Pottics – The Potts Topic Model for Semantic Image Segmentation

Christoph Dann[1], Peter Gehler[2], Stefan Roth[1] and Sebastian Nowozin[3]

[1]Technische Universität Darmstadt, [2]Max Planck Institute for Intelligent Systems,
[3]Microsoft Research Cambridge

Abstract. We present a novel conditional random field (CRF) for semantic segmentation that extends the common Potts model of spatial coherency with latent topics, which capture higher-order spatial relations of segment labels. Specifically, we show how recent approaches for producing sets of figure-ground segmentations can be leveraged to construct a suitable graph representation for this task. The CRF model incorporates such proposal segmentations as topics, modelling the joint occurrence or absence of object classes. The resulting model is trained using a structured large margin approach with latent variables. Experimental results on the challenging VOC'10 dataset demonstrate significant performance improvements over simpler models with less spatial structure.

1 Introduction

Semantic segmentation of natural images aims to partition the image into semantically meaningful regions. Depending on the task, each region represents high-level information such as individual object instances or object parts, types of surfaces, or object class labels. Semantic segmentation is a challenging research problem in computer vision; major progress has been made in the last decade, originating from three developments: First, larger benchmark data sets with thousands of annotated training images are now common [6]. Second, conditional random fields (CRFs), estimated from training data, have become a standard tool for modeling the spatial relations of segment labels. Algorithmic advances in inference and estimation for these models, as well as insights in how to build these models efficiently has enabled further performance gains [11, 15, 20]. Third, segmentation and object detection priors that are independent of the object class have been developed [1, 5]. Incorporating such priors into a segmentation model has proven to yield large performance gains.

In this paper we build on these recent developments and present a simple, but effective model for the task of semantic scene segmentation. The model is a conditional random field with two groups of random variables: segments and regions. *Segments* are large overlapping parts of the image, which we extract using constrained parametric min-cuts [5]. Each segment is more likely to be homogeneous with respect to the semantic labeling of the image, but because different segments overlap there may exist multiple contradicting segments. *Regions* are smaller entities that partition the image; we use superpixels to define these regions. Since regions are small, we can assume that they are pure in the semantic labeling. Yet, regions are typically too small to cover an entire semantic instance such as an object. Therefore they provide only limited evidence for a semantic class. Regions and segments represent two complementary levels of groupings of image pixels. This is illustrated in Fig. 1.

Our CRF model combines the two primitives into one coherent semantic segmentation model. To that end, segments represent latent topics and regions represent the actual consistent image labeling. The two components are coupled by an interaction term that associates semantic labels of regions with latent segment-level topics. The topics thus describe how region labels are spatially arranged. By learning the interaction terms from training data we jointly discover latent topics and their relation to the region-level class label.

Our contributions are the following: (1) A novel semantic segmentation model that integrates latent segment-level topics with a consistent image labeling, and (2) an efficient latent structural SVM training method for the model. Experiments on the challenging PASCAL VOC 2010 challenge demonstrate significant accuracy gains over various baselines, including Potts-based label smoothing.

2 Related Work

Our approach integrates a segmentation prior into a random field model. This follows a range of previous work, which has remained limited in various ways. Superpixel segmentations [17], small coherent regions in the image that are likely to belong a single object class, build the foundation of several recent semantic labeling approaches. It is a standard practise to define a structured model, such as a CRF, on top of superpixel segmentations [2, 16, 13, 8, 7], because the superpixels provide computational benefits and regularization. These previous approaches differ in the way they incorporate spatial consistency between superpixels.

Flat CRF models, such as [16, 7, 2], are formulated on class assignments of superpixels and penalize different labels of neighbors by a pairwise smoothing term. In case of [2], the spatial relation is estimated from training data. [7] additionally considers image features on superpixels and their direct neighbors. These features are then classified, leading to spatially robust decisions. All these methods use a flat neighborhood relation to formulate spatial consistency.

