A Hierarchical Field Framework for Unified Context-Based Classification

Sanjiv Kumar and Martial Hebert
The Robotics Institute, Carnegie Mellon University
Pittsburgh, PA 15213, USA, {skumar, hebert}@ri.cmu.edu

Abstract

We present a two-layer hierarchical formulation to exploit different levels of contextual information in images for robust classification. Each layer is modeled as a conditional field that allows one to capture arbitrary observation-dependent label interactions. The proposed framework has two main advantages. First, it encodes both the short-range interactions (e.g., pixelwise label smoothing) as well as the long-range interactions (e.g., relative configurations of objects or regions) in a tractable manner. Second, the formulation is general enough to be applied to different domains ranging from pixelwise image labeling to contextual object detection. The parameters of the model are learned using a sequential maximum-likelihood approximation. The benefits of the proposed framework are demonstrated on four different datasets and comparison results are presented.

Figure 1. Example images demonstrating that scene context is important in different domains to achieve good classification even though the local appearance is impoverished. From left: first and second - scene labeling (*region-region* interaction), third - *object-region* interaction, fourth - *object-object* interaction.

1. Introduction

The problem of detecting and classifying regions and objects in images is a challenging task due to ambiguities in the appearance of the visual data. The use of spatial context can help alleviate this problem significantly. For example, in Figure 1, the sky and the water patches may locally look very similar but their relative spatial configuration removes this ambiguity.

There are different levels of contexts one would like to use to improve classification accuracy. For instance, for pixelwise image labeling problem, the local smoothness of pixel labels will be a local context. On the other hand, global context will refer to the fact that the image regions follow probable configurations e.g., sky tends to occur above water or vegetation (Figure 1). We denote this type of global context by *region-region* interaction. Similarly, for the problem of parts-based object detection, the local context will be the geometric relationship among parts of an object while the relative spatial configurations of different objects will provide the global contextual information. This type of global context is denoted by *object-object* interaction. As shown in Figure 1, the keyboard and the mouse may be very hard to detect because of their impoverished appearance but the relative configuration of monitor, keyboard and mouse helps disambiguate the detection. Similarly, car detection is much easier given the configuration of building and road (Figure 1). In this case, the global context is provided by *object-region* interaction.

In the past, context has been advocated for the problems of pixelwise image labeling [13][5] and object detection [2][15][12]. All these techniques are either specifically tuned for a certain application domain or use context only at a specific level. The key contribution of this paper is a framework that provides a unified approach to incorporate the local as well as the global context of any of the three types in a single model.

In [13], Singhal et al. presented an approach for labeling each region in the scene sequentially based on the labels of the previous regions. This approach will give spurious results if the previously labeled regions were assigned wrong labels. Markov Random Fields (MRFs) provide a sound theoretical approach to model contextual interactions among different components simultaneously [4]. However, a variety of applications require image observations to model such interactions. For example, different natural regions in a scene, or parts of an object are related through geometric constraints. Traditional MRFs do not allow the use of observed data to model interactions between labels. Conditional Random Fields (CRFs), proposed in [10], provide a principled approach to incorporate these data-dependent interactions. In our hierarchical approach, each layer is modeled as a CRF. Another advantage of CRFs

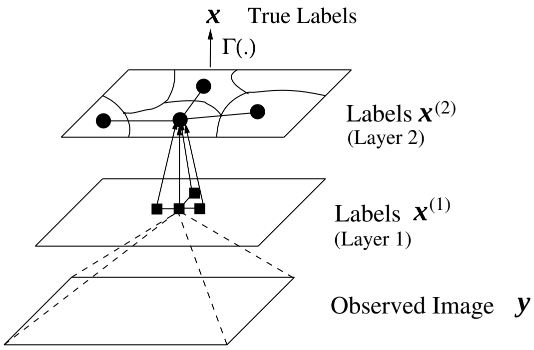


Figure 2. A simple illustration of the two-layer hierarchical field for contextual classification. Squares and circles represent sites at the two layers. Only one node along with its neighbors is shown for each layer for clarity. Layer 1 models short-range interactions while layer 2 long range dependencies in images. The true labels x are obtained from the top layer by a simple replication mapping $\Gamma(.)$. Note that the partition shown in the top layer is not necessarily a partition on the image.

over the traditional MRFs is that they use a discriminative approach for classification rather than spending the efforts in modeling the generation of the observed data.

