Combined Object Detection and Segmentation

Jarich Vansteenberge, Masayuki Mukunoki, and Michihiko Minoh

***Abstract*—We develop a method for combined object detection and segmentation in natural scene. In our approach segmentation and detection are considered as two faces of the same coin that should be combined into a single framework. There are two main steps in our strategy. First we focus on the learning of a visual vocabulary that efficiently encompasses objects' appearance, spatial configuration and underlying segmentation. This vocabulary is used within a Hough voting framework to produces object's configuration. The second step consists in searching for valid objects' configurations by interpreting and scoring them in terms of both detection and segmentation. This allows us to prune false detections and hallucinated object-like segmentation. Experiments show the advantage of the combined approach and the improvements over recent related methods.**

***Index Terms*—Object recognition, random forest, hough votes.**

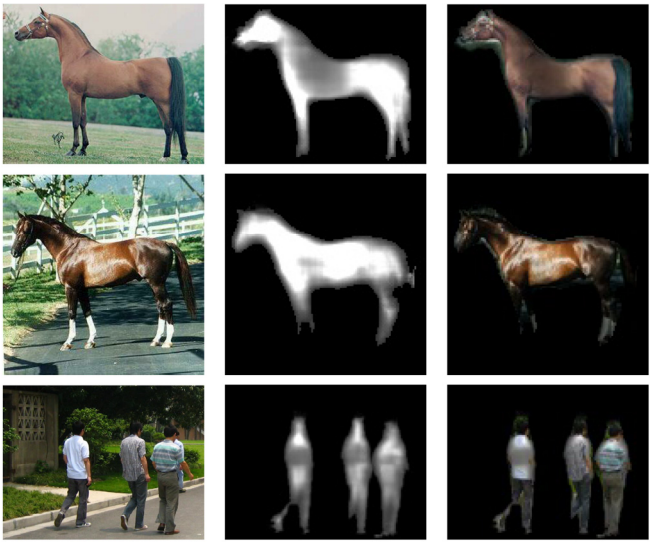

Fig. 1. Example results of our proposed combined object detection and segmentation approach. The combination of segmentation and detection improves the accuracy of both detection and segmentation processes.

I. Introduction

Mimicking the human vision system's ability to identify an object and to isolate it from its environment is one of the most challenging tasks in the field of computer vision. A tremendous amount of work has been done over the years, leading to different formulation of the problem and different approaches in handling this task.

One of the major approaches is the object detection problem in which objects' scales and locations within the images are to be discovered. Another major approach is the object segmentation problem where images are divided into regions with some of them being the objects' boundaries. One can see that while being different these two problems are strongly related. Several approaches have been proposed to combine detection and segmentation [1]-[4]. A straightforward approach consists in performing the segmentation within the object bounding box provided by a strong detector [5], [6]. While offering accurate segmentation, such method heavily depends on the quality of the detection results. Moreover no feedbacks from the segmentation process are provided to the detector.

Manuscript received September 14, 2012; revised December 27, 2012. This work was supported in part by the Academic Center for Computing and Media Studies, Kyoto University, Japan.

J. Vansteenberge is with the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University, Yoshida Honmachi, Sakyo-ku, Kyoto, Japan. (e-mail: vansteenberge@mm.kyoto-u.ac.jp).

M. Mukunoki and M. Minoh are with the Academic Center for Computing and Media Studies, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, 606-8501, Japan.

Ramanan [7] has shown that by using the segmentation to verify detection hypothesis, one could significantly improve the detection performances. Nevertheless, in their approach detection and segmentation are performed sequentially thus ignoring the interactions between them.

ObjCut [4] provides an elegant method allowing detection and segmentation process to continually interact with each other. Using MRFs with a layered pictorial structure the algorithm can achieve object detection along with high accuracy segmentation within natural scene. The total number of parts in the pictorial structure and their parameters are fairly limited which makes their model very sensitive to viewpoint variations.

