Relating Things and Stuff via Object Property Interactions

Min Sun*, Byung-soo Kim*, Pushmeet Kohli, Silvio Savarese

Abstract—In the last few years, substantially different approaches have been adopted for segmenting and detecting "things" (object categories that have a well defined shape such as people and cars) and "stuff" (object categories which have an amorphous spatial extent such as grass and sky). While things have been typically detected by sliding window or Hough transform based methods, detection of stuff is generally formulated as a pixel or segment-wise classification problem. This paper proposes a framework for scene understanding that models both things and stuff using a common representation while preserving their distinct nature by using a property list. This representation allows us to enforce sophisticated geometric and semantic relationships between thing and stuff categories via property interactions in a single graphical model. We use the latest advances made in the field of discrete optimization to efficiently perform maximum a posteriori (MAP) inference in this model. We evaluate our method on the Stanford dataset by comparing it against state-of-the-art methods for object segmentation and detection. We also show that our method achieves competitive performances on the challenging PASCAL'09 segmentation dataset.

Index Terms—Scene understanding, Semantic Labeling, Segmentation, Graph-cut

1 Introduction

The last decade has seen the development of a number of methods for object detection, segmentation and scene understanding. These methods can be divided into two broad categories: methods that attempt to model and detect object categories that have distinct shape properties such as cars or humans (*things*), and methods that seek to model and identify object categories whose internal structure and spatial support are more heterogeneous such as grass or sky (*stuff*). In the first category, we find that methods based on pictorial structures (i.e., Felzenszwalb et al. [1]), pyramid structures (i.e., Grauman and Darrell [2]), generalized Hough transform [3]–[7], or multi-view model [8], [9] work best. These representations are appropriate for capturing shape or structural properties of *things*, and typically identify the object by a bounding box. The second category of methods aims at segmenting the image into semantically consistent regions [10]–[12] and work well for *stuff*, like sky or road.

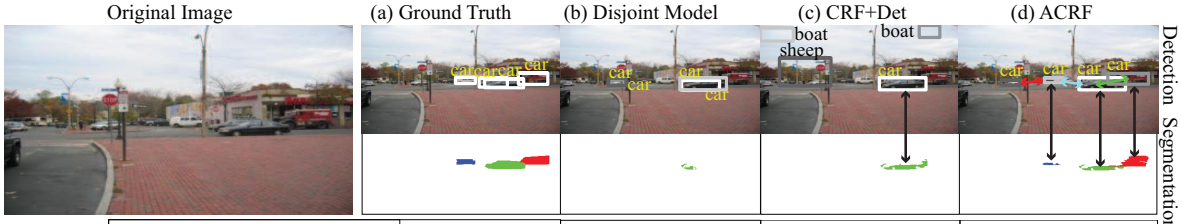
In order to coherently interpret the depicted scene, various types of contextual relationships among objects (stuff or things) have been explored. For example, co-occurrence relationships (e.g., cow and grass typically occur in the same image) have been captured in [13], [14], 2D geometric relationships (e.g., below, next-to, etc) have been utilized in [15]–[17], 2.5D geometry relationships (e.g., horizon line) have been incorporated by Hoiem et al. [18] and Bao et al. [19]. The use of such contextual relationships has inspired the development of robust algorithms for various object recognition tasks. For instance, many segmentation methods [13], [20], [21] have been proposed to capture stuff-stuff relationships in a random field formulation. Similarly, thing-thing relationships have been incorporated into a random field for jointly detecting multiple objects (Desai et al. [16]).

Recently, researchers have proposed methods to jointly detect *things* and segment *stuff*. Gould et al. [22] proposed a random field model incorporating both stuff-stuff, thing-stuff, and thing-horizon relationships. One limitation of their approach is that it cannot capture 2D geometric and co-occurrence relationships between things. Moreover, inference is computationally very demanding and typically takes around five minutes per image. To overcome this limitation, some authors have proposed inference procedures which leverage existing approaches for detection and segmentation and use the output of such approaches as input features in an iterative fashion [23]–[26]. Unfortunately, convergence is not guaranteed for most of these approaches.

We propose a novel framework for jointly detecting things and segmenting stuff that can coherently capture many known types of contextual relationships. Our contributions are three-fold. First, the model infers the geometric and semantic relationships describing the objects (i.e., object x is behind object y) via object property interactions. Second, the model enables instance base segmentation (see color coded segments in Fig. 1(d)) by associating segments of thing categories to instance-specific labels. Finally, the special design of model potentials allows us to utilize

- *M. Sun* is with University of Washington. E-mail: sunmin@cs.washington.edu,*
- *B. Kim* is with University of Michigan. E-mail: bsookim@umich.edu,*
- *P. Kohli is at Microsoft Research, Cambridge E-mail: pkohli@microsoft.com,*
- *S. Savarese is with Stanford University. E-mail: ssilvio@stanford.edu*
- ** indicates equal contributions,*

Digital Object Indentifier 10.1109/TPAMI.2013.193



Thing-Stuff (TS) Relationships	TS ↔	NA	✓	✓
Thing-Thing (TT) Relationships	TT Same plane ⟺ TT Next-to ↔ TT Behind ⟺	NA	NA	✓

Fig. 1: Our goal is to segment the image into things (e.g., cars, humans, etc.) and stuff (e.g., road, sky, etc.) by combining segmentation (bottom) with object detection (top). Results from different variants of our method (capturing a subset of critical contextual relationships) are shown from left to right columns. At the top of each column, we show the top 4 probable bounding boxes, where light and dark boxes denote the confidence ranking from high to low. Instance-based segmentation are shown in each bottom column, where different colors represent different object instances, and all stuff categories are labelled with white color to make the figure less cluttered. The CRF+Det model enforces the detections to be consistent with the segment labels. While the green car segments are reinforced by the strong (white) car detection, very weak (dark) false detections are also introduced due to noisy segment labels in the background region. On the other hand, our final ACRF captures the key relationships so that it recovers many missing detections and segmentation labels. Thing-Stuff and Thing-Thing relationships are indicated by color-coded arrows connecting bounding boxes and/or segments. Different color codes indicate different types of relationships.

a combination of state-of-the-art discrete optimization techniques to achieve efficient inference which takes a few seconds per image in average using a single core.

Object hypotheses and property lists. Our framework extends the basic conditional random field (CRF) formulations for scene segmentation (i.e., assigning an object category to each segmentation variable X) [11], [27] by introducing the concept of a "generic object hypothesis" described by a *property list* (Fig. 2-Top). Instead of only capturing the semantic property of the hypothesis using an object categorical label l in the basic conditional random field, we allow the list to include geometric properties, such as the 2D location (u, v), the distance from the camera (depth) d, and the set of segments $\mathcal{V}$ associated with the object hypothesis. Notice that the generic object hypothesis naturally describes the existence of an object instance with respect to the camera. Hence, our scene segmentation framework can not only segment a scene into different object categories but also into object instances with different properties (i.e., things) as shown in Fig. 1(d). Thus, it generalizes the work of Barinova et al. [6], [28].

We augment the above-mentioned CRF formulation with object hypothesis indicator (binary) variables which capture the presence or absence of object hypotheses (see the solid (on state) and dash (off state) nodes in Fig. 2(a)-Top). We refer to our model as the augmented CRF, or ACRF, to highlight the newly added object hypothesis indicator variables. Two additional relationships are captured in our ACRF. Firstly, the state of the indicator variable needs to be consistent with the assignment of the segmentation variables associated with the corresponding object hypothesis. We introduce a novel higher-order potential function to penalize, for the first time, both types of inconsistency: i) the indicator is off but many segments in set $\mathcal{V}$ are still assigned to the corresponding hypothesis; ii) the indicator is on but only a few segments in set $\mathcal{V}$ are assigned to the corresponding hypothesis. Secondly, the object indicator variables allow us to easily encode sophisticated semantic and geometric relationships between pairs of object hypotheses. For instance, simple pairwise potentials defined over object indicator variables can allow to incorporate i) 2D geometric relationships such as "above" which models the property that one hypothesis lies above the other (e.g., a person sitting on a bike), ii) 2.5D depth-ordering and occlusion relationships such as "in-front" which models the property that one hypothesis lies in front of the other (e.g., a person standing in front of a car). More sophisticated relationships such as a composition of these basic 2D or 2.5D relationships can also be supported. Critically, the ACRF model generalizes Ladicky et al.'s model [13] capturing stuff-stuff co-occurrence contextual relationships only. In contrast, our model can encode relationships between generic object hypotheses (both things and stuff) depending on their semantic and geometrical properties. We illustrate the efficacy of our approach in Fig. 1. As seen in the figure, detections typically do not agree with the segmentation results (Fig. 1(b)) if the detection and segmentation are applied separately. A model capturing relationships among object hypothesis and segments ensures consistency between detection and segmentation results (Fig. 1(c)). However, the relationships between object hypotheses are ignored. Hence, false object hypotheses sometime are introduced. Finally, when pairwise relationships of object hypotheses (e.g., next-to, behind, etc) are included, even small object instances, that are hard to detect and segment, can be discovered (Fig. 1(d)).

Learning. Given the property list, a pre-defined set of pair-wise relationships of object hypotheses are encoded in our model via property interactions as described in Sec. 3.3. The likelihood of the relationships are treated as model parameters that are learned from

training data. For example, the model should learn that a person is likely to sit on a motorbike, and cow and airplane are unlikely to co-occur. In our model, a likely relationship will add a negative cost to the energy of the model. On the other hand, an unlikely relationship will add a positive cost. We formulate the problem of learning these costs jointly with other parameters as a Structured SVM (SSVM) [29] learning problem with two types of loss functions related to the segmentation loss and detection loss, respectively (see Sec. 5 for details).