Hierarchical CRFs in contrast, promote spatial label consistency using higher-level entities to couple pixel classes. [14] uses a tree-structured CRF on iteratively refined superpixels that allows efficient parameter learning. Recent extensions [8, 13] augment this hierarchical CRF with higher-order potentials to incorporate image-level information and promote smoothness among many pixels.

Another class of methods [3, 16] first performs an intelligent oversegmentation of the image into likely objects, and then processes the “segment soup” to find objects. By combining the found objects, a single segmentation of the input is produced. The approach of [3] performs very well but is algorithmic, therefore has no explicit model that can be estimated from data. In [16] a manually tuned CRF is used to fuse multiple oversegmentations. Multiple attempts keep the flavour (and performance) of [3] while providing a principled probabilistic model have been made recently. In [4] and its extensions [9, 10], a pairwise graph is constructed by adding an edge between each overlapping pair of segments. By solving a maximal clique problem on the graph a coherent segmentation of the image is obtained. While principled, the approach is limited,

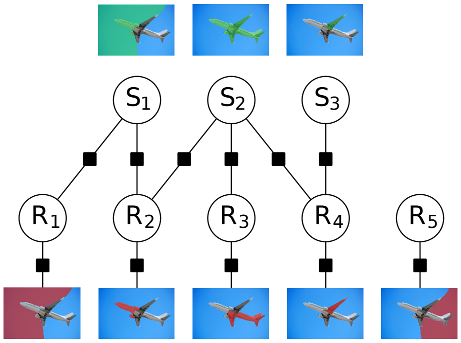

Fig. 1. CRF factor graph structure: CPMC segments (top row) represent latent topics, the regions from the corresponding decomposition of the image (bottom row) yield the semantic labels. A region is connected to a segment if it is fully contained in it; image evidence enters through unaries. Simplified example with 3 segments and 5 regions.

because it only considers pairwise terms between segments; in contrast our approach uses latent variables that can act on all regions within a segment.

3 Pottics Model

Our proposed CRF for semantic segmentation builds on three main components: *(i)* Segmentation proposals from [3], which yield the regions via a superpixelization of the input image as well as the graph structure; *(ii)* the CRF model itself including the potentials, which model bottom-up image evidence and the segment-region relations; and *(iii)* structured max-margin training with latent variables for the learning part [21]. For learning we assume that we are given a set of N training images $I^i, i = 1, \dots, N$ along with ground truth pixel-wise class labelings $Y^i, i = 1, \dots, N$. We now describe the model components in turn.

3.1 CPMC segmentation proposals

We build upon the Constrained-Parametric-Min-Cuts (CPMC) method [5] that generates a set of plausible figure-ground segmentations. The main idea behind this algorithm for the use of semantic segmentation is to pre-generate a number of plausible segmentations, from which, in a separate step, a segment is chosen as the final prediction. This largely reduces the state space and also reduces the problem of semantic segmentation to a ranking problem, which allows efficient ranking algorithms to be used. This method consistently performed with top scores on the PASCAL VOC segmentation challenges [6].

The complete CPMC algorithm is rather involved; we can thus only present a brief summary and refer to [5] for details. CPMC operates in three steps to generate a set of binary foreground-background segmentations. First a graph-cut problem is solved multiple times with different pixels in the images being forced to become foreground. Figure 2 (top row, 3rd from left) illustrates this: By forcing different pixels (green) to be foreground and solving the graph-cut problem, CPMC produces different binary segmentation masks. Parts of the image borders may be chosen as foreground in order to