The Implicit Shape Model (ISM) [1] provides an interesting way to combine detection and segmentation within a single framework. The idea is to learn a visual dictionary of local appearance and its spatial distribution over a star shaped model. At training time the visual dictionary is enriched with underlying segmentation mask and the matching location in regards to the object center. At run time the dictionary is used within a Hough voting framework to casts votes for the object location. A segmentation mask is inferred from visual words' local segmentation masks.

The ISM framework has several drawbacks. The learning of the visual dictionary of local appearances and its spatial distribution over the star shape model are learned independently. Moreover only positive samples can be used to generate the visual words (VW). Finally the aggregation of evidence within the Hough accumulator often leads to false detection on cluttered background. Over the years several modifications of the ISM framework have been proposed, mainly focusing on learning a better visual dictionary and

DOI: 10.7763/IJMLC.2013.V3.273

improving the voting procedure [8]-[10].

In our work, we present a method for combined object detection and segmentation within natural scene based on Random forest (RF) and the ISM framework. Our contribution is threefold; first we propose a way to learn a visual dictionary optimized for combined detection and segmentation. The appearance of the visual words, their distribution over the star shaped model and the underlying segmentation mask are jointly learned. Our second contribution lies in the efficient evaluation of the quality of the aggregated evidence within the accumulator. Each configuration maxima from the Hough accumulator are independently scored in terms of detection and segmentation. These scores are used to estimate the quality of the evidence combinations. Our last contribution consists of illustrating the benefits taken from performing segmentation and detection within the same framework. We show improvements in detection and segmentation performances when combining both processes.

II. Generating Object Configurations

In our approach, object recognition relies on the generation of a set of candidate object configurations which are then scored in terms of detection and segmentation (see Fig. 2). We define an object configuration $h=(\mathbf{c},\mathbf{m})$ as an assembly of parameters $\mathbf{c}$ related to the object detection i.e. the target object position and scale with parameters $\mathbf{m}$ related to the object segmentation. Similarly to [1] we use a visual dictionary to make assumptions about the objects' configurations. These assumptions, called votes, are collected into a Hough accumulator $H(x,y,s)$, where the candidate configurations are searched as local maxima.

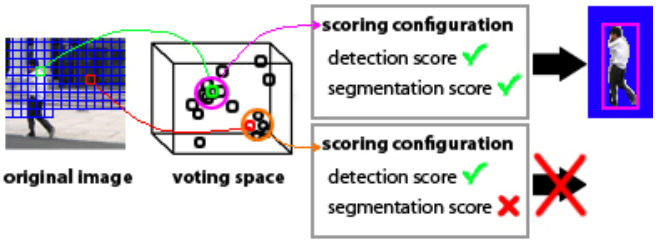

Fig. 2. Overview of our framework. The visual dictionary is matched against a test image. Each matching VW casts votes for the object configuration into an accumulator. Strongly supported configurations are scored in terms of detection and segmentation. True object configurations are required to have high scores for both detection and segmentation.

A. Voting for Detection Parameters

The voting procedure for the detection is defined as follow. Let f be an image feature extracted at location l. We match the visual dictionary against f to obtain a set of valid visual words $\{W_i\}$, hereafter called support evidences. The conditional probability of an object O existing at the position $\mathbf{c}$ within the voting space is computed as:

$$\begin{aligned} p(O,\mathbf{c}\mid f,l) &= \sum_i p(O,\mathbf{c}\mid W_i,f,l)\,p(W_i\mid f,l) \\ &= \sum_i p(\mathbf{c}\mid O,W_i,l)\,p(O\mid W_i)\,p(W_i\mid f). \end{aligned} \quad (1)$$

The first term is the distribution of the object's center position for a given VW. The second term specifies the confidence for a VW to be related to the target object. The last term describe the quality of the match between a feature and the matching visual words. So the pseudo probability of an object O at location $\mathbf{c}$ is defined as:

$$p(O,\mathbf{c}) = \sum_k p(O,\mathbf{c}\mid f_k,l_k)\,p(f_k,l_k)\ , \quad (2)$$

with $|k|$ the total number of features extracted from the test image. Setting the accumulator dimensions to be equal to the test image dimensions, we have a direct relation between positions in voting space and in the image space.

