Object Detection by Labeling Superpixels

Junjie Yan[1,2] Yinan Yu[3] Xiangyu Zhu[1] Zhen Lei [1] Stan Z. Li [1]
[1]National Laboratory of Pattern Recognition, Chinese Academy of Sciences
[2]Institute of Data Science and Technology, Alibaba Group
[3]Institute of Deep Learning, Baidu Research

Abstract

Object detection is often conducted by object proposal generation and classification sequentially. This paper handles object detection in a superpixel oriented manner instead of the proposal oriented. Specially, this paper takes object detection as a multi-label superpixel labeling problem by minimizing an energy function. It uses the data cost term to capture the appearance, smooth cost term to encode the spatial context and label cost term to favor compact detection. The data cost is learned through a convolutional neural network and the parameters in the labeling model are learned through a structural SVM. Compared with proposal generation and classification based methods, the proposed superpixel labeling method can naturally detect objects missed by proposal generation step and capture the global image context to infer the overlapping objects.

The proposed method shows its advantage in Pascal VOC and ImageNet. Notably, it performs better than the ImageNet ILSVRC2014 winner GoogLeNet (45.0% V.S. 43.9% in mAP) with much shallower and fewer CNNs.

1. Introduction

Object detection is a computer vision task to automatically localize objects in categories of interest from images. Starting from early methods which can successfully localize constrained object categories, such as face [42, 52] or pedestrian [8, 11], state-of-the-art methods [15, 20] are moving focus to the detection of varying categories with large appearance variations, such as the twenty categories in Pascal VOC [13] and two hundred categories in ImageNet [43].

While numerous works have been proposed for object detection, most of them actually transform the object detection to image classification. They first generate object proposals and then classify each proposal independently by the image classification techniques. The traditional paradigm to get proposal [38, 52] is to use the sliding window to exhaustively sample about 100, 000 bounding boxes in various scales and locations. The recently popular paradigm is to generate about 2, 000 proposals by clustering or segmentation according to low-level image cues. After that, image classification techniques are used to classify each proposal. The classification has achieved great advances recently, due to the robust low level features [8, 35], sophisticated models [40, 4, 15] and convolutional neural networks (CNN) [28, 46].

Through the transformation, the detection performance can benefit from the advances in image classification. It leads to the great improvement in detection of face, pedestrian and more general object categories in the last two decades. However, it also results in two problems. The first is that if an object is missed in object proposal step, such as an object with partially occlusion or unusual aspect ratio, the detection system would definitely miss the object. The second is that the independent classification of proposals cannot incorporate the global image context, which is very important to detect overlapped objects and distinguish object part and object itself.

To alleviate the two problems, we believe one possible solution is to move the focus in detection from proposals to superpixels. The superpixels are compact and perceptually meaningful atomic regions for images. The pixels in one superpixel can be safely assumed to belong to the same object (as long as the scale of superpixel is small enough) and superpixels can be grouped together flexibly to form objects. The interaction between objects, which is hard to model in object level, also becomes easier in superpixel level. If we know the label of each superpixel (e.g., it belongs to which object in what category), then the object detection problem becomes trivial. To this end, we conduct object detection by labeling superpixels.

However, reliable inference of a superpixel's label can be very difficult, due to the ambiguity in its appearance. In this paper, we exploit three types of information on entire image jointly by constructing an energy function on image's superpixel partition. The appearance of the superpixel is captured by a data cost term, which is propagated from classification result of the regions it belongs to by RCNN[20]. The spa-

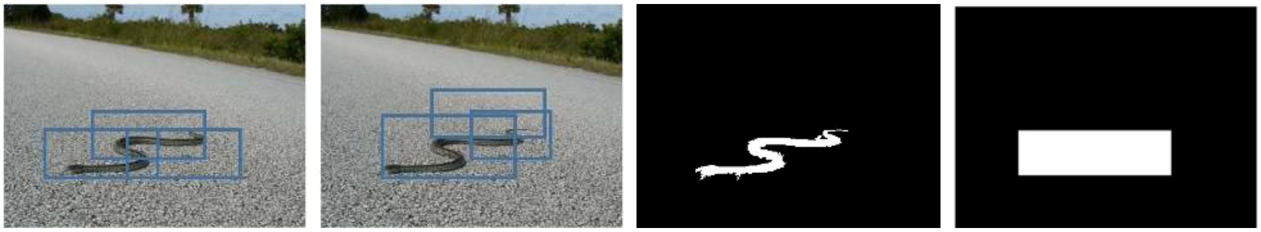

(a) sliding window (b) selective search (c) superpixel labeling (d) definition of object detection

Figure 1. Different methods for object detection. The sliding window (Fig. 1(a)) and selective search (Fig. 1(b)) based methods handle the object detection by proposal generation and independently classification in a sequential manner. The proposed superpixel labeling method (Fig. 1(c)) directly outputs the object masks for detection. The object detection problem itself can be taken as a pixel labeling problem (Fig. 1(d)), where the detection is a task the predict the labels of pixels (best viewed in color).

tial context, such as whether two superpixels belong to the same object, is captured by a smooth cost term. Since compact detection is always favored, we add a label cost term to punish the number of labels used. In this way, the detection becomes a multi-label labeling problem with label cost, and α-expansion based method such as [9] can be used for approximate inference. To learn the parameters in the energy function, such as the weight of different terms, a structural SVM is conducted to maximize the detection performance.

It should be noted that the proposed superpixel labeling method is closer to the essential definition of object detection, which infers pixels' labels of belonging objects. As shown in Fig. 1, for sliding window and selective search based method, the inference is conducted by classifying each proposal and the heuristic method like NMS is used to merge the classified proposals. Instead, the proposed method infers the labels of superpixels globally to derive the object location. One by-product of the proposed superpixel labeling based detection is that it can output a coarse mask for each detection, although only annotations of bounding boxes are used for training.