Different forms of CRFs have been used by various researchers in image modeling [7][5][15]. He et al. [5] have presented an approach where context is enforced through local and global learned features tuned to pixelwise scene labeling application. Torralba et al. [15] have combined boosting with CRFs to learn the graph structure and its potentials for contextual object detection, but do not provide a guiding framework for handling different levels of context for different applications in the same model.

Various forms of hierarchical models have been suggested under both undirected [11] as well as directed [1] graph paradigms. However, these models have been restricted to simple local contextual information such as label smoothing to obtain good segmentation. They do not use any high level global context. In addition, all the previous hierarchical models were based on MRFs. This paper presents the first work on using a hierarchy of CRFs.

2. Hierarchical Framework

In this work, we are interested in modeling interactions in images at two different levels. Thus, we propose a two-layer hierarchical field model as shown in Figure 2. Note that, in any of the two layers, the induced graph's topology is not restricted to regular 2D grid locations. In this model, each layer is a separate conditional field. The first layer models short range interactions among the sites such as label smoothing for pixelwise labeling, or geometric consistency among parts of an object. The second layer models the long range interactions between groups of sites corresponding to different coherent regions or objects. Thus, this layer can take into account interactions between different objects (monitor/keyboard) or regions (sky/water).

The two layers of the hierarchy are coupled with directed links. A node in layer 1 may represent a single pixel or a patch while a node in layer 2 represents a larger homogeneous region or a whole object. Each node in the two layers is connected to its neighbors through undirected links. In addition, each node in layer 2 is also connected to multiple nodes in layer 1 through directed links. In the present work we restrict each node in layer 1 to be connected to only one node in the layer above. As noted by Hinton et al. [6], with respect to hierarchical MRFs, the use of directed links between the two layers, instead of the undirected ones, avoids the intractability of dealing with a large partition function. Being a conditional field, each node in layer 1 can potentially use arbitrary features from the whole image to compute its bias. The top layer uses the output of layer 1 as input through the directed links.

2.1. Basic Formulation

Let the observed data from an input image be given by $\boldsymbol{y} = \{\boldsymbol{y}_i\}_{i\in S}$, where $\boldsymbol{y}_i$ is the data from i^{th} site, and S is the set of all the image sites. We are interested in finding the labels, $\boldsymbol{x} = \{x_i\}_{i\in S}$, where $x_i \in \mathcal{L}$ and $|\mathcal{L}|$ is the number of classes. For image labeling, a site is a pixel and a class may be *sky*, *grass* etc., while for contextual object detection, a site is a patch and a class may refer to objects e.g., *keyboard* or *mouse*. The set of sites in layer 1 is $S^{(1)}$ such that $S^{(1)} = S$, while that in layer 2 is denoted by $S^{(2)}$. The nodes in layer 2 induce a partition over the set $S^{(1)}$ such that a subset of nodes in layer 1 correspond to one node in layer 2. Formally, a partition h is defined as $h : S^{(1)} \rightarrow S^{(2)}$ such that, if $S_r^{(1)}$ is a subset of nodes in layer 1 corresponding to node $r \in S^{(2)}$, then $S^{(1)} = \bigcup_r S_r^{(1)}$ and $S_r^{(1)} \cap S_s^{(1)} = \phi \ \ \forall\, r, s \in S^{(2)}$. Let the space of all partitions be denoted as $\mathcal{H}$. This partition should not be confused with an image partition, since it is defined over the sites in $S^{(1)}$, which may not correspond to the image pixels (e.g., in object detection, where sites are random image patches). Let the labels on the sites in the two layers be given by $\boldsymbol{x}^{(1)} = \{x_i^{(1)}\}_{i\in S^{(1)}}$ and $\boldsymbol{x}^{(2)} = \{x_r^{(2)}\}_{r\in S^{(2)}}$, where $x_i^{(1)} \in \mathcal{L}^{(1)}$ and $x_r^{(2)} \in \mathcal{L}^{(2)}$, where $\mathcal{L}^{(2)} = \mathcal{L}$. The nodes in layer 1 may take pseudo labels that are different from the final desired labels. For instance, in object detection, a node at layer 1 may be labeled as 'a certain part' of an object rather than the object itself. In fact, the labels at this layer can be seen as noisy versions of the true desired labels .