B. Voting for Segmentation Parameters

Aside from voting for detection parameters, the support evidences need to vote for the segmentation parameters $\mathbf{m}$. Being directly related to the target object, the support evidences provide local interpretations of the image content. Assembling the local interpretations agreeing on a same object configuration allows inferring a global interpretation for the target object. Especially one can compute a back-projected segmentation mask $\mathbf{m}$ as explained in section IV.*A*. To do so we need to store additional information about the support evidences. A single vote from a feature f_k stored at $\mathbf{c}$ in the accumulator is:

$$v_{c,k} = \left(p(O,\mathbf{c}\mid f_k,l_k), l_k, \{W_i\}_k\right), \quad (3)$$

with $p(O,\mathbf{c}\mid f_k,l_k)$ the confidence of the vote and l_k the extracted feature location. The accumulation of votes from multiple features for a configuration at location $\mathbf{c}$ becomes:

$$v_c = \left(p(O,\mathbf{c}), \{l_k\}, \{\{W_i\}_k\}\right), \quad (4)$$

where $p(O,\mathbf{c})$ is the confidence for the target object O to be found at position $\mathbf{c}$. The corresponding configuration $h=(\mathbf{c},\mathbf{m})$ is composed of $\mathbf{m}=\psi(\{l_k\},\{\{W_i\}_k\})$ the back-projected segmentation mask and $\mathbf{c}=(c_x,c_y,c_s)$ the 3D location within the accumulator.

III. Learning a Visual Dictionary

The visual words are used to cast votes for the objects configurations which makes them crucial to the performances of the algorithm. A particular attention needs to be paid to their design.

A. Training Data

Our training data is a set of local patches extracted from random locations within positive and negative training images. Each training patch $P_i=(A_i,F_i,g_i,d_i)$ has a local appearance patch A_i, a ground truth local segmentation mask F_i, a class label g_i and an offset to the object center d_i. For negative patches, the local segmentation and offset vector are left undefined.

B. Random Forest

To learn the VW from the training patches, we grow a RF to act as a visual dictionary. This idea has been exploited with success by the past [11],[10], however, to the best of our knowledge, we are the first to take into account the class label, the offset vectors and the local segmentations into the learning of the visual dictionary. The idea is to grow the trees such as to learn discriminant local appearances for consistent locations relative to the object center and consistent local segmentations. A RF is composed of a set of trees trained individually as explained in the following paragraph.

Training. Starting from the root, each node of a tree splits the incoming patches $\{P_i\}$ into two subsets according to a binary test $t(A) \to \{0,1\}$. The test simply compares the pixel's values at two random positions within the appearance patch A_i .The two subsets are passed onto child nodes where further splitting is performed. This recursive splitting leads to a large number of subsets in which patches share a similar appearance. The path from the tree's root to a given leaf describes the patches' shared appearance. Thus each leaf acts as a VW. In order to product strong votes for the object configuration, a visual word should: Be highly confident in voting, vote for a precise location and vote for a consistent local segmentation mask. When growing a tree, we need to choose at each node the binary test which increases the potential vote's quality. We define separate uncertainty measurement for our three criterions. Let $Z = \{P_i\}$ be a subset of patches leaving a given node in a given tree. The uncertainty over the class labels $\{g_i\}$, offset vectors d_i and local segmentation masks F_i are defined as:

$$U_1(Z) = -\sum_{i=1}^{|Z|} p(g_i)\log(p(g_i)) \tag{5}$$

$$U_2(Z) = \sum_{i:g_i=1} (d_i - \bar{d})^2 \tag{6}$$

$$U_3(Z) = \sum_{i:g_i=1} (F_i - \bar{F})^2 , \tag{7}$$

with $g_i = 1$ if the patch is a positive sample, $\bar{d}$ and $\bar{F}$ the mean offset vector and mean segmentation patch in Z . The first measurement tends to improve the VW discriminant power. The second and third uncertainty measurements improve the votes' locations and segmentation mask accuracies.

