FOR JOINT VIDEO PEDESTRIAN SEGMENTATION AND POSE TRACKING

YANLI LI, ZHONG ZHOU* and WEI WU

State Key Laboratory of Virtual Reality Technology and Systems
Beihang University, Beijing 100191, P. R. China
**zz@vrlab.buaa.edu.cn*

Received 12 July 2012
Accepted 21 August 2013
Published 8 October 2013

Pedestrian segmentation and pose tracking are performed to infer human silhouettes and skeletons, respectively. Although the two tasks are complementary in nature, few works have been done on combining them together to improve each other, and some related methods are limited to still images. In this paper, we propose an approach to jointly solving them in monocular videos via a unified framework. Basically, the framework is built on EM-based maximum likelihood estimation, in which pose tracking is fulfilled through Bayesian filtering using body silhouette as an observation cue, and pedestrian segmentation is inferred by guided filtering with constraint of body skeleton. The two sets of parameters are alternatively updated along the video. In the initialization of the framework, we utilize a hierarchical shape matching scheme to obtain the silhouette and skeleton in the first frame. Experiments on challenging pedestrian datasets verify the approach's effectiveness to cluttered backgrounds, moving camera and various articulated bodies, and the performance is improved significantly by solving the two tasks together.

Keywords: Pedestrian segmentation; pose tracking; hierarchical shape matching; particle filtering; guided filtering.

1. Introduction

In the past decades, pedestrian segmentation and pose tracking remain hot topics due to their versatile applications in areas of surveillance, human–computer interaction, humanoid robot, computer animation and so on. Pedestrian segmentation is to infer human silhouettes, i.e. to determine which pixels in the image are generated by the foreground human, and pose tracking is to estimate human skeletons, i.e. to parse human body into several parts and estimate each part's motion. Human body analysis is notoriously difficult since body parts are highly articulated and people are dressed with various clothes that obscure the important cues to distinguish body

*Corresponding author.

parts. Uneven lighting, cluttered and dynamic backgrounds cause more variations and uncertainties.

Although pedestrian segmentation and pose tracking are often studied independently, they are in fact inseparable. For example, most pose tracking methods, e.g. Refs. 30 and 34, take human segmentation as a preprocessing step. This precondition restricts the use of these techniques to static backgrounds where good segmentation is made available by background substraction.[7,15] On the other hand, object segmentation without any prior knowledge is a well-known ill-posed problem. It is no surprise that most reliable segmentation methods employ higher level knowledge (such as object category,[18] shape,[10] or interactive scribbles[20,33]) to make the segmentation problem well posed. Realizing the complementary merits of pedestrian segmentation and pose estimation, some authors have combined them together to iteratively update each other.[5,17,22] However, these combined methods are only limited to still images. The question is then, how to extend these methods to handle video pedestrian segmentation and pose tracking using all available information.

In our early work,[21] we have tackled the two problems together within a unified framework. The main idea is to sequentially estimate human pose via a physics-based Bayesian filtering, and perform pedestrian segmentation by solving a Markov random field (MRF) energy function. The major limitation of that method is high computation time. This paper is built on the early work.[21] For speeding up, we make two advancements: (1) instead of using inefficient mincut[4] to solve the energy function, we introduce guided filtering[16] for pedestrian segmentation, which has been proven more efficient for handling MRF-based vision problems[27]; (2) instead of performing shape matching with all templates in the initial stage, we employ a hierarchical template matching scheme, which can prune out some unmatched templates and reduce computation time.

The structure of this paper is organized as follows. After reviewing the related works in Sec. 2, we outline the framework in Sec. 3. The stages of framework initialization, pose tracking and pedestrian segmentation are described in Secs. 4, 5 and 6, respectively. Experimental results are demonstrated in Sec. 7, and we conclude the paper in the last section.

2. Related Works

Pedestrian segmentation and pose tracking lie in the fields of video object segmentation and tracking, respectively. In this context, simultaneous video object segmentation and tracking is the most related theme to our work, referring to the problem of sequentially segmenting foreground objects in an unannotated video, where pedestrian is the major actor. Existing methods can be roughly classified into contour-based methods[12,25,26,32] and region-based methods.[1,3,20,23,33,35]

Contour-based methods take object boundary as the main discriminative cue, in which template matching[12,32] and active contour[25,26] are two popular techniques. Template matching methods segment objects by matching the edge maps of the

frames with shape templates. For example, Gavrila[12] performs global template matching using Distance Transformation and Chamfer matching. The matching involves simultaneous parameter estimation and edge matching. Thayananthan *et al.*[32] present a method for shape tracking by building one-to-many mapping from image features to 3D templates, in which a multivariate relevance vector machine is learned to select a sparse set of templates. Template matching methods,[12,32] however, require to store a set of templates. Considering the high variability in the shape and appearance of pedestrians, it is hard for the limited templates to capture all detailed information for a particular pedestrian.

Active contour methods,[25,26] also called level set, extract objects by attaching points of the edge maps to body boundaries using a global energy function in a level-set space. Over recent years, active contour-based methods have become very popular for object segmentation. For pedestrian segmentation, Niebles *et al.*[26] present a method which combines top-down prior model and bottom-up appearance cues to extract human regions. Temporal propagation of the identified regions is performed with bottom-up cues in a level-set framework, which takes advantage of the sparse top-down information. The framework of Mitzel *et al.*[25] employs an efficient level-set tracker in order to follow individual pedestrian over time. This low-level tracker is initialized and periodically updated by a pedestrian detector and is kept robust through a series of consistency checks. However, the pedestrians with weak edges often cause edge-based active contour methods to leak out into surrounding area. Furthermore, for highly- articulated pedestrians, the body boundaries tend to disappear in the case of self-occlusion, and hence it may produce drifting problem.

Region-based methods group foreground pixels together to formulate object regions. They can further be divided into interactive methods[20,33] and appearance-based methods.[1,3,23,35] Interactive methods require users to provide interactions like bounding boxes or scribbles for initialization. Those hard constraints confine object segmentation within the user-selected regions. For example, Li *et al.*[20] provide brush tools for users to control object boundaries, in which objects are extracted via performing 3D graph cut-based segmentation on spatial-temporal video volumes. Similarly, Wang *et al.*[33] provide a painting-based user interface that allows users to easily indicate foreground objects across space and time. The mincut solver with hierarchical mean-shift superpixels is utilized to optimize the MRF function. Obviously, the interactive figure-ground methods involve cumbersome interactions for indicating the figure and ground regions.

Appearance-based approaches segment foreground regions by representing object appearances with probabilistic models and classifying foreground pixels into individuals based on these models. In Aeschliman's method,[1] the appearance model learnt in the previous frame is used in the next frame to identify which pixels have high probabilities of being part of the target. The ensemble tracker[3] is based on the similar idea but uses AdaBoost classifier to determine which pixels make up the target. Considering that object segmentation without any prior knowledge is an ill-posed problem, some top-down cues are introduced. Malcolm *et al.*[23] use a distance penalty

term to constrain objects in regions of interest, and hence human segmentation is biased to remain in the object area. Various object category-specific methods, e.g. Ref. 35, utilize prior object shapes to obtain object-like segmentation. Basically, our framework belongs to appearance-based video object extraction. The main difference compared with other appearance-based methods[1,3,23,35] is the utilization of pose information to constrain human regions, which can be considered as strong top-down information, and thus the framework can clearly extract human bodies, avoiding the commonly existing drift problem.

