Higher Order Potentials in End-to-End Trainable Conditional Random Fields

Anurag Arnab* Sadeep Jayasumana* Shuai Zheng Philip Torr
University of Oxford, United Kingdom
{firstname.lastname}@eng.ox.ac.uk

Abstract

We tackle the problem of semantic segmentation using deep learning techniques. Most semantic segmentation systems include a Conditional Random Field (CRF) model to produce a structured output that is consistent with visual features of the image. With recent advances in deep learning, it is becoming increasingly common to perform CRF inference within a deep neural network to facilitate joint learning of the CRF with a pixel-wise Convolutional Neural Network (CNN) classifier.

While basic CRFs use only unary and pairwise potentials, it has been shown that the addition of higher order potentials defined on cliques with more than two nodes can result in a better segmentation outcome. In this paper, we show that two types of higher order potential, namely, object detection based potentials and superpixel based potentials, can be included in a CRF embedded within a deep network. We design these higher order potentials to allow inference with the efficient and differentiable mean field algorithm, making it possible to implement our CRF model as a stack of layers in a deep network. As a result, all parameters of our richer CRF model can be jointly learned with a CNN classifier during the end-to-end training of the entire network. We find significant improvement in the results with the introduction of these trainable higher order potentials.

1. Introduction

Semantic segmentation involves assigning a visual object class label to every pixel in an image, resulting in a segmentation with a semantic meaning for each segment. It can also be viewed as the task of recognizing and delineating objects in an image. While a strong pixel-level classifier is critical for obtaining high accuracy in this task, it is also important to enforce the consistency of the semantic segmentation output with visual features of the image. For example, segmentation boundaries should usually coincide with strong edges in the image, and small regions in the image with little color variation should have the same label.

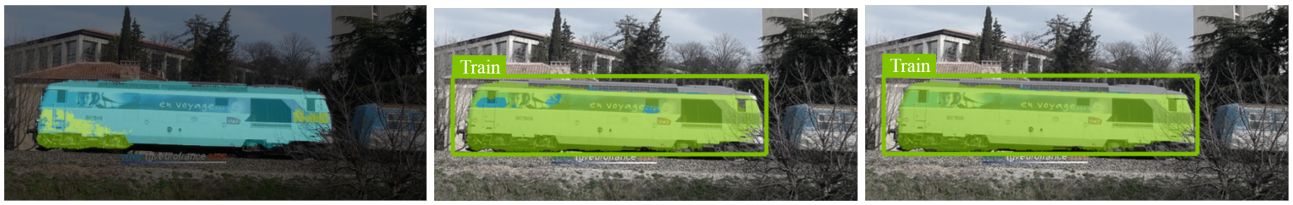
Recent advances in deep learning have enabled researchers to create stronger classifiers with automatically learned features within a Convolutional Neural Network (CNN) [23, 39, 30]. This has resulted in large improvements in semantic segmentation accuracy on widely used benchmarks such as PASCAL VOC [9]. CNN classifiers are now considered the standard choice for pixel-level classifiers used in semantic segmentation.

On the other hand, probabilistic graphical models have long been popular for structured prediction of labels, with constraints enforcing label consistency. Conditional Random Fields (CRFs) have been the most common framework, and various authors [24, 26, 42] have developed rich and expressive models based on higher order clique potentials to improve the semantic segmentation performance.

Whilst some deep learning methods showed impressive performance in semantic segmentation without incorporating graphical models [30, 16], the current methods achieving state-of-the-art performance [29, 45, 27, 4] have all incorporated graphical models into the deep learning framework in some form. However, we observe that the CRFs that have been incorporated into deep learning techniques are still rather rudimentary as they consist of only unary and pairwise potentials [45]. In this paper, we show that CRFs with carefully designed higher order potentials (potentials defined over cliques consisting of more than two nodes) can also be modelled as CNN layers when using mean field inference [20]. The advantage of performing CRF inference within a CNN is that it enables joint optimization of CNN classifier weights and CRF parameters during the end-to-end training of the complete system. Intuitively, the classifier and the graphical model learn to optimally co-operate with each other during the joint training.

We introduce two types of higher order potential into the CRF embedded in our deep network: object detection based potentials and superpixel based potentials. The primary idea of using object detection potentials is to use the outputs of an off-the-shelf object detector as additional semantic cues for finding the segmentation of an image. Intuitively, an object detector with a high recall can help the semantic segmentation algorithm by finding objects appear-

*Authors contributed equally



(a) Baseline [45] (b) Detection potentials (c) Detection and superpixel potentials

Figure 1: **Semantic segmentation with higher order CRF potentials.** (a) Segmentation with only usual unary and pairwise CRF potentials. (b) Improvements after adding object detection based higher order potentials (bounding box represents the output from an object detector). (c) Improvements after adding superpixel based higher order potentials as well.

ing in an image. As shown in Figure 1, our method is able to recover from poor segmentation unaries when we have a confident detector response. However, our method is robust to false positives identified by the object detector since CRF inference identifies and rejects false detections that do not agree with other types of energies present in the CRF. Furthermore, our CRF inference rescores the confidence output of an object detector, and we show that this in turn helps the object detector to improve its overall accuracy by rejecting false positives that are not consistent with the semantic segmentation of the image.