MAP Inference. Jointly estimating the segmentation variables X and indicator variables Y (See nodes in Fig. 2(c)) is challenging due to the intrinsic difference of the indicator and segmentation variable space, and newly added complex relationships between them (See edges in Fig. 2(c,d)). We design an efficient graph-cut-based move making algorithm by combining state-of-the-art discrete optimization techniques. Our method is based on the α-expansion move making approach [30], which works by projecting the energy minimization problem of segmentation variables X into a binary energy minimization problem to have the same space as the indicator variables Y. We use the "probing" approach similar to the one introduced by Rother et al. [31] to handle the non-submodular function describing pair-wise relationships of object hypotheses. Our MAP inference algorithm takes only a few seconds per image in average using a single core as opposed to five minutes by Gould et al. [22].

Outline of the Paper. The rest of the paper is organized as follows. We first describe the related work in Sec. 2. Model representation, inference, and learning are elaborated in Sec. 3, 4, and 5, respectively. Implementation details and experimental results are given in Sec. 6.

2 Related Work

Our method is closely related to the following three methods which all can be considered as special cases of our model. Desai et al. [16] propose a CRF model capturing thing-thing relationships and show that object detection performance can be consistently improved for multiple object categories. Their model can be considered as a special case of our model when no segmentation variable X exists.

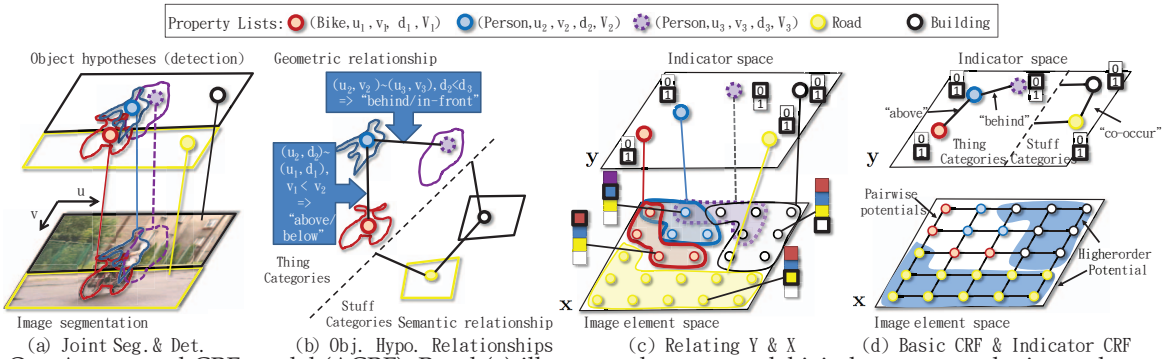
Both Ladicky et al.'s methods [13], [32] extend the basic CRF model to incorporate more sophisticated relationships. Ladicky et al. [32] incorporate things-stuff relationships and demonstrate that the information from object detection can be used to improve the segmentation performance consistently across all object categories. Their model can be considered as a special case of our model when no thing-thing relationship is incorporated. One more subtle difference is that their model only weekly enforces the consistency between things and stuff. Their model does not penalize the case when the indicator is off but many segments are still assigned to the corresponding hypothesis. Notice that the strong consistency of things-stuff in our model is crucial in combination with thing-thing relationships. Otherwise, many segments will still be assigned to the corresponding hypothesis even when the hypothesis is suppressed by thing-thing relationships. Ladicky et al. [13] propose to capture co-occurrence types of object relationships and demonstrate that the co-occurrence information can be used to improve the segmentation performance significantly. Their model can also be considered as a special case of our model when no geometric relationships of object hypotheses (i.e., above, same horizon, etc) are established. Finally, [13] cannot be used to assign segments to object instances or localize object instances. Our results on the Stanford dataset demonstrate that our model achieves superior performances than [13], [32].

Our method shares similar ideas with Yao et al. [33] to jointly model object detection and scene segmentation. However, there are two main differences. Yao et al. also jointly model scene classification problem and use the scene class to restrict the possible object categories existing is an image. Therefore, they need to apply a set of pre-trained scene classifiers in addition to the object detectors and scene segmentation methods. Moreover, their model does not incorporate thing-thing geometric relationships. Their model ignores facts such as cars are typically "next-to" each other and cups are "on-top" a table. As a result, their method works well on datasets which contains only few object instances (typically less than 3) such as the MSRC dataset [27]. On the contrary, our experimental results show that our method works well on datasets containing many more object instances such as Stanford dataset [21].

The Semantic Structure From Motion (SSFM) proposed by Bao et al. [34] also jointly models object instances and regions. However, unlike our method which utilizes one single image, their approach utilizes the correspondences of object instances and regions established across multiple images to improve object detection and segmentation performances.

Li et al. [35], [36] proposes generative models to jointly classify the scene, recognize the class of each segment, and/or annotate the images with a list of tags. However, the model cannot localize each object instances. Hence, the thing-thing and thing-stuff interactions are not incorporated in their model.

Many methods explore contextual relationships between segments and/or object hypotheses to improve a specific visual task such as detection, category discovery, etc. For instance, [37] use contextual relationships to discover object categories commonly appearing within a set of images. It was demonstrated in [38], [39] that the contextual relationships can be used to improve object detection performance.



(a) Joint Seg. & Det. (b) Obj. Hypo. Relationships (c) Relating Y & X (d) Basic CRF & Indicator CRF

Fig. 2: Our Augmented CRF model (ACRF). Panel (a) illustrates that our model jointly segments the image by assigning labels to segments (bottom layer) and detects object by determining which object hypotheses exist (top layer). The existence of the hypotheses are indicated by solid (on) and dash (off) nodes. As shown on the top panel, each "thing" object hypothesis possesses properties such as category, location (u, v), etc. On the other hand, each "stuff" object hypothesis possesses only the category label. In panel (b), we give examples of relationships established via property interactions. For "thing" categories, geometric relationships such as behind and above can be established. On the other hand, for "stuff" categories, semantic relationship such as co-occurrence can be established. Notice that the two edges, which connect to the stuff categories and end in the dashed separator line, represent the co-occurrence relationships between stuff and all thing categories. In panel (c), the figure shows the label space of the segmentation variables $\mathbf{X}$ (color-coded blocks) and the indicator variables $\mathbf{Y}$ (0 and 1 blocks). The higher-order potential capturing the relationship between indicator i and a set of segmentation variable in $\mathcal{V}_i$ is represented by an edge between a node on top layer and a set of nodes at the bottom layer. In panel (d)-Bottom, the figure shows the pairwise and higher order potential among segmentation variables $\mathbf{X}$ which are presented in the basic CRF formulation. In panel (d)-Top, the figure shows the pairwise potential between pairs of indicator variables $\mathbf{Y}$ which encodes different geometric and semantic relationships.

3 Augmented CRF

We now explain our Augmented Conditional Random Field (ACRF) model. ACRF jointly models object detection and scene segmentation (Fig. 2 (a)) by incorporating contextual relationships between things and stuff, and between multiple things (Fig. 2 (b)).

Basic CRF. Similar to other scene segmentation methods, our model is developed on top of a basic Conditional Random Fields (CRF) model. The basic CRF model is defined over a set of random variables $X = \{x_i\},\ i \in \mathcal{V}$ where $\mathcal{V}$ represents the set of image elements, which could be pixels, patches, segments, etc (Fig. 2 (c)-Bottom). Each random variable x_i is assigned to a label from a discrete label space $\mathbf{L}$, which is typically the set $\mathcal{L}$ of object categories such as grass, road, car and people.

The energy (or cost) function $E(X)$ of the CRF is the negative logarithm of the joint posterior distribution of the model and has the following form:

$$\begin{aligned} E(X) &= -\log P(X|\mathcal{E}) = -\log \phi_{eRF}(X|\mathcal{E}) + \mathrm{K} \\ &= \sum_{c \in \mathcal{C}^X} \psi_c(X_c) + \mathrm{K}\ , \end{aligned} \tag{1}$$

where $\mathcal{E}$ is the given evidence from the image and any additional information (e.g., object property lists), $\phi_{eRF}(X|\mathcal{E})$ takes the form of a CRF model with higher order potentials defined over image elements (Fig. 2 (d)-Bottom). $\phi_{eRF}(X|\mathcal{E})$ can be decomposed into potential ψ_c which is a cost function defined over a set of element variables X_c (called a clique) indexed by $c \in \mathcal{C}^X$; $\mathcal{C}^X$ is the set of cliques for image elements, and K is a constant related to the partition function. The problem of finding the most probable or maximum a posteriori (MAP) assignment of the CRF model is equivalent to solving the following discrete optimization problem: $X^* = \arg\min_{X \in \mathcal{L}^{|\mathcal{V}|}} E(X)$, where $|\mathcal{V}|$ indicates the number of elements.

The basic CRF model mostly relies on bottom-up information. It is constructed using unary potentials based on local classifiers and smoothness potentials defined over pairs of neighboring pixels. Higher-order potentials (such as the ones used in [11]) encourage labels of image elements within a group to be the same. This classic representation for object segmentation has led to excellent results for the stuff object categories, but has failed to replicate the same level of performance on the thing object categories. The reason for this dichotomy lies in the model's inability to explicitly encode the relationship between the shape and relative positions of different parts of structured object categories such as the head and the torso of a person.

In contrast, part-based models such as Pictorial Structures [40], Latent SVM (LSVM) [1], and Hough transform based models [3], [6] have shown to be much more effective at detecting "things" by generating a list of object hypotheses ordered according to their scores/likelihoods. Each hypothesis is often characterized by a property list including the category of the object l, the spatial location in the image (u, v), the depth or distance d of the object instance from the camera, and the set of segments $\mathcal{V}$ associated with the object hypothesis (Fig. 2 Top panel). In many application, a detection problem can be relaxed into an image-level classification problem. A classification method generates a hypothesis of the existence of an object category in the image without specifying the spatial configuration of the object. Since the spatial configuration of the object does not need to be spec-

ified, hypotheses for both "things" and "stuff" can be generated. Notice that, in this case, the property of such hypothesis only includes the category of the object l.