3. Problem Formulation

Given an input video with multiple pedestrians, the goal of our approach is to segment pedestrians and estimate their pose. We first obtain human bounding box sequences, each encircling an individual human by the tracking-by-detection method,[14] and then handle each sequence respectively as follows.

Mathematically, defining the sequence of pedestrian frames by $\{I_t\}, t = 1, \ldots, T$, T is the frame number, the task is to infer three pieces of information $\Phi_t = \{\Omega_t, \Delta_t, \Theta_t\}$. Ω_t specifies the segment matte, in which $\Omega_t(x) \in \{0, 1\}$ indicates that the pixel x belongs to the foreground ($\Omega_t(x) = 1$) or the background ($\Omega_t(x) = 0$). Δ_t denotes the pose parameter set, which will be described in detail in Sec. 5.1. Θ_t is the latent appearance parameter set. In our work, the observation data include the color and motion parts, i.e. $I_t = \{I_t^c, I_t^m\}$, and hence the appearance parameters involve two sets of latent parameters, $\Theta_t = \{\Theta_t^c, \Theta_t^m\}$, which are used to model the color and motion distributions respectively. We formulate the task as computing the maximum a posterior (MAP) in a first-order MRF, such that:

$$\begin{aligned} \Phi_t^* &= \arg\max_{\Phi_t} p(\Phi_t | I_1, \ldots, I_t, \Phi_1^*, \ldots, \Phi_{t-1}^*) \\ &= \arg\max_{\Phi_t} p(\Phi_t | I_1, \ldots, I_t, \Phi_{t-1}^*). \end{aligned} \tag{1}$$

Obviously, maximizing the above posterior with respect to all parameters is intractable as the state space is expensively huge. Considering that the three sets of parameters are interrelated, we have presented a unified framework[21] to iteratively solve them. This framework is based on the early one,[21] but makes two improvements to reduce computation time: (1) using hierarchial shape matching to replace brute-force matching in the initial stage; (2) performing pedestrian segmentation with guided filtering instead of mincut. More specifically, the framework (as shown in Fig. 1) is performed as follows:

1. **E-Step I (Initialization):** The initial pose and latent parameters of the first frame are estimated with a hierarchical shape matching scheme, which will be stated in Sec. 4.
2. **M-Step (Body segmentation):** The segment matte Ω_t is derived by guided filtering[16] under the constraints of $\{\Delta_t, \Theta_t\}$, which will be stated in Sec. 6.

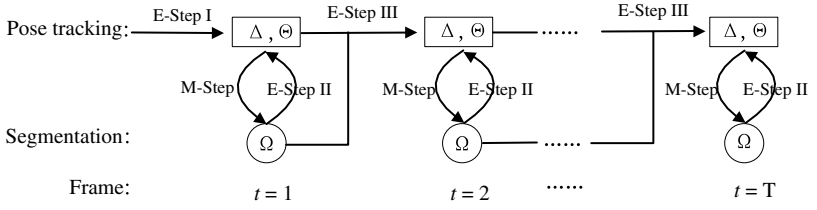


Fig. 1. The unified framework for pedestrian segmentation and pose tracking.

3. **E-Step II (Updating the latent parameters):** Based on the refined segment Ω_t, the latent parameters Θ_t are re-estimated.
4. **E-Step III (Pose tracking):** We predict the pose and latent parameters $\{\Delta_{t+1}, \Theta_{t+1}\}$ in frame $t+1$ through a Bayesian filtering process using the previous segment matte Ω_t and parameters $\{\Delta_t, \Theta_t\}$, which will be described in Sec. 5.
5. **Repeat the above steps 2, 3, 4 along the video**.

4. Hierarchical Shape Matching

Shape matching is used to search the best matched template from a set of templates and extract pedestrian model in the first frame. In general case, it is the edge map instead of the original image to be aligned with the templates. In our approach, the edge map is given by the Pb edge detector,[24] which encodes the real valued magnitude and orientation information. We use Chamfer distance to measure the similarity between the templates and the edge maps. Given two point sets, $E = \{e\}$ for the edgelets of the edge map, $S = \{s\}$ for the sample points of the shape template, the Chamfer distance is a function of relative position p:

$$d(S, E, p) = \frac{1}{|S|} \sum_{s \in S} \min_{e \in L(s+p)} (d_1(s, e, p) + \alpha * d_2(s, e, p)), \tag{2}$$

where $d_1(s, e, p) = \|(s + p) - e\|_2$, $d_2(s, e, p) = \hat{g}(e) + |\hat{o}(s + p) - \hat{o}(e)|$, $\hat{g}(\cdot)$ and $\hat{o}(\cdot)$ denote the normalized magnitude and orientation value of the Pb edge, respectively, α is a weighting constant. $L(s)$ is the one-dimensional normal line segment for the sample point s: $L(s) = \{l(i, s)|i = -M_S, \ldots, M_S\}$, where $l(i, s) = \langle i * \sin(o(s)), i * \cos(o(s))\rangle$, $2M_S + 1$ is the total length of the line segment (in pixels), and $o(\cdot)$ indicates the orientation value. The parameters in the above formula are empirically set to $\alpha = 3.0$ and $M_S = 10$.

To efficiently choose the best matched template, we organize the set of shape templates in a hierarchial tree, in which similar templates are grouped together and represented with a prototype. Note that all the templates and masks have been aligned and scale normalized. Taking each shape template as a node of an undirected complete graph (UCG) $G = \langle V, \hat{E}, W\rangle$, in which V is the node set, $\hat{E}$ is the edge set and W is the weighting set corresponding to the edge set. The edge weight is defined as: $w(i, j) = \hat{d}(i, j) + \hat{d}(j, i)$, $\hat{d}(i, j) = d(i, j, 0)$. The construction of the tree can be

considered as a problem of hierarchical graph clustering. This is a well-studied NP-hard problem in graph theory, involving some bottom-up clustering methods, e.g. Ref. 11, and top-down partition methods, e.g. Ref. 29. Here, we employ spectral clustering,[29] which partitions the graph into K subsets according to the normalized cut criterion:

$$N\text{cut}_K = \sum_{i=1}^{K} \frac{\text{cut}(A_i, V - A_i)}{\text{assoc}(A_i, V)}, \tag{3}$$

where $\text{assoc}(A, B) = \text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v)$.

An approximate solution is obtained using the K eigenvectors of the K largest eigenvalues in: $W\hat{X} = \hat{\lambda} DW$. Here $\hat{X}$ is the eigenvector matrix, $\hat{\lambda}$ is the eigenvalue matrix and $D(i, i) = \sum_j w(i, j)$ is a diagonal matrix.

Following spectral clustering,[29] we recursively divide the graph to construct a hierarchical tree. At first, nodes in the graph G are divided into K_1 subsets. Then, each subset is further divided into K_2 sub-subsets. The process is recursively implemented until the number of clustering nodes is lower than a constant value K_n. The prototype of each subset is taken as the template with the smallest mean similarity score to other templates in the subset. Taking each subset with its prototype as a subtree, the hierarchical tree is constructed.