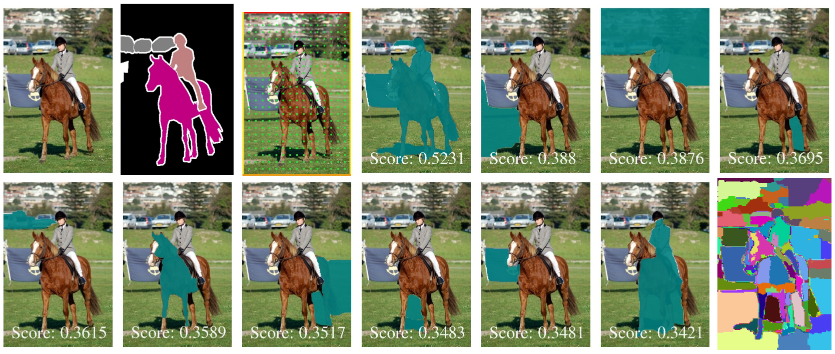

Fig. 2. *Top row* (left to right): Input image, ground truth annotation, example of seeds used for min-cut, top-four segments with their score. *Bottom row* (left to right): segments 5–10, resulting region segmentation from pairwise intersection of segments.

allow for partially occluded objects. This first step generates a large number of plausible segmentation hypotheses, but many of these segmentations are almost identical. The second step of the CPMC algorithm filters the set of segments based on a score computed from their overlap. In addition, segments that are deemed too small are removed. On average, about 300–400 segmentation masks per image are retained after this filtering procedure. In the third and last step, the segments are scored in order to predict how "object-like" they are. To this end 34 different features derived from segment shape, gestalt, and graph properties are used. For illustration, we show the ten highest scored segments for an example image in Fig. 2.

For every input image we generate a set of *segments* $S_i, i = 1, \dots, N_S$ using the CPMC procedure. From these segments we compute the set of all their intersections. We refer to these as the *regions* $R_i, i = 1, \dots, N_R$ of the image. By construction all the regions are disjoint, $R_i \cap R_j = \emptyset, \forall i, j$. For the running example in Fig. 2 the output of this step is depicted on the bottom right. Here, the image is partitioned into 1550 different regions.

3.2 CRF formulation

We use the multiple segmentations from CPMC to construct a random field model as follows (see Fig. 1 for an example): Assume that the image has been segmented into N_R regions and N_S segments, where the values N_R, N_S can vary from image to image. In order to maintain a simple notation we will ignore this. Each region i can take values in $R_i \in \{1, \dots, C\}$, where C denotes the number of semantic classes. The segments j are modeled as taking values in $S_j \in \{1, \dots, T\}$, where T is the number of latent topics, to be determined by model selection. With bold letters we will refer to the concatenation of variables, i.e. $\mathbf{R} = (R_1, \dots, R_{N_R})$. We connect all regions with the segments they are contained in and place a pairwise potential function on them. Furthermore we connect the region variables with image evidence by using unary potentials. The full CRF joint probability is now given as $p(\mathbf{R}, \mathbf{S}) \propto \exp(-E(\mathbf{R}, \mathbf{S}))$ with

$$E(\mathbf{R}, \mathbf{S}) = -\langle w, \phi(\mathbf{R}, \mathbf{S}, I) \rangle = -\sum_{i=1}^{N_R} \langle w_u, \phi_u(R_i, I) \rangle - \sum_{i=1}^{N_R} \sum_{j \sim i} \langle w_p, \phi_p(R_i, S_j, I) \rangle . \tag{1}$$

We use $j \sim i$ to denote that region i is contained in segment j, that is, whether the variables R_i and S_j share a common edge in the graph. We make a simple choice for the pairwise features, namely $\phi_p(R, S)$ being a binary vector of size $C{\cdot}T$, with a 1 at the entry that corresponds to (R, S) and 0 otherwise. This allows the parameters w_p to be represented a vector of size $C \cdot T$. For the unary features we use standard, pre-computed image descriptors; more details are given in Sec. 4.

3.3 Learning

The undirected bi-partite graph structure of the model renders computation of the normalization function intractable. Therefore, maximum likelihood learning of the model is intractable as well, and one would have to resort to approximate versions or different estimators, such as contrastive divergence or the pseudo-likelihood. We here follow a different route and use a max-margin approach.