Superpixel based higher order potentials encourage label consistency over superpixels obtained by oversegmentation. This is motivated by the fact that regions defined by superpixels are likely to contain pixels from the same visual object. Once again, the formulation is robust to the violations of this assumption and errors in the initial superpixel generation step. In practice, we noted that superpixel based potentials are effective for getting rid of small regions of spurious labels that are inconsistent with the correct labels of their surrounding pixels (Figure 1c).

We evaluate our higher order potentials on the PASCAL VOC 2012 semantic segmentation benchmark, and Cityscapes dataset, to show significant improvements over CRFs that use only unary and pairwise potentials.

2. Related Work

Before deep learning became prominent, semantic segmentation was performed with dense hand-crafted features which were fed into a per-pixel or region classifier [38]. The individual predictions made by these classifiers were often noisy as they lacked global context, and were thus post-processed with a CRF to refine the results, making use of prior knowledge such as the fact that nearby pixels, as well as pixels of the similar appearance, are likely to share the same class label.

The CRF model of [38] initially contained only unary and pairwise terms in an 8-neighbourhood, which Kohli *et al*. showed can result in shrinkage bias [19]. However, numerous improvements to this model were subsequently proposed including: densely connected pairwise potentials facilitating interactions between all pairs of image pixels [21], formulating higher order potentials defined over cliques larger than two nodes [19, 24] in order to capture more context, modelling co-occurrence of object classes [25, 33, 14], and utilizing the results of object detectors [26, 44].

Recent advances in deep learning have allowed us to replace hand-crafted features with features learned specifically for semantic segmentation. The strength of these representations was illustrated by [30] who achieved significant improvements over previous hand-crafted methods without using any CRF post-processing. The authors of [4] showed further improvements in segmentation performance, which was obtained by post-processing the results of a CNN with a CRF. More recent works [45, 27, 37, 29] have taken this idea further by incorporating a CRF as layers within a deep network and then learning parameters of both the CRF and CNN together via backpropogation.

In terms of enhancements to conventional CRF models, Ladicky [26] proposed using an off-the-shelf object detector to provide additional cues in semantic segmentation. Unlike other approaches that refine a bounding-box detection to produce a segmentation [16, 43], this method used detector outputs as a soft constraint and can thus ignore errors produced by the object detector. Their formulation, however, used graph-cut inference, which was possible due to the absence of dense pairwise potentials.

We formulate the detection potential in a different manner to [26] so that it is amenable to mean field inference. Mean field permits inference with dense pairwise connections, which results in substantial accuracy improvements [21, 4, 45]. Furthermore, mean field updates related to our potential are differentiable and its parameters can thus be learned in our end-to-end trainable architecture. Object detectors have also been employed by [44] and [40], who also modelled variables which describe the degree to which an object hypothesis is accepted or not. This was used by [44] and [40] to rescore the original detection and thus improve overall detection performance. We employ similar techniques in our model.

On a separate track, [5] have used object detection ground truth to weakly supervise the training of deep neural

networks for semantic segmentation and thus make use of the fact that there is more training data for object detection than segmentation. Gould *et al*. [15] used semantic segmentation to propose regions to detect. And by enforcing consistency between segmented regions, object detections and object instances in a graphical model, the two tasks of detection and segmentation were performed jointly in a unified model. However, a greedy move-making algorithm was used for inference. Such non-differentiable algorithms have not yet been incorporated into deep learning frameworks.

We also note that while the semantic segmentation problem has mostly been formulated in terms of pixels [38, 30, 45], some have expressed it in terms of superpixels [2, 3, 10, 6]. Superpixels can capture more context than a single pixel and computational costs can also be reduced if one considers pairwise interactions between superpixels rather than individual pixels [44]. However, such superpixel representations assume that the segments share boundaries with objects in an image, which is not always true. As a result, several authors [24, 42] have employed higher order potentials defined over superpixels that encourage label consistency over regions, but do not strictly enforce it. This approach also allows multiple layers of superpixels, which do not necessarily form a hierarchy, to be integrated. Our formulation uses this kind of higher order potentials in an end-to-end trainable CNN.

Graphical models have been used with CNNs in other areas besides semantic segmentation, such as in pose-estimation [41], deformable part models [13], and group activity recognition [7]. However, the nature of models used in these works is substantially different to ours. Some early works that advocated gradient backpropagation through graphical model inference for parameter optimization include [35, 8, 22] and [17].

Our work differentiates from the above works since, to our knowledge, we are the first to propose and conduct a thorough experimental investigation of higher order potentials that are based on detection outputs and superpixel segmentation, in a CRF which is learned end-to-end in a deep network.

3. Conditional Random Fields

We now review conditional random fields used in semantic segmentation and introduce the notation used in the paper. Take an image $\mathbf{I}$ with N pixels, indexed $1, 2, \ldots, N$. In semantic segmentation, we attempt to assign every pixel a label from a predefined set of labels $\mathcal{L} = \{l_1, l_2, \ldots, l_L\}$. Define a set of random variables $X_1, X_2, \ldots, X_N$, one for each pixel, where each $X_i \in \mathcal{L}$. Let $\mathbf{X} = [X_1\ X_2\ \ldots\ X_N]^T$. Any particular assignment $\mathbf{x}$ to $\mathbf{X}$ is thus a solution to the semantic segmentation problem.