As shown in Fig. 2, shape matching is implemented as a process of traversing the tree to find the best matched prototype. At the nonleaf level, it is the derived prototypes to be aligned with the edge map, whereas at the leaf level, all template exemplars are to be matched. If the similarity score of a prototype is above a threshold, all of its subtrees would not be visited, otherwise, the prototype is added to the list and its subtrees are traversed recursively. At last, we choose the template with the minimum similarity score in the visiting list as the best matched shape,

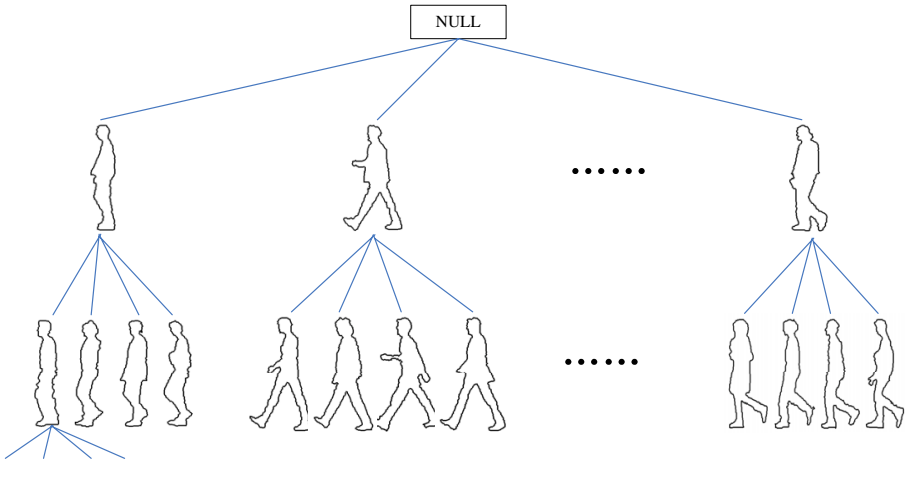
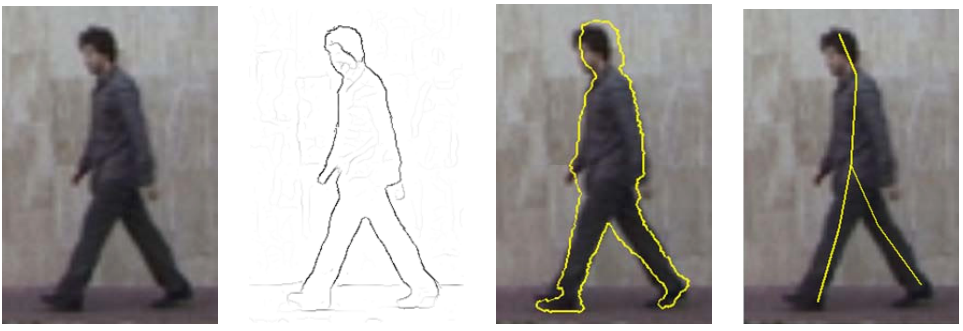


Fig. 2. The hierarchical shape tree.



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

Fig. 3. The results obtained after initialization. (a) The input image; (b) the obtained edge map is used for shape matching; (c) the derived silhouette (in yellow) of the matched template is overlaid on the image; (d) the skeleton (in yellow) of the matched template is transferred to the human body (color online).

meanwhile obtain the pedestrian silhouettes it represents. Besides, in the learning phase, we have manually clicked joints of all templates, which can be directly transferred to the frame for initializing the pose model Δ_1. Figure 3 demonstrates the shape matching results for an input frame, in which Fig. 3(b) is the Pb edge map of Fig. 3(a), Figs. 3(c) and 3(d) are the corresponding silhouette and skeleton generated after hierarchical shape matching.

5. Bayesian Filtering for Pose Tracking

Hierarchical shape matching provides initial pose information in the first frame. To further estimate spatial-temporal poses at the remaining frames, we utilize Bayesian filtering to find the best pose with the given observations. Under the assumption that the pose at time t is only dependent on the previous pose while the observations at time t are only dependent on the current pose, a recursive Bayes formula is derived as:

$$p(\Delta_t|I_{1:t}) \propto p(I_t|\Delta_t) \int p(\Delta_t|\Delta_{t-1}) p(\Delta_{t-1}|I_{1:t-1}) d\Delta_{t-1}, \tag{4}$$

where $p(I_t|\Delta_t)$ is the observation likelihood and $p(\Delta_t|\Delta_{t-1})$ is the temporal transition prior.

The multi-modality of the posterior distribution has pushed current methods towards sample techniques, in which we employ the mostly implemented particle filter. In the particle filter, the posterior is approximated by a finite set of particles with associated normalized weight $\{\Delta_t^i, \pi_t^i\}$. The posterior is generated through three steps: (1) the particles at time t are sampled from the posterior at time $t-1$, i.e. $\Delta_t^{(i)} \propto p(\Delta_t|\Delta_{t-1}^{(i)})$; (2) the weights at time t are updated according to the likelihood function, i.e. $\pi_t^{(i)} = \pi_{t-1}^{(i)} p(I_t|\Delta_t^{(i)})$; (3) normalizing all weights to make sure $\sum_i \pi_t^{(i)} = 1$. The approximated posterior is taken as $\sum_i (\pi_t^i * \Delta_t^i)$.

In the past decades, kinds of pose tracking methods have been presented based on the above general inference procedure. Their differences lie in the definition of the pose state, or the observation likelihoods, or the dynamic prior. Here, we build the prior mainly on the bipedal walking motion,[6] while present a novel observation likelihood in conjunction with a physical-based pose representation.

5.1. *Pose representation*

This physical-based pose is 2.5D, modeled with six rigid body regions, including the head, torso, upper/lower stance legs, upper/lower swing legs and parameterized with $\Delta = \{\Delta_f, \Delta_v\}$. $\Delta_f = \{sl, rp_1, rp_2, lu, ll\}$, as a fixed model, is set according to the skeleton in the first frame, in which sl indicates the walking step length, rp_1 and rp_2 are the relative positions of the head and neck joints with respect to the body center, lu and ll are the upper and lower leg lengths. Δ_v is a variation model, $\Delta_v = \{v, \theta_{hb}, \theta_{ut}, \theta_{uw}, d\theta_{ut}, d\theta_{uw}, \theta_{lt}, \theta_{lw}\}$. It is used to simulate human walking, in which v denotes the walking speed, θ_{hb} is the turning angle of the body, θ_{ut} and θ_{uw} are the angles for the upper stance and swing legs, respectively, $d\theta_{ut}$ and $d\theta_{uw}$ are the corresponding angular velocities for θ_{ut} and θ_{uw}, respectively, θ_{lt} and θ_{lw} are the angles for the lower stance and swing legs.

