PedCut: an iterative framework for pedestrian segmentation combining shape models and multiple data cues

Fabian Flohr[1,2]
fabian.flohr@daimler.com
Dariu M. Gavrila[1,2]
www.gavrila.net

[1]Environment Perception Department, Daimler R&D, Ulm, Germany
[2]Intelligent Systems Laboratory, Univ. of Amsterdam, The Netherlands

Abstract

This paper presents an iterative, EM-like framework for accurate pedestrian segmentation, combining generative shape models and multiple data cues. In the E-step, shape priors are introduced in the unary terms of a Conditional Random Field (CRF) formulation, joining other data terms derived from color, texture and disparity cues. In the M-step, the resulting segmentation is used to adapt an Active Shape Model (ASM), after which the EM process alternates.

Experiments on the public Penn-Fudan pedestrian dataset suggest that our method outperforms the state-of-the-art. We further provide results on a new Daimler pedestrian dataset, captured from on-board a vehicle, which includes disparity data. This dataset is made public to facilitate benchmarking.

1 Introduction

Person segmentation in images is a key computer vision problem in a number of application domains, such as image editing, surveillance and intelligent vehicles. It facilitates higher-level, semantic scene analysis (e.g. body part labeling, pose estimation, activity analysis) and can enhance the person detection and localization performance in itself.

In this paper, we are interested in the case where persons are observed against a complex and possibly dynamic backdrop. Such is the case when an intelligent vehicle captures images of pedestrians while driving through an urban traffic environment. The large variety of pedestrian appearances, induced by viewpoint, pose, clothing and lighting, makes the problem especially challenging. On the other hand, by focusing on a single object class, we are in a position to introduce a fair amount of prior knowledge on how pedestrians appear in images. We focus on the case where an external pedestrian detector acts as a front-end, providing regions of interest (i.e. bounding boxes) to our segmentation module. Given the intelligent vehicle context, we are interested in the optional use of disparity data obtained from stereo vision.

We present an iterative, EM-like framework, combining generative shape models and multiple data cues. In the E-step, shape priors are introduced in the unary terms of a Conditional Random Field (CRF) formulation, joining other data terms derived from color, texture

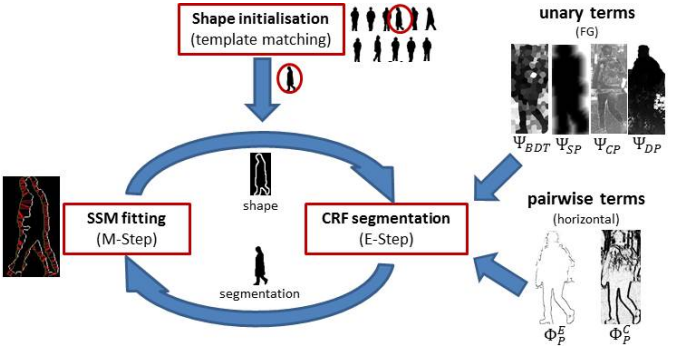


Figure 1: Overview of our iterative EM-like segmentation framework, alternating CRF-based segmentation (E-step) and SSM fitting (M-step), given shape initialisation. See Section 3.

and disparity cues. In the M-step, the resulting segmentation is used to adapt an Active Shape Model (ASM), after which the EM process alternates. Fig. 1 shows an overview of the main components of our approach.

2 Related Work

Data driven approaches [4, 7, 21, 22, 27, 29, 31] based on Conditional Random Field (CRF) formulations [7, 21] show promising segmentation results. For example, the GrabCut framework [27] involves segmentation with a minimum of user assistance based on an iterative, pixel-wise Gaussian Mixture Models (GMMs) fitting. In order to generate more discriminative and robust features, [22, 23, 29, 31] aggregate features over oversegmented regions. Superpixel-based features based on dense SIFT (dSIFT), introduced by Bosch *et al.* [4], show powerful results on a local level [22, 29]. Boosted Decision Trees (BDT) are used in [29] to classify these features. A preliminary and necessary step for these methods is to cluster the SIFT features to obtain a visual codebook representation. This is often done with k-means (e.g. in [29]). As showed in [24], generating visual codebooks using Decision Trees is more discriminative as using k-means. In [31] superpixel-based features (e.g. surface normals, planarities, distance to camera path) are constructed from a dense depth map and classified using Decision Trees.

