Semantic Segmentation with Boundary Neural Fields

Gedas Bertasius
University of Pennsylvania
gberta@seas.upenn.edu

Jianbo Shi
University of Pennsylvania
jshi@seas.upenn.edu

Lorenzo Torresani
Dartmouth College
lt@dartmouth.edu

Abstract

The state-of-the-art in semantic segmentation is currently represented by fully convolutional networks (FCNs). However, FCNs use large receptive fields and many pooling layers, both of which cause blurring and low spatial resolution in the deep layers. As a result FCNs tend to produce segmentations that are poorly localized around object boundaries. Prior work has attempted to address this issue in post-processing steps, for example using a color-based CRF on top of the FCN predictions. However, these approaches require additional parameters and low-level features that are difficult to tune and integrate into the original network architecture. Additionally, most CRFs use color-based pixel affinities, which are not well suited for semantic segmentation and lead to spatially disjoint predictions.

To overcome these problems, we introduce a Boundary Neural Field (BNF), which is a global energy model integrating FCN predictions with boundary cues. The boundary information is used to enhance semantic segment coherence and to improve object localization. Specifically, we first show that the convolutional filters of semantic FCNs provide good features for boundary detection. We then employ the predicted boundaries to define pairwise potentials in our energy. Finally, we show that our energy decomposes semantic segmentation into multiple binary problems, which can be relaxed for efficient global optimization. We report extensive experiments demonstrating that minimization of our global boundary-based energy yields results superior to prior globalization methods, both quantitatively as well as qualitatively.

1. Introduction

The recent introduction of fully convolutional networks (FCNs) [22] has led to significant quantitative improvements on the task of semantic segmentation. However, despite their empirical success, FCNs suffer from some limitations. Large receptive fields in the convolutional layers and the presence of pooling layers lead to blurring and segmentation predictions at a significantly lower resolution than the original image. As a result, their predicted segments tend to be blobby and lack fine object boundary details. We report in Fig. 1 some examples illustrating typical poor localization of objects in the outputs of FCNs.

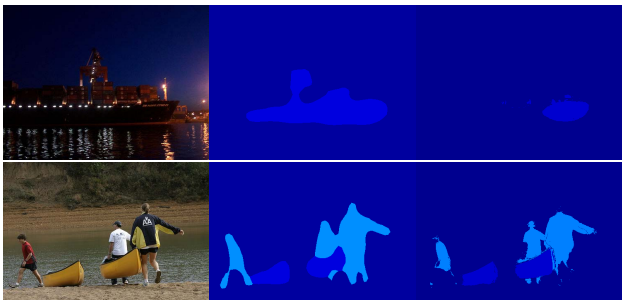

Figure 1: Examples illustrating shortcomings of prior semantic segmentation methods: the second column shows results obtained with a FCN [22], while the third column shows the output of a Dense-CRF applied to FCN predictions [19, 7]. Segments produced by FCN are blob-like and are poorly localized around object boundaries. Dense-CRF produces spatially disjoint object segments due to the use of a color-based pixel affinity function that is unable to measure semantic similarity between pixels.

Recently, Chen at al. [7] addressed this issue by applying a Dense-CRF post-processing step [19] on top of coarse FCN segmentations. However, such an approach introduces several problems of its own. First, the Dense-CRF adds new parameters that are difficult to tune and integrate into the original network architecture. Additionally, most methods based on CRFs or MRFs use low-level pixel affinity functions, such as those based on color. These low-level affinities often fail to capture semantic relationships between objects and lead to poor segmentation results (see last column in Fig. 1).

We propose to address these shortcomings by means of a Boundary Neural Field (BNF), an architecture that employs a single semantic segmentation FCN to predict semantic boundaries and then use them to produce semantic segmentation maps via a global optimization. We demonstrate that even though the semantic segmentation FCN has not been optimized to detect boundaries, it provides good

1063-6919/16 $31.00 © 2016 IEEE
DOI 10.1109/CVPR.2016.392

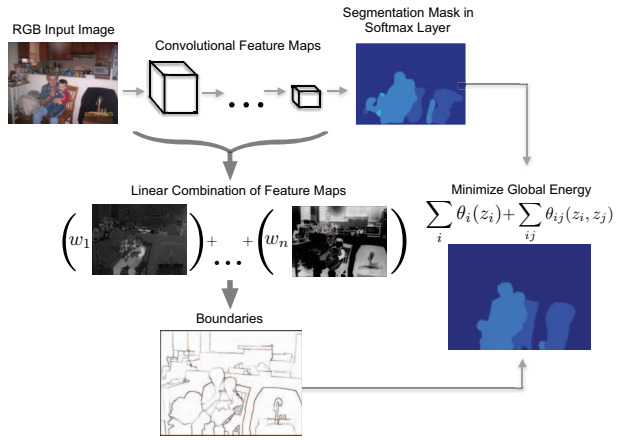

Figure 2: The architecture of our system (best viewed in color). We employ a semantic segmentation FCN [7] for two purposes: 1) to obtain semantic segmentation unaries for our global energy; 2) to compute object boundaries. Specifically, we define semantic boundaries as a linear combination of these feature maps (with a sigmoid function applied on top of the sum) and learn individual weights corresponding to each convolutional feature map. We integrate this boundary information in the form of pairwise potentials (pixel affinities) for our energy model.

features for boundary detection. Specifically, the contributions of our work are as follows:

- We show that semantic boundaries can be expressed as a linear combination of interpolated convolutional feature maps inside an FCN. We introduce a boundary detection method that exploits this intuition to predict object boundaries with accuracy superior to the state-the-of-art.
- We demonstrate that boundary-based pixel affinities are better suited for semantic segmentation than the commonly used color affinity functions.
- Finally, we introduce a new global energy that decomposes semantic segmentation into multiple binary problems and relaxes the integrality constraint. We show that minimizing our proposed energy yields better qualitative and quantitative results relative to traditional globalization models such as MRFs or CRFs.