Using the variation model Δ_v, we build a dynamic process to simulate human walking, i.e. sampling the particle state $\Delta_t^{(i)}$ based on the prior $p(\Delta_t|\Delta_{t-1}^{(i)})$. We build the prior mainly on the 2D physical formulations,[6] in which v and θ_{hb} both follow the normal distributions, that is, $v_t \sim N(v_{t-1}, \sigma_v)$, $\theta_{hb,t} \sim N(\theta_{hb,t-1}, \sigma_{hb})$. $(\theta_{ut}, \theta_{uw}, d\theta_{ut}, d\theta_{uw}, \theta_{lt})$ are induced by sl and v according to the physical Motion Laws.[6] The lower swing leg angle θ_{lw} is initialized by the skeleton in the first frame and modeled as:

$$\theta_{lw,t} \sim N(\theta_{lw,t-1} + \varepsilon(\theta_{lw,t-1} - \theta_{lw,t-2})|\theta_{ut,t} - \theta_{uw,t}|, \sigma_{\theta_{wl}}). \tag{5}$$

Here, we use $\sigma_v = 7.0$, $\sigma_{hb} = 3.0$, $\varepsilon = 0.3$, $\sigma_{\theta_{wl}} = 8.0$.

5.2. *Observation likelihoods*

The observation likelihoods are derived with multiple cues, including the color cue I_t^c, the motion cue I_t^m and the silhouette cue I_t^s. We build an independent likelihood for each cue and combine all likelihoods together to form the final likelihood:

$$p(I_t|\Delta_t) = w_c p(I_t^c|\Delta_t) + w_m p(I_t^m|\Delta_t) + w_s p(I_t^s|\Delta_t). \tag{6}$$

Here, w_c, w_m and w_s are the weighting values corresponding to the likelihoods. We now describe the likelihoods according to the cue type in the following.

Color likelihood. The color likelihood is evaluated with a stable component p_s and a wandering component p_w. The likelihood for a new observation conditioned on the previous observation is formulated by:

$$p(I_t^c|\Delta_t) = \lambda_s p_s(I_t^c|I_1^c, \sigma_s^c) + \lambda_w p_w(I_t^c|I_{t-1}^c, \sigma_w^c),$$

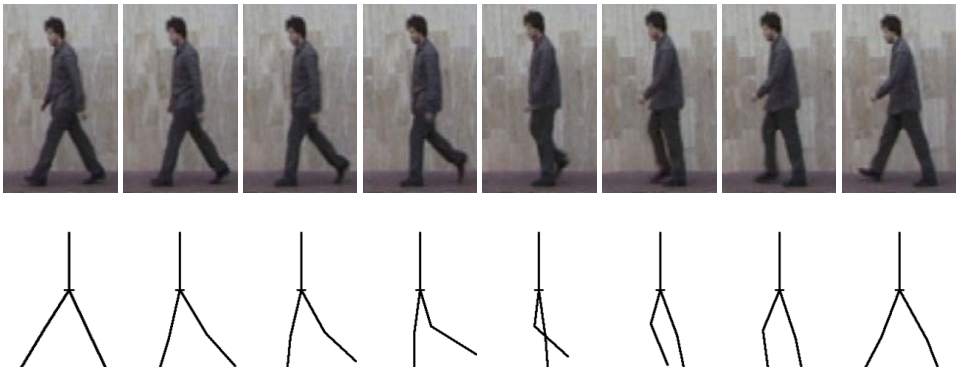


Fig. 4. The pose tracking results. The top row shows the input frames, and the second row demonstrates the corresponding poses.

where $p_s(\cdot)$ and $p_w(\cdot)$ both follow the Gaussian distributions. We set $\sigma_s^c = 0.5$, $\sigma_w^c = 0.5$, $\lambda_s = 0.8$ and $\lambda_w = 0.2$. I_1^c and I_{t-1}^c are the 3D Lab histograms of the leg regions determined by Δ_1 and Δ_{t-1}, respectively.

Motion likelihood. The motion likelihood is built on the motion field obtained by Ref. 31, which can provide the motion information about the tracked limbs between two successive frames. The likelihood of the motion cue is given by the mean square distance (MSD) of the projected positions $\{ps_i\}$ and the hypothesized positions $\{hs_i\}$ for a set of sample points:

$$p(I_t^m|\Delta_t) \propto \exp\left(-\sum_i \|ps_i - hs_i\|/\tilde{N}\right). \tag{7}$$

Here $\tilde{N}$ is the number of sample points.

Silhouette likelihood. Silhouette is a binary map indicating human foreground region, which is derived from the projection of previous silhouette Ω_{t-1} (see Sec. 6) with optical flow.[31] The negative likelihood for silhouette cue is calculated as the mean square error (MSE) of the predicted values $\{ss_i\}$ and the observed values $\{bs_i\}$ for a set of sample points inside the limb region.

$$p(I^s|\Delta_t) \propto \exp\left(-\sum_i \|ss_i - bs_i\|/\tilde{N}\right). \tag{8}$$

Here $\tilde{N}$ is the number of sample points.

Based on the above definitions, we use particle filter to sequentially estimate the pose states. Figure 4 demonstrates the tracking results for a walking cycle.

6. Guided Filtering for Segmentation

Existing object segmentation tasks are mostly formulated as global labeling problems. That is, given a collection of pixels $X = \{x_i\}$ in the image and the binary variant set $\{\Omega(x_i)\}$ associated with X, if x_i belongs to the foreground region,

$\Omega(x_i) = 1$, otherwise $\Omega(x_i) = 0$. Segmenting objects is to solve an energy function based on the MAP in Markov random field[19]:

$$E(X) = \sum_i \varphi(x_i) + \sum_{i,j} \psi(x_i, x_j). \tag{9}$$

$\varphi(x_i)$ is the data term which penalizes individual label that is not obeyed with some inherent models, and $\psi(x_i, x_j)$ is the smooth term which encourages neighboring pixels being assigned to the same label.

The energy function is then minimized via a global energy solver such as mincut.[4] A major limitation of such scheme is high computation. In this section, we utilize an alternative scheme — guided filtering[16] to tackle pedestrian segmentation. Comparing with global energy solutions, guided filtering is a local optimization method, which can be considered as a spatially smoothing operator. It achieves similar results to global methods under the assumption that the data term plays a dominant role in the energy function. Besides, the filtering has the ability to preserve edges.

The main idea of guided filtering is to perform edge-preserving smoothing under the guide of an input map, in which the filter output q is locally linear to the guidance map I: $q_i = a_x I_i + b_x, \forall i \in w_x$, where w_x is a window with radius r centered at the pixel x, (a_x, b_x) are the linear coefficients to be constant in w_x. By minimizing the difference between the filter input p and output q, i.e. $\mathrm{Err}(a_x, b_x) = \Sigma_{i \in w_x}((p_i - q_i)^2 + \epsilon a_x^2)$, we can obtain a_x, b_x and the filter out q.

Here, we take the unary potential $\varphi(x_i)$ as the filter input. It allows us to utilize multiple cues for human segmentation. In this work, four cues are integrated into the unary potential, referring to: (1) the color term $\varphi_c(x)$, (2) the motion term $\varphi_m(x)$, (3) the pose term $\varphi_p(x)$, and (4) the segment coherence term $\varphi_s(x)$, and thus the unary potential can be rewritten as:

$$\varphi(x) = \lambda_c \varphi_c(x) + \lambda_m \varphi_m(x) + \lambda_p \varphi_p(x) + \lambda_s \varphi_s(x), \tag{10}$$

where $\{\lambda_c, \lambda_m, \lambda_p, \lambda_s\}$ are the weighting values.