At each non leaf node of a tree, one of the three uncertainty measurements is randomly selected and used to score binary tests. The test which minimizes the uncertainty is kept and the tree is grown up to the next level. Inhibiting some of the uncertainty measurements will optimize the trees for either detection, with compact votes' locations or segmentation, with large back-projected areas. The Fig. 3 show the influence of the training when U2 or U3 are inhibited.

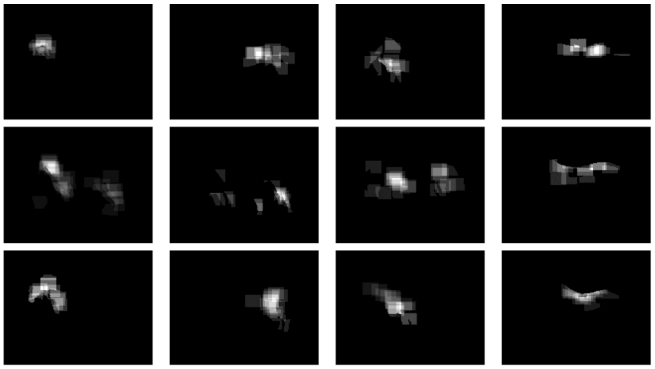

Fig. 3. Back-projections of confident leaves trained on Weizmann horse dataset. The top, middle and bottom rows shows sample leaves from RF optimized for detection only, segmentation only, combined detection and segmentation. The combined approach takes benefit from both approaches. Votes are coming from compact locations, which favor detection, while the back-projected masks cover large local areas and preserves boundaries.

When a tree's maximal depth has been reached or when the number of patches in a subset is too small, we create a leaf node and store the patches' information, that is, the number of positive patches $P^p = |\{P_i \mid g_i = 1\}|$, the total number of patch $|P_i|$, the offset vectors $\{d_i\}$ and the set of local segmentations $\{F_i\}$.

Testing. At run time patches are extracted from random locations in the test images and passed through each tree of the forest. Each time a single patch from location l , reach a leaf node W_i a vote is casted according to (1). The location distribution $p(\mathbf{c} \mid O, W_i, l)$ is determined by the set of location $\{\mathbf{c}\} = l - \{d_i\}$ weighted by a uniform probability $1/|d_i|$. The probability $p(O \mid W_i)$ is estimated by the ratio $|P^p|/|P_i|$. Finally the probability $p(W_i \mid f)$ is set uniform over the number of trees in the forest.

IV. Evaluating Proper Object Configurations

Generalized Hough transform is known to be a very robust parameter estimation method. In our framework, it allows for object detection under large occlusions and poses changes. However the additive nature of the accumulation of evidences is equivalent to assuming independence between the support evidences. This crude assumption makes the Hough framework sensitive to cluttered background, which produces falsely confident configurations.

This is where the combined approach shows its power. Obviously two adjacent support evidences are strongly correlated. When casting votes for the detection parameters the correlation of the support evidences is lost. However, the back-projected segmentation mask keeps the spatial relationship between the support evidences. One can score such mask to estimate if the spatial distribution of support evidences is compatible with a true object configuration. We end up with two scores for a single object configuration.

Assuming independence between **c** and **m** allows to simplify the computation of a candidate configuration's final score which is:

$$P(O, h_j) = P(O, \mathbf{c}_j) P(O, \mathbf{m}_j) . \tag{8}$$

The probability $P(O,\mathbf{c}_j)$ from (2) is the object configuration's detection score while $P(O,\mathbf{m}_j)$ is the segmentation score. The finals objects' configurations satisfying both detection and segmentation are defined as:

$$h^* = \{h_j \mid P(O,h_j) > \upsilon\}, \quad (9)$$

with υ the threshold controlling the strictness of the algorithm. In the following section, we show how to compute $P(O,\mathbf{m})$ from the support evidences.