2. Related Work

Boundary Detection. Spectral methods comprise one of the most prominent categories for boundary detection. In a typical spectral framework, one formulates a generalized eigenvalue system to solve a low-level pixel grouping problem. The resulting eigenvectors are then used to predict the boundaries. Some of the most notable approaches in this genre are MCG [2], gPb [1], PMI [17], and Normalized Cuts [29]. A weakness of spectral approaches is that they tend to be slow as they perform a global inference over the entire image.

To address this issue, recent approaches cast boundary detection as a classification problem and predict the boundaries in a local manner with high efficiency. The most notable examples in this genre include sketch tokens (ST) [20] and structured edges (SE) [9], which employ fast random forests. However, many of these methods are based on hand-constructed features, which are difficult to tune.

The issue of hand-constructed features have been recently addressed by several approaches based on deep learning, such as N^4 fields [11], DeepNet [18], DeepContour [27], DeepEdge [3], HFL [4] and HED [33]. All of these methods use CNNs in some way to predict the boundaries. Whereas DeepNet and DeepContour optimize ordinary CNNs to a boundary based optimization criterion from scratch, DeepEdge and HFL employ pretrained models to compute boundaries. The most recent of these methods is HED [33], which shows the benefit of deeply supervised learning for boundary detection.

In comparison to prior deep learning approaches, our method offers several contributions. First, we exploit the inherent relationship between boundary detection and semantic segmentation to predict semantic boundaries. Specifically, we show that even though the semantic FCN has not been explicitly trained to predict boundaries, the convolutional filters inside the FCN provide good features for boundary detection. Additionally, unlike DeepEdge [3] and HFL [4], our method does not require a pre-processing step to select candidate contour points, as we predict boundaries on all pixels in the image. We demonstrate that our approach allows us to achieve state-of-the-art boundary detection results according to both F-score and Average Precision metrics. Additionally, due to the semantic nature of our boundaries, we can successfully use them as pairwise potentials for semantic segmentation in order to improve object localization and recover fine structural details, typically lost by pure FCN-based approaches.

Semantic Segmentation. We can group most semantic segmentation methods into three broad categories. The first category can be described as "two-stage" approaches, where an image is first segmented and then each segment is classified as belonging to a certain object class. Some of the most notable methods that belong to this genre include [24, 6, 12, 14].

The primary weakness of the above methods is that they are unable to recover from errors made by the segmentation algorithm. Several recent papers [15, 10] address this issue by proposing to use deep per-pixel CNN features and then

classify each pixel as belonging to a certain class. While these approaches partially address the incorrect segmentation problem, they perform predictions independently on each pixel. This leads to extremely local predictions, where the relationships between pixels are not exploited in any way, and thus the resulting segmentations may be spatially disjoint.

The third and final group of semantic segmentation methods can be viewed as front-to-end schemes where segmentation maps are predicted directly from raw pixels without any intermediate steps. One of the earliest examples of such methods is the FCN introduced in [22]. This approach gave rise to a number of subsequent related approaches which have improved various aspects of the original semantic segmentation [7, 34, 8, 16, 21]. There have also been attempts at integrating the CRF mechanism into the network architecture [7, 34]. Finally, it has been shown that semantic segmentation can also be improved using additional training data in the form of bounding boxes [8].

Our BNF offers several contributions over prior work. To the best of our knowledge, we are the first to present a model that exploits the relationship between boundary detection and semantic segmentation **within a FCN framework**. We introduce pairwise pixel affinities computed from semantic boundaries inside an FCN, and use these boundaries to predict the segmentations in a global fashion. Unlike [21], which requires a large number of additional parameters to learn for the pairwise potentials, our global model only needs $\approx 5K$ extra parameters, which is about 3 orders of magnitudes less than the number of parameters in a typical deep convolutional network (e.g. VGG [30]). We empirically show that our proposed boundary-based affinities are better suited for semantic segmentation than color-based affinities. Additionally, unlike in [7, 34, 21], the solution to our proposed global energy can be obtained in closed-form, which makes global inference easier. Finally we demonstrate that our method produces better results than traditional globalization models such as CRFs or MRFs.

3. Boundary Neural Fields

In this section, we describe Boundary Neural Fields. Similarly to traditional globalization methods, Boundary Neural Fields are defined by an energy including unary and pairwise potentials. Minimization of the global energy yields the semantic segmentation. BNFs build both unary and pairwise potentials from the input RGB image and then combine them in a global manner. More precisely, the coarse segmentations predicted by a semantic FCN are used to define the unary potentials of our BNF. Next, we show that the convolutional feature maps of the FCN can be used to accurately predict semantic boundaries. These boundaries are then used to build pairwise pixel affinities, which are used as pairwise potentials by the BNF. Finally, we introduce a global energy function, which minimizes the energy corresponding to the unary and pairwise terms and improves the initial FCN segmentation. The detailed illustration of our architecture is presented in Figure 2. We now explain each of these steps in more detail.