Specifically, we learn the parameters of the CRF in Eq. (1) using a structured SVM with latent variables [21]. The optimization problem can be written as

$$\min_{w,\xi} \quad \frac{1}{2}\|w\|^2 + C\xi \tag{2}$$

$$\text{sb.t.} \quad \sum_i \left(\max_{\bar{\mathbf{R}}} \left[\max_{\bar{\mathbf{S}}} \langle w, \phi(\bar{\mathbf{R}}, \bar{\mathbf{S}}, I^i)\rangle + \varDelta(\bar{\mathbf{R}}, Y^i) \right] - \max_{\mathbf{S}} \langle w, \phi(\mathbf{R}^i, \mathbf{S}, I^i)\rangle \right) \leq \xi$$

This non-convex problem is solved with the cutting plane algorithm as implemented in the latentSVM$^{\text{struct}}$ software package.[1] With $\varDelta : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ we denote the loss function that measures the quality of our prediction.

In order to optimize Eq. (2), the following *loss-augmented inference problem* needs to be solved:

$$\bar{\mathbf{R}} = \operatorname*{argmax}_{\tilde{\mathbf{R}}} \left[\max_{\mathbf{S}} \langle w, \phi(\tilde{\mathbf{R}}, \mathbf{S}, I)\rangle + \varDelta(\tilde{\mathbf{R}}, Y) \right]. \tag{3}$$

To make a prediction on a novel test image, we need to solve the energy minimization problem

$$\mathbf{R}^* = \operatorname*{argmax}_{\mathbf{R}} \max_{\mathbf{S}} \langle w, \phi(\mathbf{R}, \mathbf{S}, I)\rangle. \tag{4}$$

Different loss functions $\varDelta$ have been used for semantic scene segmentation. The authors of the PASCAL VOC challenge [6] implement the following criterion to cope with the unbalanced number of pixels per class

$$\varDelta(\bar{Y}, Y) = \frac{1}{C} \sum_{k=1}^{C} \frac{|Y_k \cup \bar{Y}_k|}{|Y_k \cap \bar{Y}_k|}. \tag{5}$$

With Y_k we denote the binary segmentation mask with 1's where Y is of class k, 0 otherwise. However, this loss is hard to incorporate during training because it does not decompose over individual regions due to the normalizing denominator. In other words, optimally predicting segmentations for one test image alone is not possible, and all test images need to be predicted jointly. In [19] this issue is addressed by recognizing it to be a problem of inference with a higher order potential. An exact inference algorithm that empirically scales with $N \log N$ is given. We take an easier approach and circumvent

[1] http://www.cs.cornell.edu/people/tj/svm_light/svm_struct.html

this problem by substituting the loss function during training time with the Hamming loss as a proxy

$$\Delta_H(\bar{Y}, Y) = \frac{1}{|Y|} \sum_{i=1}^{|Y|} [Y(i) \neq \bar{Y}(i)], \tag{6}$$

where $[\cdot]$ denotes the Iverson bracket and $Y(i)$ the i^{th} pixel of the segmentation Y. This renders Eq. (3) to be decomposable per image, and furthermore even per region in the image. The loss-augmented inference for image-label pair (I, Y) then reduces to

$$\operatorname*{argmax}_{\bar{\mathbf{R}}} \max_{\mathbf{S}} \sum_{i=1}^{N_R} \langle w_u, \phi_u(\bar{R}_i, I)\rangle + \sum_{i=1}^{N_R} \sum_{j \sim i} \langle w_p, \phi_p(\bar{R}_i, S_j, I)\rangle + \sum_{i=1}^{N_R} \Delta_H(\bar{R}_i, Y). \tag{7}$$

Hence the loss can be seen as just another unary factor applied to the regions.

To approximately solve the inference problems in Eqs. (4) and (7), we apply Iterated Conditional Modes (ICM). With fixed region variables, we update the segment variables, then for fixed segment variables we update the region variables. Since the graph is bi-partite all regions and segments can be updated in parallel. Convergence is typically reached in about 3 iterations per image.

4 Experiments

We test our model[2] on the very challenging PASCAL VOC 2010 semantic segmentation dataset. Since the test set is not publicly available, we train on the *train* split while testing using the *val* split (964 images each).

