Structure Preserved Semantic Image Segmentation

Shuo Cheng
University of California, San Diego
scheng@ucsd.edu

Mingxiang Cai
University of California, San Diego
mcai@ucsd.edu

Dingcheng Hu
University of California, San Diego
dih015@ucsd.edu

Siman Wang
University of California, San Diego
siwang@ucsd.edu

Abstract

Semantic image segmentation plays a very important role in a wide variety of computer vision applications (e.g., robotics, autonomous driving, human-computer interaction). Previous methods (such as graph-cut and energy minimization) only consider hand-craft features (e.g., the color information, the affinity of the pixels), thus can not reach satisfying results. Recent years, with the popularity of deep learning, the performance of semantic image segmentation has been boosted a lot. However, due to the limited resolution of the prediction, the results are still unsatisfactory. In this paper, we innovatively integrate deep learning into conditional random field framework. We treat the prediction of the neural nets as a unified term, and we construct the pairwise term using the semantic features from the neural nets. Based on minimizing the global energy, we can obtain more smooth and accurate results. Exhaustive experiments demonstrate the effectiveness of our proposed method.

1. Introduction

Semantic segmentation is one of the key problems in computer vision. It is basically a a pixel-wise classification problem. Fine-grained inference is achieved by making dense predictions inferring labels for every pixel. From the high level, semantic segmentation is one of the most significant tasks to achieve complete scene understanding. The importance of scene understanding is accentuated with an increasing number of applications that infer knowledge from images. Some of these applications include autonomous driving, aerial image analysis, 3D reconstruction, medical image processing and robot-assisted surgery.

Conditional random fields (CRF) is a probabilistic framework for inferring structured and sequential data. It is widely used in natural language processing problems and have achieved a certain level of success. In the pipeline of CRF learning based image segmentation, finding a good feature representation is of great significance, and can have a profound impact on the segmentation accuracy. Most previous studies rely on hand-crafted features, e.g., using color histograms, HOG or SIFT descriptors to construct bag-of-words features, these methods have strong limitations when applied in complicated scenes. Based on these observations, we innovatively integrate deep learning method into CRF framework.

Recently, feature learning and especially deep learning methods have gained great popularity in machine learning and related fields, This type of methods typically takes raw images as an input and learn a (deep) representation of the images, and have found phenomenal success in various tasks such as classification, object detection, and tracking, etc. Deep learning methods attempt to model high-level abstractions in data at multiple layers, inspired from the cognitive processes of human brains, which generally starts from simpler concepts to more abstract ones. The advantages of deep learning methods shed light on us to integrate it into the CRF framework. On one hand, we want to take merits of the powerful features learned through the data-driven method, on the other hand, we rely on graph model to guarantee structural completeness. We innovatively incorporate the learned features to model the co-occurrence pairwise potentials, and minimize the energy function to achieve smooth and structured predictions.

2. Related works

Before deep learning took over computer vision, people used approaches like Graph-cut based method for semantic segmentation. After the image is turned into a graph, segmentation can be considered as a two-way partition problem. The two subsets of nodes after min-cut is the foreground and background in image segmentation.

One of the popular methods using deep learning technique to do semantic segmentation is using Fully Convolu-

tional Network(FCN)[1]. FCN will learn a mapping from pixels to pixels, without extracting the region proposals. As an extension of the classical CNN, FCN can take input as arbitrary-sized images. The key intuition behind FCN is that fully connected layers in classification networks can be considered as convolutions with kernels that cover the entire input regions, which is equivalent to evaluating the original classification network on overlapping input patches. However, this is more efficient since the computation over the overlapping regions of patches is shared. One issue of FCN is that after several convolutional and pooling layers, the resolution of the feature maps is down sampled. Thus, the direct predictions of FCN are usually in low resolution, resulting in relatively fuzzy object boundaries. U-Net[2] is one example of FCN.