3.1. FCN Unary Potentials

To predict semantic unary potentials we employ the DeepLab model [7], which is a fully convolutional adaptation of the VGG network [30]. The FCN consists of 16 convolutional layers and 3 fully convolutional layers. There are more recent FCN-based methods that have demonstrated even better semantic segmentation results [8, 34, 16, 21]. Although these more advanced architectures could be integrated into our framework to improve our unary potentials, in this work we focus on two aspects orthogonal to this prior work: 1) demonstrating that our boundary-based affinity function is better suited for semantic segmentation than the common color-based affinities and 2) showing that our proposed global energy achieves better qualitative and quantitative semantic segmentation results in comparison to prior globalization models.

3.2. Boundary Pairwise Potentials

In this section, we describe our approach for building pairwise pixel affinities using semantic boundaries. The basic idea behind our boundary detection approach is to express semantic boundaries as a function of convolutional feature maps inside the FCN. Due to the close relationship between the tasks of semantic segmentation and boundary detection, we hypothesize that convolutional feature maps from the semantic segmentation FCN can be employed as features for boundary detection.

3.2.1 Learning to Predict Semantic Boundaries.

We propose to express semantic boundaries as a linear combination of interpolated FCN feature maps with a non-linear function applied on top of this sum. We note that interpolation of feature maps has been successfully used in prior work (see e.g. [15]) in order to obtain dense pixel-level features from the low-resolution outputs of deep convolutional layers. Here we adopt interpolation to produce pixel-level boundary predictions. There are several advantages to our proposed formulation. First, because we express boundaries as a linear combination of feature maps, we only need to learn a small number of parameters, corresponding to the individual weight values of each feature map in the FCN. This amounts to $\approx 5K$ learning parameters, which is much smaller than the number of parameters in the entire network ($\approx 15M$). In comparison, DeepEdge [3] and HFL [4] need 17M and 6M *additional* parameters to predict boundaries.

Furthermore, expressing semantic boundaries as a linear

combination of FCN feature maps allows us to efficiently predict boundary probabilities for all pixels in the image (we resize the FCN feature maps to the original image dimensions). This eliminates the need to select candidate boundary points in a pre-processing stage, which was instead required in prior boundary detection work [3, 4].

Our boundary prediction pipeline can be described as follows. First we use use SBD segmentations [13] to optimize our FCN for semantic segmentation task. We then treat FCN convolutional maps as features for the boundary detection task and use the boundary annotations from BSDS 500 dataset [23] to learn the weights for each feature map. BSDS 500 dataset contains 200 training, 100 validation, 200 testing images, and ground truth annotations by 5 human labelers for each of these images.

To learn the weights corresponding to each convolutional feature map we first sample $80K$ points from the dataset. We define the target labels for each point as the fraction of human annotators agreeing on that point being a boundary. To fix the issue of label imbalance (there are many more non-boundaries than boundaries), we divide the label space into four quartiles, and select an equal number of samples for each quartile to balance the training dataset. Given these sampled points, we then define our features as the values in the interpolated convolutional feature maps corresponding to these points. To predict semantic boundaries we weigh each convolutional feature map by its weight, sum them up and apply a sigmoid function on top of it. We obtain the weights corresponding to each convolutional feature map by minimizing the cross-entropy loss using a stochastic batch gradient descent for 50 epochs. To obtain crisper boundaries at test-time we post-process the boundary probabilities using non-maximum suppression.

To give some intuition on how FCN feature maps contribute to boundary detection, in Fig. 3 we visualize the feature maps corresponding to the highest weight magnitudes. It is clear that many of these maps contain highly localized boundary information.

Boundary Detection Results Before discussing how boundary information is integrated in our energy for semantic segmentation, here we present experimental results assessing the accuracy of our boundary detection scheme. We tested our boundary detector on the BSDS500 dataset [23], which is the standard benchmark for boundary detection. The quality of the predicted boundaries is evaluated using three standard measures: fixed contour threshold (ODS), per-image best threshold (OIS), and average precision (AP).

In Table 1 we show that our algorithm outperforms all prior methods according to both F-score measures and the Average Precision metric. In Fig. 4, we also visualize our predicted boundaries. The second column shows the pixel-level softmax output computed from the linear combination of feature maps, while the third column depicts our final boundaries after applying a non-maximum suppression post-processing step.

Figure 3: An input image and convolutional feature maps corresponding to the largest weight magnitude values. Intuitively these are the feature maps that contribute most heavily to the task of boundary detection.

Method	ODS	OIS	AP
SCG [25]	0.739	0.758	0.773
SE [9]	0.746	0.767	0.803
MCG [2]	0.747	0.779	0.759
N^4-fields [11]	0.753	0.769	0.784
DeepEdge [3]	0.753	0.772	0.807
DeepContour [27]	0.756	0.773	0.797
HFL [4]	0.767	0.788	0.795
HED [33]	0.782	0.804	0.833
BNF	**0.788**	**0.807**	**0.851**

Table 1: Boundary detection results on BSDS500 benchmark. Our proposed method outperforms all prior algorithms according to all three evaluation metrics.

We note that our predicted boundaries achieve high-confidence predictions around objects. This is important as we employ these boundaries to improve semantic segmentation results, as discussed in the next subsection.