Augmented CRF. In order to take advantage of both the object detection and segmentation methods, we introduce a set of indicator variables (later referred to as indicators) $Y = \{y_j \in \{0,1\}; j \in \hat{\mathcal{Q}}\}$ corresponding to every object hypothesis in our ACRF model (Fig. 2 (c)-Top). Theoretically, the number of all possible object hypotheses $|\hat{\mathcal{Q}}|$ is large, since it is the Cartesian product of the space of all possible object category labels $\mathcal{L}$, all possible spatial locations $\mathbf{U} \times \mathbf{V}$ in the image, and all depth or distance values within a range $[0, D]$. For example, a sliding window detector exploring 10 scales/depths considers $369K$ hypotheses in total for a 320×240 image. Therefore, it is potentially hard to handle. Fortunately, in real world images, only a few hypotheses are actually present. Thus, most indicator variables $y_j,\ j \in \hat{\mathcal{Q}}$ are off (i.e., $y_j = 0$). We use object detectors that have been trained on achieving a high recall rate to generate a relatively small set of *plausible* object hypotheses $\mathcal{Q}_d$ (about 20 per class on Stanford dataset [1]) compared to the size of all possible object hypotheses $\hat{\mathcal{Q}}$. In addition, a set of object hypotheses $\mathcal{Q}_c$ with only object category label, similar to the ones generated by image-level classification methods, are included. As a result, we obtain the set of object hypotheses $\mathcal{Q} = \mathcal{Q}_d \cup \mathcal{Q}_c$.

Recall that variables X representing the image elements in the basic CRF formulation for object segmentation typically take values from the set of object categories $\mathcal{L}$. In contrast, in our framework, these variables take values from the set of plausible object hypotheses $x_i \in \mathbf{L} = \mathcal{Q}$ (refer as *augmented labeling space*). This allows us to obtain segmentations of instances of particular object categories which the basic CRF formulation is unable to handle.

The joint posterior distribution of the segmentation variables X and indicator variables Y can be written as:

$$P(X, Y|\mathcal{E}) \propto \phi_{eRF}(X|\mathcal{E})\ \phi_{oRF}(Y|\mathcal{E})\ \phi_{con}(X, Y|\mathcal{E})\ . \quad (2)$$

The function ϕ_{oRF} takes the form of a CRF model defined over object indicator variables as follows:

$$\phi_{oRF}(Y|\mathcal{E}) = \prod_{c \in \mathcal{C}^Y} e^{\varphi_c(Y_c)}\ , \quad (3)$$

where the potential $\varphi_c(Y_c)$ is a cost function defined over a set of indicator variables Y_c indexed by $c \in \mathcal{C}^Y$, and $\mathcal{C}^Y$ is the set of cliques of indicators. The potential function ϕ_{con} enforces that the segmentation and indicator variables take values which are consistent with each other (Fig. 2 (c)). The term is formally defined as:

$$\phi_{con}(X, Y|\mathcal{E}) = \prod_{j \in \mathcal{Q}} e^{\Phi(y_j, X_j)}\ , \quad (4)$$

where $\Phi(y_j, X_j)$ is the potential relating each indicator y_j with a set of image elements $X_j = \{x_i; i \in \mathcal{V}_j\}$ corresponding to the set of segments $\mathcal{V}_j$ of the j_{th} hypothesis. Hence, the model energy can be written as:

$$\begin{aligned} E(X, Y) &= \sum_{c \in \mathcal{C}^X} \psi_c(X_c) + \sum_{j \in \mathcal{Q}} \Phi(y_j, X_j) \\ &+ \sum_{c \in \mathcal{C}^Y} \varphi_c(Y_c)\ . \end{aligned} \quad (5)$$

The first term of the energy function is defined in a manner similar to [11]. We describe other terms of the energy function in detail in the following subsections.

3.1 Relating Y and X

The function $\Phi(y_j, X_j)$ (Fig. 2(c)) is a likelihood term that enforces consistency in the assignments of the j_{th} indicator variable y_j and a set of segmentation variables X_j. It is formally defined as:

$$\Phi(y_j, X_j) = \begin{cases} \inf & \text{if } y_j \neq \delta(X_j, l_j) \\ \gamma_{l_j} \cdot |X_j| \geq 0 & \text{if } y_j = \delta(X_j, l_j) = 1 \\ 0 & \text{if } y_j = \delta(X_j, l_j) = 0 \end{cases}\ , \quad (6)$$

where j is the index of an object hypothesis in $\mathcal{Q}$, the function $\delta(X_j, l_j)$ indicates whether the majority of X_j have object category label l_j, and is defined as:

$$\delta(X_j, l_j) = \begin{cases} 1 & \text{if } R(X_j, l_j) = \frac{|X_j(l_j)|}{|X_j|} \geq \rho(l_j) \\ 0 & \text{otherwise} \end{cases}\ , \quad (7)$$

where $X_j(l_j) = \{x_i = l_j; i \in \mathcal{V}_j\}$ is the set of image elements in X_j with label l_j, $|X_j|$ is the total number of elements in X_j, $R(X_j, l_j)$ is the consistency percentage, and $\rho(l_j) \in [0\ 1]$ is an object category-specific consistency threshold. Hence, the first condition in Eq. 6 and Eq. 7 ensures that $y_j = 1$ if and only if the j_{th} hypothesis shares an object category with at least $\rho(l_j)$ percent of image elements in X_j (i.e. $R(X_j, l_j) \geq \rho(l_j)$). The remaining conditions in Eq. 6 show that this potential is an Occam razor or MDL prior, similar to [6], [13] so that the model is penalized by $\gamma_{l_j} \cdot |X_j|$ when $y_j = 1$.

3.2 Object indicator CRF

The object indicator CRF potential $\varphi_c(Y_c)$ in Eq. 5 can be decomposed into two terms as follows,

$$\sum_{c \in \mathcal{C}^Y} \varphi_c(Y_c) = \sum_{j \in \mathcal{Q}_d} \varphi_u(y_j) + \sum_{(j,k) \in \mathcal{U}} \varphi_p(y_j, y_k)\ , \quad (8)$$

where $\mathcal{Q}_d \subset \mathcal{Q}$ is the set of hypotheses with geometric properties and $\mathcal{U}$ is the set of pairs of indicators, which interact with each other.

1. We set the pre-trained detector threshold as -0.7.

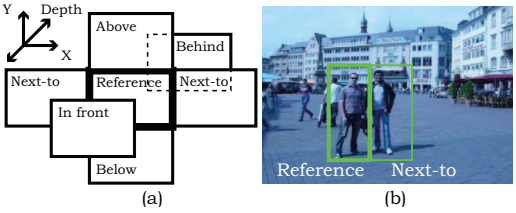


Fig. 3: (a) Pairwise relationships between object hypotheses can be determined by drawing an additional box with respect to a reference box. (b) In this example, a person (right) is "next-to" a person on the left side.

The term $\varphi_u(y_j)$ is the unary potential for an indicator variable with geometric properties. It is defined as:

$$\varphi_u(y_j) = \begin{cases} \beta_j \cdot |X_j| \geq 0, & \texttt{if } \ y_j = 0 \\ 0, & \texttt{if } \ y_j = 1 \end{cases} , \qquad (9)$$

such that the cost of suppressing hypothesis j (i.e., label y_j is off) is $\beta_j \cdot |X_j|$, and β is the calibrated detection confidence so that confidences are comparable across object categories.

The term $\varphi_p(y_j, y_k)$ (black edges in Fig. 2 (d)-Top) represents the interactions between a pair of indicator variables. Depending on the types of properties associated with the pair of indicator variables, this term can represent a number of relationships. It can not only model spatial relationship in 2D such as the ones learned and employed in the approach proposed by [16], but also model behind and in-front relationships given the depth property. The term can also encode co-occurrence relationships [13] for pairs of indicators with only category properties.

For a pair of indicators j and k, the term is formally defined as:

$$\varphi_p(y_j, y_k) = w^{r_{jk}}_{l_j,l_k}(y_j, y_k) \cdot \max(|X_j|, |X_k|) \ , \qquad (10)$$

where r_{jk} is the type of relationship that we want to enforce between the pair of object instances j and k, and is a subset of the overall relationship set $\mathcal{R}$, which is pre-defined as: $\mathcal{R} = \{$*co-occur, above, below, next-to, in-front, behind, overlap, and horizon lines agreement*$\}$.

The pseudo-boolean function

$$w^{r_{jk}}_{l_j,l_k}(y_j, y_k) : \{0,1\}^2 \rightarrow \mathbb{R} \qquad (11)$$

specifies the cost of all 4 possible combination of y_j and y_k under the relationship r_{jk} for a pair of object categories l_j, l_k. As a result, the potential can capture both attractive (i.e., $w(0,0)+w(1,1) \leq w(0,1)+w(1,0)$) and repulsive (i.e., $w(0,0)+w(1,1) \geq w(0,1)+w(1,0)$) interactions. For example, a person usually is sitting on a motorbike (attractive), and cars do not overlap with each other in 3D (repulsive). Notice that although the relationship set $\mathcal{R}$ is pre-defined, the parameters $w(y_j, y_k)$ are learned jointly with other model parameters as described in Sec. 5.

3.3 Pair-wise relationships between object hypotheses

The relationship r_{jk} is specified by the properties associated with the indicators j and k. If the indicators i and j have geometric properties, geometric relationships are determined by the following steps. Given a pair of hypotheses, we firstly set the bounding box of one hypothesis as a reference box. Then, we draw additional boxes with respect to the reference box for a certain spatial relationship (i.e. *above*: draw on top of a box; *next-to*: draw on left or right side of a box, etc. See Fig. 3 for details). If a drawn box overlaps more than 50% with the bounding box of the other hypothesis which is not selected as the reference box, we can specify a relationship to the given pair of boxes. If none of the above relationships is selected and the two original boxes overlaps each other more than 50%, we use the depth property associated to the hypotheses to select either *overlap*, *in-front*, or *behind* relationship. The "horizon lines agreement" relationship is based on whether the predicted two horizon lines from two bounding boxes are in agreement or not. In specific, horizon lines for two boxes are estimated assuming objects' average heights are known, similar to [18]. If two lines are close to each other within a certain range, which is a function of the specific class (i.e. person or car have smaller range, boat have a larger range), they are having a same horizon line. On the other hand, if the indicators i and j have only object category properties, the relationship r_{jk} models the co-occurrence cost of the object categories. In this case, we further assume

$$w^{co}_{j,k}(y_j, y_k) = \begin{cases} \gamma_{l_j,l_k} \geq 0 & \text{if } y_j = y_k = 1 \\ 0 & \text{otherwise} \end{cases} , \qquad (12)$$

where γ_{l_j,l_k} is the co-occurrence cost for object categories l_j and l_k. From the above definition, we can see that our model generalizes both CRF models proposed in [13], [16].