The rest of the paper is organized as follows. Section 2 reviews the related work. The motivation of superpixel based detecton and the details of the superpixel labeling method are described in Section 3 and Section 4. In Section 5 we show experimental results and finally in Section 6 we conclude the paper.

2. Related Work

The improvement in object detection can be divided into proposal generation and proposal classification. Typical trends are the proposal number becomes smaller and smaller and the classification method becomes more and more complex.

To generate object proposals, the most direct and commonly used procedure is the sliding window for exhaustive search. It is popularized by early works in pedestrian detection [38] and face detection [52]. The current publicly available state-of-the-art face detection [37, 5] and pedestrian detection [55, 57] methods are all based on sliding window. The deformable part model (DPM), which is the foundation of champion systems in Pascal VOC 2007-2011, is also based on sliding window. The main drawback of the sliding window is that the number of proposals can be about $O(10^6)$ for a 640×480 image, which limits the complexity of classification due to the evaluation efficiency.

Various methods are proposed to reduce the number of proposals. It is proven useful in [23] and popularized by [50]. In [50], the superpixels generated by [16] are hierarchically grouped to form object proposals. The number of proposals can be about 2, 000 with a recall rate of 98% on Pascal VOC and 92% on ImageNet. Besides the small number, another advantage is that proposals at arbitrary scale and aspect ratio can be generated, which provides more flexibility for general object detection. This method is widely used by leading object detection methods on Pascal VOC [20] and ImageNet [46]. Recently, many methods are further proposed to get more compact and efficient object proposals, including the unsupervised approach [23, 50, 2] and the supervised approach [1, 62, 6, 36]. An evaluation and survey on recent object proposal method can be found in [24].

When the proposals are fixed, detection becomes classification of each proposal. It involves how to represent the proposal and how to classify the representation. The feature representation becomes more and more sophisticated, from hand-crafted Haar [52] and HOG [8] to learning based CNN [20]. Built on top of these feature representations, carefully designed models can be incorporated. The two popular models are the deformable part model (DPM [15]) and the bag of words (BOW [40, 4]). Given the feature representation, classifier such as Boosting [17] and SVM [7] are commonly used for classification. Structural SVM [49, 26] and its latent version [60] are recently widely used when the learning data has structural loss, such as DPM. A recent work [21] also shows that the DPM can be interpreted as a CNN. The CNN based representation has shown great advantages and has been adopted by all the leading methods in ImageNet [43].

Previous works have noticed the problems in proposal based detection. In [10, 41], context models are built to learn the context information to improve the heuristic non-maximum suppression. In [19, 48], spatial models are used to inference the occlusion. In [15, 20], regression is used to refine the bounding box. However, all of these methods cannot generate new object proposals and their performance is limited by the proposal used.

A small number of methods which do not use the proposal generation and classification paradigm have been proposed. The implicit shape model [32] generalizes the hough transform to combine object shape information of training samples for object detection and probabilistic segmentation. This method is further improved in [18, 3]. [47, 12] use deep neural network to simultaneously regress the detection bounding boxes and their detection scores. [22, 58, 59] infer whether an off-the-shelf detection is right or wrong by jointly optimize the detection and segmentation. Although promising directions are provided, the performance still does not match the leading proposal generation and classification method, such as the RCNN [20].

Our superpixel labeling method for object detection is related to semantic image segmentation and scene parsing. [30] captures the object co-occurrence by the label cost term for semantic segmentation. [31] uses the conditional random field (CRF) to combine object detection and segmentation. However, these works are designed for Pascal VOC segmentation task, where overlapped objects of the same category are taken as one segment. In [14], CRF is built on top of CNN features for scene parsing. [29] proposes to use detection annotation to infer the segmentation mask. Very Recently, [34] releases the Microsoft COCO dataset with object level mask, which can be used to improve our method.

3. Motivation

We use superpixel as the atom in further operations. The ideal superpixel partition for detection is that the superpixel number is small enough for the efficiency in inference and each superpixel does not span in multiple objects. In this paper, we use the superpixel generation algorithm proposed in [16], which well satisfies this requirement. To increase the diversity of superpixels, four parameter settings are used to generate superpixels, as the setting of "fast mode" suggested in [50]. Throughout this paper, the four superpixel partitions are handled independently, and we only describe operations in one superpixel partition for the simplicity in notation.

Table 1. Comparison of labeling pixels, superpixels and proposals for object detection on ILSVRC2014 val2. The N_p, N_s and N_r are the number of pixels, superpixels and proposals, respectively. K is the possible number of objects in one category for an image, for example 5.

Method	Recall @0.9	Recall @0.5	Solution Space
Pixel	100%	100%	N_p^K ($\sim 10^{26}$)
Superpixel	99.8%	100%	N_s^K ($\sim 10^{13}$)
Proposal	25.5%	91.7%	N_r (~ 2000)

We compare the superpixel based method for detection with proposal based method and pixel based method on val2 of ILSVRC2014 [1]. If we can successfully label each pixel, superpixel and proposal (this is to say, we know it belongs to which object in what category), the recall rates at 0.9 and 0.5 overlap ratio [2] are listed in Tab. 1. The pixel based method can naturally get 100% recall rate at any overlap ratio, but the output space is too large and becomes infeasible. To our best knowledge, no successful methods have been reported on pixel based object detection. The proposal based methods have very small output space, but the recall ratio is not enough, especially when the requirement of overlap ratio is high. The proposed superpixel based method, can be taken as a trade-off between the pixel based method and proposal based method. It has nearly 100% recall with a reasonable output space.