Approaches matching shape models [2, 9, 10, 14, 15, 16] can be distinguished by whether they use global or part-based shape representations. Global shape representations can be discrete, in terms of a Set of Exemplars (i.e. shape templates from a training set). Hierarchical representations can be derived on top of these for increased matching efficiency [15]. Active Shape Models (ASMs) [10] combine Statistical Shape Models (SSMs) (compact, linear-subspace probabilistic representations) with means to match these to images. [16] represent shapes by multiple SSMs (MSSMs) to account for different shape aspects (pedestrian feet apart vs. feet closed). Active Appearance Models (AAMs) [9] extend ASMs by capturing shape and texture information jointly. ASMs and AAMs require feature correspondence, unlike exemplar-based representations. Fitting them to an image can result in sub-optimal solutions, due to convergence to local minima. Part-based representations, like pictorial structures [2, 14] offer a modular representation; tree-structured graphical models can be used for modeling the dependency between parts [2].

In terms of combining object models with data driven cues [3, 8, 11, 19, 20, 26], a number of approaches [8, 13, 20] are based on GrabCut. Kumar *et al*. [20] introduces the Object Category Specific CDRF (Contrast Dependent Random Field) and shows that multiple shape potentials can be used by formulating a linear weighted sum of energies, which makes solution by efficient methods (e.g. Graph Cut) still possible. Non-articulated objects are represented by a set of exemplars, which capture shape and appearance. For articulated objects, a PS model is used. Interaction between parts is additionally modeled by a Markov Random Field (MRF). A layer indicator for each part is added to handle occlusion. Based on multiple views, Bray *et al*. [8] describes an iterative pose estimation approach using a stick model joined with data cues in a CRF formulation (similar to [20]). Kokkinos and Maragos [19] describe an EM segmentation framework, in which the segmentation result from the E-step is used to estimate parameters of an AAM in the M-step. Eslami and Williams [11] describe a generative framework based on a Shape Boltzmann Machine combined with appearance cues. In [26], a Metropolis-Hastings sampler takes the results of an edge-based part detector as starting point to generate proposals of each body part. Learned appearance histograms for all parts are used for segmentation. Bo and Fowlkes [3] use hierarchical decomposition of parts and compute scores of matching part-based mean templates with additional color cues.

3 Our Segmentation Framework

For an overview of our segmentation framework, see Fig. 1. Our iterative process starts with a MSSM instantiation supplied by a shape initialisation module (Subsection 3.1). This MSSM instantiation is used as shape potential, joining data-driven cues in a CRF-based segmentation (Subsection 3.2), i.e. the E-step. The resulting binary segmentation allows to update the parameters of the MSSM instantiation (Subsection 3.3), i.e. the M-step. The process alternates until the CRF-based segmentation does not change appreciable any more (average Hamming distance between subsequent segmentations is less than 10%) or a maximum of N_{it} iterations is reached.

We consider our main contribution to be the beforementioned iterative, EM-like framework for accurate pedestrian segmentation, combining generative shape models and multiple data cues. It is able to cope with the large variation of pedestrian appearances, across cluttered backgrounds. Our iterative segmentation framework is in spirit most related to that of Kokkinos and Maragos [19]. However, our objects of interest, pedestrians, feature a larger appearance variation than the frontal faces and sideway cars of [19]. We are thus unable to deploy generative color/texture models like the AAMs, relying for these cues on discriminative classification of superpixels, within a CRF approach, among other data terms. We cope with the stronger shape aspect variations using a MSSM. All in all, our E- and M-steps are defined differently to [19].

3.1 Shape initialisation

Our shape training set consists of a set of $N_T = 10946$ pedestrian shape exemplars, obtained by manual labeling. A Multi Statistical Shape Model (MSSM) is derived from this training set, based on shape registration and clustering, as described in [16]. A total of $N_C = 12$ clusters are obtained; each involve a SSM for a particular shape aspect (frontal pose feet

closed, rightwards feet open, etc.). The dimensionality of the linear subspace (i.e. the number of eigenvectors) were chosen dynamically to cover 95% of total variance.