A. Back-projected top-down segmentation mask.

Similarly to the detection parameters $\mathbf{c}$, the segmentation parameters $\mathbf{m}$ which corresponding to the foreground labeling, are also extracted from the support evidences' votes. We start by collecting all the votes for an object configuration h within a circular region centered at the configuration's location $\mathbf{c}_h$. The collected votes contain a set of support evidences $\{W_i\}$ but also their matching locations $\{l\}$. Reminding that local ground truth segmentations F_i are available for each VW word, one can produce a global segmentation mask by assembling these local segmentations. We closely follow the probabilistic formulation of [1] where the backprojected segmentation mask $\mathbf{m}$ is computed as a weighted sum of the local segmentation masks. Fig. 4 shows samples back-projected segmentation masks.

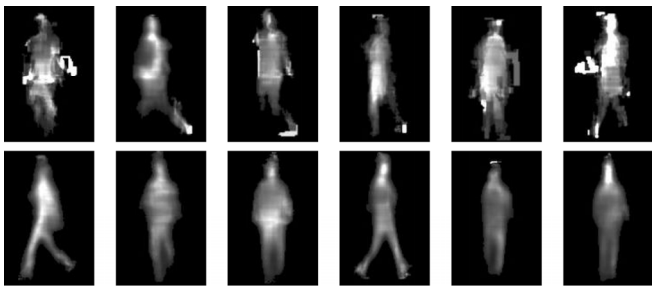

Fig. 4. Back-projected segmentation masks. The first and second rows show the back-projected masks for respectively, false and true object configurations.

B. Scoring the Segmentation Mask

Once a back-projected segmentation mask is available, it can be used to score the combination of support evidence from which it originated. Indeed, we can observe from Fig. 4 that proper combinations of support evidences lead to object-like segmentations while some of the poor combinations produce ill shaped segmentations. To estimate the quality of the support evidence combination we learn a scoring function on positive and negative back-projected masks. HoG [12] feature descriptors are extracted from each segmentation mask and serves as training samples for learning a scoring function. We used the libSVM package to learn the segmentation class probability $P(O,\mathbf{m})$. The model's output gives the probability for a segmentation mask to be the target object foreground.

V. Results

We have tested our algorithm for detection and segmentation on three challenging datasets, the Weizmann Horse dataset [13], the TUD pedestrian dataset [14] and the PennFudan pedestrian dataset [2].For each dataset, the positive training images have been resized so that each object's bounding box would have its largest dimension approximately equal to 120 pixels. All the trees were trained on 18000 positive and 18000 negative patches. The training patches of 16 by 16 pixels were extracted at random locations within the bounding box of positive images and anywhere within negative images. For both pedestrian datasets, we used a subset of 600 images from the INRIA dataset [12] as negative training set. Each RF was composed of 5 trees.

The SVMs used to score the segmentation masks are using RBF kernel and have been trained on 50 positives and 50 negatives masks for the Horse dataset and 200 positives and 200 negatives masks for both pedestrian dataset. A true detection should overlap the ground truth bounding box by more than 0.5. To avoid multiple detections of the same instance we use non-maxima suppression.

Due to the lack of standardized evaluation measures in the segmentation community, we use two measurements to evaluate the segmentation performances. The Fscore defined as $Fscore = (2\times\text{precision}\times\text{recall})/(\text{precision}+\text{recall})$ and the foreground accuracy computed as $Acc = (\text{intersection})/(\text{union})$.

The Table I show the superiority of the combined approach over specialized approach. All the segmentation results are given for a detection recall of 97.7%. We can see the combined approach performs the best for both detection and segmentation. This dataset was originally built for segmentation which makes it not very challenging for detection. To better illustrate the detection's improvements, we used the TUD pedestrian dataset. Fig. 5 shows the combined approach increased both precision and recall when comparing to RF optimized for detection only. We also improves over two Hough-based approach, the 4DISM [15] and the Hough Forest [10] retrained from the code available on their website.