By moving the focus from proposal to superpixel, it is possible to achieve higher recall and larger overlap ratio, but it also confront challenges due to the large output space. In the following part, we show how to regularize the model for effective inference and learning.

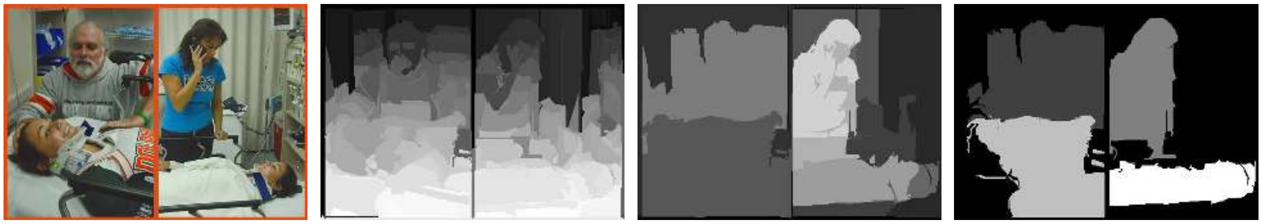
4. Methodology

For each superpixel generation setting, we can get a superpixel partition of an image and denote it as $\mathcal{P} = \{p_1, p_2, \cdots, p_N\}$, where p_i is the i-th superpixel and N is the superpixel number. Based on the partition, we also have a neighborhood system $\mathcal{N}$, where $(p_i, p_j) \in \mathcal{N}$ if p_i and p_j are spatially connected. The detection is conducted by finding a label configuration for each superpixel $\mathcal{L} = \{l_1, l_2, \cdots, l_N\}$, where the label $l_i \in \{0, 1, 2, \cdots, \infty\}$. Here $l_i = 0$ means p_i belongs to the background, $l_i = j$ means p_i belongs to the j-th object and the object number can be any non-negative integer. For the simplicity, we handle each category independently at the labeling step.

For each labeling configuration, we define an energy function $E(\mathcal{L})$ to measure its cost and can find the best label configuration $\mathcal{L}^*$ with the smallest cost by minimizing $E(\mathcal{L})$. Now let us think what an appropriate label configuration should be. When considering each superpixel independently, its label should be based on the fitness between its appearance and the appearance model learned from the

[1] https://github.com/rbgirshick/rcnn/tree/ilsvrc.

[2] The overlap ratio is based on the definition in Pascal VOC [13], which is the intersection of two regions against the union of the two regions.



(a) Input Image (b) Superpixel Partition (c) Data Cost (d) Superpixel Labeling

Figure 2. Example of the proposed superpixel labeling approach. We generate superpixel partitions for input images, and then calculate data cost for each superpixel by propagating the score of regions. However, the data cost term is always not enough for interacting objects, and we need smooth term and label cost term. The final superpixel labeling result is shown in Fig. 2(d).

training data of this category. Considering the smoothness nature of objects in image, the labels of neighborhood superpixels should be correlated and punished for varying labels. If two neighborhood superpixels have the same label and thus be taken as the same object, their appearance should also be correlated. Finally, the label configuration should favor fewer labels for compact detection. To this end, we use the following energy function,

$$E(\mathcal{L}) = \sum_{p_i \in \mathcal{P}} D(l_i, p_i) + \sum_{(p_i,p_j)\in\mathcal{N}} V(l_i, l_j, p_i, p_j) + C(\mathcal{L}), \quad (1)$$

where we always ignore the image notation I to simplify the notation. $D(l_i, p_i)$ is the data cost to capture the appearance of p_i and assign a cost based on the conflict between the appearance model and the label l_i. $V(l_i, l_j, p_i, p_j)$ is the pairwise smooth cost defined on the neighborhood system $\mathcal{N}$. $C(\mathcal{L})$ is the label cost term, which is defined on the label configurations $\mathcal{L}$ and is image invariant. It is motivated by the MDL prior and plays an important role to get objects in detection instead of object parts. In the following part, we show how to define the three terms in order to make them meaningful for detection and then show the inference and learning details.

4.1. Data Cost

The data cost for each superpixel should only be calculated by its appearance. However, appearance of a superpixel usually does not have enough semantic information, considering that it may only have a small number of pixels and corresponds to an ambiguous object part. One observation is that the regions (proposals), which are grouped neighborhood superpixels, provide more semantic object level information and the appearance model of regions can be well learned from annotation of detection. To make the superpixel data cost term more reliable, we classify regions and then propagate their costs to superpixels.

To get scores of regions, we use the RCNN approach proposed in [20], where output of the penultimate layer of a CNN trained for multi-category classification is used as feature extraction. For each category, a binary SVM is trained to distinguish object regions from the background and objects of other categories. Different CNN features can largely affect the final performance and we leave the details in the Section 5.2. Suppose the region set is $\mathcal{R} = \{r_1, \cdots, r_T\}$, and the classification score of r_t by RCNN is s_t, we use the sigmoid function to map it to the data cost ranging in $(0, 1)$,

$$D(l_t, r_t) = \begin{cases} \frac{1}{1+exp(-\alpha \cdot s_t)}, & \text{if } l_t > 0 \\ \frac{exp(-\alpha \cdot s_t)}{1+exp(-\alpha \cdot s_t)}, & \text{if } l_t = 0 \end{cases} \quad (2)$$

where α is set to be 1.5 empirically. The costs of all labels except 0 are the same since they indicate the region belongs to objects of a special category. One superpixel can belong to different regions, so that we need to pool the costs of different regions to a single value. For each superpixel, we use the weighted sum of T smallest costs,

$$D(l_i, p_i) = \sum_{t=1}^{T} w_{d_t} \cdot D(l_t, R(p_i)_t), \quad (3)$$

where $R(p_i)_t$ is the i-th regions p_i belongs to with the t-th smallest cost. The weight w_d is learned from the training data and T is set to be 3 empirically.