4 Inference

We now show that the MAP inference problem in our ACRF model can be solved by minimizing the energy function using an efficient graph cut based expansion move making algorithm [30].

Standard move making algorithms repeatedly project the energy minimization problem into a smaller subspace in which a sub-problem is efficiently solvable. Solving this sub-problem produces a change to the solution (referred to as a move) which results in a solution having lower or equal energy. The *optimal* move leads to the largest possible decrease in the energy.

The *expansion* move algorithm projects the problem into a Boolean label sub-problem. In an α-expansion move, every segmentation variable X can either retain its current label or transit to the label α. One iteration

of the algorithm involves making moves for all α in $\mathbf{L}$ successively. Under the assumption that the projection of the energy is pairwise and submodular, it can be exactly solved using graph cuts [41], [42]. Since our ACRF model is built on top of the basic CRF proposed in [11], [30], in the following, we derive the graph construction only for energy terms related to indicator variables Y. For the constructions of the basic CRF, please refer to [11], [30].

4.1 Functions of indicator variables Y with only category property.

The energy terms related to the indicator variables, whose only property is a category label, are $\Phi(y_j, X_j)$ in Eq. 6, $\varphi_p(y_j, y_k)$ in Eq. 10, and assumption in Eq. 12. Observing that we can represent the combination of these terms as a function, $\mathcal{F} : \mathbf{L} \rightarrow \mathbb{R}$ as

$$\mathcal{F}(L(Y)) = \min_X \sum_{j \in \mathcal{Q}_c} \Phi(y_j, X_j) + \sum_{(j,k) \in \mathcal{U}_c} \varphi_p(y_j, y_k) \ , \tag{13}$$

where $L(Y) = \{l_j; j \in \mathcal{Q}_c, y_j = 1\}$ is a set of existing object categories (i.e., $y_j = 1$), $\mathcal{Q}_c$ is any subset of the indicator variables, whose only property is a category label, and $\mathcal{U}_c$ is a subset of $\mathcal{U}$ such that $j, k \in \mathcal{Q}_c$. From the definition of the term in section 3.1 and 3.2, we can see that $\mathcal{F}(\{l_j\}) = \gamma_{l_j}|X_j|$. Furthermore,

$$\begin{aligned} \mathcal{F}(\{l_j, l_k\}) &= \mathcal{F}(\{l_j\}) + \gamma_{l_k}|X_k| \\ &\quad + \gamma_{l_j,l_k} \max(|X_j|, |X_k|) \qquad (14) \\ \mathcal{F}(\{l_j, l_k, l_q\}) &= \mathcal{F}(\{l_j, l_k\}) + \gamma_{l_q}|X_q| \\ &\quad + \gamma_{l_j,l_q} \max(|X_j|, |X_q|) \\ &\quad + \gamma_{l_k,l_q} \max(|X_k|, |X_q|) \ . \qquad (15) \end{aligned}$$

This shows that the above function satisfies the properties of the co-occurrence potential:

$$L_1 \subset L_2 \Longrightarrow \mathcal{F}(L_1) \leq \mathcal{F}(L_2) \ , \tag{16}$$

proposed by [13] which allows us to use their graph construction for minimizing this energy function. We estimate γ's from the training data as described in [13].

4.2 Functions of indicator variables Y with geometric properties.

The energy terms related to the indicator variables with geometric properties are $\Phi(y_j, X_j)$ in Eq. 6 and $\varphi_p(y_j, y_k)$ in Eq. 10. Since it is essential to capture both repulsive and attractive pair-wise relationships for object instance hypotheses in $\varphi_p(y_j, y_k)$, it cannot be combined with $\Phi(y_j, X_j)$ in Eq. 6 to form a co-occurrence potential satisfying Eq. 15. Hence, we need to introduce a new graph construction approach to guarantee submodularity. In the following, we show that by approximating $\Phi(y_j, X_j)$, projecting the segmentation variables X using α-expansion, and negating indicator variables Y dynamically in the expansion algorithm, the projected function is guaranteed to be submodular.

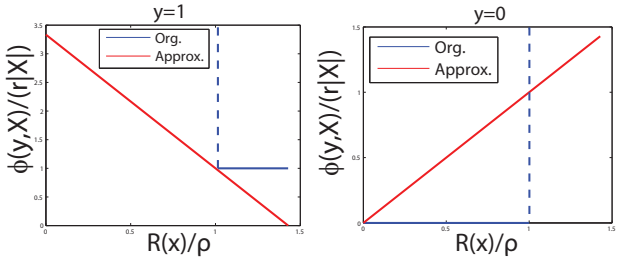


Fig. 4: Comparison between the original function $\Phi(y, X_j)$ (blue lines) and the approximated function (red lines) in Eq. 18 and 17. The left panel shows the case when $y = 1$. The right panel shows the case when $y = 0$. Notice that the dash blue lines indicate the sharp transition from finite values to infinite values.

4.2.1 *Approximating* $\Phi(y_j, X_j)$

We observe in Eq. 6, when $y_j = 1$

$$\begin{aligned} \Phi(y_j, X_j) &= \begin{cases} \inf & \text{if } \delta(X_j, l_j) = 0 \\ \gamma_j|X_j| & \text{if } \delta(X_j, l_j) = 1 \end{cases} \\ &\approx \gamma_j|X_j| \frac{1 - R(X_j, l_j)}{1 - \rho(l_j)} \ . \end{aligned} \tag{17}$$

When $y_j = 0$

$$\Phi(y_j, X_j) = \begin{cases} \inf & \text{if } \delta(X_j, l_j) = 1 \\ 0 & \text{if } \delta(X_j, l_j) = 0 \end{cases} \approx \gamma_j|X_j| \frac{R(X_j, l_j)}{\rho(l_j)} \ . \tag{18}$$

Hence, $\Phi(y_j, X_j)$ becomes,

$$\Phi(y_j, X_j) \approx \gamma_j|X_j| \left(y_j \frac{1 - R(X_j, l_j)}{1 - \rho(l_j)} + (1 - y_j) \frac{R(X_j, l_j)}{\rho(l_j)} \right) \ . \tag{19}$$

The effect of the approximation in Eq. 18 and 17 are shown in Fig. 4. Instead of imposing an infinite cost when $\delta(X, l) \neq y$, our approximation imposes a cost which is linearly proportional to the consistency percentage $R(X, l)$. When $y = 1$, the ratio between the consistency percentage and the consistency threshold $R(X, l)/\rho(l)$ are encouraged to be large, which means that the more elements in X are labeled as l, the better (Fig. 4-Left). In contrast, when $y = 0$, the ratio between the consistency percentage and the consistency threshold $R(X, l)/\rho(l)$ is encouraged to be small, which means that the less elements in X are labeled as l, the better (Fig. 4-Right). Next, we introduce the α-expansion and how to negate the indicator variables Y dynamically in the expansion algorithm so that the projected function is guaranteed to be submodular.

4.2.2 *α-expansion move and dynamical negation algorithm*

We define the transformation function $T_\alpha(x_i; t_i)$ for the α-expansion move which transforms the label of a random variable x_i as:

$$T_\alpha(x_i, t_i) = \begin{cases} \alpha, & \text{if } t_i = 0 \\ x_i, & \text{if } t_i = 1 \end{cases} \tag{20}$$

The corresponding α-expansion move energy for the term in Eq. 19 can be written as: $\Phi(y_j, T_j) =$

$$\begin{cases} \gamma_j|X_j|\{\frac{y_j}{1-\rho(l_j)}\left(1 - R(X_j, l_j) + \sum_{i\in\mathcal{V}_j(l_j)}\frac{(1-t_i)}{|X_j|}\right) \\ +\frac{1-y_j}{\rho(l_j)}\sum_{i\in\mathcal{V}_j(l_j)}\frac{(t_i)}{|X_j|}\}, \text{if } \alpha \neq l_j \\ \gamma_j|X_j|\{\frac{1-y_j}{\rho(l_j)}\left(R(X, l_j) + \sum_{i\in\mathcal{V}_j\setminus\mathcal{V}_j(l_j)}\frac{(1-t_i)}{|X_j|}\right) \\ +\frac{y_j}{1-\rho(l_j)}\sum_{i\in\mathcal{V}_j\setminus\mathcal{V}_j(l_j)}\frac{(t_i)}{|X_j|}\}, \text{if } \alpha = l_j \end{cases} \tag{21}$$

where $T_j = \{t_i; i \in \mathcal{V}_j\}$ is a set of transformed binary variables, $\mathcal{V}_j(l_j) = \{i; x_i = l_j\} \cap \{i \in \mathcal{V}_j\}$ is the set of image elements with label l_j, $\mathcal{V}_j \setminus \mathcal{V}_j(l_j)$ is the remaining set of elements in $\mathcal{V}_j$ with labels other than l_j (i.e.,$\{i; x_i \neq l_j\} \cap \{i \in \mathcal{V}_j\}$). Most importantly, we observed that when $\alpha \neq l_j$ the function is submodular in (y_j, t_i), but when $\alpha = l_j$ it is submodular in $(\overline{y}_j, t_i)$, where $\overline{y}_j = 1 - y_j$ is the negation of y_j. This motivates us to dynamically negate a subset of the indicator variables according to the category labels $\{l_j\}$.

4.2.3 *Probing* $\varphi_p(y_j, y_k)$

The graph construction of the pair-wise instance indicators in Eq. 10 is equivalent to the construction of binary variables which is described in [30]. However, there is one issue that we need to resolve in order to ensure submodularity of the pair-wise binary function. First of all, since it is essential to capture both attractive (i.e., both indicators having the same labels) and repulsive (i.e., both indicators having different labels) relationships, some pair-wise functions $\varphi_p(y_j, y_k)$ will be submodular and others will be non-submodular in (y_j, y_k). Therefore, we need to fix the negating pattern of the indicator variables. However, this contradicts with the dynamic negating approach described in the previous section.

