

Pedestrian Image Segmentation via Shape-Prior Constrained Random Walks

Ke-Chun Li, Hong-Ren Su, and Shang-Hong Lai

Department of Computer Science,
National Tsing Hua University Hsinchu, Taiwan

Abstract. In this paper, we present an automatic and accurate pedestrian segmentation algorithm by incorporating pedestrian shape prior into the random walks segmentation algorithm. The random walks [1] algorithm requires user-specified labels to produce segmentation with each pixel assigned to a label, and it can provide satisfactory segmentation result with proper input labeled seeds. To take advantage of this interactive segmentation algorithm, we improve the random walks segmentation algorithm by incorporating prior shape information into the same optimization formulation. By using the human shape prior, we develop a fully automatic pedestrian image segmentation algorithm. Our experimental results demonstrate that the proposed algorithm significantly outperforms the previous segmentation methods in terms of pedestrian segmentation accuracy on a number of real images.

Keywords: human segmentation, random walks, shape prior.

1 Introduction

Pedestrian segmentation is an important problem in computer vision, especially for video surveillance [2]. Human detection is usually the first step in video surveillance. The traditional systems can only supply rough human locations, but more precise human segmentation information is required for some advanced applications, such as gait recognition, human identification or human motion analysis [3]. Therefore, pedestrian image segmentation which segments pedestrian from an image is a critical problem with several potential applications.

Object segmentation is the key technique in many applications, including interactive video editing, content-based image retrieval, video surveillance, medical image analysis, and so on. This problem has been researched in computer vision for decades, but automatic object segmentation is still very challenging for general objects whose appearance is difficult to model, such as humans. A finely segmented human image can provide important and precise information of the human, which is very helpful for a number of higher level tasks on human motion analysis. It remains a great challenge because of the highly articulated human body postures, viewpoint changes, large appearance variations, and cluttered background, especially when pedestrians have similar color or texture with

the connected background regions. Recently, interactive segmentation techniques [1][20] becomes popular due to their flexibility in handling the difficult cases.

In recent years, simultaneous detection and segmentation of pedestrians become a popular problem. In general, these methods obtain rough shape or silhouette of human from different ways. In [4], Lin et al. proposed a hierarchical part-template matching approach [5] and learned a human detector which consists of elementary part detectors for head-torso, upper legs, and lower legs. This algorithm provides accurate human detection and a rough human segmentation from an image. Gao et al. [6] presented a novel feature representation called Adaptive Contour Feature (ACF) that is robust against reasonable object deformation like HOG, and the detection and segmentation of human was trained by a cascade framework [7] and Real AdaBoost [8]. For human segmentation, each pixel is classified as human or background from the ACFs.

The random walker algorithm was originally developed for interactive image segmentation by Grady [1]. With input of some user-specified seeds, the algorithm provides image segmentation result by solving a sparse linear system. Grady [21] extended the random walker algorithm by incorporating an intensity-based prior model into the energy minimization formulation to relieve the requirement of user-specified labels in the original algorithm. The prior model proposed in [21] is based on imposing the prior intensity distribution constraint as an additional quadratic energy term into the original energy minimization formulation. In this work, we propose to impose a pedestrian shape prior, instead of the intensity-based prior, into the random walks image segmentation algorithm. This shape prior is a mixture of Gaussians distribution, which is quite different from [21], and the image segmentation can still be obtained via solving similar sparse linear systems.

The overall flow of the proposed pedestrian segmentation algorithm is depicted in Fig. 1. It mainly consists of two components; namely, shape prior estimation and pedestrian segmentation. The shape prior estimation is to learn the human shape prior distribution from a training set of human shapes. The pedestrian segmentation is based on a shape-prior constrained random walks

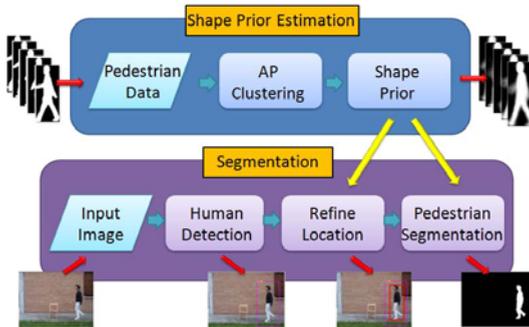


Fig. 1. System flow of the proposed pedestrian segmentation algorithm

image segmentation algorithm that incorporates the learned human shape prior into the random walks segmentation framework. We will detail these two main components in the subsequent sections.