4.2. Smooth Cost

The smooth cost is used to encode the pairwise information. For the detection task, two kinds of information are useful. The first is that adjacent superpixels are often positively correlated and should be encouraged to have the same label. The second is that when the two adjacent superpixels have the same label and thus belong to the same object, they should be similar in appearance. To this end, the pairwise term is defined as:

$$V(l_i, l_j, p_i, p_j) = w_{s_l} V_l(l_i, l_j) + V_a(l_i, l_j, p_i, p_j), \quad (4)$$

where the $V_l(l_i, l_j)$ captures the first information and the $V_a(l_i, l_j, p_i, p_j)$ captures the second information.

For the $V_l(l_i, l_j)$, we set it to be a boolean variable. If $l_i = l_j$ and $(p_i, p_j) \in \mathcal{N}$, the cost is zero, otherwise the cost is a punishment 1. It can be denoted as $\delta(l_i \neq l_j)$. This term has a weight w_{s_l}.

For the $V_a(l_i, l_j, p_i, p_j)$, we need a cost to measure the appearance consistency of two neighborhood superpixels which are assigned with the same label. In this paper, we use the color and texture as two complementary criteria. We calculate a histogram with 25 bins for each color channel and then concatenate them to be a histogram with 75 bins. For the texture, we use the SIFT histogram as suggested in [50]. The cost is defined as,

$$\begin{aligned} V_a(l_i,l_j,p_i,p_j) = {} & w_{s_c}(1-\sum_q min(c_i^q,c_j^q)) \\ & +w_{s_t}(1-\sum_q min(t_i^q,t_j^q)), \end{aligned} \tag{5}$$

where c_i^q and t_i^q are the values in the q-th bin of color and texture histogram of superpixel p_i. $\sum min(c_i^q,c_j^q)$ and $\sum min(t_i^q,t_j^q)$ are the intersection distances of color and texture, ranging in $[0,1]$. The weights w_{s_c} and w_{s_t} will be learned automatically in the training step.

4.3. Label Cost

By introducing the similarity part in the smooth term, the final labeling result may contain many labels, such as parts of an object may have varying appearance and may be labeled as different objects. To this end, we need a term to favor compact detection by punishing the number of labels. The idea is related to the minimizing description length (MDL). In this paper, we use the following definition,

$$C(\mathcal{L}) = \sum_{i=1}^{K} w_{l_i}\cdot\delta(i,\mathcal{L}), \tag{6}$$

where $\delta(\cdot)$ is an indicator function defined as,

$$\delta(i,\mathcal{L}) = \begin{cases} 1, & \text{if } i\in\mathcal{L} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where the weight w_l will also be learned from the data. It is always the need that the weight w_i increases monotonically with i. Here we constrain the max number of objects for each category K to be 5 and set the weight of background label to be 0. Note that this cost is only related to the label configuration $\mathcal{L}$ and does not depend on image.

4.4. Inference and Learning

When the smooth cost term is a metric, the energy function can be solved by the extended α-expansion algorithm with well characterized optimal bounds as proved in [9]. Unfortunately, the smooth cost term used in this paper does not satisfy this, and we can only find the solution in a heuristic manner. To get the reliable labeling result, we need a good initialization. In this paper, we use the RCNN [20] detection result (the details of CNN can be found in Section 5.2) for initialization. For each detection of RCNN, we assign labels of superpixels in this detection to the detections' order number. The superpixel number for each partition is always no more than 500 for an image, so that the α-expansion is usually very efficient. After we get the superpixel labeling configuration $\mathcal{L}^*$, we simply connect superpixels with the same labels and use the corresponding bounding box as the detection result, where the score is the average score of its superpixels. If two regions are formed by superpixels of the same label but are not connected, we take them as two different instances in detection. An example of the superpixel labeling procedure can be found in Fig. 2.

The energy function defined above has the parameters w_d, w_s and w_l, where $w_s = [w_{s_l}, w_{s_c}, w_{s_t}]$. We learn them from the training data to optimize the detection performance. For each category in each image, the energy can be rewritten as a linear form in terms of w_d, w_s and w_l,

$$E(\mathcal{L}) = w^T\Phi(\mathcal{P},\mathcal{L}), \tag{8}$$

where w is the concatenation of w_d, w_s and w_l. $\Phi(\mathcal{P},\mathcal{L})$ is the concatenation of the costs on the entire image, which is defined as,

$$\begin{aligned} \Phi(\mathcal{P},\mathcal{L}) = [&\underbrace{\sum_{p_i\in\mathcal{P}} D(l_t,R(p_i)_t),}_{i=1,\cdots,T} \sum_{(p_i,p_j)\in\mathcal{N}} \delta(l_i\neq l_j), \\ &\sum_{(p_i,p_j)\in\mathcal{N}} (1-\sum_q min(c_i^q,c_j^q)), \sum_{(p_i,p_j)\in\mathcal{N}} (1-\sum_q min(t_i^q,t_j^q)) \\ &\underbrace{\delta(i,\mathcal{L})}_{i=1,\cdots,K}]^T. \end{aligned} \tag{9}$$