We use notations $\{\mathbf{V}\}$, and $\mathbf{V}^{(i)}$ to represent the set of elements of a vector $\mathbf{V}$, and the i^{th} element of $\mathbf{V}$, respectively. Given a graph G where the vertices are from $\{\mathbf{X}\}$ and the edges define connections among these variables, the pair $(\mathbf{I}, \mathbf{X})$ is modelled as a CRF characterised by $\Pr(\mathbf{X} = \mathbf{x}|\mathbf{I}) = (1/Z(\mathbf{I}))\exp(-E(\mathbf{x}|\mathbf{I}))$, where $E(\mathbf{x}|\mathbf{I})$ is the *energy* of the assignment $\mathbf{x}$ and $Z(\mathbf{I})$ is the normalization factor known as the partition function. We drop the conditioning on $\mathbf{I}$ hereafter to keep the notation uncluttered. The energy $E(\mathbf{x})$ of an assignment is defined using the set of cliques $\mathcal{C}$ in the graph G. More specifically,

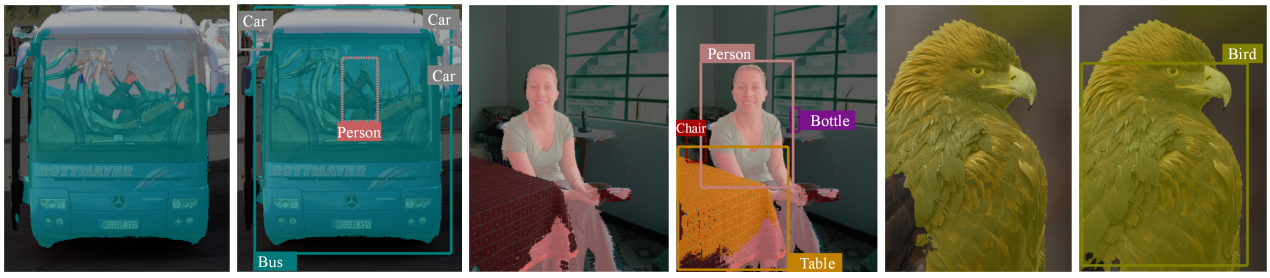
$$E(\mathbf{x}) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c), \tag{1}$$

where $\mathbf{x}_c$ is a vector formed by selecting elements of $\mathbf{x}$ that correspond to random variables belonging to the clique c, and $\psi_c(.)$ is the cost function for the clique c. Note that the $\psi_c(.)$ function usually uses prior knowledge about a good segmentation, as well as information from the image, the observation the CRF is conditioned on.

Minimizing the energy in Eq. (1) yields the maximum a posteriori (MAP) labelling of the image, *i.e*., the most probable label assignment given the observation (image). When dense pairwise potentials are used in the CRF to obtain higher accuracy, exact inference is impracticable, and one has to resort to an approximate inference method such as mean field inference [21]. Mean field inference is particularly appealing in a deep learning setting since it is possible to perform mean field inference inside a deep network using a Recurrent Neural Network [45].

4. CRF with Higher Order Potentials

Many CRF models that have been incorporated into deep learning frameworks [4, 45] have so far used only unary and pairwise potentials. However, potentials defined on higher order cliques have been shown to be useful in earlier works such as [19, 42]. The key contribution of this paper is to show that a number of explicit higher order potentials can be added to CRFs to improve image segmentation, while staying compatible with deep learning techniques. We formulate these higher order potentials in a manner that mean field inference can still be used to solve the CRF. Advantages of mean field inference are twofold: First, it enables efficient inference when using densely-connected pairwise potentials in a CRF. Multiple works, [21, 22, 4, 45] have shown that dense pairwise connections result in substantial accuracy improvements, particularly at image boundaries. Secondly, mean field inference can be performed within an end-to-end trainable deep network [45]. This is because mean field inference consists of a number of repeated, differentiable operations. In our formulations, we keep all mean field updates differentiable with respect to inputs as well as to the different CRF parameters introduced. This



(a) (b) (c) (d) (e) (f)

Figure 2: **Utility of object detections as another cue for semantic segmentation.** For every pair, segmentation on the left was produced with only unary and pairwise potentials. Detection based potentials were added to produce the result on the right. Note how we are able to improve our segmentations for the bus, table and bird over their respective baselines. Furthermore, our system is able to reject erroneous detections such as the person in (b) and the bottle and chair in (d). Images were taken from the Pascal VOC 2012 reduced validation set. Baseline results were produced using the public code and model of [45].

design enables us to use backpropagation to automatically learn all the parameters in the introduced potentials.