2 Shape Prior Estimation

In order to estimate the pedestrian shape prior model, we collect a large set of pedestrian silhouettes, which were extracted from videos with uniform background. We took several pedestrian sequences, with each sequence containing one person walking along different directions. We applied a background subtraction procedure to segment the pedestrian regions from videos and the extracted regions are normalized to 64×128 , as depicted in Fig. 2. Finally, the dataset consists of 1,439 pedestrian silhouettes extracted from videos of 10 walking persons. With the left-right reflections, there are totally 2,878 pedestrian silhouettes in the training dataset.

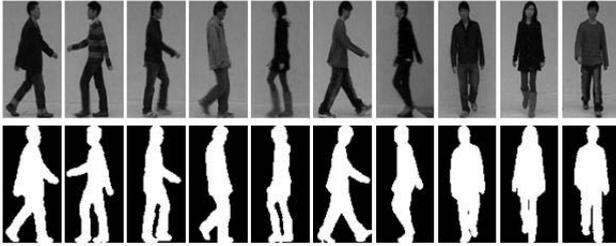


Fig. 2. Samples of pedestrian silhouettes extracted from images. The silhouettes are normalized to 64×128 .

2.1 Pedestrian Shape Prior Model

The prior pedestrian shape model is estimated from the binary pedestrian silhouette data. To cluster the set of human silhouette data, we employ the Affinity Propagation (AP) [9] clustering algorithm in this paper. The AP clustering is an iterative algorithm that works by finding a set of exemplars in the data and assigning other data points to the exemplars.

In this work, all the training pedestrian silhouettes are divided into 7 clusters after applying the AP clustering. Next, we estimate the shape prior model $\boldsymbol{\mu}^s = (\mu_1^s, \mu_2^s, \dots, \mu_N^s)$ by taking the averages for all pixels of the silhouettes for each cluster. In addition to the 7 prior models constructed from all the associated pedestrian silhouettes, we also compute the model with all the pedestrian images $\boldsymbol{\mu}^0 = (\mu_1^0, \mu_2^0, \dots, \mu_N^0)$. Fig. 3. shows the probability map of each cluster. All the 8 shape models are employed to form a mixture of Gaussians distribution for the human shape prior distribution; namely,

$$p(\mathbf{x}) = c \sum_{s=0}^7 G(\mathbf{x} - \boldsymbol{\mu}^s, D^s), \quad (1)$$

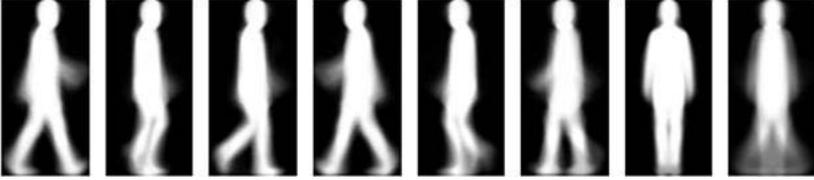


Fig. 3. The probability maps corresponding to all clusters are shown here, and the last one is the probability map computed from all training data

where $\mathbf{x} = (x_1, x_2, \dots, x_N)$ with x_i representing a continuous random variable for the i -th pixel that indicates the likelihood to be a pedestrian region, G is a N -dimensional Gaussian function, D^s is a diagonal covariance matrix to be determined later, and c is a normalization factor.

3 Pedestrian Segmentation

Given an image I , each pixel $v_i \in V$ is assigned to a label $l_i \in \{0, 1\}$ representing background and pedestrian, respectively, in the pedestrian segmentation problem. The pedestrian segmentation problem is simply to assign v_i the label based on the posteriori probability $P(x_i > \alpha | I)$, where α is set to 0.5 in this problem. Note that the shape prior $p(\mathbf{x})$ is assumed to be a mixture of Gaussian functions given in eq. 1. We will formulate the pixel likelihood estimation problem by using a graphical model framework, and derive the energy function for the image segmentation.