3.2.2 Constructing Pairwise Pixel Affinities.

We can use the predicted boundaries to build pairwise pixel affinities. Intuitively, we declare two pixels as similar (i.e., likely to belong to the same segment) if there is no boundary crossing the straight path between these two pixels. Conversely, two pixels are dissimilar if there is a boundary crossing their connecting path. The larger the boundary magnitude of the crossed path, the more dissimilar the two pixels should be, since a strong boundary is likely to mark the separation of two distinct segments. Similarly to [1], we encode this intuition with a following formulation:

$$w_{ij}^{sb} = \exp\left(\frac{-M_{ij}}{\sigma_{sb}}\right) \quad (1)$$

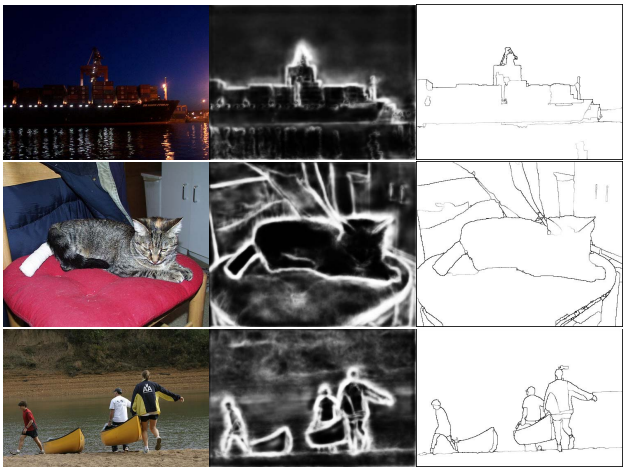

Figure 4: A figure illustrating our boundary detection results. In the second column, we visualize the raw probability output of our boundary detector. In the third column, we present the final boundary maps after non-maximum suppression. While most prior methods predict the boundaries where the sharpest change in color occurs, our method captures semantic object-level boundaries, which we subsequently use to aid semantic segmentation.

where M_{ij} denotes the maximum boundary value that crosses the straight line path between pixels i and j, σ_{sb} depicts the smoothing parameter and w_{ij}^{sb} denotes the semantic boundary-based affinity between pixels i and j.

Similarly, we want to exploit high-level object information in the network to define another type of pixel similarity. Specifically, we use object class probabilities from the *softmax* (SM) layer to achieve this goal. Intuitively, if pixels i and j have different hard segmentation labels from the *softmax* layer, we set their similarity (w_{ij}^{sm}) to 0. Otherwise, we compute their similarity using the following equation:

$$w_{ij}^{sm} = \exp\left(\frac{-D_{ij}}{\sigma_{sm}}\right) \tag{2}$$

where D_{ij} denotes the difference in *softmax* output values corresponding to the most likely object class for pixels i and j, and σ_{sm} is a smoothing parameter. Then we can write the final affinity measure as:

$$w_{ij} = \exp\left(w_{ij}^{sm}\right) w_{ij}^{sb} \tag{3}$$

We exponentiate the term corresponding to the object-level affinity because our boundary-based affinity may be too aggressive in declaring two pixels as dissimilar. To address this issue, we increase the importance of the object-level affinity in (3) using the exponential function. However, in the experimental results section, we demonstrate that most of the benefit from modeling pairwise potentials comes from w_{ij}^{sb} rather than w_{ij}^{sm}.

We then use this pairwise pixel affinity measure to build a global affinity matrix W that encodes relationships between pixels in the entire image. For a given pixel, we sample $\approx 10\%$ of points in the neighborhood of radius 20 around that pixel, and store the resulting affinities into W.

3.3. Global Inference

The last step in our proposed method is to combine semantic boundary information with the coarse segmentation from the FCN *softmax* layer to produce an improved segmentation. We do this by introducing a global energy function that utilizes the affinity matrix constructed in the previous section along with the segmentation from the FCN *softmax* layer. Using this energy, we perform a global inference to get segmentations that are well localized around the object boundaries and that are also spatially smooth.

Typical globalization models such as MRFs [31], CRFs [19] or Graph Cuts [5] produce a discrete label assignment for the segmentation problem by jointly modeling a multi-label distribution and solving a non-convex optimization. The common problem in doing so is that the optimization procedure may get stuck in local optima.

We introduce a new global energy function, which overcomes this issue and achieves better segmentation in comparison to prior globalization models. Similarly to prior globalization approaches, our goal is to minimize the energy corresponding to the sum of unary and pairwise potentials. However, the key difference in our approach comes from the relaxation of some of the constraints. Specifically, instead of modeling our problem as a joint multi-label distribution, we propose to decompose it into multiple binary problems, which can be solved concurrently. This decomposition can be viewed as assigning pixels to foreground and background labels for each of the different object classes. Additionally, we relax the integrality constraint. Both of these relaxations make our problem more manageable and allow us to formulate a global energy function that is differentiable, and has a closed form solution.

In [35], the authors introduce the idea of learning with global and local consistency in the context of semi-supervised problems. Inspired by this work, we incorporate some of these ideas in the context of semantic segmentation. Before defining our proposed global energy function, we introduce some relevant notation.