Input to shape initialisation is an image region of interest (i.e. a bounding box) provided by a pedestrian detector front-end (optionally, this includes the associated disparity values). As ASMs defined on SSMs can get stuck in local minima, template matching in the region of interest is performed using the individual pedestrian shape exemplars. We use chamfer matching differentiated by gradient direction (in our case: four discretization intervals, not encoding the gradient sign), as in [15]. The best matching shape exemplar is converted to its MSSM representation (SSM representation of respective cluster); it acts as a shape prior in the following CRF segmentation step.

3.2 CRF segmentation (E-step)

In the following, we define $\boldsymbol{I}_i$ as the value in Lab [25] color space and $\boldsymbol{D}_i$ as the disparity value at pixel i. We use Semi Global Matching (SGM) [17] for disparity computation. Furthermore, let $\boldsymbol{S}_i$ be the feature vector of the superpixel containing pixel i. We now describe the four unary and two pairwise potentials in our CRF formulation. One of these, the BDT potential (see below), remains constant over the EM iterations, the others are being refined.

3.2.1 Defining the unary terms

Boosted Decision Tree superpixel classification (BDT): We use oversegmented regions produced by SLIC superpixels [1] to train a BDT classifier as in [18]. For the classifier we used $N_C = 100$ trees pruned to a maximum depth of 10. Dense SIFT (dSIFT) are extracted over the given image with a step width of four. We train a visual codebook similar to [24], but instead of using fully Randomized Decision Trees, we use an ensemble of $N_V = 100$ BDTs. We prune each tree to get approx. 100 leaf nodes per tree. Our visual word is created from all leaf nodes over the N_V trees. At testing, a visual word vector contains N_V one entries for a tested feature. These vectors are summed up over the area of a superpixel. By this, we get a histogram which captures a discriminative representation of a superpixel. Additionally to the SIFT features we also use Textons, calculated similar to the dSIFT features. The resulting Texton superpixel histogram is appended to the dSIFT superpixel histogram. This vector is then used as classification feature.

The classifier output $f_{\mathcal{B}}(\boldsymbol{S}_i)$ for the superpixel-based feature $\boldsymbol{S}_i$, is a log-likelihood ratio score [18] which is used as potential in the CRF after sigmoid conversion:

$$\Psi_{BDT}(x_i, \boldsymbol{S}_i) = -logP(x_i|\mathcal{B}, \boldsymbol{S}_i). \tag{1}$$

where

$$P(x_i = 1|\mathcal{B}, \boldsymbol{S}_i) = \frac{1}{1 + exp(-f_{\mathcal{B}}(\boldsymbol{S}_i))} \tag{2}$$

and $P(x_i = 0|\mathcal{B}, \boldsymbol{S}_i) = 1 - P(x_i = 1|\mathcal{B}, \boldsymbol{S}_i)$. Above $\mathcal{B}$ denotes our trained BDT classifier.

Shape Potential (SP): In the first iteration we directly use the shape template found in the shape initialisation. We calculate a distance transformation from the shape contour Ω - denoting with $dist(loc_i, \Omega)$ the distance of the pixel location loc_i on the grid, to the nearest contour point on Ω. If pixel i lies inside the shape contour, $dist(loc_i, \Omega)$ is negative, otherwise $dist(loc_i, \Omega)$ is positive (see also [20]).

The resulting shape potential is

$$\Psi_{SP}(x_i,\Omega) = -\log P(x_i|\Omega), \tag{3}$$

where

$$P(x_i = 1|\Omega) = \frac{1}{1+exp(\mu_s \cdot dist(loc_i,\Omega))}. \tag{4}$$

and $P(x_i = 0|\Omega) = 1 - P(x_i = 1|\Omega)$. The parameter μ_s determines the penalization of points outside, compared to points inside the shape. From the second iteration on, we use the actual result of the previous iteration (binary image) for fitting the selected MSSM (see section 3.3).