3.1 Graphical Model

We first describe the notion of a graph for an image. A graph $G = (V, E)$ has vertices (nodes), represented by set V , with each vertex corresponding to a pixel and $V = \{v_i\}_{i=1, \dots, N}$, and edges, denoted by $e \in E \subseteq V \times V$. An edge e connecting two vertices v_i and v_j is denoted by e_{ij} . A weighted graph has a value assigned to each edge, and it is called a weight. The weight of edge e_{ij} , is denoted by $w(e_{ij})$ and we express it as w_{ij} . The degree of a vertex is $d_i = \sum w_{ij}$ for all edges e_{ij} incident on v_i . In this work, we assume that the graph is undirected ($w_{ij} = w_{ji}$).

3.2 Edge Weights

In order to represent the image structure, one must define a function that maps a change in image intensities to edge weights. This is a common feature of graph-based algorithms for image analysis. In this work, we implement the typical Gaussian weighting function given below. Let N_i be the neighborhood of a pixel v_i . In this paper, we employ an 8-connected neighborhood structure,

$$w_{ij} = \begin{cases} \exp(-\beta(g_i - g_j)^2) & v_j \in N_i, \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where g_i indicates the image intensity at pixel v_i and β is a constant that controls the strength of the weight. This equation could be modified to handle color vector data by replacing $(g_i - g_j)^2$ by $\|g_i - g_j\|^2$. The definition of weighting function is the same as random walks [1] and [10], and it provides a numerical measure for the label similarity between two neighboring pixels.

Besides the above Gaussian weighting function is used here, we also present a weighting function for the shape prior,

$$w_{ij}^s = \begin{cases} \exp(-\theta(\mu_i^s - \mu_j^s)^2) & v_j \in N_i, \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where μ_i^s denotes the probability at pixel v_i for the s-th shape prior model, and θ is a free parameter. It is similar to eq. 2. The higher value of the weight is computed when the probability of μ_i^s is very close to μ_j^s , which means that they should be labeled with the same class, and vice versa. The degree of the s-th shape prior model is $d_i^s = \sum w_{ij}^s$ for all edges e_{ij} incident on v_i . Finally, the importance for each node, denoted by D_i^s , is determined by multiplying the associated degree of the prior shape model and the degree of the same node determined by its weight, i.e.

$$D_i^s = d_i \times d_i^s \quad (4)$$

Thus, the diagonal covariance matrix D^s for the s-th Gaussian in eq. 1 is formed from the diagonal entries D_i^s .

3.3 Likelihood Estimation

Given a weighted graph, there are a set of marked nodes V_M (seeds), and a set of unmarked nodes V_U , such that $V_U \cup V_M = V$ and $V_U \cap V_M = \phi$. Therefore, we would like to label each node $v_i \in V_U$ with a class, pedestrian or background. The set of seeds V_M in previous random walker segmentation is usually obtained by user interaction. In this work, the seeds are placed automatically by using the prior human shape model, and more details will be described subsequently.

In the random walks segmentation approach, the problem is to assign a label to each node $v_i \in V_U$, the likelihood x_i , such that a random walker starting from that node first reaches a seed $v_j \in V_M$, and assigns a label for v_j . The segmentation is then completed by assigning each free node to the label corresponding to the highest likelihood. In our implementation, the node $v_i \in V_U$ will be labeled to 1 (pedestrian) if $x_i > 0.5$.

Now we review the quadratic energy function to be minimized in random walks segmentation [1],

$$E_{RW}(\mathbf{x}) = \sum_{e_{ij} \in E} w_{ij}(x_i - x_j)^2 \quad (5)$$

This energy function is called Dirichlet integral [11] in random walks. It is similar to an electrical problem, which includes three fundamental equations of circuit theory (Kirchhoff's current and voltage law and Ohm's law). Here is the explanation for this energy function in likelihood: the energy function will be minimum if the likelihood x_i and x_j at node v_i and v_j are very close when w_{ij} is a large value, thus the node v_i and v_j should be labeled to the same class.