For the purpose of illustration, suppose that we only have two classes: foreground and background. Then we can denote an optimal continuous solution to such a segmentation problem with variable z^*. To denote similarity between pixels i and j we use w_{ij}. Then, d_i indicates the degree of a pixel i. In graph theory, the degree of a node denotes the number of edges incident to that node. Thus, we set the degree of a pixel to $d_i = \sum_{j=1}^{n} w_{ij}$ for all j except $i \neq j$. Finally, with f_i we denote an initial segmentation proba-

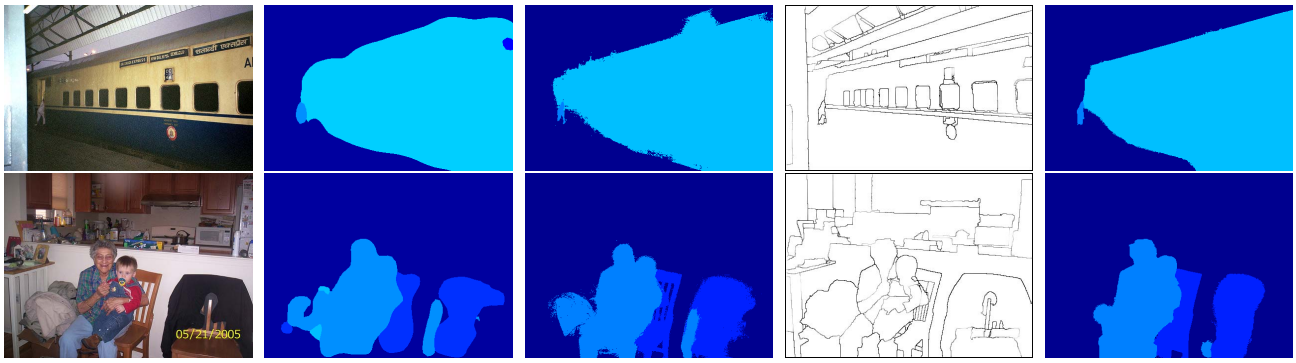

Figure 5: A figure illustrating semantic segmentation results. Images in columns two and three represent FCN *softmax* and Dense-CRF predictions, respectively. Note that all methods use the same FCN unary potentials. Additionally, observe that unlike FCN and Dense-CRF, our methods predicts segmentation that are both well localized around object boundaries and that are also spatially smooth.

bility, which in our case is obtained from the FCN *softmax* layer.

Using this notation, we can then formulate our global inference objective as:

$$z^* = \underset{z}{\operatorname{argmin}} \frac{\mu}{2} \sum_i d_i (z_i - \frac{f_i}{d_i})^2 + \frac{1}{2} \sum_{ij} w_{ij} (z_i - z_j)^2 \quad (4)$$

This energy consists of two different terms. Similar to the general globalization framework, our first term encodes the unary energy while the second term includes the pairwise energy. We now explain the intuition behind each of these terms. The unary term attempts to find a segmentation assignment (z_i) that deviates little from the initial candidate segmentation computed from the *softmax* layer (denoted by f_i). The z_i in the unary term is weighted by the degree d_i of the pixel in order to produce larger unary costs for pixels that have many similar pixels within the neighborhood. Instead, the pairwise term ensures that pixels that are similar should be assigned similar z values. To balance the energies of the two terms we introduce a parameter μ and set it to 0.025 throughout all our experiments.

We can also express the same global energy function in matrix notation:

$$\mathbf{z}^* = \underset{\mathbf{z}}{\operatorname{argmin}} \frac{\mu}{\mathbf{2}} \mathbf{D}(\mathbf{z} - \mathbf{D^{-1}f})^{\mathbf{T}} (\mathbf{z} - \mathbf{D^{-1}f}) + \frac{\mathbf{1}}{\mathbf{2}} \mathbf{z}^{\mathbf{T}} (\mathbf{D} - \mathbf{W}) \mathbf{z} \quad (5)$$

where $\mathbf{z}^*$ is a $n \times 1$ vector containing an optimal continuous assignment for all n pixels, $\mathbf{D}$ is a diagonal degree matrix, and $\mathbf{W}$ is the $n \times n$ pixel affinity matrix. Finally, $\mathbf{f}$ denotes a $n \times 1$ vector containing the probabilities from the *softmax* layer corresponding to a particular object class.

An advantage of our energy is that it is differentiable. If we denote the above energy as $E(z)$ then the derivative of this energy can be written as follows:

$$\frac{\partial E(z)}{\partial z} = \mu \mathbf{D}(\mathbf{z} - \mathbf{D^{-1}f}) + (\mathbf{D} - \mathbf{W})\mathbf{z} = \mathbf{0} \quad (6)$$

With simple algebraic manipulations we can then obtain a closed form solution to this optimization:

$$\mathbf{z}^* = (\mathbf{D} - \alpha \mathbf{W})^{-\mathbf{1}} \beta \mathbf{f} \quad (7)$$

where $\alpha = \frac{1}{1+\mu}$ and $\beta = \frac{\mu}{1+\mu}$. In the general case where we have k object classes we can write the solution as:

$$\mathbf{Z}^* = (\mathbf{D} - \alpha \mathbf{W})^{-\mathbf{1}} \beta \mathbf{F} \quad (8)$$