The main objective of this work is to utilize proposal segmentations and CRFs to improve over unary or region-wise prediction methods. Therefore, we use a single, competitive unary [12] from the literature as a baseline, which has been pre-trained on the training portion of VOC'10 and is publicly available[3]. The unary potentials are based on TextonBoost [18] augmented by color, pixel location, histogram of gradients (HOG) information, as well as the outputs of bounding box object detectors. While unary feature functions can be learned jointly with the pairwise components, this is not the main motivation here. Since, moreover, piecewise training has been shown to be a competitive speed-up (e.g. [14]) we fix the pre-trained unaries as our unary feature ϕ_u and train only a scaling factor w_u as well as the pairwise potentials.

4.1 Baseline methods

We test against various baseline methods: First, we use the unary potential function to predict pixel class membership directly (UO). Next, pixel-predictions are accumulated over regions and all pixels within a region are assigned to the maximal class score (RP). A third method makes use of the CPMC segments. To that end, all segments accumulate the pixel-predictions of the unary factor and the highest scoring class is chosen as the "segment-label". Since a pixel may be contained in multiple segments, its class is predicted to be the majority vote of all segment labels of segments that it is contained in. We call this method segment-prediction (SP).

[2] Part of the code used in our experiments is available at http://github.com/chrodan/pottics

[3] http://graphics.stanford.edu/projects/densecrf/unary/

Besides these simple voting strategies, we evaluate the classic Potts model as another baseline. That is we connect all neighboring regions (in the sense of a pixel being in a 4-connected neighbourhood relationship with another region) with a pairwise factor of the simple form $\phi_p(R_i, R_j) = \alpha[R_i = R_j]$. We choose α by model selection, but found that any value $\alpha \neq 0$ deteriorates the performance. We still include this method (Potts) with $\alpha = 0.5$. Last, we consider our model with the number of topics T being set to 1 as a generalized form of the Potts model that makes use of the special graph structure that we are building. This model (1T) is also trained using the max-margin learning. All methods use the same set of $N_S = 100$ segments and the resulting region decomposition. We found that fewer segments cannot represent details in images well enough and additional segments deteriorate performance because of their low quality.

4.2 Empirical results

The empirical results are summarized in Table 1. We report both accuracy using the Hamming loss (left, Eq. (6)) and the VOC loss (right, Eq. (5)). Here the Pottics model has been trained with the number of topics set to $T = 50$ (we tested with setting $T \in \{25, 100\}$ and found qualitatively the same results). We make the following observations: The Pottics model outperforms the other baselines on most classes, and in the class-averaged total score. This confirms our intuition that the CPMC segments provide valuable information on how to connect different regions in the image. We also note that turning this additional information into better performance is not immediate. All other baseline methods that use the constructed graph (SP, Potts, 1T) perform worse than the simple UO baseline. Against our intuition, Potts performs worse as well. We suspect that the unary pixel prediction is sufficiently strong to already incorporate neighbourhood information, hence a simple class smoothing performs worse.

The gain of Pottics over UO is quite substantial with an 5.2% increase in average VOC performance. In Fig. 3 we show images where the improvement in terms of accuracy using the Hamming loss is highest, and in Fig. 4 example images where performance drops compared to UO.

We note that other approaches obtain higher absolute scores (e.g. [12]), but they are computationally more involved and also include elaborate feature tuning and stacking. We restrict ourselves to a simple, efficient base method to show the desired effects, but expect similar improvements when the Pottics model is used in combination with more advanced approaches.

Training of the Pottics model takes about 3 hours with $T = 50$ on the 964 training images. ICM typically converges in 3 iterations, which in our implementation requires about 5 seconds per image. Most time is spent in generating the CPMC segments, about 5-10 minutes for a single image using the public implementation[4] of the authors of [5].