Fortunately, a simple approach "probing" the indicator variables similar to the one described in [31] can effectively handle the non-submodular function, since each indicator only interacts with a small number of nearby indicators. The probing approach randomly fixes a small set of indicator variables $\{y_j; j \in \mathcal{Q}_{fix}\}$, where the contradiction between the fix negating pattern and dynamic negating algorithm takes place. As a result, the pair-wise function is ensured to be submodular. Notice that our inference algorithm does not rely on sophisticated techniques such as QPBO [31] which requires more memory and computation time.

4.3 Overall projected energy function

At each iteration of the α-expansion, the terms of the original model energy (Eq. 5) becomes a pairwise and submodular function of T, Y, and $\overline{Y}$. The overall projected energy function (except the function in Eq. 13) becomes,

$$\begin{aligned} E(T, Y, \overline{Y}) = &\sum_{c\in\mathcal{C}^X}\psi_c(T_c) + \sum_{j\in\mathcal{Q}_\alpha}\Phi(y_j, T) + \sum_{j\in\mathcal{Q}_{\overline{\alpha}}}\Phi(\overline{y}_j, T) \\ &+ \sum_{(j\in\mathcal{Q}_\alpha, k\in\mathcal{Q}_\alpha)\cup(j,k)\in\mathcal{U}_d}\varphi_p(y_j, y_k) \\ &+ \sum_{(j\in\mathcal{Q}_{\overline{\alpha}}, k\in\mathcal{Q}_{\overline{\alpha}})\cup(j,k)\in\mathcal{U}_d}\varphi_p(\overline{y}_j, \overline{y}_k) \\ &+ \sum_{(j\in\mathcal{Q}_\alpha, k\in\mathcal{Q}_{\overline{\alpha}})\cup(j,k)\in\mathcal{U}_d}\varphi_p(y_j, \overline{y}_k)\ . \end{aligned} \tag{22}$$

where $\mathcal{Q}_\alpha = \{j; l_j \neq \alpha\} \setminus \mathcal{Q}_{fix}$, $\mathcal{Q}_{\overline{\alpha}} = \{j; l_j = \alpha\} \setminus \mathcal{Q}_{fix}$, and $\mathcal{U}_d = \mathcal{U} \setminus \mathcal{U}_c$. Therefore, we will construct the graph using T, partially using indicator $Y = \{y_j; j \in \mathcal{Q}_\alpha\}$, and partially using the negation of indicator $Y = \{\overline{y}_j; j \in \mathcal{Q}_\alpha\}$ depending on whether $l_j = \alpha$. Notice that $\mathcal{Q}_{fix}$ is randomly selected at every iteration. Therefore, no indicator variable is always fixed. The α-expansion algorithm eventually converges to a local optimal solution.

5 Learning

The full ACRF model in Eq. 5 contains several terms. In order to balance the importance of different terms, we introduce a set of linear weights for each term as follows,

$$\begin{aligned} W^T\Psi(X, Y) = &\sum_{c\in\mathcal{C}^X} w_c\psi_c(X_c) + \sum_{(j,k)\in\mathcal{U}_d} w_{l_j,l_k}^{r_{jk}}(y_j, y_k) \\ &+ \sum_{j\in\mathcal{Q}_d} w^u(l_j)(\Phi(y_j, X_j) + \varphi_u(y_j)) \\ &+ w^{co}(\sum_{(j,k)\in\mathcal{U}_c}\varphi_p(y_j, y_k) + \sum_{j\in\mathcal{Q}_c}\Phi(y_j, X_j))\ , \end{aligned} \tag{23}$$

where w_c models weights for unary, pair-wise, and higher-order terms in X. $w^u(l)$ is the object category specific weight for unary term in y, w^{co} is the weight for hypothesis with only category label, and $w_{l_j,l_k}^{r_{jk}}$ is the weight for a pair of object categories l_j, l_k with the relationship type r_{jk} in Eq. 10. Recall from Sec. 3.2 and 4 that $\mathcal{Q}_d$ and $\mathcal{Q}_c$ are the set of hypotheses with geometric properties and with only category label, respectively. Similarly, $\mathcal{U}_d$ and $\mathcal{U}_c$ are the subset of $\mathcal{U}$ such that $j, k \in \mathcal{Q}_d$ and $j, k \in \mathcal{Q}_c$, respectively. Notice that the function $\Psi(X, Y, I)$ also depends on the image evidence I. We typically omit the image evidence I for simplicity. Since all these weights are linearly related to the energy function, we formulate the problem of jointly training these weights as a Structured SVM (SSVM) learning problem [29] similar to [16].

Assuming that a set of example images, ground truth segment object category labels, and ground truth object bounding boxes $\{I^n, X^n, Y^n\}_{n=1,\ldots,N}$ are given. The SSVM problem is as follows,

$$\begin{aligned} \min_{W,\xi\geq 0}\quad & W^TW + C\sum_n \xi^n(X, Y) \\ \text{s.t.} \\ \xi^n(X, Y)\quad & = \max_{X,Y}(\triangle(X, Y; X^n, Y^n) \\ & \quad + W^T\Psi(X^n, Y^n, I^n) - W^T\Psi(X, Y, I^n)), \forall n\ , \end{aligned} \tag{24}$$

where W concatenates all the model parameters which are linearly related to the potentials $\Psi(X,Y)$; C controls the relative weight of the sum of the violated terms $\{\xi^n(X,Y)\}$ with respect to the regularization term; $\triangle(X,Y;X^n,Y^n)$ is the loss function that generates large loss when the X or Y is very different from X^n or Y^n. Depending on the performance evaluation metric, we design different loss functions as described in the Sec. 5.1

Following the SSVM formulation, we propose to use a stochastic subgradient descent method to solve Eq. 24. The subgradient of $\partial_W \xi^n(X,Y)$ can be calculated as $\Psi(X^n,Y^n) - \Psi(X^*,Y^*)$, where $(X^*,Y^*) = \arg\min_{X,Y}(W^T\Psi(X,Y) - \triangle(X,Y;X^n,Y^n))$. When the loss function can be decomposed into a sum of local losses on individual segments and individual detections, (X^*,Y^*) can be solved using graph-cut similar to the inference problem (Sec. 4). For other complicated loss functions, we found that it is effective to set (X^*,Y^*) approximately as $\arg\min_{X,Y} W^T\Psi(X,Y)$, when the loss is bigger than a threshold.

The remaining model parameters are set as follows. The object category-specific consistency threshold $\rho(l)$ in Eq. 6 are estimated using the median values observed in training data. The γ involved in Eq. 13 are estimated from the MSE as described in [13]. The β in Eq. 9 are set to be the calibrated detection confidence such that $\beta \geq 0$. The unary potentials in $\psi_c(X_c)$ are obtained from off-the-shelf stuff classifiers [13], [43] (see details in Sec. 6). The pair-wise potentials in $\psi_c(X_c)$ are modelled using codebook representations similar to [40].

5.1 Loss Function

For the experiments on Stanford dataset, the performance is measured by the pixel-wise classification accuracy (i,e, percentage of pixels correctly classified), and the detection accuracy. We define the loss function $\triangle(X,Y;X^n,Y^n)$ as the sum of segmentation loss $\triangle(X;X^n)$ and the detection loss $\triangle(Y;Y^n)$.

The segmentation accuracy is measured by

$$\frac{\text{true positive}}{\text{true positive+ false negative}} . \tag{25}$$

Hence, the segmentation loss $\triangle(X;X^n)$ is defined as

$$\triangle(X;X^n) = \frac{1}{Q}\sum_{i\in\mathcal{V}} \mathbf{1}\{x_i \neq x_i^n\} c_x(l_i) , \tag{26}$$

where $\mathcal{V}$ captures the indices for the set of segments, $\mathbf{1}\{STATEMENT\}$ is 1 if the $STATEMENT$ is true, $c_x(l_i)$ is the object category l_i specific cost (used to reweight the loss contributed from different object categories), and $Q = \sum_{i\in\mathcal{V}} c_x(l_i)$. Therefore, the overall segmentation loss can be decomposed into a sum over local loss for each segment $\frac{1}{Q}\mathbf{1}\{x_i \neq x_i^n\}c_x(l_i)$.

The detection loss $\triangle(Y;Y^n)$ is defined as

$$\triangle(Y;Y^n) = \frac{1}{M}\sum_{i\in\mathcal{Q}_d} \mathbf{1}\{y_i \neq y_i^n\} c_y(l_i) , \tag{27}$$

where $\mathcal{Q}_d$ captures the indices for the set of detections, $M = \sum_{i\in\mathcal{B}} c_y(l_i)$. Similarly, the overall detection loss can be decomposed into a sum over local loss for each detection $\frac{1}{M}\mathbf{1}\{y_i \neq y_i^n\}c_y(l_i)$.

For the experiments on the PASCAL dataset, the overall loss function $\triangle(X,Y;X^n,Y^n)$ is similarly decomposed into sum of the segmentation loss $\triangle(X;X^n)$ and the detection loss $\triangle(Y;Y^n)$. The detection loss is the same as before. However, since the segmentation performance is measured differently by

$$\frac{\text{true positive}}{\text{true positive+ false positive + false negative}} , \tag{28}$$

the segmentation loss is defined as 1-segmentation performance. Notice that the segmentation loss cannot be decomposed into a per segment loss. Therefore, we obtain (X^*,Y^*) approximately as $\arg\min_{X,Y} W^T\Psi(X,Y)$, when $\triangle(X^*,Y^*;X^n,Y^n)$ is bigger than a threshold.