In this work, we use two types of higher order potential, one based on object detections and the other based on superpixels. These are discussed in detail in Section 4.1 and Section 4.2 respectively. Our complete CRF model is represented by

$$E(\mathbf{x}) = \sum_i \psi_i^U(x_i) + \sum_{i<j} \psi_{ij}^P(x_i, x_j) + \sum_d \psi_d^{\text{Det}}(\mathbf{x}_d) + \sum_s \psi_s^{\text{SP}}(\mathbf{x}_s), \quad (2)$$

where the first two terms $\psi_i^U(.)$ and $\psi_{ij}^P(.,.)$ are the usual unary and densely-connected pairwise energies [21] and the last two terms are the newly introduced higher order energies. Energies from the object detection take the form $\psi_d^{\text{Det}}(\mathbf{x}_d)$, where vector $\mathbf{x}_d$ is formed by elements of $\mathbf{x}$ that correspond to the foreground pixels of the d^{th} object detection. Superpixel label consistency based energies take the form $\psi_s^{\text{SP}}(\mathbf{x}_s)$, where $\mathbf{x}_s$ is formed by elements of $\mathbf{x}$ that correspond to the pixels belonging to the s^{th} superpixel.

4.1. Object Detection Based Potentials

Semantic segmentation errors can be classified into two broad categories [5]: recognition errors and boundary errors. Boundary errors occur when semantic labels are incorrect at the edges of objects, and it has been shown that densely connected CRFs with appearance consistency terms are effective at combating this problem [21]. On the other hand, recognition errors occur when object categories are recognised incorrectly or not at all. A CRF with only unary and pairwise potentials cannot effectively correct these errors since they are caused by poor unary classification. However, we propose that a state-of-the-art object detector [12, 34] that is capable of recognising and localising objects, can provide important information in this situation and help reduce the recognition error, as shown in Figure 2.

A key challenge in feeding in object detection potentials to semantic segmentation are false detections. A naïve approach of adding an object detector's output to a CRF formulated to solve the problem of semantic segmentation would confuse the CRF due to the presence of the false positives in the detector's output. Therefore, a robust formulation, which can automatically reject object detection false positives when they do not agree with other type of potentials in the CRF, is desired. Furthermore, since we are aiming for an end-to-end trainable CRF which can be incorporated into a deep neural network, the energy formulation should permit a fully differentiable inference procedure. We now propose a formulation which has both of these desired properties.

Assume that we have D object detections for a given image, and that the d^{th} detection is of the form (l_d, s_d, F_d), where $l_d \in \mathcal{L}$ is the class label of the detected object, s_d is the confidence score of the detection, and $F_d \subseteq \{1, 2, \ldots, N\}$, is the set of indices of the pixels that belong to the foreground of the detection. The foreground within a detection bounding box could be obtained using a foreground/background segmentation method such as GrabCut [36], and represents a very rough segmentation of the detected object. Through our detection potentials, we would like to encourage the set of pixels represented by F_d, to take the label l_d. However, this should not be a hard constraint since the foreground segmentation could be inaccurate and the whole detection itself could be a false detection. We therefore seek a soft-constraint that assigns a penalty if a pixel in F_d takes a label other than l_d. Moreover, if other energies used in the CRF strongly suggest that many pixels in F_d do not belong to the class l_d, the detection d should be identified as invalid.

An approach to accomplish this is described in [26] and [44]. However, in both cases, dense pairwise connections were absent and different inference methods were used. In contrast, we would like to use the mean field approximation to enable efficient inference with dense pairwise connections [21]. Furthermore, as shown in [45], mean field inference steps are fully differentiable and can be represented as layers within a neural network that can be trained end-to-end. We therefore use a detection potential formulation quite different to the ones used in [26] and [44].

In our formulation, as done in [26, 44], we first introduce latent binary random variables $Y_1, Y_2, \dots Y_D$, one for each detection. The interpretation for the random variable Y_d that corresponds to the d^{th} detection is as follows: If the d^{th} detection has been found to be valid after inference, Y_d will be set to 1, it will be 0 otherwise. Mean field inference probabilistically decides the final value of Y_d. Note that, through this formulation, we can account for the fact the initial detection could have been a false positive: some of the detections obtained from the object detector may be identified to be false following CRF inference.

All Y_d variables are added to the CRF which previously contained only X_i variables. Let each $(\mathbf{X}_d, Y_d)$, where $\{\mathbf{X}_d\} = \{X_i \in \{\mathbf{X}\} | i \in F_d\}$, form a clique c_d in the CRF. We define the detection-based higher order energy associated with a particular assignment $(\mathbf{x}_d, y_d)$ to the clique $(\mathbf{X}_d, Y_d)$ as follows:

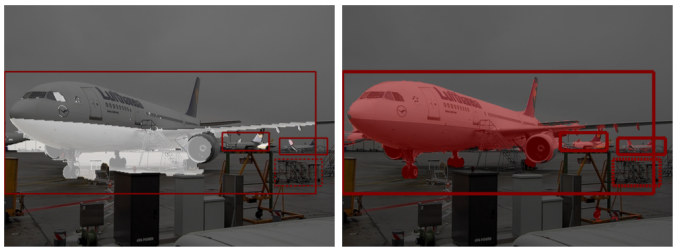
$$\psi_d^{\text{Det}}(\mathbf{X}_d = \mathbf{x}_d, Y_d = y_d) = \begin{cases} w_{\text{Det}} \frac{s_d}{n_d} \sum_{i=1}^{n_d} [x_d^{(i)} = l_d] & \text{if } y_d = 0, \\ w_{\text{Det}} \frac{s_d}{n_d} \sum_{i=1}^{n_d} [x_d^{(i)} \neq l_d] & \text{if } y_d = 1, \end{cases} \tag{3}$$