4.3 Effect of segment topics

In Fig. 6 we show the learned parameters of the Pottics model when using $T = 50$ topics per segment. High values correspond to less probable combinations, e.g. topic #15 has low values for *bus* (class 6) and *car* (7), since those are likely to appear together, while

[4] http://sminchisescu.ins.uni-bonn.de/code/cpmc/ [Version 1.0]

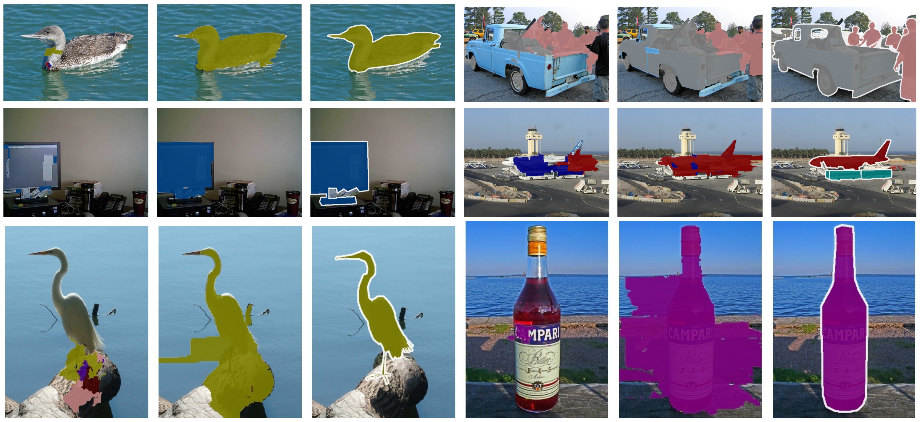

Fig. 3. Example segmentations for which the Pottics model improves most. From left to right: UO output, Pottics prediction, ground truth. In the second row, the monitor is almost perfectly recovered. In the airplane example the Pottics model corrects the boat class (blue) to be an airplane.

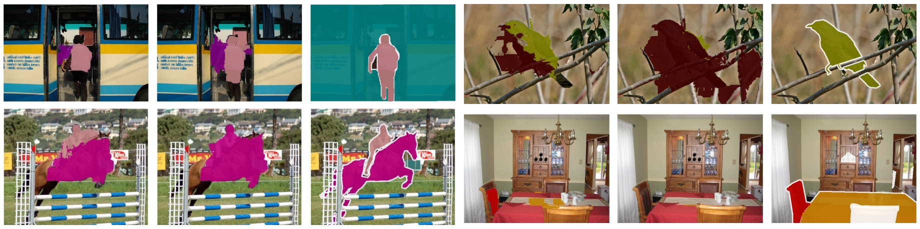

Fig. 4. Example segmentations for which the performance deteriorates. Same ordering as in Fig. 3. In the lower left, the person segment is wrongly joined with the horse segment. In the lower right example, the table is mistaken for a chair.

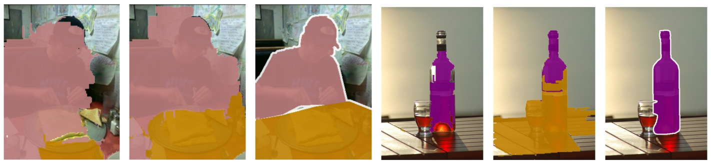

Fig. 5. Two examples for the topic impact of tables. From left to right: UO prediction, Pottics prediction, ground truth. Classes *person*, *table* and *bottle*.

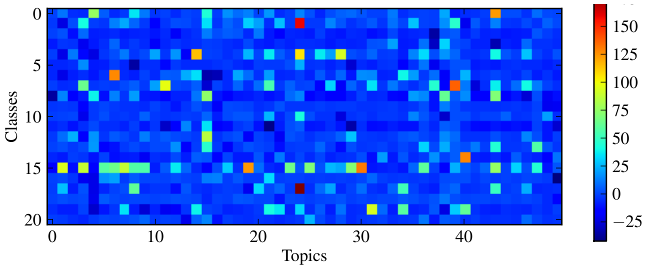

Fig. 6. Learned parameters w_p. The value of the unary parameter is $w_u = -156.6$. High values correspond to less probable combinations. The class ordering corresponds to the listing in Table 1.