Given an image $\boldsymbol{y}$, we are interested in obtaining the conditional distribution $P(\boldsymbol{x}|\boldsymbol{y})$ over the true labels. Given

$\boldsymbol{y}$, let us define a space of valid partitions, $\mathcal{H}_v$, such that $\forall\ h \in \mathcal{H}_v, x_i = x_r^{(2)}\ \ \forall\, i \in S_r^{(1)}$ where $r = h(i)$. This implies that multiple nodes in layer 1 make a hypothesis about a single *homogeneous* region or an object in layer 2. Further, we define a replication mapping, $\Gamma(.)$, which takes any value (discrete or continuous) on node r and assigns it to all the nodes in $S_r^{(1)}$. Thus, given a partition $h \in \mathcal{H}_v$, and the corresponding labels $\boldsymbol{x}^{(2)}$, the labels $\boldsymbol{x}$ can be obtained simply by replication. This implies, $P(\boldsymbol{x}|\boldsymbol{y}) \equiv P(\boldsymbol{x}^{(2)}|h, \boldsymbol{y})$ if $h \in \mathcal{H}_v$. However, given an observed image $\boldsymbol{y}$, the constraint $h \in \mathcal{H}_v$ is too restrictive. Instead, we define a distribution, $P(h|\boldsymbol{y})$, that prefers partitions in $\mathcal{H}_v$ over all possible partitions, and,

$$P(\boldsymbol{x}|\boldsymbol{y}) \cong \sum_{h\in\mathcal{H}} P(\boldsymbol{x}^{(2)}|h, \boldsymbol{y})P(h|\boldsymbol{y})$$
$$= \sum_{h\in\mathcal{H}} \sum_{\boldsymbol{x}^{(1)}} P(\boldsymbol{x}^{(2)}|h, \boldsymbol{x}^{(1)})P(h|\boldsymbol{x}^{(1)})P(\boldsymbol{x}^{(1)}|\boldsymbol{y}), \qquad (1)$$

where both $P(\boldsymbol{x}^{(1)}|\boldsymbol{y})$ and $P(\boldsymbol{x}^{(2)}|h, \boldsymbol{x}^{(1)})$ are modeled as conditional fields which will be explained in Sections 2.2 and 2.3. In (1), computing the sum over all the possible configurations of $\boldsymbol{x}^{(1)}$ is a NP-hard problem. One way to reduce complexity is to do inference in layer 1 until equilibrium is reached and then using this configuration $\hat{\boldsymbol{x}}^{(1)}$ as input to the next layer, i.e., $P(\boldsymbol{x}^{(1)}|\boldsymbol{y}) = \delta(\boldsymbol{x}^{(1)} - \hat{\boldsymbol{x}}^{(1)})$. However, by doing this, one loses the power of modeling the uncertainty associated with the labels in layer 1, which was included explicitly in (1) through $P(\boldsymbol{x}^{(1)}|\boldsymbol{y})$. In principle, one can use Monte Carlo sampling or a variational approach to approximate the sum in (1), but they may be computationally expensive. In this work, instead, we wanted to examine what could be achieved by making a very simplifying assumption, where along with the equilibrium configuration, we also propagate the uncertainty associated with it to the next layer. We use the sitewise maximum marginal configuration as $\hat{\boldsymbol{x}}^{(1)}$. Let the marginals at each site i be $b_i(x_i^{(1)}) = \sum_{\boldsymbol{x}^{(1)}\setminus x_i^{(1)}} P(\boldsymbol{x}^{(1)}|\boldsymbol{y})$, and $\boldsymbol{b}(\boldsymbol{x}^{(1)}) = \{b_i(x_i^{(1)})\}_{i\in S^{(1)}}$. The belief set, $\boldsymbol{b}(\boldsymbol{x}^{(1)})$ is propagated as an input to the next layer. Note that the configuration $\hat{\boldsymbol{x}}^{(1)}$ can be obtained directly from $\boldsymbol{b}(\boldsymbol{x}^{(1)})$ by taking its sitewise maximum configuration. Thus, in the future, we will omit explicit conditioning on $\hat{\boldsymbol{x}}^{(1)}$. Now, we can write

$$P(\boldsymbol{x}|\boldsymbol{y}) \approx \sum_{h\in\mathcal{H}} P(\boldsymbol{x}^{(2)}|h, \boldsymbol{b}(\boldsymbol{x}^{(1)}))P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)})). \qquad (2)$$

Note that both terms in the summation implicitly include the transition probabilities $P(x_r^{(2)}|\hat{x}_i^{(1)})$. For the first term, these are absorbed in the unary potential of the conditional field in layer 2 as explained in Section 2.3. Section 2.4 will describe a simple design choice for $P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)}))$. We first describe the modeling of the conditional field in layer 1.