Color term. The color distribution across human body is typically compact, thus it is considered as a vital cue for segmentation. We define the color model as K-Means clusters: $\Theta_t^c = \{\mu_{k,t}^{c,J} | k = 1, \ldots, K_c, J \in \{B, F\}\}$, in which K_c is the color cluster number (set to 3 in experiments) and $\mu_{k,t}^{c,J}$ is the mean color of the cluster (k, J), B indicates the background while F indicates the foreground. The color term is defined by:

$$\varphi_c(x) = \begin{cases} d_c^F(x)/(d_c^F(x) + d_c^B(x)), & \Omega(x) = 1 \\ d_c^B(x)/(d_c^F(x) + d_c^B(x)), & \Omega(x) = 0. \end{cases}$$

Here, $d_c^J(x) = \min_k \|I_t^c(x) - \mu_{k,t}^{c,J}\|$, $I_t^c(x)$ denotes the color data in Lab color space.

Motion term. Pedestrians typically preserve relative motion with the static background scene. Motion cue, which is invariant to illumination changes, seems to

be more natural and robust. For building the motion models, we first obtain the motion field[31] by comparing the current frame with its subsequent frame, and then estimate the mean motion values within the foreground and background regions, obtaining the motion model $\Theta_t^m = \{\mu_t^{m,J} | J \in \{B, F\}\}$, in which $\mu_t^{m,J}$ is the mean motion value. The motion term is calculated as:

$$\varphi_m(x) = \begin{cases} d_m^F(x)/(d_m^F(x) + d_m^B(x)), & \Omega(x) = 1 \\ d_m^B(x)/(d_m^F(x) + d_m^B(x)), & \Omega(x) = 0. \end{cases}$$

Here, $d_m^J(x) = \|I_t^m(x) - \mu_t^{m,J}\|$, $I_t^m(x)$ involves the motion vector in pixel x.

Pose term. The pose cue ensures that pixels falling near to the skeleton would more likely be assigned with human label and vice versa. In our case the skeleton is modeled as a puppet of skeleton lines. As shown in Fig. 5(d), we use the distance field along the skeleton to represent the pose term. The pose term takes the form:

$$\varphi_p(x) = \min(\|x - q^*\|/(r_i|L_{q^*}|), 1.0). \tag{11}$$

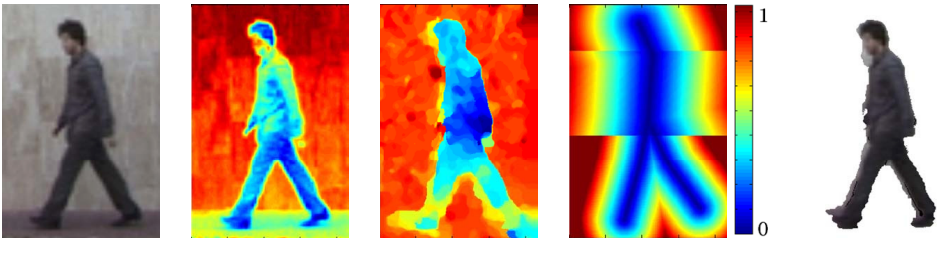
Here, $q^* = \arg\min_{q \in \{L_i\}} \|x - q\|$. $\{L_i | i = 1, \ldots, 6\}$ are the skeleton lines, indicating the head, torso, two upper legs and two lower legs. $|L_i|$ is the line length, $\{r_i\}$ is the width/height ratio for the skeleton region, empirically set to $\{1.0, 0.5, 0.34, 0.34, 0.3, 0.3\}$ in our experiments.

Segment coherence term. This term is used to maintain temporal coherence of segmentation along the video sequence, which is defined by:

$$\varphi_s(x) = \begin{cases} c_s, & \Omega(x) = \Omega(x') \\ 1 - c_s, & \Omega(x) \neq \Omega(x'), \end{cases}$$

where x' is the matched pixel of x in the subsequent frame, c_s is a constant value (empirically set to 0.3). Note that the coherence term $\varphi_s(x)$ in the first frame is unavailable.

Based on guided filtering,[16] we perform body segmentation with three steps: (1) obtaining the foreground likelihood map $L_F = \{\varphi(x)\}$ and the background likelihood map $L_B = \{1 - \varphi(x)\}$; (2) taking the grayscale image of I as the guidance map, the



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

Fig. 5. The human segmentation results. (a) the input frame; (b) the color map; (c) the motion map; (d) the pose map; (e) the final refined segmentation result.

two likelihood maps are filtered respectively (denoting the filter outputs as $\hat{L}_F(x)$ and $\hat{L}_B(x)$); (3) defining the body matte $\Omega(x_i)$ as: $\Omega(x_i) = \delta(\hat{L}_F(x_i) < \hat{L}_B(x_i))$.

At E-Step II, we re-estimate the mean values $\Theta_t = \{\mu_{k,t}^{c,J}, \mu_t^{m,J} | k = 1, \ldots, K_c, J \in \{B, F\}\}$ within the segment matte as follows: (1) sampling some pixels in the foreground and background regions individually; (2) the foreground and background pixels are clustered into K_c components using the K-Means method; (3) for each component, its parameters $\{\mu_{k,t}^{c,J}, \mu_t^{m,J}\}$ are statistically obtained.

The re-estimated parameters Θ_t are used to refine segmentation again. For further refining the extracted silhouette, we invoke the Bayesian matting[8] to soft-segment an eroded narrow region along the silhouette boundaries. Figures 5(b)–5(d) demonstrate the color, motion and pose terms to the input image (Fig. 5(a)), and Fig. 5(e) is the final refined segmentation result.

7. Experimental Results

To illustrate the performance of the proposed method, we apply it with the Ethz dataset,[2] the Weizmann dataset,[13] and some sequences we captured with a hand-hold camera. The Ethz dataset[2] consists of five pedestrian sequences, and the Weizmann dataset[13] is composed of 15 pedestrian sequences. Both of the two datasets have ground truth masks, which allows us to make quantitative comparison. The sequences we captured with a hand-hold camera are used to verify our framework's robustness to moving camera. In those sequences, all human windows are resized to 320 pixels in width. We use the Ethz dataset[2] with 200 shape templates in the initialization stage, and set the parameters as: $K_1 = K_2 = 4$ and $K_n = 10$, resulting in a 4-level tree. The other parameters are set as: guided filtering's parameters $r = 2$ and $\epsilon = 0.1$, the weighting values $\{\lambda_c, \lambda_m, \lambda_p, \lambda_s\} = \{0.3, 0.3, 0.2, 0.2\}$, $\{w_c, w_m, w_s\} = \{0.3, 0.3, 0.4\}$.

For quantitative evaluation, we measure the segmentation accuracy in form of *F-measure*.[a] $F\text{-}measure = 2 * pre * rec/(pre + rec)$, where *pre* is defined as the ratio of true positive pixels (i.e. pixels labeled as foreground actually belong to foreground) to all labeled foreground pixels, and *rec* is defined as the ratio of true positive pixels to ground truth pixels. The pose accuracy is estimated by the Mean Square Distance (MSD) between the lower body joints and the corresponding hand-marked joints.