3. Method

3.1. Conditional random field revisiting

Before fomulating our problem, we will first introduce some preliminaries of CRF model. CRF[3] is a undirected graph consists of label nodes and observation node. In the image segmentation settings, every pixel in the image is related to one label node and one observation node. Two label nodes are connected if and only if the corresponding pixels are connected. Mathematically, we define each pixel is a random variable from $\mathbf{X} = \{X_1, X_2, X_3, ..., X_n\}$ and each X_a takes a label x_a from $\mathcal{L} = \{x_1, x_2, x_3, ..., x_d\}$, where n and d are the total number of pixels and labels respectively. In order to abtain the final label assignment of each pixel, we will minimize the Gibbs enegy function of CRF, which has the following form:

$$E(\mathbf{x}) = \sum_a \phi\left(x_a\right) + \sum_{a,b\neq a} \psi\left(x_a, x_b\right)$$

where $\phi\left(x_a\right)$ denotes the unary potential of a pixel and $\psi\left(x_a, x_b\right)$ denotes the binary potential of a pair of pixels. The unary term and the binary term represents the data cost and the smoothness cost respectively. Mathematically, they can be written as:

$$\phi\left(x_a\right) = -\log p\left(x_a\right)$$

$$\psi\left(x_a, x_b\right) = u\left(x_a, x_b\right) \sum_c \omega^{(c)} k^{(c)}\left(\mathbf{f}_a, \mathbf{f}_b\right)$$

Here, the unary term is defined as the negative log likelihood of the assigned labels, and the binary term is defined as an weighted sum of different kernels. Note that $u(x_a, x_b)$ measures that compatibility of label x_a and label x_b, for simplicity, we use the potts model and set $u(x_a, x_b) = \mathbf{1}(x_a \neq x_b)$. Intuitively, the binary term guarantees that when the total energy is minimized, same labels will be assigned to pixels that are similar. The simialarity is related to pixels' relative positions and also other features, depending on the kernels that are used.

3.2. Energy function generalization

In previous studies, RGB values from the original image is widely used as feature input[4][5]. While using RGB features has achieved certain results, in our study, we innovatively build a universal framework that allows the energy function to incorporate general features.

In this framework, multiple bilateral kernels are used. Without loss of generality, a bilateral kernel can be represented as the followings:

$$k^{(i)}\left(\mathbf{f_a}, \mathbf{f_b}\right) = \exp\left(-\frac{\left|p_a - p_b\right|^2}{2\theta_\alpha^2} - \frac{\left|I_a - I_b\right|^2}{2\theta_\beta^2}\right)$$

Here, p_a, p_b represent the coordinate information of pixels X_a, X_b, and I_a, I_b are holders for other arbitrary features. While the coordinate information is fixed, we can utilize different pixel-level features to model the co-occurrence pairwise potentials. For instance, If we let I_a be a constant, the kernel reduces to a smoothness kernel; if we let I_a be RGB values of the original image, it forms an appearance kernel; if we let I_a be learned features from the deep network, it forms a textural kernels. A simple schematic diagram is shown as Figure. 1.

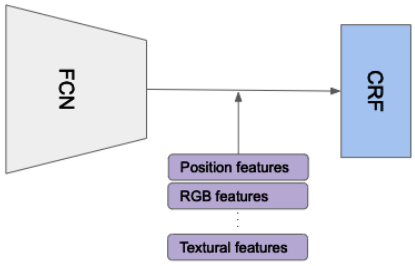

Figure 1. General framework diagram

In our paper, we test the framework with RGB features, with textural features extracted from deep network layer and with the combination of the two.

3.3. Problem definition

With the gibbs energy function defined, we formulate the optimization problem as following:

$$\begin{array}{ll} \min_{\mathbf{y}} & E(\mathbf{y}) = \sum_a \sum_i \phi_{a:i} y_{a:i} + \sum_{a,b\neq a} \sum_{i,j} \psi_{ab:ij} y_{a:i} y_{b:j} \\ \text{s.t.} & \sum_i y_{a:i} = 1 \quad \forall a \in \{1 \dots n\} \\ & y_{a:i} \in \{0,1\} \quad \forall a \in \{1 \dots n\}, \quad \forall i \in \mathcal{L} \end{array} \tag{1}$$

Here, $\mathbf{y}$ is the assignement matrix for $\mathbf{X}$, where $y_{a:i} = 1$ if and only if the label assignment of random variable X_a

is i, or namely $x_a = i$. Also notice that we use $\phi_{a:i} = \phi(x_a = i)$, $\psi_{ab:ij} = \psi(x_a = i, x_b = j)$ for short.