Color Potential (CP): Based on the segmentation at the previous iteration (or the initial shape at the first iteration), we fit two GMMs, one for background and one for foreground, each with K_C (here $K_C = 5$) components in Lab color space. With the additional vector $\boldsymbol{k} = \{k_1,..k_i,..k_N\}$ we assign each pixel a unique component with $k_i \in \{1,..,K_C\}$ for foreground ($x_i = 1$) or background ($x_i = 0$). This approach was successfully used in [27]. Thus the color potential is defined as

$$\Psi_{CP}(x_i,\boldsymbol{I}) = -\log P\left(\boldsymbol{I}_i|x_i,k_i,\theta_{k_i}\right) - \log \pi(x_i,k_i), \text{ with } x_i = \{0,1\}. \tag{5}$$

Above, k_i is the best component of the GMM chosen for pixel i with learned Gaussian parameters θ_{k_i} and component weight $\pi(x_i,k_i)$.

Disparity Potential (DP): The disparity potential for the foreground is defined with one Gaussian distribution

$$\Psi_{DP}(x_i = 1,\boldsymbol{D}) = -\log P(\boldsymbol{D}_i|x_i = 1,\theta_d) \tag{6}$$

with parameters $\theta_d = \{\tilde{d},\sigma_d\}$. Here $\tilde{d}$ denotes the median value over all disparity values labeled as pedestrian in the current segmentation ($x_i = 1$). The disparity variance σ_d^2 was learned from data, over all pixels and their neighborhoods in the ground truth segmentation.

The background potential $\Psi_{DP}(x_i = 0,\boldsymbol{D})$ is modeled based on all disparity values $\boldsymbol{D}_i$ with values in the range $\boldsymbol{D}_i < \tilde{d} - 3\sigma_d$ and $\boldsymbol{D}_i > \tilde{d} + 3\sigma_d$ using a GMM as in the color potential. Like in the color potential we select for each pixel only the best out of K_D (here $K_D = 3$) components of the learned GMM.

3.2.2 Defining the pairwise terms

We define two pairwise potentials, which take the form of generalized Potts models [5]. The first is a color-sensitive potential, specified such, that it increases the costs of an edge inversely proportional to the color difference in Lab color space of two neighbored pixels i and j. The second potential is a contour-sensitive potential, which increases the cost inversely proportional to the edge magnitude between pixels i and j. For the second potential we use an additional weighting term based on disparity information. The resulting potentials have the form:

$$\Phi_P^C(x_i,x_j,\boldsymbol{I}) = exp(\frac{-||\boldsymbol{I}_i - \boldsymbol{I}_j||}{2\sigma_c^2})\frac{1}{dist(loc_i,loc_j)} \times \delta(x_i \neq x_j) \tag{7}$$

and

$$\Phi_P^E(x_i,x_j,\boldsymbol{I},\boldsymbol{D}) = exp(\frac{-\max_{l\in\overline{ij}}[|\nabla \boldsymbol{I}_l| \cdot P(c_l|\boldsymbol{U})]}{2\sigma_e^2})\frac{1}{dist(loc_i,loc_j)} \times \delta(x_i \neq x_j) \tag{8}$$

The variances σ_c^2 and σ_e^2 can be set according to the camera noise [7]. The notation $\overline{ij}$ denotes the line containing all pixels between pixel i and j, while $|\nabla \boldsymbol{I}_l|$ denotes the edge magnitude at pixel l. The weighting term $P(c_l|\boldsymbol{U})$ in eq. (8) denotes the probability that there is a pedestrian contour between i and j, given the contour points $\boldsymbol{U} = \{u_1, u_2, .., u_n\}$ of the disparity based segmentation. This disparity segmentation is calculated using the median disparity $\tilde{d}$ over all pixels from the current segmentation, labeled as foreground. $P(c_l|\boldsymbol{U})$ is defined with

$$P(c_l|\boldsymbol{U}) = exp(\frac{-(\min_k[dist(loc_{u_k}, loc_l)] - \mu_{dp})^2}{2\sigma_{dp}^2}), \tag{9}$$

where loc_i denoting again the location of a given pixel i on the grid. We learned the mean distance μ_{dp} (and variance σ_{dp}^2) of disparity segmentation contours to the ground truth contours from training data. The combination of these potentials forces consistent regions and assigns a lower cost to edges that lie on true contours.