6 Experiments

We compare our full ACRF model with [13], [21], [32], [44], [45] on the Stanford Background (referred to as Stanford) dataset [21] as well as with several state-of-the-art techniques on PASCAL VOC 2009 segmentation (referred to as PASCAL) dataset [46]. As opposed to other datasets, such as MSRC [27], Stanford dataset contains more cluttered scenes and more object instances per image. Hence, segmenting and detecting "things" is particularly challenging. The challenging PASCAL segmentation dataset contains a large number of "things" labels with a single "stuff/background" label. However, the dataset contains a limited number of object instances in each image which is less ideal to demonstrate the importance of pair-wise relationships between object hypotheses.

Implementation details. For all the experiments below, we use the same pre-trained LSVM detectors [1] to obtain a set of object hypotheses with geometric properties for "things" categories (e.g., car, person, and bike). The object depths are inferred by combining both cues from the size and the bottom positions of the object bounding boxes similar to [18], [19], [24]. The subset of segments $\mathcal{V}$ associated to each object hypothesis is obtained by using the average object segmentation in the training set. In detail, for each mixture component in LSVM, we estimate the average object segmentation and use it to select the set of segmentation variables overlapped with the average object segmentation. The responses from off-the-shelf stuff classifiers are used as the unary stuff potentials in our model. On Stanford dataset, we use the STAIR Vision Library [43] that was earlier used in [21]. On PASCAL dataset, we use only the pixel-wise unary responses from the first layer of the hierarchical CRF [13]. We model different types of pair-wise stuff relationships using a codebook representation similar to [47]. The following geometric pair-wise relationships

	Sky	Tree	Road	Grass	Water	Building	Mountain	Background	Car	Person	Motorbike	Bus	Boat	Cow	Sheep	Bicycle	Global	Average
CRF	90.4	72.8	86.1	**91.8**	63.0	81.2	**56.7**	77.4	49.1	39.9	15.3	76.3	18.9	65.0	**70.4**	17.3	79.9	47.7
C+D	**92.4**	74.2	**88.0**	88.6	67.2	**84.9**	44.5	77.1	56.7	**61.7**	9.3	69.7	**36.9**	88.1	62.8	64.2	82.0	58.5
ACRF	91.9	**74.4**	87.3	88.5	**69.4**	84.6	44.5	77.2	**74.9**	60.1	**17.2**	**79.4**	**36.9**	**88.6**	58.2	**64.7**	**82.4**	**61.9**

TABLE 1: System analysis of our model on the Stanford dataset. The CRF row shows the results of the basic CRF model which uses only the stuff-stuff relationship component (first term in Eq. 5) of our ACRF model. The C+D row shows results by adding object hypothesis indicators obtained from pre-trained detectors to the CRF model (first two terms in Eq. 5). The last row shows results of the full ACRF model. Notice that in the "Background" column, we treat all background classes as a general background class. Our ACRF model improves the average accuracy for a significant 14% compared to the basic CRF model.

(a) Global Accuracy

[44]	[21]	[45]	[32]	[13]	ACRF
77.5	76.4	76.9	80.2	80.0	**82.4**

Recall	Background	Car	Person	Motorbike	Bus	Boat	Cow	Sheep	Bicycle	Global	Avg.
60%	77.2	71.5	60.3	17.2	80.7	35.1	88.6	58.2	61.5	82.3	61.1
40%	77.2	59.1	53.8	17.2	77.7	31.2	79.2	58.2	40.5	81.0	54.9
20%	77.3	53.2	47.8	17.2	75.8	24.2	73.6	58.2	37.1	80.5	51.6

TABLE 2: Segmentation performance comparison on the Stanford dataset. (a) Global accuracy of our ACRF model compared to state-of-the-art methods. (b) Sensitivity analysis of our segmentation accuracy affected by the quality of the detectors on Stanford dataset. Notice that the average accuracy decreases only gradually when the maximum recall decreases.

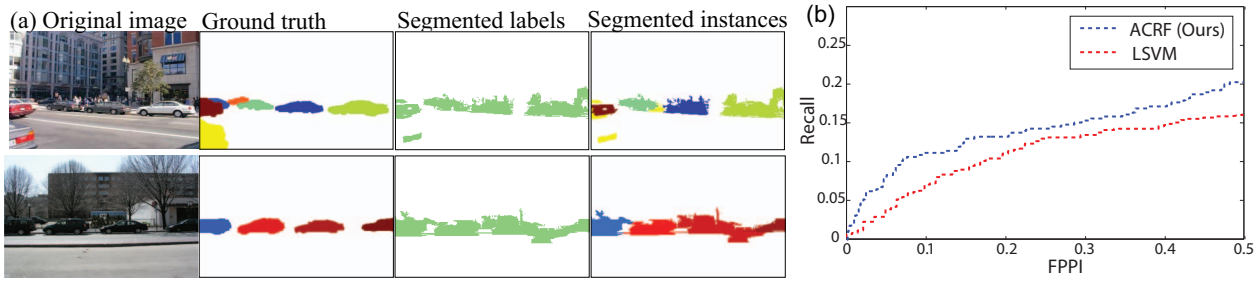


Fig. 5: (a) Typical thing segmentation results on the Stanford dataset. Notice that our model can obtain instance-based segmentations (last column) due to the ability to reason in the augmented labeling space $\mathcal{Q}$. (b) Recall v.s. FPPI curves of our ACRF and LSVM on the Stanford dataset. Our ACRF achieves better recall at different FPPI values.

are used for the experiments to incorporate geometric relationship between two object hypotheses: *next-to*, *above*, *below*, *in-front*, *behind*, *overlap*. On top of that, we have one additional geometric relationship based on horizon lines agreement between two hypotheses. The types of relationships are determined as described in Sec. 3.3. All models are trained with object bounding boxes and pixel-wise class supervisions.

6.1 Stanford dataset

The Stanford dataset [21] contains 715 images from challenging urban and rural scenes. On top of 8 background ("stuff") categories, we annotate 9 foreground ("things") object categories - car, person, motorbike, bus, boat, cow, sheep, bicycle, others. We follow the 5-fold cross-validation scheme which splits the data into different set of 572 training and 143 testing images. In Table 2(a)[2], our ACRF model outperforms state-of-the-art methods [13], [21], [32], [44], [45] in the percentage of pixels correctly classified as either one of the eight background classes, or a general foreground class (referred to as global accuracy).

Global accuracy v.s. average accuracy. The global accuracy is not ideal to highlight the accuracy gain of our method in foreground classes, since it ignores classification errors in fine foreground classes and the number of background pixels clearly outnumbers the number of foreground pixels. Hence, we report per class accuracy and the uniform average accuracy across a general background class and 8 foreground classes (referred to as average accuracy) in Table 1, Table 2(b), and Table 5.

In Table 1, the performances of most foreground classes (seven out of eight) are significantly improved when additional components are added on top of the basic CRF model. As a result, the full ACRF model obtains a 14.2% average improvement over the basic CRF model. Typical results are shown in Fig. 9-Top. Using our efficient inference algorithm, inference takes in average 1.33 seconds for each image which has 200 to 300 super-pixels on a Intel(R) Xeon(R) CPU @ 2.40GHz. We highlight that our model can generate object instance-based segmentations due to the ability to reason in the *augmented labeling space* $\mathcal{Q}$ (Fig. 5(a)).

Another advantage of using our model is to improve detection accuracy. We measure detection performance in terms of Recall v.s. False Positive Per Image (FPPI) in Fig. 5(b), where detection results from 5-fold validations are accumulated and shown in one curve. The performance of the proposed model is compared with the pre-trained LSVM [1]. Our model achieves consistent higher recall than the LSVM base-

2. We implement [13], [32] by ourselves and evaluate the performance.

	Background	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dining table	Dog	Horse	Motor bike	Person	Potted plant	Sheep	Sofa	Train	TV/monitor	Average
CRF	69.0	19.7	2.4	8.8	6.8	8.8	21.6	17.5	13.1	0.5	8.6	7.1	6.5	6.5	13.8	19.2	5.8	13.6	3.3	21.3	9.5	13.5
CRF+Det	71.0	21.8	**14.8**	21.1	**18.7**	**34.8**	**48.3**	33.3	16.7	**12.3**	**27.5**	10.5	14.1	27.5	35.4	31.2	29.8	**28.7**	17.5	31.8	27.7	27.3
ACRF	**75.5**	**29.1**	14.7	**23.0**	18.2	34.0	47.8	**40.4**	**17.2**	11.4	27.0	**12.6**	**17.5**	**30.1**	**40.1**	**34.9**	**30.6**	28.2	**20.7**	**31.0**	**30.1**	**29.4**

TABLE 3: The segmentation accuracy of different variants of our model (i.e., CRF, CRF+ Detection, and full ACRF models) on PASCAL dataset.

BONN	CVC	NECUIUC_CLS	ACRF	UoCTTI	NECUIUC_SEG	LEAR	BROOKES	UCI	MPI
36.3	34.5	29.7	29.4	29.0	28.3	25.7	24.8	24.7	15.0

TABLE 4: Average segmentation accuracy of our ACRF model compared to other state-of-the-art methods on PASCAL dataset.

Remove	Background	Car	Person	Motorbike	Bus	Boat	Cow	Sheep	Bicycle	Global	Avg.
CarPerson	77.1	66.9	58.5	14.6	79.2	36.9	88.6	58.2	64.8	82.1	60.7
CarBus	77.2	73.9	58.2	17.2	67.3	36.9	88.6	58.3	64.7	82.2	60.3
All	77.2	56.8	59.9	9.1	69.5	37.2	90.3	61.2	64.1	82.0	58.4

TABLE 5: Segmentation accuracy on the Stanford dataset when the pair-wise geometric relationships are partially (e.g., car and person, or car and bus geometric relationships removed) or totally (i.e., all geometric relationships) removed.

line at small number of FPPI as shown in Fig. 5(b).

Since our method utilizes pre-trained object detectors to obtain a set of plausible object hypotheses, we evaluate the segmentation accuracy given worse detectors to see how our method depends on the quality of the detectors. We simulate a worse detector by reducing the number of recalled objects in the set of plausible detections. As shown in Table 2(b), the average accuracy decreases only gradually when the maximum recall decreases. Notice that, even when the recall is only 20%, our ACRF model still achieves accuracy better than the basic CRF.