In addition to the random walks energy function given in eq. 5, we propose another energy function to incorporate the prior shape model. The nodewise priors μ_i^s , which represents the probability of the s -th pedestrian prior model at node v_i , and the energy function of the s -th pedestrian prior model can be written as:

$$E_{Prior}^s(\mathbf{x}^s) = \sum_{v_i \in V} D_i^s(x_i^s - \mu_i^s)^2 \quad (6)$$

where x_i^s is the likelihood at pixel v_i of the s -th shape prior model.

The above two energy functions, given in eq. 5 and 6, are combined to approximate the total energy function corresponding to the MAP estimation of \mathbf{x} with the introduction of a parameter λ that controls the weighting between the two energy functions; i.e.

$$E_{Total}(\mathbf{x}) \approx E_{RW}(\mathbf{x}) + \lambda \min_{0 \leq s \leq 7} E_{Prior}^s(\mathbf{x}) \quad (7)$$

where the first term E_{RW} is the label-continuity constraint borrowed from the original Random Walks formulation enforcing that two neighboring pixels in the small neighborhood system should have the same label if their colors or intensities are similar, and the energy E_{Prior}^s is the unary constraint that each pixel tends to the s -th prior model. The weighting parameter λ is a positive coefficient measuring how much we want to fit the prior models. If $\lambda = 0$, the energy function E_{Total} is completely the same as that used in the random walks segmentation algorithm.

3.4 Convex Optimization

There is no closed-form solution to directly minimize the energy function in eq. 7. Instead, we minimize the individual combined energy function $E_{Total}^s = E_{RW} + \lambda E_{Prior}^s$ for each individual shape prior model, and find the one with the minimal cost to be the solution. For each individual combined energy function, it can be formulated as a quadratic form of \mathbf{x} as follows:

$$\begin{aligned} E_{Total}^s(\mathbf{x}^s) &= E_{RW}(\mathbf{x}^s) + \lambda E_{Prior}^s(\mathbf{x}^s) \\ &= \sum_{e_{ij} \in E} w_{ij}(x_i^s - x_j^s)^2 + \lambda \sum_{v_i \in V} D_i^s(x_i^s - \mu_i^s)^2 \\ &= \mathbf{x}^{sT} L \mathbf{x}^s + (\mathbf{x}^s - \boldsymbol{\mu}^s)^T \lambda D^s (\mathbf{x}^s - \boldsymbol{\mu}^s) \end{aligned} \quad (8)$$

where D^s is a diagonal matrix with the values D_i^s on the diagonal, $D^s = \text{diag}([D_1^s, \dots, D_N^s])$, and L represents the combinatorial Laplacian matrix [12] defined by

$$L_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -w_{ij} & \text{if } v_i \text{ and } v_j \text{ are adjacent nodes,} \\ 0 & \text{otherwise,} \end{cases} \quad (9)$$

where L_{ij} is indexed by vertices v_i and v_j .

Partitioning the vertices into two sets, marked node set V_M (seeds) and unmarked node set V_U , we may decompose eq. 8 into

$$\begin{aligned} E_{Total}^s(\mathbf{x}^s) &= \mathbf{x}^{sT} L \mathbf{x}^s + (\mathbf{x}^s - \boldsymbol{\mu}^s)^T \lambda D^s (\mathbf{x}^s - \boldsymbol{\mu}^s) \\ &= \mathbf{x}_M^{sT} L_M \mathbf{x}_M^s + 2\mathbf{x}_U^{sT} B^T \mathbf{x}_M^s + \mathbf{x}_U^{sT} L_U \mathbf{x}_U^s \\ &\quad + (\mathbf{x}_M^s - \boldsymbol{\mu}_M^s)^T \lambda D_M^s (\mathbf{x}_M^s - \boldsymbol{\mu}_M^s) \\ &\quad + (\mathbf{x}_U^s - \boldsymbol{\mu}_U^s)^T \lambda D_U^s (\mathbf{x}_U^s - \boldsymbol{\mu}_U^s) \end{aligned} \quad (10)$$

where $\mathbf{x}^s = [\mathbf{x}_M^s, \mathbf{x}_U^s]$ and $\boldsymbol{\mu}^s = [\boldsymbol{\mu}_M^s, \boldsymbol{\mu}_U^s]$ correspond to the partitioning of the labels and potentials into the seeded and unseeded nodes, respectively.