7.1. *Performance comparison with the early work*

This work builds on our early work.[21] The main limitation of the early work[21] is the computational complexity. Here we make two improvements to reduce computation time: (1) using hierarchical shape matching to replace brute-force matching in the initial stage; (2) using guided filtering instead of mincut[4] for optimizing pedestrian segmentation. In order to verify these improvements, we compare this work with the early work[21] in terms of the two parts.

[a] http://www.dcs.gla.ac.uk/Keith/Preface.html.

7.1.1. *Hierarchical shape matching*

Given 200 templates, brute-force matching would require 200 correlations. The time needed for silhouette extraction by matching a 320 * 240 image with a template, on a 2.2 GHz CPU, 3 GB RAM machine, is about 0.18 seconds per frame. Thus, 36.0 s is expected to find the best matched template with brute-force matching. Instead, hierarchical shape matching manages to prune out most of significantly unmatched templates, typically about 15 templates are matched and the matching time is reduced to 2.7 s.

7.1.2. *Guided filtering-based optimization*

As illustrated in our early work,[21] mincut[4] achieves high quality results on pedestrian segmentation, while preserves a well-known drawback, i.e. lower efficiency. The main reason is that it employs time-consuming a-expansion and a-b swapping. In contrast, guided filtering[16] works as local smoothing with high efficiency. The time complexity is $O(N)$, N is the pixel number. To verify if guided filtering-based optimization can produce similar accuracy as mincut, we evaluate our approach on the Ethz and Weizmann datasets. The comparison results are demonstrated in Table 1. As can be seen, the guided filtering-based optimization scheme leads to accuracy comparable to those of mincut. Besides, we measured speed-ups of three times.

7.2. *Framework evaluation*

7.2.1. *Influence of silhouette and skeleton*

The main characteristic of the framework is the combination of pose tracking and pedestrian segmentation, in which silhouette produced by pedestrian segmentation serves as a useful cue for pose tracking, and skeleton produced by pose tracking is used to constrain human extraction. To verify if silhouette and skeleton have influence on pose tracking and pedestrian segmentation, respectively, we compared the performance of pose tracking with/without silhouette, and the performance of pedestrian segmentation with/without skeleton. As shown in Tables 2 and 3, pedestrian segmentation gets improvement (about 5%) with skeleton, and pose tracking can infer more correct limb configurations with silhouette.

Table 1. Comparison of the segmentation performances for mincut and guided filtering.

	Segmentation Accuracy		Running Time	
Method	Ethz	Weizmann	Ethz	Weizmann
Mincut[4]	89.6%	88.1%	9.7 s	10.2 s
Guided filtering[16]	90.0%	88.7%	3.0 s	3.1 s

Table 2. The segmentation accuracies (measured by *F*-Measure) obtained with and without skeleton.

Dataset	With Skeleton	Without Skeleton
Ethz	90.0%	85.7%
Weizmann	88.7%	83.2%

Table 3. The mean pose errors (measured by MSD) obtained with and without silhouette.

Dataset	With Silhouette	Without Silhouette
Ethz	5.1 pixels	9.5 pixels
Weizmann	6.4 pixels	11.7 pixels

7.2.2. *Comparison with several related works*

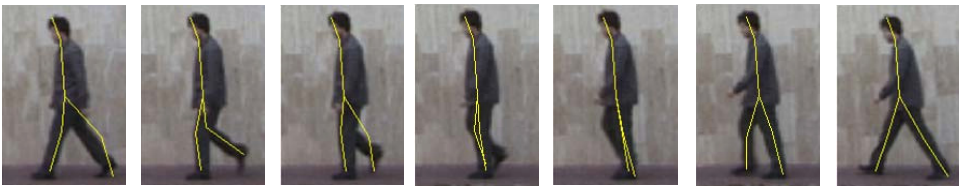
To further analyze the framework performance, we compared our method with several related methods, including NBest,[9] template matching[12] and GrabCut.[28] NBest[9] is used as a baseline for comparison on pose tracking, while GrabCut[28] and template matching[12] are used for comparison on pedestrian segmentation.

NBest[9] is a N-Best-based pose estimation method, which first generates multiple candidate body configurations, and then uses nonmaximal suppression cues to prune out near-identical configurations. Table 4 summarizes the quantitative evaluation of our method and NBest.[9] The qualitative comparison over one sequence is demonstrated in Fig. 6. The results show that our approach outperforms NBest,[9] especially when there are inter-occlusions between the two legs.

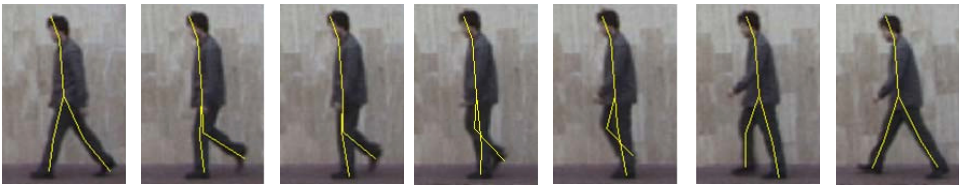
Template matching[12] performs silhouette extraction by matching the input image with a set of templates. It is an automatic foreground segmentation method, yet the segmentation results are sensitive to local variations since it does not consider local appearance. As shown in Fig. 7(b), the head and hip regions are typically inaccurately segmented by template matching. GrabCut[28] is an interactive MRF-based object cutout method, which commonly requires users to initially provide a bounding box and then draw figure and ground scribbles. Figure 7(c) demonstrates some figure and ground scribbles we drew on the frames, and Fig. 7(d) shows the corresponding results obtained by GrabCut.[28] Obviously, GrabCut[28] requires cumbersome interactions, and its results are sensitive to the interactions. Comparably, our framework can automatically extract human silhouettes (see Fig. 7(f)) based on the inferred

Table 4. Comparison of the mean pose errors (measured by MSD) for NBest and our method.

Method	Ethz	Weizmann
NBest[9]	21.2 pixels	18.5 pixels
Our method	5.1 pixels	6.4 pixels



(a)

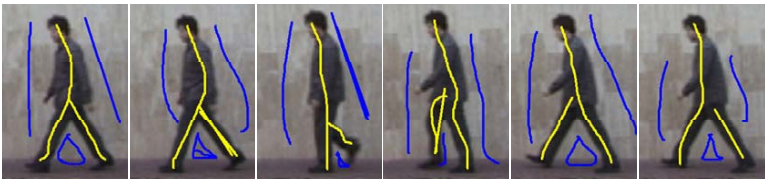


(b)

Fig. 6. Pose tracking results for NBest and our method. (a) NBest; (b) our method.

(a)

(b)



(c)

Fig. 7. Qualitative comparison for template matching, GrabCut and our method. (a) The input frames; (b) the silhouettes (in yellow) obtained by template matching are overlaid on the frames; (c) the foreground (in yellow) and background (in blue) scribbles drew on the frames, which are interactive inputs to GrabCut; (d) the results obtained by GrabCut based on the scribbles in (c); (e) the poses inferred by our framework; (f) the extracted pedestrians by our framework based on the poses in (e).

(d)

(e)

(f)

Fig. 7. (*Continued*)

Table 5. Comparison of the segmentation performances (measured by *F*-Measure) for template matching, GrabCut and our method.