2.2. Conditional Field - Layer 1

The conditional distribution of the labels given the observed data, i.e., $P(\boldsymbol{x}^{(1)}|\boldsymbol{y})$ is directly modeled as a homogeneous pairwise conditional random field proposed by [10] as,

$$P(\boldsymbol{x}^{(1)}|\boldsymbol{y}) = \frac{1}{Z} \prod_{i\in S^{(1)}} \phi(x_i^{(1)}, \boldsymbol{y}) \prod_{i,j\in\mathcal{N}_i} \psi(x_i^{(1)}, x_j^{(1)}, \boldsymbol{y}),$$

where Z is a normalizing constant known as the partition function, $\mathcal{N}_i$ is the set of neighbors of site i. Here, $\phi(x_i^{(1)}, \boldsymbol{y})$ and $\psi(x_i^{(1)}, x_j^{(1)}, \boldsymbol{y})$ are the unary and the pairwise potentials.

Generalizing the binary form in [7][14] to multiclass problems, we model the unary potential as,

$$\log \phi(x_i^{(1)}, \boldsymbol{y}) = \sum_{k\in\mathcal{L}^{(1)}} \delta(x_i^{(1)} = k) \log P'(x_i^{(1)} = k|\boldsymbol{y}), \qquad (3)$$

where $\delta(x_i^{(1)} = k)$ is 1 if $x_i^{(1)} = k$ and 0 otherwise, and $P'(x_i^{(1)} = k|\boldsymbol{y})$ is an arbitrary domain-specific discriminative classifier. This form of unary potential gives us the desired flexibility to integrate different applications preferring different types of local classifiers in a single framework. Let $\boldsymbol{h}_i(\boldsymbol{y})$ be a feature vector (possibly in a kernel-projected space), that encodes appearance based features for the i^{th} site (a pixel, a patch or an object). To model $P'(x_i^{(1)} = k|\boldsymbol{y})$, in this paper we generalize the logistic classifier used in [7] to a softmax function,

$$P'(x_i^{(1)}\!= k|\boldsymbol{y}) = \begin{cases} \dfrac{\exp(\boldsymbol{w}_k^T \boldsymbol{h}_i(\boldsymbol{y}))}{1+\sum_{l=1}^{|\mathcal{L}^{(1)}|-1} \exp(\boldsymbol{w}_l^T \boldsymbol{h}_i(\boldsymbol{y}))} & \text{if } k < |\mathcal{L}^{(1)}| \\[2ex] \dfrac{1}{1+\sum_{l=1}^{|\mathcal{L}^{(1)}|-1} \exp(\boldsymbol{w}_l^T \boldsymbol{h}_i(\boldsymbol{y}))} & \text{if } k = |\mathcal{L}^{(1)}| \end{cases}$$

Here, $\boldsymbol{w}_k$ are the model parameters for $k = 1 \ldots |\mathcal{L}^{(1)}| - 1$. For a $|\mathcal{L}^{(1)}|$ class classification problem, one needs only $|\mathcal{L}^{(1)}| - 1$ independent hyperplanes.

The pairwise potential predicts how the labels at two sites should interact given the observations. Generalizing the interaction potential in [7] for multiclass field,

$$\log \psi(x_i^{(1)}, x_j^{(1)}, \boldsymbol{y}) = \sum_{k,l\in\mathcal{L}^{(1)}} \boldsymbol{v}_{kl}^T \boldsymbol{\mu}_{ij}(\boldsymbol{y}) \delta(x_i^{(1)} = k)\delta(x_j^{(1)} = l) \qquad (4)$$

where, $\boldsymbol{\mu}_{ij}(\boldsymbol{y})$ is the pairwise feature vector, and $\boldsymbol{v}_{kl}$ are the model parameters. For example, in the case of object detection, the vector $\mu_{ij}(\boldsymbol{y})$ encodes the pairwise features required for modeling geometric and possibly photometric consistency of a pair of parts. The sitewise label smoothing can be achieved by forcing $\mu_{ij}(\boldsymbol{y})$ to be 1.