Differentiating the above matrix form for the energy function E_{Total}^s given in eq. 10 with respect to \mathbf{x}_U^s , and setting it to zero yields

$$\frac{\partial E_{Total}^s}{\partial \mathbf{x}_U^s} = B^T \mathbf{x}_M^s + L_U \mathbf{x}_U^s + \lambda D_U^s (\mathbf{x}_U^s - \boldsymbol{\mu}_U^s) = 0 \quad (11)$$

then the system of linear equations can be written as

$$(L_U + \lambda D_U^s) \mathbf{x}_U^s = -B^T \mathbf{x}_M^s + \lambda D_U^s \boldsymbol{\mu}_U^s \quad (12)$$

Since the matrix $A = L_U + \lambda D_U^s$ is positive definite, the linear system in 12 can be solved easily by an iterative numerical algorithm, such as conjugate gradient, to obtain the likelihood \mathbf{x}_U^s for all unmarked pixels.

3.5 Prior Model Decision

The score function is presented for deciding the prior model automatically. The correlation coefficient is used as a score function $Score(\mathbf{x}^s, \boldsymbol{\mu}^s)$ that evaluates the normalized correlation between the prior model $\boldsymbol{\mu}^s$ and its corresponding segmentation result \mathbf{x}^s . Let s^* denote the prior model with the maximum score, i.e.

$$s^* = \text{argmax}_s (Score(\mathbf{x}^s, \boldsymbol{\mu}^s)) \quad (13)$$

Finally, after we compute the likelihood $x_i^{s^*}$ in eq. 12 for the s^* -th prior model, the decision rule of each pixel v_i for image segmentation is given as follows

$$l_{v_i}^{s^*} = \begin{cases} 1(\text{pedestrian}) & \text{if } x_i^{s^*} > T_1, \\ 0(\text{background}) & \text{otherwise,} \end{cases} \quad (14)$$

where T_1 is the threshold. After assigning the label $l_{v_i}^{s*}$ to each pixel v_i , the pedestrian segmentation is accomplished.

3.6 Human Detection and Refinement

In order to make use of the prior model appropriately, it is necessary to find the precise location of pedestrian in an image. First of all, a human detector is applied to obtain a rough position, I_r , of pedestrian in this system. Fig. 4 (a) shows the result of human detection in [13], the bounding box is regarded as a coarse pedestrian location in this procedure.

Refining the precise location of pedestrian based on the rough position is proposed in this section. This problem is formulated as a binary sliding-window search problem. A multiple-size window is scanned over the bounding box which is determined from the human detection result, and template matching provides estimates of the pose model parameters for every detected window. We define a score function, which is the same as that given in eq. 14, to find precise location of pedestrian based on the maximal score.

The random walks segmentation without prior model is applied to I_r to obtain the ordinary probability of pedestrian (Fig. 4 (c)), and the seeds are placed using $\mu_i^0 \in \boldsymbol{\mu}^0$ (prior model with whole training data), i.e.

$$Seed_{ped} = \{v_i | \mu_i^0 > T_2\} \quad (15)$$

where T_2 is threshold for shape prior, and $Seed_{background}$ is determined with respect to the rectangular boundary of I_r . An example of human detection refinement is depicted in Fig. 4 (b).

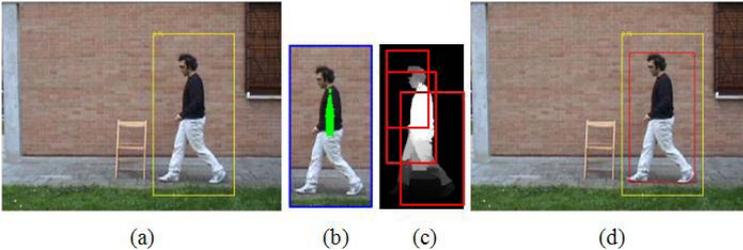


Fig. 4. Example of human detection and refinement procedure. (a) Result of human detection, the bounding box with yellow line shows I_r . (b) Seeds are placed in I_r . The green and blue dots represent the pedestrian and background seeds, respectively. (c) Result of the random walks segmentation with the seeds from (b), and the bounding boxes with red lines indicate the multiscale sliding window. (d) The final result with human detection and refinement procedure.