where $n_d = |F_d|$ is the number of foreground pixels in the d^{th} detection, $x_d^{(i)}$ is the i^{th} element of the vector $\mathbf{x}_d$, w_{Det} is a learnable weight parameter, and $[\,.\,]$ is the Iverson bracket. Note that this potential encourages $X_d^{(i)}$s to take the value l_d when Y_d is 1, and at the same time encourages Y_d to be 0 when many $X_d^{(i)}$s do not take l_d. In other words, it enforces the consistency among $X_d^{(i)}$s and Y_d.

An important property of the above definition of $\psi_d^{\text{Det}}(.)$ is that it can be simplified as a sum of pairwise potentials between Y_d and each $X_d^{(i)}$ for $i = 1, 2, \dots, n_d$. That is,

$$\psi_d^{\text{Det}}(\mathbf{X}_d = \mathbf{x}_d, Y_d = y_d) = \sum_{i=1}^{n_d} f_d(x_d^{(i)}, y_d), \text{ where,}$$

$$f_d(x_d^{(i)}, y_d) = \begin{cases} w_{\text{Det}} \frac{s_d}{n_d} [x_d^{(i)} = l_d] & \text{if } y_d = 0, \\ w_{\text{Det}} \frac{s_d}{n_d} [x_d^{(i)} \neq l_d] & \text{if } y_d = 1. \end{cases} \tag{4}$$



(a) (b)

Figure 3: **Robustness to imperfect foreground segmentation.** (a) Detected objects, as well as their putative foreground segmentations obtained using GrabCut. (b) The output using detection potentials. Incorrect parts of the foreground segmentation of the big aeroplane have mostly been ignored by CRF inference as they did not agree with the other energy terms.

We make use of this simplification in Section 5 when deriving the mean field updates associated with this potential.

For the latent Y variables, in addition to the joint potentials with X variables, described in Eq. (3) and (4), we also include unary potentials, which are initialized from the score s_d of the object detection. The underlying idea is that if the object detector detects an object with high confidence, the CRF in turn starts with a high initial confidence about the validity of that detection. This confidence can, of course, change during the CRF inference depending on other information (*e.g.* segmentation unary potentials) available to the CRF.

An example of an input image with multiple detections and GrabCut foreground masks is shown in Figure 3. Note how false detections are ignored and erroneous parts of the foreground mask are ignored.

4.2. Superpixel Based Potentials

The next type of higher order potential we use is based on the idea that superpixels obtained from oversegmentation [11, 1] quite often contain pixels from the same visual object. It is therefore natural to encourage pixels inside a superpixel to have the same semantic label. Once again, this should not be a hard constraint in order to keep the algorithm robust to initial superpixel segmentation errors and to violations of this key assumption.

We use two types of energies in the CRF to encourage superpixel consistency in semantic segmentation. Firstly, we use the P^n-Potts model type energy [42, 18], which is described by,

$$\psi_s^{\text{SP}}(\mathbf{X}_s = \mathbf{x}_s) = \begin{cases} w_{\text{Low}}(l) & \text{if all } x_s^{(i)} = l, \\ w_{\text{High}} & \text{otherwise,} \end{cases} \tag{5}$$

where $w_{\text{Low}}(l) < w_{\text{High}}$ for all l, and $\{\mathbf{X}_s\} \subset \{\mathbf{X}\}$ is

(a) (b) (c)

Figure 4: **Segmentation enhancement from superpixel based potentials.** (a) The output of our system without any superpixel potentials. (b) Superpixels obtained from the image using the method of [11]. Only one "layer" of superpixels is shown. In practice, we used four. (c) The output using superpixel potentials. The result has improved as we encourage consistency over superpixel regions. This removes some of the spurious noise that was present previously.

a clique defined by a superpixel. The primary idea is that assigning different labels to pixels in the same superpixel incurs a higher cost, whereas one can get away with a lower cost if the labelling is consistent throughout the superpixel. Costs $w_{\text{Low}}(l)$ and w_{High} are learnable during the end-to-end training.

Secondly, to make this potential stronger, we average initial unary potentials from the classifier (the CNN in our case), across all pixels in the superpixel and use the average as an additional unary potential for those pixels. During experiments, we observed that superpixel based higher order energy helps in getting rid of small spurious regions of wrong labels in the segmentation output, as shown in Fig. 4.

5. Mean Field Updates and Their Differentials

In this section we discuss the mean field updates for the higher order potentials discussed in the previous section. We also show that these update operations are differentiable with respect to the $Q_i(X_i)$ distribution inputs at each iteration, as well as the parameters used in the formulation of higher order potentials. The implication is that it is possible to use our CRF model in the CRF-RNN setting proposed in [45], to build an end-to-end trainable deep network for pixel-wise label prediction with CRF inference embedded.