For an image I_m, suppose the ground truth superpixel labeling configuration is $\mathcal{L}_m$ and the labeling configuration inferred from the energy function is $\mathcal{L}_m^*$. We want to find the combination of $\{w_d, w_s, w_l\}$ that, given the image I_m, it tends to get $\mathcal{L}_m^* = \mathcal{L}_m$. Given M training images, the objective function can be defined as,

$$\begin{aligned} &\arg\min_{w,\xi_m\geq 0} w^T w + C\sum_{m=1}^{M}\xi_m \\ &s.t. \forall m\in[1,M], \forall\mathcal{L}'_m \\ &\quad w^T\Phi(\mathcal{P}_m,\mathcal{L}'_m) - w^T\Phi(\mathcal{P}_m,\mathcal{L}_m) \geq l(\mathcal{L}_m,\mathcal{L}'_m) - \xi_m \end{aligned} \tag{10}$$

where $w^T w$ is the regularization term. The constraint in Eq. 10 is specified as follows. Let us consider the m-th image with superpixel partition $\mathcal{P}_m$ and its ground truth label configuration is $\mathcal{L}_m$. We want the $\mathcal{L}_m$ to have smaller cost than all other label configurations $\mathcal{L}'_m$. However, not all the incorrect label configurations are equally bad. The loss function $l(\mathcal{L}_m,\mathcal{L}'_m)$ measures how incorrect $\mathcal{L}'_m$ is and penalizes the slack variable ξ_m according to the difference between $\mathcal{L}_m$ and $\mathcal{L}'_m$.

We decompose the loss $(\mathcal{L}_m,\mathcal{L}'_m)$ of superpixel labeling configurations to object level. Given the labeling configuration $\mathcal{L}'$, we can naturally get the object detection configuration. We calculate the number of true negative and false positive according to the Pascal VOC criterion [13] and use it as the cost. After the loss function and inference method are provided, the objective function defined in Eq. 10 can be solved by a cutting plane procedure and we use the package in [51] and refer the theory to [26].

5. Experiments

We evaluate the proposed method on ImageNet ILSVRC2014 detection task, which is currently the most challenging large scale detection dataset with 200 categories collected from the Internet. For the best practice, annotation of the testing set is not publicly available and the detection results are submitted to the testing server to get the performance. We compare our method with current state-of-the-art methods and then diagnose contribution of each step. We also report the performance on the widely used Pascal VOC 2007.

5.1. Comparison on ImageNet Detection

For the ImageNet object detection, we follow the training, validation and testing set partition in ILSVRC2014 [43]. We uses the CNNs, which are trained on the 1000 category classification data for initialization and fine-tuned on the detection data, as the setting in [20, 50, 56, 46]. As in [20], the proposal which overlaps a ground truth window with at most 0.3 is taken as a negative sample. We train four CNN models with the depth of 9, 10, 11 and 12, respectively. For the four CNNs, the final convolution layer is followed by a spatial pyramid pooling layer [27] and the output of the penultimate layer (the dimension is 4096) is used as the feature representation. Features of the four CNNs are concatenated as the final feature representation and fed into the binary linear SVM classifier. The final classification results are used to initialize the data cost term. After that, we use the proposed energy function to infer superpixel labels and get the detection result. We list mean average precision of the leading methods from 2013 to 2014 on the testing set, as well as our method, in Tab. 2. Since the number of models used for ensemble may significantly affect the final results, we also report the performance of single model to fairly compare each detection method.

Our best single CNN based model has a detection mAP of 42.5%. After ensemble of four CNNs, the mAP increases to be 45.0%. Our method improves one times compared with the champion in ILSVRC2013 and has already been better than the ILSVRC2014 champion GoogLeNet. We only use 4 CNNs while the GoogLeNet uses 7 CNNs, and our CNNs are not as sophisticated as the GoogLeNet. Our method shows that by carefully designing new detection method, there exists potentials to get better detection result although the CNN is not good enough. From Tab. 2, we find that large improvement from 2013 to 2014, mainly comes from the adoption of RCNN framework, which was originally proposed in [20]. Actually, all the 2014 methods listed above use the RCNN framework. The reasons of the different results are the proposals used and the features generated by different CNNs. The proposed superpixel labeling method can be naturally incorporated with these methods (by using them to enhance the data cost term in the energy function) to get further improvement.

Table 2. Results on the testing set of ILSVRC2014 detection task, which are merged by mean average precision (mAP) on 200 categories. The numbers of our method are got from the testing server, while numbers of other entries are directly from the ILSVRC2014 result page and corresponding papers. The methods marked with * do not use classification data for pre-training and marked with $^+$ only use the 2013 data.

Method	single model	# CNNs	Combined
NEC-RegionLet [54] $^+$	20.8	1	20.8
NYU-OverFeat [44] $^+$	-	7	24.3
UvA-Euvision [50] $^+$	22.6	6	22.6
MSRA-SPP-Net [27] *	31.8	6	35.1
NUS-NIN [33] *	35.6	3	37.2
Berkeley Vision [20]	34.5	1	34.5
UvA-Euvision [50]	35.4	1	35.4
Deep Insight [56]	40.2	3	40.5
CUHK-DeepID-Net [39]	37.7	10	40.7
GoogLeNet [46]	38.0	7	43.9
Superpixel Labeling	**42.5**	4	**45.0**

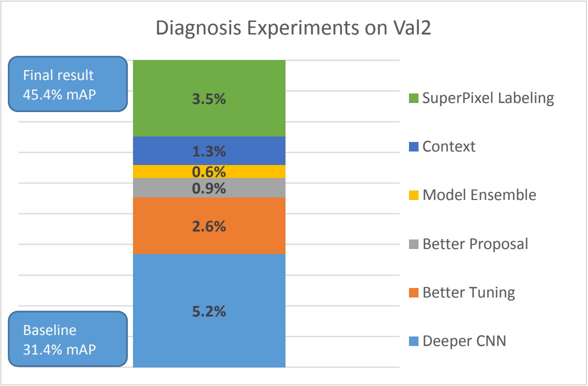


Figure 3. Diagnosis experiments on val2 of ILSVRC2014 detection (best viewed in color).