Method	Ethz	Weizmann
Template matching[12]	82.2%	76.5%
GrabCut[28]	75.4%	85.5%
Our method	90.0%	88.7%

poses (see Fig. 7(e)). Table 5 summarizes the quantitative comparison of our method with GrabCut[28] and template matching.[12] To automatically evaluate GrabCut,[28] we provided no scribbles and used pedestrian windows as the bounding boxes. As can be seen, our method achieves best performance for the Ethz and Weizmann datasets.

7.3. *Robustness analysis*

We demonstrate the segmentation results for six sequences in Fig. 8, in which the first three sequences come from the Ethz dataset, and the others are from the Weizmann dataset. The first image of each row indicates the background environment where the pedestrian locates in. For saving space, we omit the remaining

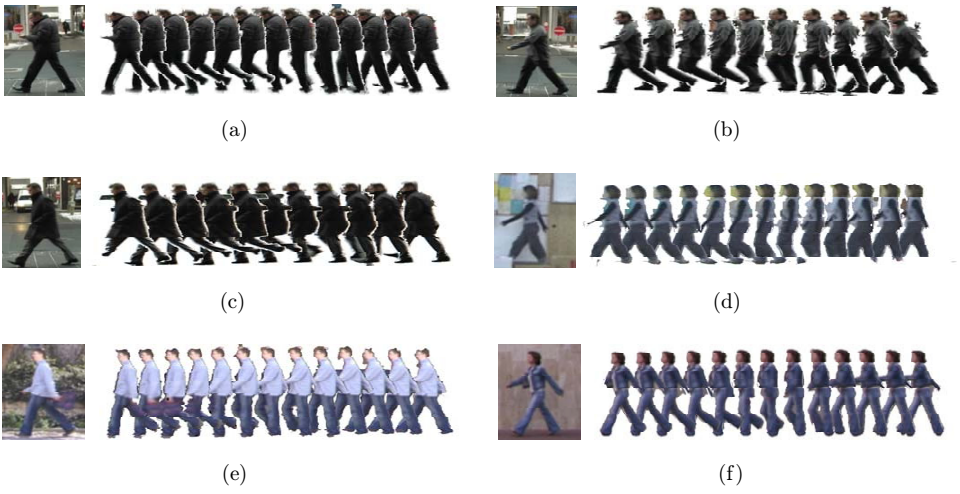


Fig. 8. Segmentation results for pedestrian sequences from the Ethz and Weizmann datasets.

frames. Following up is the segmentation results for the input sequence. As can be seen, although the human poses in the walking cycles are continuous varying under cluttered backgrounds, the coherent optimization of our framework ensures accurate segmentation across time. We attribute this to the utilization of pose information and pixel correspondence (i.e. motion field) along the sequences.

Compared with the background substraction technique,[7] one advantage of this framework is its robustness to moving camera. Theoretically, this is because the stages of hierarchical shape matching and pedestrian segmentation can extract silhouettes without constraint of static scene. Hierarchical shape matching performs body extraction in the first frame of each sequence by matching the edge map with a set of templates. It is used for still images and applied here to initialize the body information of the input sequences, no matter the sequences are captured by a static or moving camera. The stage of pedestrian segmentation obtains body silhouettes by optimizing foreground regions with multiple cues. Generally speaking, all the cues can be successfully inferred when the method handles moving cameras, and thus this stage can be applied to moving cameras directly. We conducted experiments over several sequences captured with a hand-hold camera. Figure 9 demonstrates the results on two sequences. From the demonstration, we can see that even the sequences contain arbitrary camera motion, our method still acquired clear human silhouettes. Yet if the frequency of the camera is too low, the optical flow will be unavailable, and the method will fail to estimate motion cues for silhouette extraction.

7.4. *Failure cases*

Our method works under the assumption that the pedestrian in the first frame of each sequence can be successfully extracted via hierarchical shape matching. Yet

(a) (b)

Fig. 9. Segmentation results for two pedestrian sequences captured by a hand-hold camera.

Fig. 10. Failure cases for segmenting pedestrians via hierarchical shape matching, in which the pedestrian silhouettes (in yellow) are overlaid on the images.

such an assumption may not hold in some cases. Figure 10 illustrates some fail cases of our method over the images from the Fudan-Penn dataset. As illustrated, those pedestrians were not successfully extracted under the cases of similar clothes color to the backgrounds, severe occlusion or cluttered backgrounds.

Fig. 11. Failure cases for segmenting pedestrians under occlusion.

Fig. 12. Failure cases for segmenting pedestrians with similar color.

Pedestrians tend to be occluded by each other in real scenes. In order to test if our method can be applied to multi-human segmentation, we considered two sequences with multiple pedestrians, in which temporal and spatial occlusions exist between the pedestrians. Figure 11 shows an example of segmenting pedestrians with severe occlusion, and Fig. 12 demonstrates an example for segmenting pedestrians with similar color. As illustrated, the pedestrians are wrongly segmented under severe occlusion in Fig. 11, and that the leg and head regions are often mislabeled in Fig. 12. This is expected since our method segments each pedestrian individually and takes no consideration of their occlusion.

8. Conclusion and Discussion

In this paper, we have proposed a method for joint pedestrian segmentation and pose tracking along monocular videos, in which pose tracking and pedestrian segmentation interact closely to create positive feedbacks to improve performance. This method extends the original EM style framework.[21] As described earlier, the major limitation of the original framework[21] is the computation time. For solving this problem two improvements are presented in this paper. First, using hierarchical shape matching to replace brute-force shape matching in the initial stage, our method can extract pedestrians by matching with only a few templates. Second, the guided filtering-based optimization scheme is employed to replace the inefficient mincut in the stage of pedestrian segmentation, which achieves comparable results and obtains speed-ups of several orders of magnitude. As to the future application, we hope this kind of method can be used to drive the 3D human animations to augment virtual reality systems.[36]

The main limitation of the method lies in its sensitiveness to occlusion, thus one possible direction is to extend the framework to deal with occlusion so that the further method can handle multiple pedestrians' segmentation in crowded scenes. To

develop a real-time system, the stages of pose tracking and pedestrian segmentation should be re-designed for implementation in parallel graphics hardware.

Acknowledgments

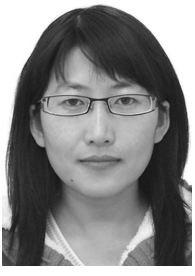
We thank Dr. Jiangjian Xiao for his insightful discussions. This work is supported by the National 863 Program of China under Grant No. 2012AA011801 and the Natural Science Foundation of China under Grant No. 61170188.