4 Experimental Results

All of our experiments were performed on a PC equipped with Intel i5 CPU 750 (2.67 GHz) and 2 GB memory. The proposed pedestrian segmentation algorithm

was implemented in MATLAB. We have used the MIT pedestrian dataset [14], INRIA person dataset [15], and ViSOR surveillance video dataset [16] to evaluate the results of our algorithm. The ViSOR dataset consists of surveillance videos, and we trimmed a short pedestrian sequence from it for the testing.

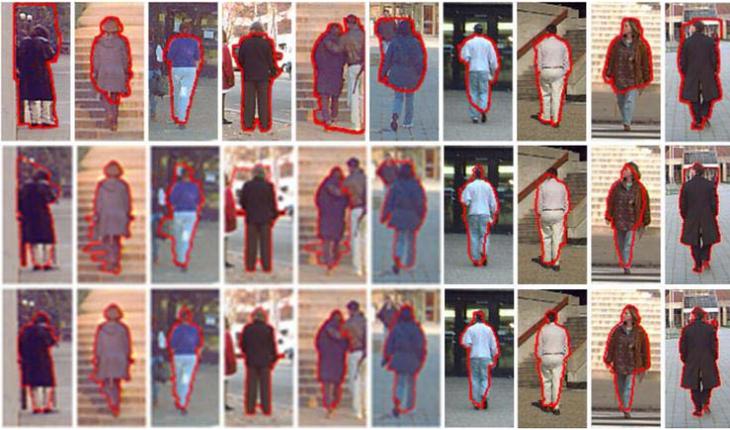
Refining the precise location of pedestrian is formulated as a sliding-window search problem. We refine the detection window by using the NCC score with 7 scales of the sliding windows. In order to quantify the results of the pedestrian segmentation, we applied our algorithm on MIT, INRIA, and ViSOR datasets, and compare the segmentation results with manually labeled masks, which are regarded as the ground truth.

The percentage of overlap area between the ground truth mask and the segmented region is evaluated as the segmentation accuracy in this experiment. The manually labeled segmentation ground truths are sometimes ambiguous around the boundary pixels. Hence we mark a two-pixel width do-not-care (DNC) boundary for accuracy assessment of the human segmentation results. This strategy is similar to that used in [18] [19].

We select 60 testing images from MIT and INRIA datasets, and cut a short video from the "Man with a dog" video sequence in the ViSOR dataset, the trimmed video contains 30 testing images, Table 1 shows comparison of the segmentation accuracies for the GrabCut [20] algorithm, the random walks algorithm and the proposed algorithm. Fig. 6 depicts some segmentation results from the testing images for the comparison of the proposed pedestrian segmentation algorithm with the GrabCut [20] algorithm and the random walks segmentation algorithm. In our experiments, all of the three algorithms have with the same initialized windows of detected humans for a fair comparison. It is obvious from Table 1 and Fig. 6 that the proposed pedestrian segmentation algorithm significantly outperforms the other two well-known segmentation algorithms in our experiments.



Fig. 5. The testing image samples and the corresponding segmentation ground truth masks



(a)



(b)



(c)

Fig. 6. Pedestrian segmentation results by using three different methods with the same initializations on testing images in (a) MIT, (b) INRIA, and (c) ViSOR datasets. The first rows give the results by using the GrabCut method, the second rows are the results by using the random walks algorithm, and the third rows show the results by using the proposed pedestrian segmentation algorithm.