where $\mathbf{Z}$ now depicts a $n \times k$ matrix containing assignments for all k object classes, while $\mathbf{F}$ denotes $n \times k$ matrix with object class probabilities from *softmax* layer. Due to the large size of $\mathbf{D} - \alpha \mathbf{W}$ it is impractical to invert it. However, if we consider an image as a graph where each pixel denotes a vertex in the graph, we can observe that the term $\mathbf{D} - \mathbf{W}$ in our optimization is equivalent to a Laplacian matrix of such graph. Since we know that a Laplacian matrix is positive semi-definite, we can use the preconditioned conjugate gradient method [28] to solve the system in Eq. (9). Alternatively, because our defined global energy in Eq. (5) is differentiable, we can efficiently solve this optimization problem using stochastic gradient descent. We choose the former option and solve the following system:

$$(\mathbf{D} - \alpha \mathbf{W})\mathbf{z}^* = \beta \mathbf{f} \quad (9)$$

To obtain the final discrete segmentation, for each pixel we assign the object class that corresponds to the largest column value in the row of $\mathbf{Z}$ (note that each row in $\mathbf{Z}$ represents a single pixel in the image, and each column in $\mathbf{Z}$

Metric	Inference Method	RGB Affinity	**BNF Affinity**
PP-IOU	Belief Propagation [31]	75.4	**75.6**
	ICM	74.2	**75.8**
	TRWS [32]	75.9	**76.7**
	QPBO [26]	76.9	**77.2**
	BNF	74.6	**77.6**
PI-IOU	Belief Propagation [31]	45.9	**46.2**
	ICM	45.7	**48.8**
	TRWS [32]	51.5	**52.0**
	QPBO [26]	55.3	**57.2**
	BNF	53.0	**58.5**

Table 2: We compare semantic segmentation results when using a color-based pixel affinity and our proposed boundary-based affinity. We note that our proposed affinity improves the performance of all globalization techniques. Note that all of the inference methods use the **same FCN unary potentials**. This suggests that for every method our boundary-based affinity is more beneficial for semantic segmentation than the color-based affinity.

represents one of the object classes). In the experimental section, we show that this solution produces better quantitative and qualitative results in comparison to commonly used globalization techniques.

4. Experimental Results

In this section we present quantitative and qualitative results for semantic segmentation on the SBD [13] dataset, which contains objects and their per-pixel annotations for 20 Pascal VOC classes. We evaluate semantic segmentation results using two evaluation metrics. The first metric measures accuracy based on pixel intersection-over-union averaged per pixels (PP-IOU) across the 20 classes. According to this metric, the accuracy is computed on a per-pixel basis. As a result, the images that contain large object regions are given more importance. However, for certain applications we may need to accurately segment small objects. Therefore, similar to [4] we also consider the PI-IOU metric (pixel intersection-over-union averaged per image across the 20 classes), which gives equal weight to each of the images.

We compare Boundary Neural Fields with other commonly used global inference methods. These methods include Belief Propagation [31], Iterated Conditional Mode (ICM), Graph Cuts [5], and Dense-CRF [19]. Note that in all of our evaluations we use the same FCN unary potentials for every model.

Our evaluations provide evidence for three conclusions:

- In Subsection 4.1, we show that our boundary-based pixel affinities are better suited for semantic segmentation than the traditional color-based affinities.
- In Subsection 4.2, we demonstrate that our global minimization leads to better results than those achieved by other inference schemes.
- In Fig. 5, we qualitatively compare the outputs of FCN and Dense-CRF to our predicted segmentations. This comparison shows that the BNF segments are better localized around the object boundaries and that they are also spatially smooth.

4.1. Comparing Affinity Functions for Semantic Segmentation

In Table 2, we consider two global models. Both models use the same unary potentials obtained from the FCN *softmax* layer. However, the first model uses the popular color-based pairwise affinities, while the second employs our boundary-based affinities. Each of these two models is optimized using several inference strategies. The table shows that using our boundary based-affinity function improves the results of all global inference methods according to both evaluation metrics. Note that we cannot include Dense-CRF [19] in this comparison because it employs an efficient message-passing technique and integrating our affinities into this technique is a non-trivial task. However, we compare our method with Dense-CRF in Subsection 4.2.

The results in Table 2 suggest that our semantic boundary based pixel affinity function yields better semantic segmentation results compared to the commonly-used color based affinities. We note that we also compared the results of our inference technique using other edge detectors, notably UCM [1] and HFL [4]. In comparison to UCM edges, we observed that our boundaries provide 1.0% and 6.0% according to both evaluation metrics respectively. When comparing our boundaries with HFL method, we observed similar segmentation performance, which suggests that our method works best with the high quality semantic boundaries.

4.2. Comparing Inference Methods for Semantic Segmentation

Additionally, we also present semantic segmentation results for both of the metrics (PP-IOU and PI-IOU) in Table 3. In this comparison, all the techniques use the same FCN unary potentials. Additionally, all inference methods except Dense-CRF use our affinity measure (since the previous analysis suggested that our affinities yield better performance). We use BNF-SB to denote the variant of our method that uses only semantic boundary based affinities. Additionally, we use BNF-SB-SM to indicate the version of our method that uses both boundary and *softmax*-based affinities (see Eq. (3)).

Based on these results, we observe that our proposed technique outperforms all the other globalization methods according to both metrics, by 0.3% and 1.3% respectively.