Take a CRF with random variables $V_1, V_2, \ldots, V_N$ and a set of cliques $\mathcal{C}$, which includes unary, pairwise and higher order cliques. Mean field inference approximates the joint distribution $\Pr(\mathbf{V} = \mathbf{v})$ with the product of marginals $\prod_i Q(V_i = v_i)$. We use $Q(\mathbf{V}_c = \mathbf{v}_c)$ to denote the marginal probability mass for a subset $\{\mathbf{V}_c\}$ of these variables. Where there is no ambiguity, we use the short-hand notation $Q(\mathbf{v}_c)$ to represent $Q(\mathbf{V}_c = \mathbf{v}_c)$. General mean field updates of such a CRF take the form [20, 42]

$$Q^{t+1}(V_i = v) = \frac{1}{Z_i} \exp\left(-\sum_{c \in \mathcal{C}} \sum_{\{\mathbf{v}_c | v_i = v\}} Q^t(\mathbf{v}_{c-i})\, \psi_c(\mathbf{v}_c)\right), \quad (6)$$

where Q^t is the marginal after the t^{th} iteration, $\mathbf{v}_c$ an assignment to all variables in clique c, $\mathbf{v}_{c-i}$ an assignment to all variables in c except for V_i, $\psi_c(\mathbf{v}_c)$ is the cost of assigning $\mathbf{v}_c$ to the clique c, and Z_i is the normalization constant that makes $Q(V_i = v)$ a probability mass function after the update.

5.1. Updates from Detection Based Potentials

Following Eq. (4) above, we now use Eq. (6) to derive the mean field updates related to ψ_d^{Det}. The contribution from ψ_d^{Det} to the update of $Q(X_d^{(i)} = l)$ takes the form

$$\sum_{\{(\mathbf{x}_d, y_d) | x_d^{(i)} = l\}} Q(\mathbf{x}_{d-i}, y_d)\, \psi_d^{\text{Det}}(\mathbf{x}_d, y_d) = \begin{cases} w_{\text{Det}}\, \frac{s_d}{n_d}\, Q(Y_d = 0) & \text{if } l = l_d, \\ w_{\text{Det}}\, \frac{s_d}{n_d}\, Q(Y_d = 1) & \text{otherwise,} \end{cases} \quad (7)$$

where $\mathbf{x}_{d-i}$ is an assignment to $\mathbf{X}_d$ with the i^{th} element deleted. Using the same equations, we derive the contribution from the energy ψ_d^{Det} to the update of $Q(Y_d = b)$ to take the form

$$\sum_{\{(\mathbf{x}_d, y_d) | y_d = b\}} Q(\mathbf{x}_d)\, \psi_d^{\text{Det}}(\mathbf{x}_d, y_d) = \begin{cases} w_{\text{Det}}\, \frac{s_d}{n_d}\, \sum_{i=1}^{n_d} Q(X_d^{(i)} = l_d) & \text{if } b = 0, \\ w_{\text{Det}}\, \frac{s_d}{n_d}\, \sum_{i=1}^{n_d} (1 - Q(X_d^{(i)} = l_d)) & \text{otherwise.} \end{cases} \quad (8)$$

It is possible to increase the number of parameters in $\psi_d^{\mathrm{Det}}(.)$. Since we use backpropagation to learn these parameters automatically during end-to-end training, it is desirable to have a high number of parameters to increase the flexibility of the model. Following this idea, we made the weight w_{Det} class specific, that is, a function $w_{\mathrm{Det}}(l_d)$ is used instead of w_{Det} in Eqs. (3), (7) and (8). The underlying assumption is that detector outputs can be very helpful for certain classes, while being not so useful for classes that the detector performs poorly on, or classes for which foreground segmentation is often inaccurate.

Note that due to the presence of detection potentials in the CRF, error differentials calculated with respect to the X variable unary potentials and pairwise parameters will no longer be valid in the forms described in [45]. However, since all operations involved in the mean field updates described in Eq. (7) and (8) are differentiable, it is possible to compute the error differentials with respect to the unary potentials of both the X and Y variables, and with respect to the class-specific detection potential weights $w_{\mathrm{Det}}(l)$.

5.2. Updates for Superpixel Based Potentials

The contribution from the P^n-Potts type potential to the mean field update of $Q(X_i = l)$, where pixel i is in the superpixel clique s, was derived in [42] and takes the form

$$\sum_{\{\mathbf{x_s}|x_s^{(i)}=l\}} Q(\mathbf{x}_{s-i})\, \psi_s^{\mathrm{SP}}(\mathbf{x}_s) = w_{\mathrm{Low}}(l) \prod_{j\in s, j\neq i} Q(X_j = l) + w_{\mathrm{High}} \left(1 - \prod_{j\in s, j\neq i} Q(X_j = l)\right). \quad (9)$$

Note that the mean field update operation in Eq. (9) is differentiable with respect to the parameters $w_{\mathrm{Low}}(l)$ and w_{High}. Therefore, it is possible to calculate error differentials with respect to these parameters and optimize them via backpropagation. Furthermore, since the update is differentiable with respect to $Q(X)$ values, we can pass the differentials back to the previous iterations and layers, enabling backpropagation based optimization in those stages as well.

Mean field updates for the additional unary type superpixel-based potentials and their differentials take the same form as the updates and differentials for traditional unary terms [45].