3.2.3 Energy minimization

We minimize following energy functional:

$$\begin{aligned} &E(\boldsymbol{x}, \Omega, \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{\omega}) = \\ &\sum_{i \in \mathcal{V}} \omega_{BDT} \Psi_{BDT}(x_i, \boldsymbol{S}) + \omega_{SP} \Psi_{SP}(x_i, \Omega) + \omega_{CP} \Psi_{CP}(x_i, \boldsymbol{I}) + \omega_{DP} \Psi_{DP}(x_i, \boldsymbol{D}) \\ &+ \sum_{i,j \in \mathcal{E}} \omega_P^C \Phi_P^C(x_i, x_j, \boldsymbol{I}) + \omega_P^E \Phi_P^E(x_i, x_j, \boldsymbol{I}, \boldsymbol{D}) \end{aligned} \tag{10}$$

defined on the index set $\mathcal{V}$ with an eight-connected edge neighborhood $\mathcal{E}$. The posterior is defined with $P(\boldsymbol{x}, \Omega | \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{\omega}) = \frac{1}{Z} exp(-E(\boldsymbol{x}, \Omega, \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{S}, \boldsymbol{\omega})$, where Z is the partition function. Main CRF parameters $\boldsymbol{\omega}$ are the weights for the specified unary and pairwise terms (ω_{BDT}, ω_{SP}, ω_{CP}, ω_{DP}, ω_P^C and ω_P^E). As our pairwise terms stay submodular, we can perform inference with Graph Cut [6].

3.3 Fitting the SSM (M-step)

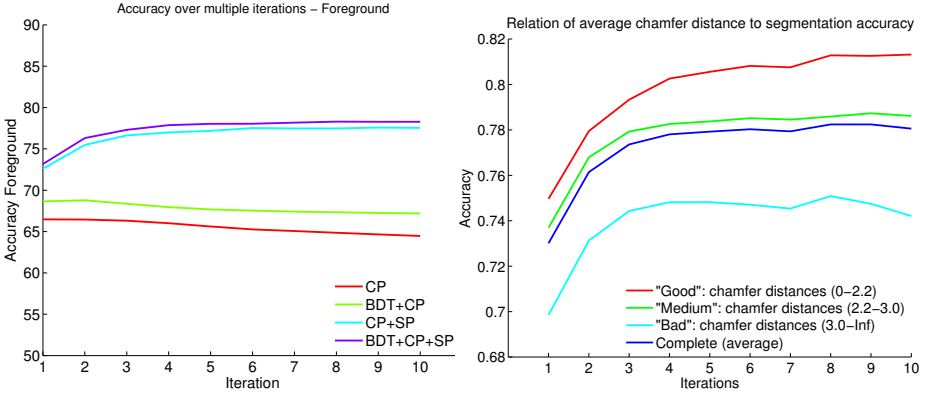
We use an ASM approach [10] for fitting the SSM model to the obtained CRF segmentation. Point correspondences between SSM and image are given by chamfer matching [15]. As in shape initialisation (Section 3.1), we can differentiate chamfer matching by gradient direction. But note that since we have a binary segmentation, we can here utilize information about the gradient sign to improve matching (i.e. eight discretization intervals for gradient direction).

4 Experiments

Segmentation accuracy is measured by the intersection/union criterium of the PASCAL VOC challenge [12]. For evaluation, we use the public Penn-Fudan dataset [30]. It contains 170 color images with 345 box/shape-labeled pedestrians from which 169 labels are used in [3, 11]. We use this same data subset. Due the scarcity of public available pedestrian

	Daimler [28] (train BDT)	Daimler [this paper] (validation/test)	Penn-Fudan [3, 30] (test)
#images	300	228	169
#pedestrians	521 (sel. 521)	785 (sel. 30/300)	169 (sel. 169)
sel. min BB [h,w]	no BB	[121,34] pixel	[186,63] pixel
sel. max BB [h,w]	no BB	[468,267] pixel	[373,207] pixel
color	no	yes	yes
disparity	yes	yes	no

Table 1: Used datasets and their characteristics.



a) b)

Figure 2: FG segmentation accuracy over EM iterations a) using various cue combinations b) Dependence on shape initialisation (BDT+CP+SP)

segmentation datasets containing stereo and color information, we created a new Daimler dataset (subdivided in validation/test subsets)[1]. See Table 1 for the used datasets.