Hamm. %	UO	RP	SP	Potts	1T	Pottics
background	96.1	96.8	97.1	95.0	**96.8**	92.5
plane	29.2	28.5	13.0	26.0	10.2	**45.9**
bicycle	0.7	0.6	0.0	0.6	0.1	**4.8**
bird	14.4	13.7	2.5	12.4	3.0	**35.6**
boat	17.0	16.8	4.9	15.7	4.8	**27.0**
bottle	16.2	15.6	2.3	12.8	3.9	**27.9**
bus	36.4	35.9	31.3	36.3	23.2	**59.5**
car	47.0	47.1	40.6	47.9	32.3	**62.5**
cat	49.6	50.1	43.3	50.2	**72.1**	70.9
chair	6.7	6.8	0.3	6.4	2.8	**8.7**
cow	11.5	11.1	2.2	9.0	2.6	**23.3**
table	6.5	5.5	1.0	5.3	2.9	**20.2**
dog	16.0	15.7	8.0	14.7	4.9	**16.0**
horse	19.4	19.1	3.9	16.2	6.5	**21.7**
motorbike	33.6	34.1	24.2	32.2	17.8	**42.2**
person	48.3	48.0	41.4	50.4	33.4	**51.6**
plant	11.5	10.7	9.0	10.5	5.9	**19.8**
sheep	**24.8**	25.1	10.4	21.6	11.3	23.4
sofa	**14.9**	14.7	6.0	13.9	7.9	14.4
train	33.1	33.0	20.4	31.1	19.0	**46.4**
tv	26.5	24.6	8.2	22.2	4.2	**33.8**
average	26.6	26.4	17.6	25.3	17.4	**35.6**
total	79.2	**79.7**	77.6	78.2	77.3	78.9

VOC %	UO	RP	SP	Potts	1T	Pottics
background	80.3	80.5	77.5	78.8	78.7	**80.8**
plane	27.6	27.4	12.8	22.4	10.1	**41.0**
bicycle	0.6	0.6	0.0	0.6	0.1	**3.9**
bird	11.9	11.9	2.3	10.7	2.8	**22.1**
boat	16.0	16.1	4.8	13.7	4.8	**25.3**
bottle	15.2	14.9	2.3	12.7	3.8	**24.2**
bus	33.0	33.1	29.0	32.2	22.4	**41.3**
car	43.3	44.2	37.3	43.2	31.4	**52.8**
cat	28.8	**30.4**	25.4	26.5	25.0	25.3
chair	5.2	5.5	3.4	5.4	2.5	**6.4**
cow	10.7	10.5	2.2	7.8	2.5	**20.2**
table	5.4	4.7	1.0	4.5	2.7	**12.5**
dog	12.2	**12.3**	7.3	11.9	4.5	11.5
horse	16.1	16.3	3.7	13.4	6.1	**18.6**
motorbike	28.4	29.4	20.9	26.9	16.5	**34.7**
person	34.6	35.7	33.2	35.2	28.4	**37.1**
plant	11.0	10.3	8.4	9.7	5.7	**16.2**
sheep	20.0	20.9	9.9	17.4	10.8	**21.0**
sofa	12.8	**12.9**	5.7	11.7	7.4	12.3
train	30.1	30.3	19.1	28.8	18.2	**39.9**
tv	23.2	22.0	8.0	20.9	4.2	**27.6**
total	22.2	22.4	14.8	20.7	13.8	**27.4**

Table 1. Performance on the VOC 2010 validation set: *(left)* accuracy using Hamming loss, *(right)* VOC accuracy. The different methods are described in the text.

at the same time making joint occurrence with classes such as *cat*, *table*, *dog*, *horse*a unlikely and thus down-weighting their score.

The class *table* is one that performs better under the Pottics model, with an increase of 7.1% in VOC accuracy. In Fig. 5 we show two examples of this class. The topic #21 is dominant, its value having the effect of placing higher scores on the labels *table*. Most probably this accounts for the effect of tables not being segmented well by the CPMC algorithm, which the Pottics model can address. In the right example of Fig. 5 a *table* segment emerges, since the topic has a positive effect on the two objects *bottle* and *table* appearing jointly. The *table* is not present in the ground truth annotation though, hence this example deteriorates performance.