6. Experiments

We evaluate our new CRF formulation on two different datasets using the CRF-RNN network [45] as the main baseline. The present work essentially introduces a richer CRF model to [45]'s deep network, in place of the limited CRF that has only unary and pairwise potentials. This is the rationale for using the CRF-RNN network as our main baseline.

6.1. Experimental Setup

Our deep network consists of two conceptually different, but jointly trained, stages. The first stage is formed by the FCN-8s network of [30], which is initialized with the Imagenet-trained VGG-16 network [39], and fine-tuned with Pascal VOC data [45]. This part of the network is used to obtain segmentation unaries for our CRF.

The output of this first stage is fed into the second stage, which is the CRF inference network. It is implemented using the mean field update operations and their differentials described in Section 5. We use five levels of recurrence to mimic five iterations of mean field inference during training, and increased it to 10 during testing. Our CRF network takes two more inputs in addition to the segmentation unaries obtained from the FCN-8s network: data from the object detector and superpixel oversegmentations of the image.

We used the publicly available code and model of the Faster RCNN [34] object detector. The fully automated version of GrabCut [36], was then used to obtain foregrounds from detection bounding boxes. These choices were made after conducting preliminary experiments with alternative methods for detection and foreground segmentation.

We used four levels of superpixel oversegmentations, with increasing superpixel size to define the cliques used in this potential. Four levels of oversegmentations were used since performance on the PASCAL VOC validation set stopped increasing after this number. We used the superpixel method of [11] since they were shown to adhere to object boundaries the best [1], but our method generalises to any oversegmentation algorithm.

During training, the full network was trained end-to-end, optimizing the weights of the CNN classifier (FCN-8s), and the CRF parameters jointly. We initialized our network using the publicly available weights of [45], and then trained with loss normalization turned off, a learning rate of 10^{-10}, and full-image batches. A low learning rate was used because the loss normalization was turned off.

6.2. Pascal VOC 2012 Dataset

To keep our results comparable with the baseline [45], we fine-tune on the Pascal VOC 2012 extended train set used in [45] and report results on the reduced validation set and the test set.

Since the allowed number of test submission to the evaluation server is limited, and Pascal VOC guidelines discourage the use of the test set for ablation studies. We use the same reduced validation set used in [45] to evaluate the improvements gained by the addition of each higher order potential and report test set results only for our best method on the validation set.

As shown in Table 1, adding either detection potentials, or superpixel based potentials results in an accuracy increase over our baseline, which only has unary and pairwise

Table 1: Mean IoU accuracy on the PASCAL VOC 2012

Method	Reduced val. set	Test set
Baseline (unary + pairwise) [45]	72.9	74.7
Superpixels	73.6	–
Detections	74.4	–
Detections and superpixels	75.1	75.8
DPN [29]	–	77.5
BoxSup [5]	–	75.2
DeepLab [4]	–	73.9
Message learning[28]	–	73.4
DeconvNet [32]	–	72.5
Piecewise [27]	–	70.7
FCN-8s [30]	–	62.2

potentials in the CRF. The best accuracy is obtained when both detection potentials and superpixels are incorporated into our model. We test the performance of only adding detection potentials, and only adding superpixel potentials, on the reduced validation set since the VOC evaluation server limits the number of test set submissions that can be made.

Our method outperforms other state-of-the-art techniques barring the DPN method of [29]. A possible reason for this difference is that the CNN part of the DPN network includes significant modifications to the VGG-16 network to improve its performance in semantic segmentation, whereas we use the VGG-16 network used in [30] and [45].

Rescoring of detections

As mentioned in Section 4.1, the unary potentials of the latent Y detection variables in our CRF are obtained from the confidence score of the object detector, and are then updated during mean-field inference. We view the final value of the Y variables after inference as the rescored or calibrated confidence value of the object detector.

We performed semantic segmentation on the images in the test set of the VOC 2012 detection challenge using initial bounding boxes from Faster R-CNN [34]. Although our network does not change the bounding box predictions of the detector, it does adjust the confidence scores. As shown in Table 2, we observe a slight improvement of 0.25% in the mean average precision when using our recalibrated scores.

This suggests that our CRF inference is able to evaluate object detection inputs in light of other potentials (unary, pairwise, and superpixels). Inference increases the relative score of detections which agree with the segmentation, and decreases the score of detections that do not agree with other energies in the CRF. Figures 2b) and d) show examples of false positive detection that have been ignored and correct detections that have been used to refine our segmentation. Note that we used the publicly available version (code and model) of Faster R-CNN, which is not same as the model they obtain best performance with.

Table 2: Comparison between the adjust detection scores as a result of CRF inference and original detection scores

	Faster RCNN	Faster RCNN with rescored confidences
Mean Average Precision (%)	64.34	64.59

6.3. Pascal Context

Table 3 shows the results of our method on the recently released Pascal Context dataset [31]. This dataset augments the Pascal VOC dataset with annotations for the whole scene. As a result, there are 59 categories as opposed to the 20 in the VOC dataset. Many of the new labels are "stuff" classes such as "grass" and "sky". Our object detectors are therefore only trained for 20 of the 59 labels in this dataset. Nevertheless, we are able to show an improvement of approximately 0.8% over the previous state-of-the-art [5], and 2% over our baseline [45], since we also make use of superpixel-based potentials. We trained on the provided training set and evaluated our method on the validation set.