References

1. C. Aeschliman, J. Park and A. C. Kak, A probabilistic framework for joint segmentation and tracking, in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2010* (San Francisco, CA, USA, 2010), pp. 1371–1378.
2. M. Andriluka, S. Roth and B. Schiele, People-tracking-by-detection and people-detection-by-tracking, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2008* (Anchorage, Alaska, 2008), pp. 1–8.
3. S. Avidan, Ensemble tracking, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2005* (San Diego, CA, USA, 2005), pp. 494–501.
4. Y. Boykov, O. Veksler and R. Zabih, Fast approximate energy minimization via graph cuts, *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(11) (2001) 1222–1239.
5. M. Bray, P. Kohli and P. Torr, PoseCut: Simultaneous segmentation and 3D pose estimation of humans using dynamic graph-cuts, in *Proc. Euro. Conf. Computer Vision, ECCV 2006* (Graz, Austria, 2006), pp. 642–655.
6. M. A. Brubaker, D. J. Fleet and A. Hertzmann, Physics-based person tracking using the anthropomorphic walker, *Int. J. Comput. Vis.* **87**(1–2) (2010) 140–155.
7. S. Brutzer, B. Hoferlin and G. Heidemann, Evaluation of background subtraction techniques for video surveillance, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2011* (Colorado Springs, CO, USA, 2011), pp. 1937–1944.
8. Y. Y. Chuang, B. Curless, D. Salesin and R. Szeliski, A bayesian approach to digital matting, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2001* (Kauai, HI, USA, 2001), pp. 264–271.
9. P. Dennis and R. Deva, N-best maximal decoders for part models, in *Proc. IEEE Inter. Conf. Computer Vision, ICCV 2011* (Barcelona, Spain, 2011), pp. 2627–2634.
10. P. Etyngier, F. Segonne and R. Keriven, Shape priors using manifold learning techniques, in *Proc. IEEE Inter. Conf. Computer Vision, ICCV 2009* (Rio de Janeiro, Brazil, 2007), pp. 1–8.
11. P. F. Felzenszwalb and D. P. Huttenlocher, Efficient graph-based image segmentation, *Int. J. Comput. Vis.* **59**(2) (2004) 167–181.
12. D. M. Gavrila, A exemplar-based approach to hierarchical shape matching, *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(8) (2007) 1408–1421.
13. L. Gorelick, M. Blank, E. Shechtman, M. Irani and R. Basri, Actions as space-time shapes, *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(12) (2007) 2247–2253.
14. H. Grabner and H. Bischof, On-line boosting and vision, in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2006* (NY, USA, 2006), pp. 260–267.
15. B. K. Hamidreza, S. Y. Hadi and A. S. Seyed, Background estimation in kernel space, *Int. J. Pattern Recogn. Artif. Intell.* **25**(1) (2011) 1–35.
16. K. He, J. Sun and X. Tang, Guided image filtering, in *Proc. Euro. Conf. Computer Vision, ECCV 2010* (Heraklion, Crete, Greece, 2010), pp. 1–14.

17. M. P. Kumar, P. Torr and A. Zisserman, OBJ CUT, in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition, CVPR 2005* (San Diego, CA, USA, 2005), pp. 18–25.
18. D. Larlus and F. E. E. Jurie, Combining appearance models and markov random fields for category level object segmentation, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2008* (Anchorage, Alaska, USA, 2008), pp. 1–8.
19. S. Z. Li, Markov random field modeling in image analysis, 3rd ed. (Springer-Verlag, New York, 2009).
20. Y. Li, J. Sun and H. Y. Shum, Video object cut and paste, *ACM Trans. Graphics* **24**(3) (2005) 595–600.
21. Y. Li, Z. Zhou and W. Wu, Iterative pedestrian segmentation and pose tracking under a probabilistic framework, in *Proc. IEEE Inter. Conf. Robotics and Automation, ICRA 2012* (Minnesota, USA, 2012), pp. 1206–1211.
22. Z. Lin, L. S. Davis, D. Doermann and D. DeMenthon, An interactive approach to pose-assisted and appearance-based segmentation of humans, in *Proc. IEEE Int. Conf. Computer Vision, ICCV 2007* (Rio de Janeiro, Brazil, 2007), pp. 1–8.
23. J. Malcolm, Y. Rathi and A. Tannenbaum, Multi-object tracking through clutter using graph cuts, in *Proc. IEEE Int. Conf. Computer Vision, ICCV 2007* (Rio de Janeiro, Brazil, 2007), pp. 1–5.
24. D. Martin, C. Fowlkes and J. Malik, Learning to detect natural image boundaries using brightness and texture, *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(5) (2000) 530–549.
25. D. Mitzel, E. Horbert, A. Ess and B. Leibe, Multi-person tracking with sparse detection and continuous segmentation, in *Proc. Euro. Conf. Computer Vision, ECCV 2010* (Heraklion, Crete, Greece, 2010), pp. 397–410.
26. J. C. Niebles, B. Han and F. F. Li, Efficient extraction of human motion volumes by tracking, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2010* (San Francisco, CA, USA, 2010), pp. 655–662.
27. C. Rhemann, A. Hosni, M. Bleyer, C. Rother and M. Gelautz, Fast cost-volume filtering for visual correspondence and beyond, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2011* (Colorado Springs, CO, USA, 2011), pp. 3017–3024.
28. C. Rother, V. Kolmogorov and A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, *ACM Trans. Graphics* **27**(3) (2004) 309–314.
29. J. Shi and J. Malik, Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8) (2000) 888–905.
30. L. Sigal, A. O. Balan and M. J. Black, HUMANEVA: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion, *Int. J. Comput. Vis.* **70**(1–2) (2010) 41–54.
31. D. Sun, S. Roth and M. J. Black, Secrets of optical flow estimation and their principles, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2010* (San Francisco, CA, USA, 2010), pp. 2432–2439.
32. A. Thayananthan, R. Navaratnam, B. Stenger, P. H. S. Torr and R. Cipolla, Multivariate relevance vector machines for tracking, in *Proc. Euro. Conf. Computer Vision, ECCV 2006* (Graz, Austria, 2006), pp. 124–138.
33. J. Wang, P. Bhat, R. A. Colburn, M. Agrawala and M. F. Cohen, Interactive video cutout, *ACM Trans. Graphics* **24**(3) (2005) 585–594.
34. X. Xu and B. Li, Learning motion correlation for tracking articulated human body with a rao-blackwellised particle filter, in *Proc. IEEE Int. Conf. Computer Vision, ICCV 2007* (Rio de Janeiro, Brazil, 2007), pp. 1–8.

35. Z. Yin and R. T. Collins, Shape constrained figure-ground segmentation and tracking, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR 2009* (Miami, FL, USA, 2009), pp. 731–738.
36. Q. Zhao, A survey on virtual reality, *Sci. China F* **52**(3) (2009) 348–400.



Yanli Li received her B.S. and M.S. degrees in Computer Science from Shandong University, Shandong, China, in 2004 and 2007, respectively. She is currently a Ph.D. candidate at State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China. Her main interests lie in image and video processing, especially involving scene parsing, object segmentation and 3D modeling.

Zhong Zhou received his B.S. degree from Nanjing University, Nanjing, China in 1999, and his Ph.D. from Beihang University in 2004. He is currently an Associate Professor at State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. His main research interests include augmented virtual environment, natural phenomena simulation, distributed virtual environment and Internet-based VR technologies.

Wei Wu received his Ph.D. from Harbin Institute of Technology, Harbin, China, in 1995. Since 1998, he has been working at Beihang University, China. He is currently a Full Professor at Beihang University, and the chair of the Technical Committee on Virtual Reality and Visualization, China Computer Federation. His current research interests include virtual reality, wireless networking, and distributed interactive system.