6.2 PASCAL dataset

This dataset contains $14,743$ images with 21 categories including 20 thing categories and 1 stuff category. Only a subset of images have segmentation labels, and we used the standard split for training (749 images), validation (750 images), and testing (750 images). A system analysis of our model (Table 3) shows that the performances of most classes were improved when additional components are added on top of the basic CRF model. However, our ACRF model is able to significantly boost up the performance and achieves competitive accuracy compared to other teams in the challenge (ranked in 4_{th} in Table 4). Typical results are shown in Fig. 9-Bottom.

6.3 Relationship Analysis

We found that the pair-wise geometric relationships contribute to the accuracy improvement of our ACRF model more than the co-occurrence relationships, since CRF+Det+co-occurrence relationships (Last row in Table 5) do not consistently improve the accuracy of all categories compared to CRF+Det (middle row in Table 1). Moreover, all geometric relationships collectively contribute to the accuracy improvement, since our model is still better than CRF+Det when the geometric relationships of two most frequently co-occurred pairs of object categories, namely car v.s. person and car v.s. bus, are removed, respectively (First two rows in Table 5). The learned relation parameters of the two most frequently co-occurred pairs of object categories are visualized in Fig. 6(a,b). Nevertheless, the learned relationships sometime introduce errors. We show the failure cases in Fig. 7.

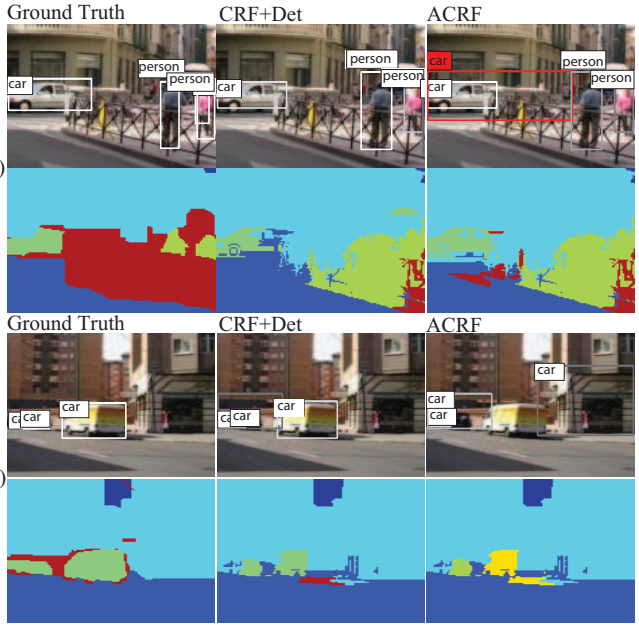
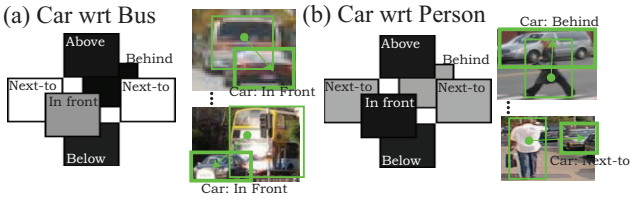


Fig. 7: Failure cases analysis. Panel (a) shows the case when the learned pairwise relationship between a car and a person next-to each other does not match to the existing relationship in the test image. As a result, the false alarm of a car (red box) appears with ACRF. (b) The typical example when depth heuristic fails. The yellow car in the center of the image is successfully detected and segmented with a CRF+Det model. However, it fails to detect with ACRF model, because the depth is not correctly inferred due to the fact that the height of the yellow car is not close to the average height of the cars in the training set.

In Fig. 6(c), we evaluate the percentage of pairs of true positive object detections for each relationship.



(c)	Before	BL1	BL2	ACRF
next-to	5.4%	25.9%	32.2%	**92.5%**
above/below	6.1%	51.72%	61.2%	**76.2%**
in-front/behind	5.7%	50.0%	60.4%	**72.3%**
different horizon	0.1%	38.5%	15.0%	**60.0%**

Fig. 6: Examples of the learned pair-wise relationships between object hypotheses are visualized in panel (a,b). The grayscale color code indicates to what degree the relationship is likely (white means it is likely, black means it is unlikely). Our model learned that (a) a car is likely to be in front of a bus, and a car is unlikely to be below a bus, (b) a car is likely to be behind a person. (c) Prediction accuracy of the objects co-occurrence for each type of relationship averaged over 5-fold validations. The first and last columns show the accuracy before and after applying inference on our full ACRF model, respectively. Notice that there is a consistent improvement across all types. The performance of two baseline methods are reported in the middle two columns which are all inferior then our results.

Before (i.e., raw detections from LSVM [1]) applying inference, the percentage is fairly low since there are many false positive detections. After applying our ACRF model, the percentage increases dramatically as expected. We also outperform two stronger baseline methods aiming at pruning out incorrect pairs of object hypotheses for each relationship as defined below. BL1 uses only the detection confidence to prune out detections. In specific, for each pair of detections with a certain relationship, we assign a score as a sum of scores for both bounding boxes from LSVM. Then, we sample $p\%$ of pairs with highest scores, where p is the percentage of pairs of true positive detections for a certain relationship from the training set. BL2 incorporates pairwise object interactions and prune out detections. Again, for each pair of detections with a certain relationship, we assign a score as a sum of detection scores for both detections. Then, we sample pairs within top $p(c_1, c_2)\%$, where $p(c_1, c_2)$ is a class-pair specific percentage of pairs of true positive detections from the training set, and c_1 and c_2 is classes corresponding to two bounding boxes.

Using the inferred relationships we can provide high level geometrical description of the scene and determine properties such as: object x is behind object y. Finally, we can obtain 3D pop-up models of the scene from a single image as in Fig. 8

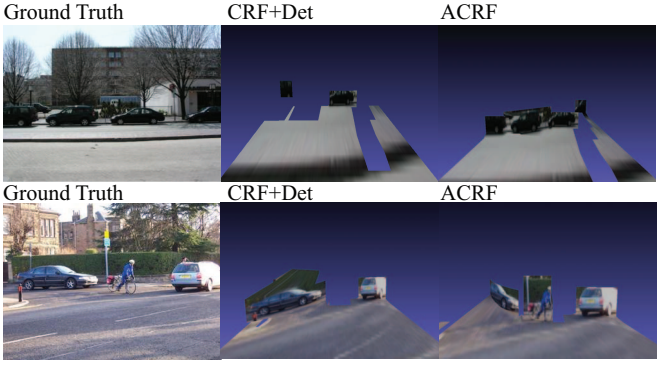


Fig. 8: 3D pop-up models from the Stanford dataset. Videos related to above 3D pop-up models can be found in the project page: www.eecs.umich.edu/vision/projects/ACRF/ACRFproj.html

7 Conclusion

We have presented a unified CRF-based framework for jointly detecting and segmenting "things" and "stuff" categories in natural images. We have shown that our framework incorporates in a coherent fashion various types of (geometrical and semantic) contextual relationships via property interactions. Our new formulation generalizes previous results based on CRF where the focus was to capture the co-occurrence between stuff categories only. We have quantitatively and qualitatively demonstrated that our method: i) produces better segmentation results than state-of-the art on the Stanford dataset and competitive results on PASCAL'09 dataset; ii) improves the recall of object instances on Stanford dataset; iii) enables the estimation of contextual relationship among things and stuff. Extensions for future work include incorporating more sophisticated types of properties.

References

[1] Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: CVPR. (2008)
[2] Grauman, K., Darrell, T.: The pyramid match kernel: discriminative classification with sets of image features. In: ICCV. (2005)
[3] Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV workshop on statistical learning in computer vision. (2004)
[4] Gall, J., Lempitsky, V.: Class-specific hough forests for object detection. In: CVPR. (2009)
[5] Maji, S., Malik, J.: Object detection using a max-margin hough tranform. In: CVPR. (2009)
[6] Barinova, O., Lempitsky, V., Kohli, P.: On detection of multiple object instances using hough transforms. In: CVPR. (2010)
[7] Sun, M., Bradski, G., Xu, B.X., Savarese, S.: Depth-encoded hough voting for coherent object detection, pose estimation, and shape recovery. In: ECCV. (2010)
[8] Sun, M., Su, H., Savarese, S., Fei-Fei, L.: A multi-view probabilistic model for 3d object classes. In: CVPR. (2009)
[9] Xiang, Y., Savarese, S.: Estimating the aspect layout of object categories. In: CVPR. (2012)
[10] He, X., Zemel, R.S., Carreira-Perpiñán, M.Á.: Multiscale conditional random fields for image labeling. In: CVPR. (2004)
[11] Kohli, P., Ladicky, L., Torr, P.H.: Robust higher order potentials for enforcing label consistency. In: CVPR. (2008)
[12] Shotton, J., Blake, A., Cipolla., R.: Semantic texton forests for image categorization and segmentation. In: CVPR. (2008)
[13] Ladicky, L., Russell, C., Kohli, P., Torr, P.H.: Graph cut based inference with co-occurrence statistics. In: ECCV. (2010)