TABLE I: Detection and Segmentation Results on the Horse Dataset.

	RF Det. only	RF Det. + Seg.	RF Seg. only
Det. EER	98.8%	98.8%	97.7%
Seg. Fscore	78.2%	79.2%	78.9%

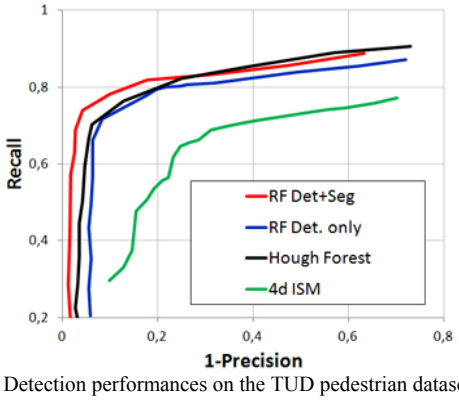

Fig. 5. Detection performances on the TUD pedestrian dataset. The combined approach perform the best among Hough based approach.

The Table II shows the comparison of performance for the Weizmann horse dataset. At detection EER, we achieve better detection results and improved the segmentation quality by 10.1% compared to the recent related method of Torrent et al. [16]. We also get higher segmentation quality than the early bottom-up segmentation method of Ren et al. [17]. Finally we achieve worse segmentation but similar detection compared to the state of the art method [18].

TABLE II: DETECTION AND SEGMENTATION RESULTS ON THE WEIZMANN HORSE DATASET AT DETECTION EER.

Methods	Seg. Fscore	Det. EER	Image
Zhu [18]	89.2%	99.1%	228
Ren et al. [17]	80.2%	-	172
Torrent et al. [16]	69.1%	97.0%	262
RF Det. + Seg.	80.7%	98.8%	262

Comparative results for the PennFudan dataset can be seen in Table III. We compared our results with two works having reported detection and segmentation results for this dataset.

TABLE III: DETECTION AND SEGMENTATION RESULTS ON THE PENNFUDAN PEDESTRIAN DATASET AT DETECTION EER.

Methods	Fscore	Acc.	EER	Image
Bo [5]	82.9 %	73.2 %	85.5%	101
RF Det. + Seg.	83.7 %	72.8 %	85.4%	101
Wang et al. [2]	-	-	59.5 %	345
RF Det. + Seg.	78.4 %	64.7 %	80.7 %	345

We improve detection EER by more than 20% over Wang et al. [2] results. They did not provide quantitative results for the segmentation. However visual comparisons of the produced masks show the higher segmentation accuracy of our approach (see Fig. 6). Recent results [5] have been published for a subset of the original database containing only 101 fully un-occluded pedestrians from the original 345. We achieve very similar performances for both the detection and segmentation. It should be noticed that Bo et al. [5] are using state of the art bottom up segmentation algorithm, while our segmentation is a purely top-down. Furthermore their segmentation results are heavily depending on the detection's bounding box. Our approach doesn't suffer from this flaw. When tested on the full dataset, we observe a decrease in our performances due to the heavy occlusion that appears within the test set.

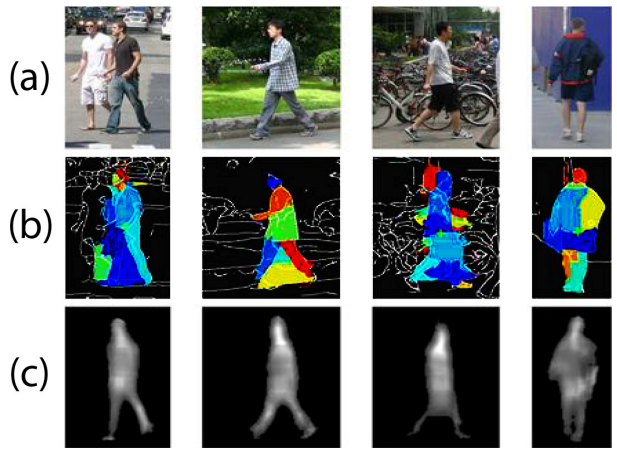

Fig. 6. Visual comparison of segmentation results for the PennFudan dataset. (a) Original images. (b) Wang et al. [2] segmentation results. (c) Our results.