The detection performance varies a lot in 200 categories. We show the top 24 categories and bottom 24 categories in Tab. 5.1. Most categories of good performance are from the nature, while some manufacture categories still have poor performance. It is mainly because that the manufacture categories can have large aspect ratio and usually have much occlusion. The category with the highest performance is the butterfly with a AP of 92.7%, which is already better than the well-explored pedestrian detection on INRIA [8] (88.2%) and approaching that of face detection on AFW [61] (93.7%).

top 24	butterfly	92.7	rabbit	83.9	frog	80.4	fox	75.9	snowmobile	73.9	elephant	72.8	tiger	70.4	tennis ball	68.2
	volleyball	86.2	basketball	82.8	bear	78.5	skunk	75.3	scorpion	73.5	iPod	71.4	armadillo	70.0	harp	67.6
	dog	85.9	bird	82.1	snowplow	77.7	zebra	74.3	turtle	73.0	red panda	70.7	antelope	68.3	whale	67.2
bottom 24	head cabbage	23.9	swimming trunks	21.3	ruler	20.9	purse	18.1	stove	16.9	lamp	14.1	microphone	12.9	horizontal bar	11.3
	bookshelf	23.7	diaper	21.2	bench	20.1	pencil box	18.0	plastic bag	14.9	ski	14.0	nail	12.5	ladle	9.3
	miniskirt	23.3	flute	21.2	screwdriver	19.5	water bottle	18.0	binder	14.5	eraser	12.9	spatula	11.8	backpack	6.8

Table 3. The average precision of top and bottom 24 categories by the superpixel labeling method in ILSVRC2014 testing set.

	plane	bicycle	bird	board	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mean
SS-BOW [50]	43.5	46.5	10.4	12.0	9.3	49.4	53.7	39.4	12.5	36.9	42.2	26.4	47.0	52.4	23.5	12.1	29.9	36.3	42.2	48.8	33.8
DPM v5 [15]	33.2	60.3	10.2	16.1	27.3	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5	33.7
RegionLet [54]	54.2	52.0	20.3	24.0	20.1	55.5	68.7	42.6	19.2	44.2	49.1	26.6	57.0	54.5	43.4	16.4	36.6	37.7	59.4	52.3	41.7
RCNN [20]	68.1	72.8	56.8	43.0	36.8	66.3	74.2	67.6	34.4	63.5	54.5	61.2	69.1	68.6	58.7	33.4	62.9	51.1	62.5	64.8	58.5
RCNN-gt	68.8	73.6	55.6	50.1	51.7	71.1	77.0	61.3	38.5	60.4	48.5	58.9	69.0	69.2	69.2	39.0	60.0	49.6	61.2	67.2	60.0
Proposed Method	71.8	70.3	58.1	46.2	39.8	70.2	75.2	71.9	38.3	69.0	56.7	66.9	73.5	71.8	59.0	31.9	67.3	56.0	64.3	69.6	61.4

Table 4. Average-Precision of different methods on 20 categories of Pascal VOC 2007 testset.

5.2. Diagnosis Experiments

Our current system except the superpixel labeling is based on the framework of RCNN. In this part, besides the proposed superpixel labeling detection algorithm, we also expose the details which significantly improve the baseline RCNN implementation [20].

The baseline RCNN implementation[3] uses a CNN with AlexNet [28] which is trained on imageNet classification data and fine-tuned on detection data. We independently find that the depth of CNN plays a key role to the final performance, which is in consistent with [46, 45] for classification and [46, 56] for detection. In our experiment, directly deepening the 7 layer AlexNet to 12 layer model can get a 5.2% mAP gain. Further improvement comes from better model tuning, including larger mini-batch sizes and more iterations. A cascade, which prunes many easy background proposals, enables the classifier to focus on the most difficult and has a 0.9% improvement. It also helps to accelerate the training and inference procedure. This observation is in consistent with the bounding box rejection in [39]. When multiple models are combined, a 0.6% performance gain is obtained. Further performance gain comes from the image level context. We find that simply weighted sum the image classification score and the detection score could reliably improve the performance. The proposed superpixel labeling based detection method finally brings a 3.5% improvement, which enables our system to perform better than the GoogLeNet. By accumulating these techniques, we get about 50% relative performance gain over the baseline.

Due to the limitation in time and machine (and a highly optimized code), we only have four CNNs for model ensemble, but we find that they are enough to achieve the leading performance. Empirically, better classification CNN (which is used for fine-tuning), more fine-tuning iterations and more model ensemble lead to better detection results. Currently, the CNNs used for initialization get the 13% accuracy of top 5 classification accuracy on classification data with single center test, while the GoogLeNet is about 10%. Directly changing the CNN used in this paper to GoogLeNet could further improve the detection performance[4]. We plan to release these models.

5.3. Experiments on Pascal VOC

We finally evaluate our method on Pascal VOC 2007 [13], which is a widely used benchmark for object detection. We use the “comp4” protocol since that the CNN trained on additional ImageNet classification data is used to initialize the CNN. To fairly compare our method with the RCNN baseline, we use exactly the same CNN feature extractor and the same object proposals, as in [20]. We also add the result of “RCNN-gt”, where the ground truth bounding boxes are added to the proposals and can be taken as an upper bound of the RCNN. The standard DPM, selective search proposal with bag-of-words classifier and RegionLet are used for comparison. The results are listed in Tab. 4.