Evaluation on the Penn-Fudan dataset. Fig. 2a) shows the FG segmentation accuracies using various cue combinations over the EM iterations (a similar plot applies for the BG). The incorporation of SP can be seen to have a major beneficial effect. The benefit of adding BDT is only substantial when SP is not available (compare blue vs. cyan, and green vs. red plots). Fig. 2b) shows the dependence of segmentation accuracy on shape initialisation (i.e. availability of good contrast object contours). Table 2 shows the results for different cue combinations. A comparison with the state-of-the-art [3, 11] is given in Table 3. We achieve outperformance both in terms of foreground and background segmentation accuracy. Fig. 3 shows representative segmentation results on the Penn-Fudan dataset with the best performing cues BDT+CP+SP. We can use the available body-part label information associated with SSM points to establish a basic component-based segmentation into head, upper and lower body. Note that our results cannot be compared to [3] and [11] for the upper and lower body, since they segment and label body components based on clothing, while we do it on true body proportions. For head segmentation, a comparison is possible: we obtain an accuracy of 57.1 compared to 51.8 [3] and 54.1 [11].

[1] This dataset is available for non-commercial research purposes. Follow the links from http://isla.science.uva.nl/ or contact the 2nd author.

	BDT	CP	BDT+CP	SP	SP+CP	BDT+SP+CP
FG	42.8	64.3	67.1	67.2	77.5	**78.5**
BG	54.4	66.7	70.6	71.6	80.3	**81.5**
average	48.6	65.5	68.9	69.4	78.9	**80.0**

Table 2: Segmentation accuracy for various cue combinations on the Penn-Fudan dataset.

	Ours (BDT+SP+CP)	Bo & Fowlkes [3]	Eslami & Williams [11]
FG	**78.5**	73.3	71.6
BG	**81.5**	81.1	73.8
average	**80.0**	77.2	72.7

Table 3: Comparison with the state-of-the-art on the Penn-Fudan dataset

a) b) c) d) e) f) g)

Figure 3: Results on the Penn-Fudan dataset after four EM iterations (BDT+SP+CP). First row: input images with initial/final SSM fit (red/white). Second row: correct/missing/excessive segmentation (white/red/cyan). Columns: a)-d) decent segmentations from decent shape initialisations, e)-f) decent segmentations from poor shape initialisations, and g) poor segmentation from poor shape initialisation.

	SP	CP	DP	CP+DP	SP+CP	SP+DP	SP+CP+DP
FG	68.9	60.2	63.8	70.2	73.5	67.8	**76.4**
BG	72.6	62.9	56.3	70.9	77.4	69.0	**78.6**
average	70.7	61.6	60.1	70.6	75.5	68.4	**77.5**

Table 4: Segmentation accuracy for different cue combinations without BDT on our dataset.

	BDT	BDT+CP	BDT+CP+DP	BDT+CP+SP	BDT+CP+SP+DP
FG	56.6	65.4	74.1	74.9	**77.4**
BG	59.7	68.1	75.2	78.2	**79.6**
average	58.1	66.7	74.6	76.5	**78.5**

Table 5: Segmentation accuracy for different cue combination with BDT on our dataset.

a) b) c) d) e) f) g)

Figure 4: Results on our dataset after four EM iterations (BDT+SP+CP+DP). First row: input images with initial/final SSM fit (red/white). Second row: correct/missing/excessive segmentation (white/red/cyan). Columns: a)-d) decent segmentations from decent shape initialisations, e)-f) decent segmentations from poor shape initialisations, and g) poor segmentation from poor shape initialisation.

Evaluation on our dataset. Tables 4 and 5 show results with different cue combinations with and without the BDT classifier. When not having disparity data, the combination BDT+CP+SP performs best, as in the Penn-Fudan case. Adding DP improves average segmentation accuracy by 2%. Fig. 4 shows representative segmentation results on our dataset with the best performing BDT+CP+SP+DP cues. Our unoptimized Matlab implementation requires about 2 s for segmenting a pedestrian in four EM iterations, running on a 3.33 GHz i7-CPU processor. Main bottleneck is the matching of templates during shape initialisation (Section 3.1); we aim to replace this by the more efficient hierarchical approach of [15].