3.4. Optimization details

Notice that the problem defined above is an integer programming problem, which is ingeneral NP-hard. To solve this, a commonly used techique is linear programming relaxation, that is, relaxing the integer constrains on $y_{a:i}$ to allow it to be fractional values in $[0, 1]$[4]. The resulted linear programming problem is written as the following:

$$\min_{\mathbf{y}} \tilde{E}(\mathbf{y}) = \sum_a \sum_i \phi_{a:i} y_{a:i} + \sum_{a,b \neq a} \sum_i K_{ab} \frac{|y_{a:i} - y_{b:i}|}{2}$$

$$\text{s.t.} \quad \mathbf{y} \in \mathcal{M} = \left\{ \begin{array}{l} \sum_i y_{a:i} = 1, a \in \{1 \dots n\} \\ y_{a:i} \geq 0, a \in \{1 \dots n\}, i \in \mathcal{L} \end{array} \right\} \quad (2)$$

where $K_{ab} = \sum_c w^{(c)} k\left(\mathbf{f}_a^{(c)}, \mathbf{f}_b^{(c)}\right)$. Notice that we can easily substitute $|y_{a:i} - y_{b:i}|$ and add linear constrains to eliminate the absolute function, the fomulation above is indeed a linear programming problem. Although the above relaxatoin can be solved by a standard solver, the computation involves $\mathcal{O}\left(n^2\right)$ unknown variables and hence is unmanageable.

In order to accerlerate computation, we employ PROX-LP algorithm[5] proposed by Ajanthan et al. The algorithm minimizes $\tilde{E}(y)$ iteratively to increase speed and save space. Its advantages include smooth updates and monotone descent of objective function. Particularly, suppose at time t we have assignment matrix y_t, the next update involves solving:

$$\begin{array}{ll} \min_{\mathbf{y}} & \tilde{E}(\mathbf{y}) + \frac{1}{2\lambda} \left\|\mathbf{y} - \mathbf{y}^k\right\|^2 \\ \text{s.t.} & \mathbf{y} \in \mathcal{M} \end{array} \quad (3)$$

where measures the strength of the proximal term. To formulate the dual problem, We introduce shawdow price variables $\boldsymbol{\alpha} = \left\{\alpha_{ab:i}^1, \alpha_{ab:i}^2 | a, b \neq a, i \in \mathcal{L}\right\}, \boldsymbol{\beta} = \left\{\beta_a | a \in \{1 \dots n\}\right\}, \boldsymbol{\gamma} = \left\{\gamma_{a:i}, a \in \{1 \dots n\}, i \in \mathcal{L}\right\}$, and define $(A\boldsymbol{\alpha})_{a:i} = -\sum_{b \neq a} \left(\alpha_{ab:i}^1 - \alpha_{ab:i}^2 + \alpha_{ba:i}^2 - \alpha_{ba:i}^1\right)$ $(B\boldsymbol{\beta})_{a:i} = \beta_a$ for simplicity. With the notations defined, the dual problem of (3) can be written as:

$$\begin{aligned} \min_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} g(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) &= \frac{\lambda}{2} \|A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}\|^2 \\ &\quad + \left\langle A\boldsymbol{\alpha} + B\boldsymbol{\beta} + \boldsymbol{\gamma} - \boldsymbol{\phi}, \mathbf{y}^k \right\rangle - \langle \mathbf{1}, \boldsymbol{\beta} \rangle \\ \text{s.t.} \quad \gamma_{a:i} &\geq 0 \quad \forall a \in \{1 \dots n\} \quad \forall i \in \mathcal{L} \\ \boldsymbol{\alpha} \in \mathcal{C} &= \left\{ \boldsymbol{\alpha} \;\middle|\; \begin{array}{l} \alpha_{ab:i}^1 + \alpha_{ab:i}^2 = \frac{K_{ab}}{2}, a, b \neq a, i \in \mathcal{L} \\ \alpha_{ab:i}^1, \alpha_{ab:i}^2 \geq 0, a, b \neq a, i \in \mathcal{L} \end{array} \right\} \end{aligned} \quad (4)$$

Finally, we employ iterate gradient descent to solve the dual problem.