All the methods except the DPM in Tab. 4 use the selective search for proposal generation. The performance increases with better classification, from BOW, RegionLet to CNN. An interesting observation is that when the ground truth bounding boxes are added, the performance only has a 1.5% improvement. It indicates that the proposals with small overlaps, instead of the missed objects, mostly harm the performance. Our method can reduce the influence by exploring the global image information to more clearly infer the overlapped objects and reduce the influence of localization problem. It is even better than the RCNN with ground truth by 1.4%. Similar to the observations on ImageNet, the superpixel labeling algorithm has a 3% improvement compared to the RCNN when using the same CNN feature.

The speed of our system depends on the algorithm used to initialize scores of superpixels. In our current implementation, we use the RCNN framework with new CNN feature extractor based on the open source software Caffe [25]. It

[3] publicly available in https://github.com/rbgirshick/rcnn/tree/ilsvrc

[4] In preparing the camera ready version, we find that by adding a GoogLeNet, the mAP on val2 improves to be 48.0%.

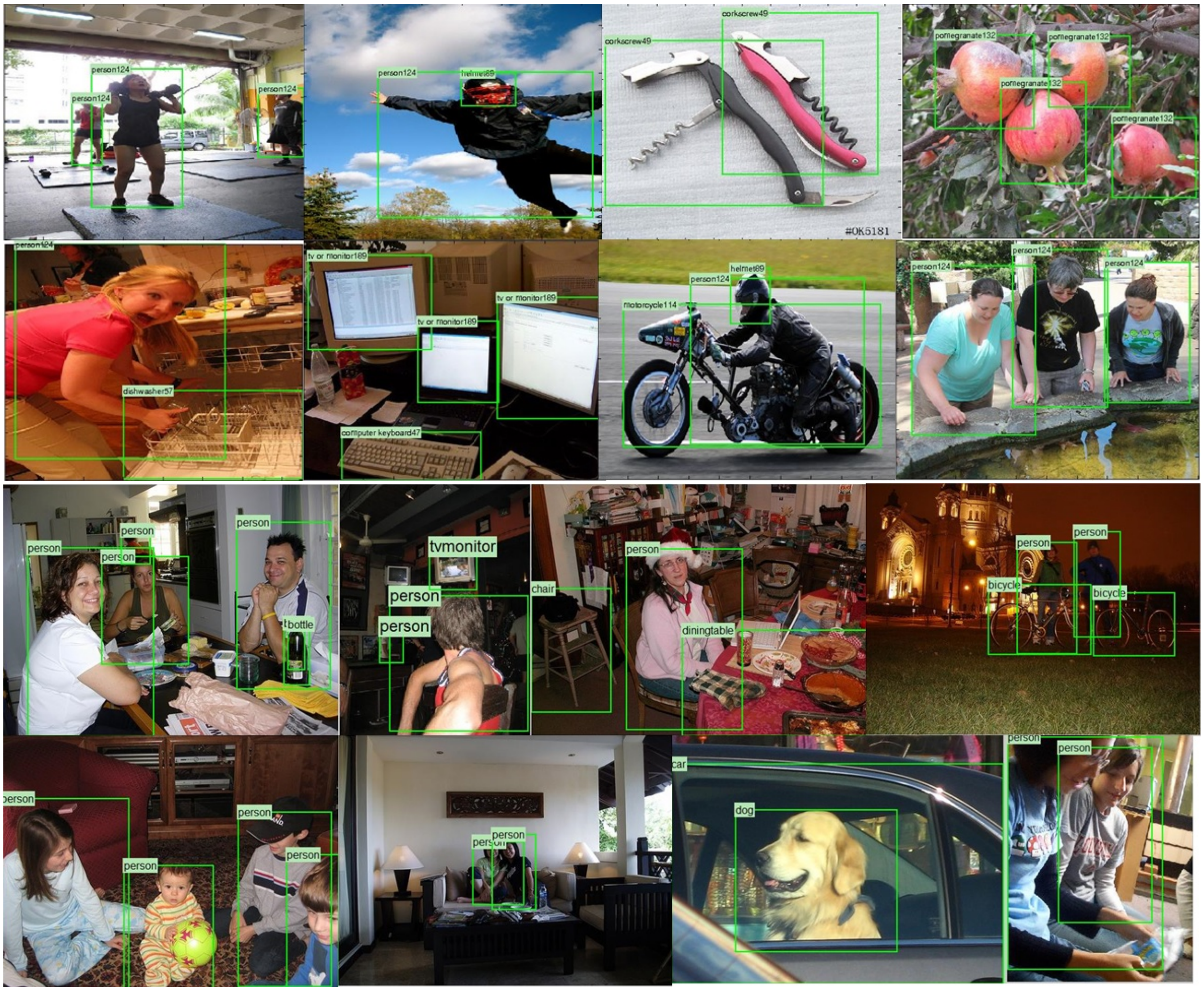


Figure 4. Qualitative results of Superpixel Labeling based object detection on ImageNet and Pascal VOC (best viewed in color).

runs at 1fps for each 128 object proposals on a NVIDIA Telsa K40 GPU. We note that it can be significantly accelerated by the spatial pyramid pooling method proposed in [27]. Benefiting from the efficient α-expansion based graph cut implementation in [9], the superpixel labeling procedure is very efficient. The qualitative detection result of the proposed superpixel labeling method on ImageNet and Pascal VOC are shown in Fig. 4.