7. Conclusion

We presented a CRF model with two types of higher order potential to tackle the semantic segmentation problem. The first potential is based on the intuitive idea that object detection can provide useful cues for semantic segmentation. Our formulation of this potential is capable of automatically rejecting false object detections that do not agree at all with the semantic segmentation. Secondly, we used a potential that encourages superpixels to have consistent labelling. These two new potentials can co-exist with usual unary and pairwise potentials in the CRF.

Importantly, we showed that efficient mean field inference is still possible in the presence of the new higher order potentials and derived the explicit forms of mean field updates and their differentials. This enabled us to implement the new CRF model as a stack of CNN layers and to train it end-to-end in a unified deep network with the pixel-wise CNN classifier. We experimentally showed that the addition of higher order potentials results in a significant increase in semantic segmentation accuracy.

Table 3: Intersection over Union (IoU) results on Pascal Context validation set compared to other current methods. The results of O2P [3] were reported in the errata of [31].

Method	Ours	BoxSup [5]	CRF-as-RNN [45]	FCN-8s [30]	CFM [6]	O2P [3]
Mean IoU (%)	41.3	40.5	39.3	37.8	34.4	18.1

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *PAMI*, 34(11):2274–2282, 2012.

[2] D. Batra, R. Sukthankar, and T. Chen. Learning class-specific affinities for image labelling. In *CVPR*, pages 1–8. IEEE, 2008.

[3] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *ECCV*, pages 430–443. 2012.

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *ICLR*, 2015.

[5] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *ICCV*, 2015.

[6] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. *CVPR*, 2015.

[7] Z. Deng, M. Zhai, L. Chen, Y. Liu, S. Muralidharan, M. J. Roshtkhari, and G. Mori. Deep structured models for group activity recognition. In *BMVC*, 2015.

[8] J. Domke. Learning graphical model parameters with approximate marginal inference. *PAMI*, 2013.

[9] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010.

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

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

[12] R. Girshick. Fast r-cnn. In *ICCV*, 2015.

[13] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. In *CVPR*, 2015.

[14] J. M. Gonfaus, X. Boix, J. Van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *CVPR*, pages 3280–3287. IEEE, 2010.

[15] S. Gould, T. Gao, and D. Koller. Region-based segmentation and object detection. In *NIPS*, pages 655–663, 2009.

[16] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *ECCV*, pages 297–312. Springer, 2014.

[17] M. Kiefel and P. V. Gehler. Human pose estimation with fields of parts. In *ECCV*, pages 331–346. Springer, 2014.

[18] P. Kohli, M. P. Kumar, and P. H. Torr. P3 & beyond: Solving energies with higher order cliques. In *CVPR*, 2007.

[19] P. Kohli, P. H. Torr, et al. Robust higher order potentials for enforcing label consistency. *IJCV*, 82(3):302–324, 2009.

[20] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[21] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with Gaussian edge potentials. In *NIPS*, 2011.

[22] P. Krähenbühl and V. Koltun. Parameter learning and convergent inference for dense random fields. In *ICML*, 2013.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105. 2012.

[24] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Associative hierarchical crfs for object class image segmentation. In *ICCV*, pages 739–746, 2009.

[25] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*, pages 239–253. 2010.

[26] L. Ladický, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where and how many? combining object detectors and crfs. In *ECCV*, pages 424–437, 2010.

[27] G. Lin, C. Shen, I. Reid, et al. Efficient piecewise training of deep structured models for semantic segmentation. *arXiv preprint arXiv:1504.01013*, 2015.

[28] G. Lin, C. Shen, I. D. Reid, and A. van den Hengel. Deeply learning the messages in message passing inference. *arXiv preprint arXiv:1506.02108*, 2015.

[29] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *ICCV*, 2015.

[30] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[31] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, et al. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, pages 891–898. IEEE, 2014.

[32] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. *arXiv preprint arXiv:1505.04366*, 2015.

[33] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *ICCV*, pages 1–8, 2007.

[34] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[35] S. Ross, D. Munoz, M. Hebert, and J. A. Bagnell. Learning message-passing inference machines for structured prediction. In *CVPR*, 2011.

[36] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM TOG*, 2004.

[37] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015.

[38] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *IJCV*, 2009.

[39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[40] M. Sun, B.-s. Kim, P. Kohli, and S. Savarese. Relating things and stuff via objectproperty interactions. *PAMI*, 36(7):1370–1383, 2014.

[41] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *NIPS*, pages 1799–1807, 2014.

[42] V. Vineet, J. Warrell, and P. H. Torr. Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. *IJCV*, 2014.

[43] Y. Yang, S. Hallman, D. Ramanan, and C. C. Fowlkes. Layered object models for image segmentation. *PAMI*, 2012.

[44] J. Yao, S. Fidler, and R. Urtasun. Describing the Scene as a Whole: Joint Object Detection, Scene Classification and Semantic Segmentation. In *CVPR*, pages 702–709, 2012.

[45] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *ICCV*, 2015.