5 Conclusion

We introduced a novel CRF model for semantic image segmentation that goes beyond Potts-like spatial smoothing using latent topics. The model is inspired by the success of methods that generate class-independent figure-ground proposal segmentations, which are used to define the graph structure between regions to be labeled and segments, which represent the latent topics. Training relied on a structured max-margin formulation with latent variables. We evaluated our model on the challenging VOC'10 dataset and found it to significantly improve performance over non-spatial as well as simple spatial baselines.

This paper focused on showing the effect of incorporating segmentation information into the prediction. There are many different ways on how the proposed Pottics model can be extended. First, the unary potential function of the regions could be jointly trained with the entire model. The segment variables could further be made image-dependent and equipped with unary factors. While we find that both the Hamming and VOC accuracy are positively correlated, prediction remains sub-optimal when the inference is done using the Hamming loss.

References

1. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: CVPR (2010)
2. Batra, D., Sukthankar, R., Chen, T.: Learning class-specific affinities for image labelling. In: CVPR (2008)
3. Carreira, J., Li, F., Sminchisescu, C.: Object Recognition by Sequential Figure-Ground Ranking. IJCV August (2011)
4. Carreira, J., Ion, A., Sminchisescu, C.: Image segmentation by discounted cumulative ranking on maximal cliques. arXiv CoRR abs/1009.4823 (2010)
5. Carreira, J., Sminchisescu, C.: Constrained parametric min-cuts for automatic object segmentation. In: CVPR. pp. 3241–3248 (2010)
6. Everingham, M., Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. IJCV 88(2), 303–338 (Sep 2009)
7. Fulkerson, B., Vedaldi, A., Soatto, S.: Class segmentation and object localization with superpixel neighborhoods. In: ICCV (2009)
8. Gonfaus, J., Boix, X., van de Weijer, J., Bagdanov, A., Serrat, J., Gonzalez, J.: Harmony potentials for joint classification and segmentation. In: CVPR (2010)
9. Ion, A., Carreira, J., Sminchisescu, C.: Image segmentation by figure-ground composition into maximal cliques. In: ICCV (2011)
10. Ion, A., Carreira, J., Sminchisescu, C.: Probabilistic joint image segmentation and labeling. In: NIPS (2011)
11. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. PAMI 28(10), 1568–1583 (2006)
12. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: NIPS (2011)
13. Ladicky, L., Russell, C., Kohli, P., Torr, P.H.S.: Associative hierarchical CRFs for object class image segmentation. In: ICCV. IEEE (Sep 2009)
14. Nowozin, S., Gehler, P., Lampert, C.: On parameter learning in CRF-based approaches to object class image segmentation. In: ECCV. Springer (2010)
15. Nowozin, S., Lampert, C.: Structured Learning and Prediction in Computer Vision. Foundations and Trends in Computer Graphics and Vision 6(3-4), 185–365 (2011)
16. Pantofaru, C., Schmid, C.: Object recognition by integrating multiple image segmentations. In: ECCV. Springer (2008)
17. Ren, X., Malik, J., Division, C.S.: Learning a classification model for segmentation. In: ICCV (2003)
18. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost for image understanding: Multiclass object recognition and segmentation by jointly modeling texture, layout, and context. In: IJCV (2007)
19. Tarlow, D., Zemel, R.: Big and tall: Large margin learning with high order losses. In: CVPR Ws. on Inference in Graphical Models with Structured Potentials (2011)
20. Vishwanathan, S.V.N., Schraudolph, N.N., Schmidt, M.W., Murphy, K.P.: Accelerated training of conditional random fields with stochastic gradient methods. In: ICML (2006)
21. Yu, C.N., Joachims, T.: Learning structural svms with latent variables. In: ICML (2009)