Table 1. Overlap area percentages between ground truth and segmented regions

Dataset	Methods	Accuracy
MIT	GrabCut	68.06 %
	Random Walks	81.16 %
	Our method	84.36 %
INRIA	GrabCut	46.56 %
	Random Walks	69.78 %
	Our method	83.36 %
ViSOR	GrabCut	46.14 %
	Random Walks	71.03 %
	Our method	87.08 %

5 Conclusion

In this paper, we presented an automatic pedestrian segmentation algorithm by incorporating pedestrian shape prior into random walks segmentation. Our experimental results show that the proposed algorithm can provide good segmentation results for the cases with slight occlusion, similar background and illumination changes. In addition, a new pedestrian dataset with labeled silhouettes is produced.

For the directions of future research, the over-segmentation can be incorporated into this framework to obtain more accurate segmentation and reduce the computational time by taking each region as a node. It could be a possible direction for developing a more accurate and robust pedestrian segmentation technique.

References

1. Grady, L.: Random Walks for Image Segmentation Journal. *IEEE Trans. Pattern Analysis and Machine Intelligence* 28(11), 1768–1783 (2006)
2. Juan, C.-F., Chang, C.-M., Wu, J.-R., Lee, D.: Computer Vision-Based Human Body Segmentation and Posture Estimation. *IEEE Trans. Systems, Man and Cybernetics, Part A: Systems and Humans* 39(1), 119–133 (2009)
3. Cucchiara, R., Grana, C., Prati, A., Vezzani, R.: Probabilistic posture classification for Human-behavior analysis. *IEEE Trans. Systems, Man and Cybernetics, Part A: Systems and Humans* 35(1), 42–54 (2005)
4. Lin, Z., Davis, L.S.: Shape-Based Human Detection and Segmentation via Hierarchical Part-Template Matching. *IEEE Trans. Pattern Analysis and Machine Intelligence* 32(4), 604–618 (2010)
5. Lin, Z., Davis, L.S., Doermann, D., DeMenthon, D.: Hierarchical Part-Template Matching for Human Detection and Segmentation. In: *International Conf. on Computer Vision*, pp. 1–8 (2007)
6. Gao, W., Ai, H., Lao, S.: Adaptive Contour Features in oriented granular space for human detection and segmentation. In: *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1786–1793 (2009)

7. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. In: International Conf. on Computer Vision, vol. 2, pp. 734–741 (2003)
8. Schapire, R., Singer, Y.: Improved Boosting Algorithms Using Confidence-rated Predictions. *Machine Learning* 37, 297–336 (1999)
9. Givoni, I.E., Frey, B.J.: A Binary Variable Model for Affinity Propagation. *Neural Computation* 21, 1589–1600 (2009)
10. Kim, T.H., Lee, K.M., Lee, S.U.: Nonparametric higher-order learning for interactive segmentation. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 32201–3208 (2010)
11. Courant, R., Hilbert, D.: *Methods of Math. Physics*, vol. 2. John Wiley and Sons (1989)
12. Merris, R.: Laplacian Matrices of Graphs: A Survey. *Linear Algebra and Its Applications* 197,198, 143–176 (1994)
13. Maji, S., Berg, A.C., Malik, J.: Classification using intersection kernel support vector machines is efficient. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
14. <http://cbcl.mit.edu/cbcl/software-datasets/PedestrianData.html>
15. howpublished, <http://pascal.inrialpes.fr/data/human/>
16. howpublished, <http://www.openvisor.org/>
17. <http://iris.usc.edu/Vision-Users/OldUsers/bowu/DatasetWebpage/dataset.html>
18. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: TextonBoost: Joint Appearance, Shape and Context Modeling for Multi-class Object Recognition and Segmentation. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 1–15. Springer, Heidelberg (2006)
19. Wu, B., Nevatia, R.: Simultaneous Object Detection and Segmentation by Boosting Local Shape Feature based Classifier. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
20. Rother, C., Kolmogorov, V., Blake, A.: Interactive Foreground Extraction using Iterated Graph Cuts. *ACM Trans. on Graphics* 23, 309–314 (2004)
21. Grady, L.: Multilabel Random Walker Image Segmentation Using Prior Models. In: IEEE Conf. on Computer Vision and Pattern Recognition, vol. 1, pp. 763–770 (2005)