Metric	Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
PP-IOU	FCN-Softmax	80.7	71.6	80.7	71.3	72.9	88.1	81.8	86.6	47.4	82.9	57.9	83.9	79.6	80.4	81.0	64.7	78.2	54.5	80.9	69.9	74.8
	Belief Propagation [31]	81.4	72.2	82.4	72.2	74.3	88.8	82.4	87.2	48.4	83.8	58.4	84.6	80.5	80.9	81.5	65.1	79.5	55.5	81.5	71.2	75.6
	ICM	81.7	72.2	82.8	72.1	75.3	89.6	83.4	87.7	46.3	83.3	58.4	84.6	80.6	81.4	81.5	65.8	79.5	56.0	80.7	74.1	75.8
	TRWS [32]	81.6	70.9	83.8	72.0	75.1	89.5	82.5	88.0	**51.7**	86.6	61.9	85.8	83.3	80.8	81.1	65.3	81.5	**58.8**	77.6	75.9	76.7
	Graph Cuts [5]	82.5	72.4	84.6	73.3	**77.2**	89.7	83.3	88.8	49.3	84.0	60.3	85.4	82.2	81.2	81.9	66.7	79.8	58.0	82.3	74.9	76.9
	QPBO [26]	82.6	72.3	84.7	73.1	76.7	89.9	**83.6**	89.3	49.7	85.0	61.1	86.2	82.9	81.3	**82.3**	67.1	80.5	**58.8**	82.2	75.1	77.2
	Dense-CRF [19]	**83.4**	71.5	84.9	72.6	76.2	89.5	83.3	89.1	50.4	**86.7**	61.0	**86.8**	**83.5**	**81.8**	**82.3**	66.9	**82.2**	58.2	81.9	75.1	77.3
	BNF-SB	81.9	72.5	84.9	73.3	76.0	90.3	83.1	89.2	51.2	**86.7**	**61.5**	86.6	83.2	81.3	81.9	66.2	81.7	58.6	81.6	**75.8**	77.4
	BNF-SB-SM	82.2	**73.1**	**85.1**	**73.8**	76.7	**90.6**	83.4	**89.5**	51.3	**86.7**	61.4	**86.8**	83.3	81.7	**82.3**	**67.7**	81.9	58.4	**82.4**	75.4	**77.6**
PI-IOU	FCN-Softmax	56.9	35.1	47.8	41.1	27.4	51.1	43.4	52.7	22.2	43.1	29.2	54.2	40.5	45.6	59.1	24.2	43.6	24.8	55.9	37.2	41.8
	Belief Propagation [31]	68.0	38.6	52.9	45.8	31.9	55.9	47.2	58.2	24.6	49.9	31.7	60.2	44.9	50.1	62.4	25.2	49.9	27.6	62.3	42.2	46.2
	ICM	65.3	40.9	56.4	45.3	33.7	58.9	49.5	61.9	25.8	53.5	33.2	62.1	48.0	53.2	63.4	24.1	54.8	34.0	63.7	47.7	48.8
	TRWS [32]	67.5	40.7	60.3	46.3	35.6	63.4	49.6	69.3	29.7	58.9	37.8	67.4	57.3	53.8	64.1	26.3	62.0	36.9	63.1	49.9	52.0
	Graph Cuts [5]	72.1	47.8	64.5	50.8	36.0	70.8	51.4	71.6	31.7	65.8	34.4	71.8	62.0	59.4	64.8	29.0	60.9	38.7	70.3	51.6	55.3
	QPBO [26]	71.6	46.8	65.6	49.6	38.0	72.6	52.7	76.7	32.5	69.6	38.9	74.4	61.4	61.0	**66.2**	30.3	**68.7**	**41.4**	72.2	52.8	57.2
	Dense-CRF [19]	68.0	39.5	58.0	45.0	33.4	62.8	47.7	66.0	29.4	60.9	36.0	68.5	54.6	51.4	63.7	28.3	57.6	37.1	65.9	48.2	51.1
	BNF-SB	71.6	48.1	**67.2**	52.3	37.8	**79.5**	52.9	**80.8**	**33.3**	71.5	**39.5**	**75.1**	65.7	63.4	65.1	31.1	67.5	39.6	**73.2**	**54.7**	**58.5**
	BNF-SB-SM	**72.0**	**48.9**	66.5	**52.9**	**39.1**	79.0	**53.4**	78.6	32.9	**72.2**	39.4	74.6	**65.9**	**64.2**	65.8	**31.7**	66.9	39.0	73.1	53.9	**58.5**

Table 3: Semantic segmentation results on the SBD dataset according to PP-IOU (per pixel) and PI-IOU (per image) evaluation metrics. We use BNF-SB to denote the variant of our method that uses only semantic boundary based affinities. Additionally, we use BNF-SB-SM to indicate our method that uses boundary and *softmax* based affinities (See Eq. (3)). We observe that our proposed globalization method outperforms other globalization techniques according to both metrics by at least 0.3% and 1.3% respectively. Note that in this experiment, all of the inference methods use **the same FCN unary potentials**. Additionally, for each method except Dense-CRF (it is challenging to incorporate boundary based affinities into the Dense-CRF framework) we use our boundary based affinities, since those lead to better results.

Additionally, these results indicate that most benefit comes from the semantic boundary affinity term rather than the *softmax* affinity term.

In Fig. 5, we also present qualitative semantic segmentation results. Note that, compared to the segmentation output from the *softmax* layer, our segmentation is much better localized around the object boundaries. Additionally, in comparison to Dense-CRF predictions, our method produces segmentations that are much spatially smoother.

