1 Introduction

We propose an alternative discriminative approach to CRFs for scene labeling by extending the large margin principle underpinning SVMs to incorporate spatial correlations. Our main contributions are as follows:

1. We propose an alternative discriminative approach to CRFs for image labeling by extending the large margin principle to incorporate spatial correlations among neighboring pixels.

2. By explicitly executing the submodular condition, we are able to utilize a graph-cuts inference engine that promises to obtain the optimal solution efficiently.

3. Our model is also capable of integrating with higher-order scene context. We demonstrate this by incorporating superpixel prediction cue in the experiments.

2 Our Approach

Large-margin Formulation During training we have access to a set of ground-truth images \( \{X_i, Y_i\}_{i=1}^{n} \). Our aim is to learn a \( W \)-parameterized conditional distribution model \( p(Y|X;W) \), over the label graph \( Y \) for each image \( X_i \), where the maximum probability is attained at its ground-truth label \( Y_i \). An often-made assumption is \( p(Y|X;W) \) follows an exponential family model

\[
\log p(Y|X;W) = \langle F(Y), h(X;W) \rangle - A_Y(Y),
\]

where \( F(Y) \) is a feature map over the joint instance-label space. Given an image-label pair \( \{X_i, Y_i\} \), we would like the logarithmic conditional probability at the ground-truth label \( Y_i \) to be the largest over all candidate label \( Y \), i.e., \( \log p(Y|X;W) \geq \log p(Y|X;W) \) for each \( X \), and at the same time bound \( W \) to avoid trivial solutions. By invoking (1), the optimization problem reads,

\[
\begin{align*}
\min \langle F(Y), h(X;W) \rangle + \sum_{i} \beta \sum_{t} \langle \lambda ti \rangle \quad \text{s.t.} \quad & (\forall W) \in \mathbb{R} \\
& \text{max} \langle F(Y), h(X;W) \rangle = \langle F(Y), h(X;W) \rangle.
\end{align*}
\]

The label loss, \( \Delta(Y) \), plays the margin between the true label and its competitive assignments.

Now, presented with an unseen image \( X \), the problem of scene labeling can be formally described as predicting the graph label \( Y \) by maximizing the conditional probability likelihood (ML) in (1)

\[
Y^* = \arg \max_{Y} \langle F(Y), h(X;W) \rangle .
\]

Denote the discriminative function \( F(Y|X) = \langle F(Y), h(X;W) \rangle \), therefore, the optimal label assignment \( Y^* \) can be regarded as the one which comes with the maximum \( F(Y|X) \) score.

The graph label \( Y \) enables us to decompose the discriminant function into local (node and edge) parts, and similarly the feature function, which yields

\[
F(Y|X) = \sum_{i} \varphi_i(X) + \sum_{ij} \varphi_{ij}(X;Y) = \sum_{i} \langle \varphi_i, \delta_i(Y|X) \rangle + \sum_{ij} \langle \varphi_{ij}, \delta_{ij}(X,Y) \rangle .
\]

3 Experiments

Features Constructed from low-level descriptors, our node features are designed to capture local appearance as well as scene contextual information, while edge features encode local compatibilities. We build up conventional filter banks consisting of oriented Gaus- sians and Laplacian of Gausians at 3 scales. We then learn a series of one-vs-all AdaBoost classifiers (one for each class label) based on these filter responses, and normalize over all categories. This results in 7-dim of texton-shape potentials. We also compute 15-dim of color and 7-dim of location features. These three components, together with 7-dim superpixel context features and a one dimensional bias constant, are concatenated to form our local appearance node features.

Our edge features \( \omega \) contain three components. The first one is the element-wise absolute difference of texton and contextual features from node \( i \) and \( j \), which has 14 dimensions. In addition, we include the one dimensional edge feature of color distance, together with a one dimensional bias constant. These features are further expanded over different class labels (as a tensor product), which allows distinct weights \( \omega_{ij} \) and \( \omega_{ij} \) to be applied on different labels \( y \) and \( y \) on a node or pair of labels \( y \) and \( y \). In an edge, our feature construction produces features of 1043 dimensions, in which 259 and 784 dimensions for node and edge respectively. This naturally leads to a rich set of model parameters to be learned.

Efficient Learning to Label Images

Incorporating Higher-order Scene Context Often we encounter situation where pixels a few blocks away are necessary to facilitate a proper prediction of the current pixel. Superpixels are useful to circumvent this issue as a set of nearby pixels (often belong to the same class) can be considered to help the prediction. Our strategy of incorporating higher-order scene context is to take multi-scale superpixel segmentations as additional node features. By considering multiple scale-spaces, we have access to a series of prediction maps, corresponds to coarser throughout to finer granularity. Experimental evaluation suggests that it helps to improve our prediction accuracy over the state-of-the-arts.

Running Time The Sowerby dataset consists of 7 labels on 104 images sized 96 × 64 pixels. And the Corel dataset is a 100-image dataset of African and Arctic wildlife natural scenes. Each image is 180 × 120 and each pixel is labeled as one of 7 classes. During the experiments, images in both datasets are split into the train, validation and test sets using ratio of 5:1:4. Despite of learning a model with thousands of parameters (1043 dimensions in total), by exploiting parallel computation over multiple GPUs in training, we still manage to train efficiently real-world datasets, meanwhile graph-cuts allows efficient inference in test run. The train/test time of our approach is 1h/1.1s on Sowerby and 3h/2.5s on Corel dataset, measured on a Beowulf cluster and used 20 GPUs.

Figure 1: Examples of labeling results on Sowerby dataset. From row 1-5: images, ground-truth, superpixels, superpixel segmentations, and predictions of our full-fledged labeling algorithm.

Figure 2: Examples of labeling results on Corel dataset. From row 1-5: images, ground-truth, superpixels, superpixel segmentations, and predictions of our full-fledged labeling algorithm.