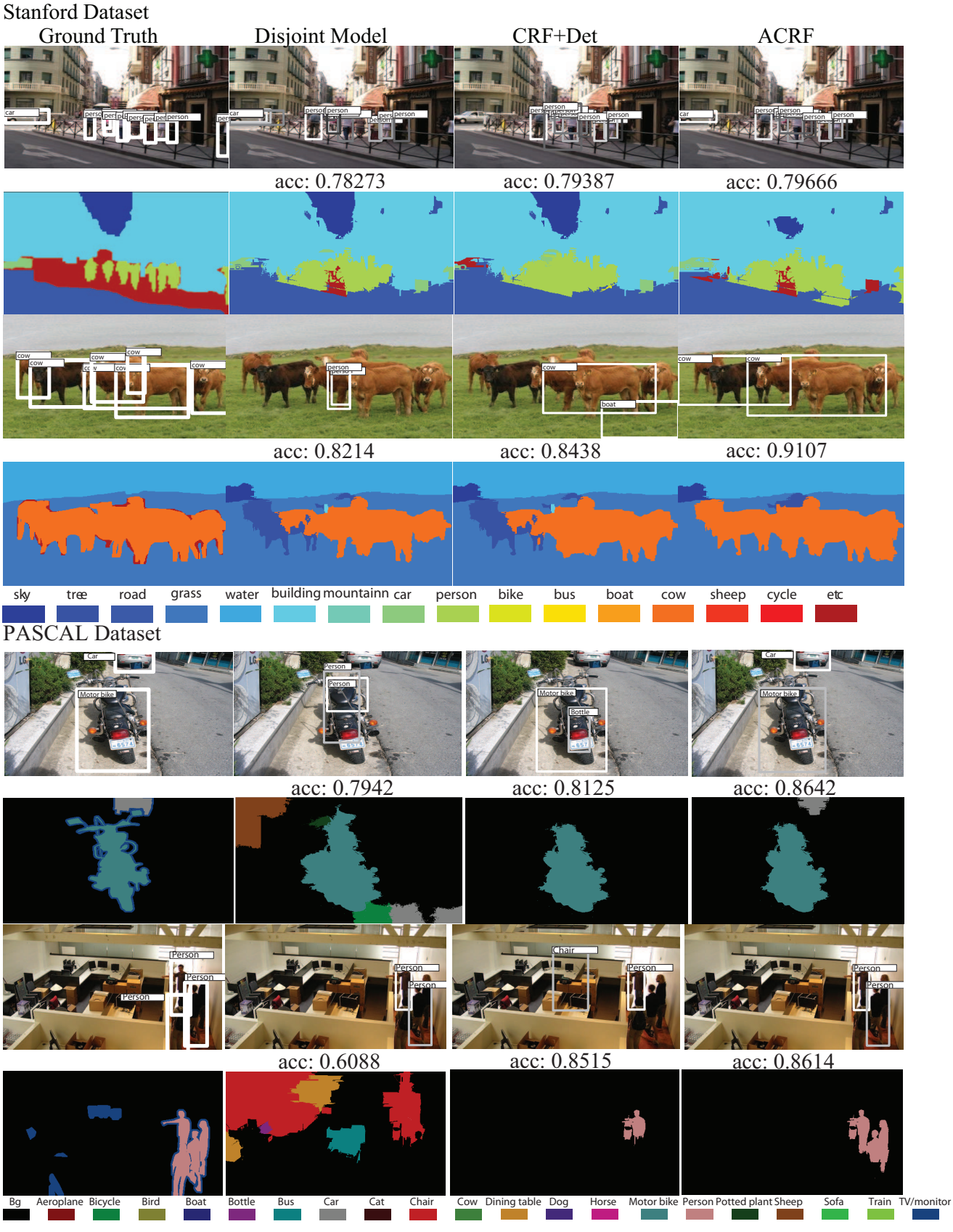
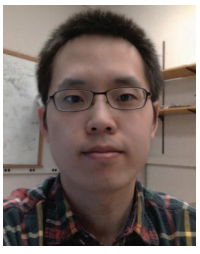


Fig. 9: Typical results on the Stanford (top 4 rows) and the PASCAL datasets (bottom 4 rows). Every set of results compare ground truth annotation, disjointed model (disjointedly applied object detection and segmentation methods), CRF+Det, ACRF, from left to right, respectively. The odd rows show the top K object hypotheses (color-coded bounding boxes representing the confidence ranking from light (hight confidence) to dark (low confidence)), where K is the number of recalled objects in the ACRF result. The even rows show the segmentation results (color-code is shown at the bottom).

[14] Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: ICCV. (2007)
[15] Gupta, A., Davis, L.: Beyond nouns: Exploiting prepositions and comparators for learning visual classifiers. In: ECCV. (2008)
[16] Desai, C., Ramanan, D., Fowlkes, C.: Discriminative models for multi-class object layout. In: ICCV. (2009)
[17] Heitz, G., Koller, D.: Learning spatial context: Using stuff to find things. In: ECCV. (2008)
[18] Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: CVPR. (2006)
[19] Bao, S.Y., Sun, M., Savarese, S.: Toward coherent object detection and scene layout understanding. In: CVPR. (2010)
[20] Winn, J., Shotton, J.: The layout consistent random field for recognizing and segmenting partially occluded objects. In: CVPR. (2006)
[21] Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: ICCV. (2009)
[22] Gould, S., Gao, T., Koller, D.: Region-based segmentation and object detection. In: NIPS. (2009)
[23] Heitz, G., Gould, S., Saxena, A., Koller, D.: Cascaded classification models: Combining models for holistic scene understanding. In: NIPS. (2008)
[24] Sun, M., Bao, S.Y., Savarese, S.: Object detection with geometrical context feedback loop. In: BMVC. (2010)
[25] Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop on scene interpretation. In: CVPR. (2008)
[26] Li, C., Kowdle, A., Saxena, A., Chen, T.: Toward holistic scene understanding: Feedback enabled cascaded classification models. PAMI **34** (2012) 1394–1408
[27] Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV. (2006)
[28] Barinova, O., Lempitsky, V., Kohli, P.: On detection of multiple object instances using hough transforms. PAMI (2012)
[29] Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector learning for interdependent and structured output spaces. In: ICML. (2004)
[30] Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. PAMI (2001)
[31] Rother, C., Kolmogorov, V., Lempitsky, V., Szummer, M.: Optimizing binary mrfs via extended roof duality. In: CVPR. (2007)
[32] Ladicky, L., Sturgess, P., Alahari, K., Russell, C., Torr, P.H.: What, where, and how many? combining object detectors and CRFs. In: ECCV. (2010)
[33] Yao, Y., Fidler, S., Urtasun, R.: Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In: CVPR. (2012)
[34] Bao, S.Y., Savarese, S.: Semantic structure from motion. In: CVPR. (2011)
[35] Li, L.J., Socher, R., Fei-Fei, L.: Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In: CVPR. (2009)
[36] Fei-Fei, L., Li, L.J.: What, Where and Who? Telling the Story of an Image by Activity Classification, Scene Recognition and Object Categorization. Studies in Computational Intelligence-Computer Vision (2010) 157–171
[37] Lee, Y.J., Grauman, K.: Object-graphs for context-aware category discovery. In: CVPR. (2010)
[38] Galleguillos, C., McFee, B., Belongie, S., Lanckriet, G.R.G.: Multi-class object localization by combining local contextual interactions. In: CVPR. (2010)
[39] Divvala, S.K., Hoiem, D., Hays, J., Efros, A.A., Hebert, M.: An empirical study of context in object detection. In: CVPR. (2009)
[40] Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. IJCV (2005)
[41] Boros, E., Hammer, P.: Pseudo-boolean optimization. Discrete Applied Mathematics (2002)
[42] Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts. PAMI (2004)
[43] Gould, S., Russakovsky, O., Goodfellow, I., Baumstarck, P., Ng, A., Koller, D.: The stair vision library (v2.3) (2009)
[44] Tighe, J., Lazebnik, S.: Superparsing: Scalable nonparametric image parsing with superpixels. In: ECCV. (2010)
[45] Munoz, D., Bagnell, J.A., Hebert, M.: Stacked hierarchical labeling. In: ECCV. (2010)
[46] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results (2009)
[47] Bosch, X.B., Gonfaus, J.M., van de Weijer, J., Bagdanov, A.D., Serrat, J., Gonz'alez, J.: Harmony potentials for joint classification and segmentation. In: CVPR. (2010)

Min Sun received his B.Sc. from National Chiao Tung University, his M.Sc. from Stanford University, and his Ph.D. from the University of Michigan, Ann Arbor. He is currently a postdoctoral researcher at the University of Washington. His research interests include object recognition, image understanding, human pose estimation, and machine learning. He has won best paper awards in 3DRR 2007.



Byung-soo Kim received his B.Sc. from Korea Advanced Institute of Science and Technology, his M.Sc. from the University of Michigan, Ann Arbor and he is currently a Ph.D. candidate at the University of Michigan, Ann Arbor. His research interests include object recognition and image understanding from color/depth images.

Pushmeet Kohli Pushmeet Kohli is a research scientist in the Machine Learning and Perception group at Microsoft Research Cambridge, and an associate of the Psychometric Centre, University of Cambridge. His PhD thesis written at Oxford Brookes University was the winner of the British Machine Vision Associations Sullivan Doctoral Thesis Award, and a runner-up for the British Computer Society's "Distinguished Dissertation Award". Pushmeets research revolves around Intelligent Systems and Computational Sciences with a particular emphasis on algorithms and models for scene understanding and human pose estimation. His papers have appeared in SIGGRAPH, NIPS, ICCV, AAAI, CVPR, PAMI, IJCV, CVIU, ICML, AISTATS, AAMAS, UAI, ECCV, and ICVGIP and have won best paper awards in ECCV 2010, ISMAR 2011 and ICVGIP 2006, 2010.

Silvio Savarese Silvio Savarese is an Assistant Professor of Electrical Engineering at the University of Michigan, Ann Arbor. After earning his Ph.D. in Electrical Engineering from the California Institute of Technology in 2005, he joined the University of Illinois at Urbana-Champaign from 20052008 as a Beckman Institute Fellow. He is recipient of the James R. Croes Medal in 2013, a TWR Automotive Endowed Research Award in 2012, an NSF Career Award in 2011 and Google Research Award in 2010. In 2002 he was awarded the Walker von Brimer Award for outstanding research initiative. He served as workshops chair and area chair in CVPR 2010, and as area chair in ICCV 2011 and CVPR 2013. He was editor of the Elsevier Journal in Computer Vision and Image Understanding, special issue on "3D Representation for Recognition" in 2009. He co-authored a book on 3D object and scene representation published by Morgan and Claypool in 2011. His work with his students has received several best paper awards including a best student paper award in the IEEE CORP workshop in conjunction with ICCV 2011, a Best Paper Award from the Journal of Construction Engineering and Management in 2011and the CETI Award at the 2010 FIATECH's Technology Conference. His research interests include computer vision, object recognition and scene understanding, shape representation and reconstruction, human activity recognition and visual psychophysics.