4. Experiments

4.1. Data set

We utilized the Cityscape dataset, which contains video sequences recorded in street scenes from 50 different cities, with high quality pixel-level annotations. The training set includes 2975 RGB frames and the validation set includes 500 RGB frame.

4.2. Quantitative analysis

	w/o CRF	w/ CRF
Accuracy	0.884	**0.892**

Table 1. Overall accuracy

catagory	accuracy w/ CRF (%)	accuracy boost (%)
building	96.75	1.41
terrain	25.51	0.97
vegetation	96.65	0.78
wall	74.34	0.72
road	89.30	0.67
bicycle	97.82	0.16
rider	11.97	0.14
truck	47.74	0.13
sidewalk	87.18	0.13
fence	0.00	0.00
train	0.00	0.00
motorcycle	0.00	0.00
bus	0.00	0.00
car	98.60	0.00
traffic light	0.17	-0.09
person	85.15	-0.92
traffic sign	19.74	-1.50

Table 2. Accuracy by categories

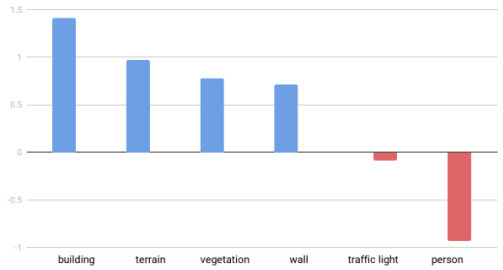

Figure 2. Accuracy boost by categories (in percentage)

4.3. Qualitative analysis

See figure 2.

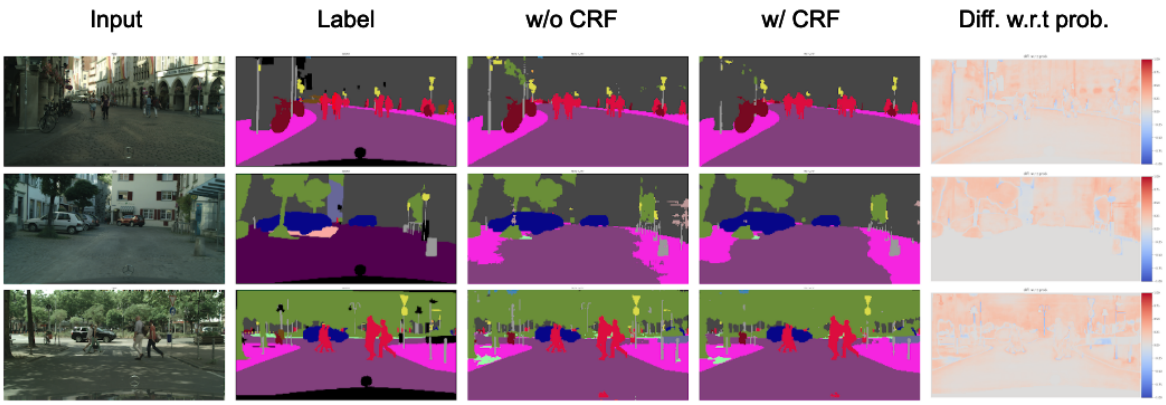

Figure 3. Qualitative results comparison.

5. Conclusion and future work

In this work, we presented a method for improving the semantic segmentation results. Experiments demonstrate that some errors can be removed through minimizing our proposed energy function. Also, our method can be applied on new data which are out of the training domain.

For the future work, we can extend the proposed energy function to temporal domain, so as to achieve spatial-temporal continuity.

References

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[3] Pradeep D. Ravikumar and John D. Lafferty. Quadratic programming relaxations for metric labeling and markov random field map estimation. pages 737–744, 01 2006.

[4] Alban Desmaison, Rudy Bunel, Pushmeet Kohli, Philip H. S. Torr, and M. Pawan Kumar. Efficient continuous relaxations for dense CRF. *CoRR*, abs/1608.06192, 2016.

[5] Thalaiyasingam Ajanthan, Alban Desmaison, Rudy Bunel, Mathieu Salzmann, Philip H. S. Torr, and M. Pawan Kumar. Efficient linear programming for dense crfs. *CoRR*, abs/1611.09718, 2016.