VI. CONCLUSION AND DISCUSSION

We have presented a simple method to efficiently combine detection and segmentation into a single framework. Both elements have been taken into account at training time, in the building of the visual vocabulary and the model but also at run time by scoring object configurations in term of detection and segmentation. The experiments have shown the performance improvements of the combined approach over specialized approach and clear improvements in comparison to closely related methods. The algorithm performs on par with state of the art for some of the tested dataset. As future work, we are planning to include pose estimation within our framework and introducing occlusion within our model.

REFERENCES

[1] B. Leibe, A. Leonardis, and B. Schiele. "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision,* vol. 77, pp. 259–289, 2008.

[2] L. Wang, J. Shi, G. Song, and I. fan Shen, "Object detection combining recognition and segmentation," in *Proceedings of the 8th Asian Conference on Computer Vision*, vol. 4843 of Lecture Notes in Computer Science, pp. 189–199. Springer, 2007.

[3] Z. Tu, X. Chen, A. L. Yuille, and S. C. Zhu. Image parsing: Unifying segmentation, detection, and recognition. *In Toward Category-Level Object Recognition*, pp. 545–576, 2006.

[4] M. P. Kumar, P. H. S. Torr, and A. Zisserman, "OBJ CUT," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, vol. 1, pp. 18–25, 2005.

[5] Y. Bo and C. Fowlkes, "Shape-based pedestrian parsing," *CVPR*, 2011.

[6] V. S. Lempitsky, P. Kohli, C. Rother, and T. Sharp, "Image segmentation with a bounding box prior," in *Proc. ICCV*, pp. 277–284, 2009.

[7] D. Ramanan, "Using segmentation to verify object hypotheses," in *Proc. CVPR*, pp. 1–8, 2007.

[8] S. Maji and J. Malik, "Object detection using a max-margin hough transform," in *CVPR'09*, pp. 1038–1045, 2009.

[9] B. Ommer and J. Malik, "Multi-scale object detection by clustering lines," in *Proc. ICCV* 2009.

[10] J. Gall and V. Lempitsky, "Class-specific hough forests for object detection," in *Proc. IEEE Conference Computer Vision and Pattern Recognition*, 2009.

[11] J. Shotton, M. Johnson, and R. Cipolla, "Semantic texton forests for image categorization and segmentation," in *Proc. CVPR*, 2008.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. International Conference on Computer Vision & Pattern Recognition*, vol. 2, pp. 886–893, June 2005.

[13] E. Borenstein, "Combining top-down and bottom-up segmentation," in *Proc. IEEE workshop on Perceptual Organization in Computer Vision, CVPR*, pp. 46, 2004.

[14] M. Andriluka, S. Roth, and B. Schiele, "People tracking by detection and people detection by tracking," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08)*, 2008.

[15] E. Seemann, B. Leibe, and B. Schiele, "Multi-aspect detection of articulated objects," in *Proc. CVPR,* vol. 2, pp. 1582–1588, 2006.

[16] A. Torrent, X. Llad´o, J. Freixenet, and A. Torralba, "Simultaneous detection and segmentation for generic objects," in *Proc. ICIP*, pp. 653–656, 2011.

[17] X. Ren, C. Fowlkes, and J. Malik, "Cue integration for figure/ground labeling," in Y. Weiss, B. Sch¨olkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems,* vol. 18, pp. 1121–1128. MIT Press, Cambridge, MA, 2006.

[18] L. Zhu, Y. Chen, and A. L. Yuille, "Learning a hierarchical deformable template for rapid deformable object parsing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 6, pp. 1029–1043, 2010.