4.3. Semantic Boundary Classification

We can also label our boundaries with a specific object class, using the same classification strategy as in the HFL system [4]. Since the SBD dataset provides annotations for semantic boundary classification, we can test our results against the state-of-the-art HFL [4] method for this task. Due to the space limitation, we do not include full results for each category. However, we observe that our produced results achieve mean Max F-Score of 54.5% (averaged across all 20 classes) whereas HFL method obtains 51.7%.

5. Conclusions

In this work we introduced a Boundary Neural Field (BNF), an architecture that employs a semantic segmentation FCN to predict semantic boundaries and then uses the predicted boundaries and the FCN output to produce an improved semantic segmentation maps a global optimization. We showed that our predicted boundaries are better suited for semantic segmentation than the commonly used low-level color based affinities. Additionally, we introduced a global energy function that decomposes semantic segmentation into multiple binary problems and relaxes an integrality constraint. We demonstrated that the minimization of this global energy allows us to predict segmentations that are better localized around the object boundaries and that are spatially smoother compared to the segmentations achieved by prior methods. We made the code of our globalization technique available at `http://www.seas.upenn.edu/~gberta/publications.html`.

The main goal of this work was to show the effectiveness of boundary-based affinities for semantic segmentation. However, due to differentiability of our global energy, it may be possible to add more parameters inside the BNFs and learn them in a front-to-end fashion. We believe that optimizing the entire architecture jointly could capture the inherent relationship between semantic segmentation and boundary detection even better and further improve the performance of BNFs. We will investigate this possibility in our future work.

6. Acknowledgements

This research was funded in part by NSF award CNS-1205521.

References

[1] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011. 2, 4, 7

[2] Pablo Arbelaez, J. Pont-Tuset, Jon Barron, F. Marqués, and Jitendra Malik. Multiscale combinatorial grouping. In *Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 4

[3] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2, 3, 4

[4] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 2, 3, 4, 7, 8

[5] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, November 2001. 5, 7, 8

[6] João Carreira, Rui Caseiro, Jorge Batista, and Cristian Sminchisescu. Semantic segmentation with second-order pooling. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part VII*, ECCV'12, pages 430–443, Berlin, Heidelberg, 2012. Springer-Verlag. 2

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully. In *ICLR*, 2015. 1, 2, 3

[8] Jifeng Dai, Kaiming He, and Jian Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 3

[9] Piotr Dollár and C. Lawrence Zitnick. Fast edge detection using structured forests. *PAMI*, 2015. 2, 4

[10] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 2013. 2

[11] Yaroslav Ganin and Victor S. Lempitsky. N^4-fields: Neural network nearest neighbor fields for image transforms. *ACCV*, 2014. 2, 4

[12] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from RGB-D images for object detection and segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 2

[13] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *International Conference on Computer Vision (ICCV)*, 2011. 4, 7

[14] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision (ECCV)*, 2014. 2

[15] Bharath Hariharan, Pablo Andrés Arbeláez, Ross B. Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2, 3

[16] Seunghoon Hong, Hyeonwoo Noh, and Bohyung Han. Decoupled deep neural network for semi-supervised semantic segmentation. In *NIPS)*, December 2015. 3

[17] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H. Adelson. Crisp boundary detection using pointwise mutual information. In *ECCV*, 2014. 2

[18] Jyri J Kivinen, Christopher KI Williams, and Nicolas Heess. Visual boundary prediction: A deep neural prediction network and quality dissection. *AISTATS*, 1(2):9, 2014. 2

[19] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 109–117. Curran Associates, Inc., 2011. 1, 5, 7, 8

[20] Joseph Lim, C. Lawrence Zitnick, and Piotr Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *CVPR*, 2013. 2

[21] Guosheng Lin, Chunhua Shen, Ian D. Reid, and Anton van den Hengel. Efficient piecewise training of deep structured models for semantic segmentation. *CoRR*, abs/1504.01013, 2015. 3

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 1, 3

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001. 4

[24] Mohammadreza Mostajabi, Payman Yadollahpour, and Gregory Shakhnarovich. Feedforward semantic segmentation with zoom-out features. *CoRR*, abs/1412.0774, 2014. 2

[25] X. Ren and L. Bo. Discriminatively Trained Sparse Code Gradients for Contour Detection. In *Advances in Neural Information Processing Systems*, December 2012. 4

[26] Carsten Rother, Vladimir Kolmogorov, Victor Lempitsky, and Martin Szummer. Optimizing binary mrfs via extended roof duality. In *Proc Comp. Vision Pattern Recogn. (CVPR)*, June 2007. 7, 8

[27] Wei Shen, Xinggang Wang, Yan Wang, Xiang Bai, and Zhijiang Zhang. Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. June 2015. 2, 4

[28] Jonathan R Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, Pittsburgh, PA, USA, 1994. 6

[29] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 1997. 2

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 3

[31] Marshall F. Tappen and William T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 900–, Washington, DC, USA, 2003. IEEE Computer Society. 5, 7, 8

[32] Martin Wainwright, Tommi Jaakkola, and Alan Willsky. Map estimation via agreement on (hyper)trees: Message-passing and linear programming approaches. *IEEE Transactions on Information Theory*, 51:3697–3717, 2002. 7, 8

[33] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 2, 4

[34] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip Torr. Conditional random fields as recurrent neural networks. In *International Conference on Computer Vision (ICCV)*, 2015. 3

[35] Dengyong Zhou, Olivier Bousquet, Thomas N. Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In S. Thrun, L.K. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 321–328. MIT Press, 2004. 5