5 Conclusion

This paper presented an iterative, EM-like framework for accurate pedestrian segmentation, combining generative shape models and multiple data cues. We showed the benefit of different cue combinations and the ability of the framework to improve results with each additional cue, on various datasets. On the public Penn-Fudan dataset, we showed to outperform the state-of-the-art by more than 5% on foreground accuracy while remaining ahead on back-

ground accuracy. Further work involves body-part labelling and pose estimation, as well as enhanced pedestrian recognition.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34:2274–2282, 2012.

[2] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1014–1021. IEEE, 2009.

[3] Y. Bo and C. C. Fowlkes. Shape-based pedestrian parsing. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2265–2272. IEEE, 2011.

[4] A. Bosch, A. Zisserman, and X. Munoz. Scene classification via pLSA. *Proc. of the European Conf. on Computer Vision (ECCV)*, pages 517–530, 2006.

[5] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 648–655. IEEE, 1998.

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[7] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In *Proc. of the International Conf. on Computer Vision (ICCV)*, volume 1, pages 105–112. IEEE, 2001.

[8] M. Bray, P. Kohli, and P. H. S. Torr. Posecut: Simultaneous segmentation and 3D pose estimation of humans using dynamic graph-cuts. In *Proc. of the European Conf. on Computer Vision (ECCV)*, pages 642–655, 2006.

[9] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(6):681–685, 2001.

[10] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[11] S. A. Eslami and C. Williams. A Generative Model for Parts-based Object Segmentation. In *Advances in Neural Information Processing Systems (NIPS)*, pages 100–107, 2012.

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[13] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2008.

[14] M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computers*, 100(1):67–92, 1973.

[15] D. M. Gavrila. A Bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(8):1408–1421, 2007.

[16] J. Giebel and D. M. Gavrila. Multimodal shape tracking with point distribution models. *Pattern Recognition*, pages 1–8, 2002.

[17] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[18] D. Hoiem, A. A. Efros, and M. Hebert. Recovering surface layout from an image. *International Journal of Computer Vision*, 75(1):151–172, 2007.

[19] I. Kokkinos and P. Maragos. Synergy between object recognition and image segmentation using the expectation-maximization algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(8):1486–1501, 2009.

[20] M. Kumar, P. Torr, and A. Zisserman. Objcut: Efficient segmentation using top-down and bottom-up cues. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(3):530–545, 2010.

[21] J. Lafferty, A. McCallum, and F. C. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proc. of the International Conf. on Machine Learning (ICML)*, pages 282–289, 2001.

[22] D. Larlus and F. Jurie. Combining appearance models and markov random fields for category level object segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1–7. IEEE, 2008.

[23] A. Monroy and B. Ommer. Beyond Bounding-Boxes: Learning Object Shape by Model-Driven Grouping. *Proc. of the European Conf. on Computer Vision (ECCV)*, Part 3:580–593, 2012.

[24] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. *Advances in Neural Information Processing Systems (NIPS)*, 19:985–992, 2007.

[25] M. Pointer. A comparison of the CIE 1976 colour spaces. *Color Research & Application*, 6(2):108–118, 1981.

[26] I. Rauschert and R. Collins. A Generative Model for Simultaneous Estimation of Human Body Shape and Pixel-Level Segmentation. *Proc. of the European Conf. on Computer Vision (ECCV)*, pages 704–717, 2012.

[27] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *Proc. of the ACM Transactions on Graphics (SIGGRAPH)*, 23(3):309–314, 2004.

[28] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth. Efficient Multi-Cue Scene Segmentation. In *Lecture Notes in Computer Science (Proc. of the German Conf. on Pattern Recognition (GCPR))*, volume 8142. Springer, 2013.

[29] J. Tighe and S. Lazebnik. Superparsing. *International Journal of Computer Vision*, 101(2):329–349, 2013.

[30] L. Wang, J. Shi, G. Song, and I.-F. Shen. Object detection combining recognition and segmentation. In *Asian Conf. on Computer Vision*, pages 189–199. Springer, 2007.

[31] C. Zhang, L. Wang, and R. Yang. Semantic segmentation of urban scenes using dense depth maps. *Proc. of the European Conf. on Computer Vision (ECCV)*, pages 708–721, 2010.