6. Conclusion

This paper proposes to handle object detection by labeling superpixels. Compared with the traditional proposal generation and classification based methods, the superpixel based method has a much larger output space and provides more flexibility. It can alleviate the problems in proposal based method. For example, it can infer overlapped objects by encoding global image information. Current leading methods, such as RCNN with very deep CNN, can be incorporated into the superpixel labeling by providing a strong data cost term. The CNN used in RCNN and the parameters in the energy function are learned sequentially, and we plan to jointly learn them for further performance gain. Our work can also give a rough mask and can be extended to semantic segmentation, which is taken as a future work. We believe our approach can also be used for other applications, such as detection based visual tracking[53].

Acknowledgement

This work was supported by the Chinese National Natural Science Foundation Projects #61203267, #61375037, #61473291, National Science and Technology Support Program Project #2013BAK02B01, Chinese Academy of Sciences Project No. KGZD-EW-102-2. Junjie Yan thanks for a fellowship from Baidu, a GPU donation from Nvidia and many insights from Dr. Chang Huang and Naiyan Wang.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *PAMI*, 2012. 2

[2] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. CVPR, 2014. 2

[3] O. Barinova, V. Lempitsky, and P. Kholi. On detection of multiple object instances using hough transforms. *PAMI*, 2012. 3

[4] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, 2011. 1, 2

[5] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *ECCV*. Springer, 2014. 2

[6] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *IEEE CVPR*, 2014. 2

[7] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 1995. 2

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*. IEEE, 2005. 1, 2, 6

[9] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *IJCV*, 2012. 2, 5, 8

[10] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. *IJCV*, 2011. 3

[11] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *PAMI*, 34, 2012. 1

[12] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*. IEEE, 2014. 3

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 1, 3, 5, 7

[14] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *PAMI*, 2013. 3

[15] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 2010. 1, 2, 3, 7

[16] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 2, 3

[17] J. Friedman, T. Hastie, R. Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 2000. 2

[18] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *CVPR*. IEEE, 2009. 3

[19] G. Ghiasi, Y. Yang, D. Ramanan, and C. C. Fowlkes. Parsing occluded people. In *CVPR*, 2014. 3

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1, 2, 3, 4, 5, 6, 7

[21] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *arXiv preprint arXiv:1409.5403*, 2014. 2

[22] S. Gould, T. Gao, and D. Koller. Region-based segmentation and object detection. In *NIPS*, 2009. 3

[23] C. Gu, J. J. Lim, P. Arbeláez, and J. Malik. Recognition using regions. In *CVPR*. IEEE, 2009. 2

[24] J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? In *BMVC*, 2014. 2

[25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 7

[26] T. Joachims, T. Finley, and C.-N. Yu. Cutting-plane training of structural svms. *Machine Learning*, 2009. 2, 6

[27] H. Kaiming, Z. Xiangyu, R. Shaoqing, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014. 6, 8

[28] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 7

[29] D. Kuettel, M. Guillaumin, and V. Ferrari. Segmentation propagation in imagenet. In *ECCV*. Springer, 2012. 3

[30] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*. Springer, 2010. 3

[31] L. Ladickỳ, P. Sturgess, K. Alahari, C. Russell, and P. H. Torr. What, where and how many? combining object detectors and crfs. In *ECCV*. Springer, 2010. 3

[32] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 2008. 3

[33] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013. 6

[34] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. *ECCV*, 2014. 3

[35] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 1

[36] S. Manen, M. Guillaumin, and L. V. Gool. Prime object proposals with randomized prim's algorithm. In *ICCV*. IEEE, 2013. 2

[37] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *ECCV*. Springer, 2014. 2

[38] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *CVPR*. IEEE, 1997. 1, 2

[39] W. Ouyang, P. Luo, X. Zeng, S. Qiu, Y. Tian, H. Li, S. Yang, Z. Wang, Y. Xiong, C. Qian, et al. Deepid-net: multi-stage and deformable deep convolutional neural networks for object detection. *arXiv preprint arXiv:1409.3505*, 2014. 6, 7

[40] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*. Springer, 2010. 1, 2

[41] R. Rothe, M. Guillaumin, and L. Van Gool. Non-maximum suppression for object detection by passing messages between windows. In *ACCV*. Springer, 2014. 3

[42] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *PAMI*, 1998. 1

[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge, 2014. 1, 3, 6

[44] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 6

[45] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*, 2014. 1, 2, 6, 7

[47] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *NIPS*, 2013. 3

[48] S. Tang, M. Andriluka, and B. Schiele. Detection and tracking of occluded people. *IJCV*, 2014. 3

[49] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. In *JMLR*, 2005. 2

[50] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 2013. 2, 3, 5, 6, 7

[51] A. Vedaldi. A MATLAB wrapper of SVM$^{\text{struct}}$, 2011. 6

[52] P. Viola and M. Jones. Robust real-time face detection. *IJCV*, 2004. 1, 2

[53] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung. Transferring rich feature hierarchies for robust visual tracking. *arXiv preprint arXiv:1501.04587*, 2015. 8

[54] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *ICCV*. IEEE, 2013. 6, 7

[55] J. H. H. Woonhyun Nam, Piotr Dollár. Local decorrelation for improved pedestrian detection. In *NIPS*, 2014. 2

[56] J. Yan, N. Wang, Y. Yu, S. Li, and D.-Y. Yeung. Deeper vision and deep insight solutions. In *ECCV workshop on ILSVRC2014*, 2014. 6, 7

[57] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li. Robust multi-resolution pedestrian detection in traffic scenes. In *CVPR*. IEEE, 2013. 2

[58] Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes. Layered object detection for multi-class segmentation. In *CVPR*. IEEE, 2010. 3

[59] J. Yao, S. Fidler, and R. Urtasun. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *CVPR*. IEEE, 2012. 3

[60] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*. ACM, 2009. 2

[61] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*. IEEE, 2012. 6

[62] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*. Springer, 2014. 2