2.3. Conditional Field - Layer 2

The formulation of the conditional field for layer 2 can be obtained in the same way as described in the previous section by changing the observations to $\boldsymbol{b}(\boldsymbol{x}^{(1)})$, the set of sites to $S^{(2)}$, and the label set to $\mathcal{L}^{(2)}$. The main difference lies in the form of the unary potential. Each node $r \in S^{(2)}$ in this layer receives beliefs as input from the nodes contained in set $S_r^{(1)}$ from the layer below. Taking into consideration the transition probabilities on the directed links between node r and the nodes in $S_r^{(1)}$, the unary potential can be written as,

$$\log \phi(x_r^{(2)}, \boldsymbol{b}(\boldsymbol{x}^{(1)})) = \sum_{k \in \mathcal{L}^{(2)}} \Big\{ \delta(x_r^{(2)} = k) \Big(\log P'(x_r^{(2)} = k | \boldsymbol{b}(\boldsymbol{x}^{(1)})) + \frac{1}{|S_r^{(1)}|} \sum_{i \in S_r^{(1)}} \log P(x_r^{(2)} = k | \hat{x}_i^{(1)}) \Big) \Big\}$$

Here, $|S_r^{(1)}|$ is a normalizer that takes into account the different cardinalities of sets $S_r^{(1)}$.

2.4. Modeling Partitioning

The distribution $P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)}))$ should be designed such that it gives high weight to a partition $h \in \mathcal{H}_v$, given the belief set from layer 1. Since a good partition should drive all the nodes in a set $S_r^{(1)}$ to take the same true labels, the conditional distribution over the partitions is modeled as,

$$P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)})) \propto \Big\{ \prod_{r \in S^{(2)}} \Big[\max_{x_r^{(2)} \in \mathcal{L}^{(2)}} \prod_{i \in S_r^{(1)}} \sum_{x_i^{(1)} \in \mathcal{L}^{(1)}} \Big(b_i(x_i^{(1)}) P(x_r^{(2)} | x_i^{(1)}) \Big) \Big]^{1/|S_r^{(1)}|} \Big\}^{1/|S^{(2)}|}$$

The term in the product over i is the probability that the node r, connected to site i, will take label $x_r^{(2)}$. Also, $|S_r^{(1)}|$ and $|S^{(2)}|$ compensate for the differences in the number of nodes in set $S_r^{(1)}$ and the overall number of nodes induced by the partition respectively.

3. Parameter Learning and Inference

The set of parameters Θ, to be learned in the hierarchical model, includes the parameters of the conditional fields at layer 1 and layer 2, and the transition probability matrices $P(x_r^{(2)} | \hat{x}_i^{(1)})$. The field parameters for each layer are the parameters of the unary and pairwise potentials i.e., $\theta^{(\alpha)} = \Big\{ \boldsymbol{w}_k^{(\alpha)}, \boldsymbol{v}_{kl}^{(\alpha)} \Big\}_{\forall k,l}^{\alpha=1,2}$.

Given M i.i.d. labeled training images, the maximum likelihood estimates of the parameters are given by maximizing the log-likelihood $L(\Theta) = \sum_{m=1}^{M} \log P(\boldsymbol{x}^m | \boldsymbol{y}^m, \Theta)$, where the conditional distribution in the sum for each image m is given by (1). Since this likelihood is hard to evaluate, following the assumption made in Section 2.1, we use a sequential learning approach in which, first the parameters of layer 1 are estimated separately. Fixing these estimates, the parameters of the next layer and the transition matrices are estimated by maximizing the likelihood for the conditional distribution given in (2). Although suboptimal, the drawbacks of the sequential approach are somewhat moderated by the fact that the partition functions for the fields in the two layers are decoupled due to the directed connections.

Starting with parameter learning in layer 1, since the labels at this layer are not known, we assign pseudo labels $\boldsymbol{x}^{(1)}$ on S using the true labels $\boldsymbol{x}$. In the image labeling applications, since the nodes at both the layers take the labels from the same set, one can assume the pseudo labels to be the same as the true labels. For object detection, where the labels at layer 1 are part identifiers rather than being object identifiers, one possible way to generate pseudo labels will be to use soft clustering on the object parts and assign a part label to each node as in [8]. It is clear that the labels generated in this way are going to be noisy. That is where the hierarchical model becomes more relevant, where the top layer refines the label estimates from the layer below and the directed connections incorporate the transition probabilities from the noisy labels to the true labels.

To learn the parameters of the conditional field in layer 1 using gradient ascent, the derivative of the log-likelihood from the distribution $P(\boldsymbol{x}^{(1)} | \boldsymbol{y}, \theta^{(1)})$ can be written as,

$$\frac{\partial l(\theta^{(1)})}{\partial \boldsymbol{w}_k^{(1)}} = \sum_m \sum_{i \in S^{(1)}} \Big(\delta(x_i^{(1)m} = k) - \Big\langle \delta(x_i^{(1)} = k) \Big\rangle \Big) \boldsymbol{h}_i(\boldsymbol{y}^m) \tag{5}$$

$$\frac{\partial l(\theta^{(1)})}{\partial \boldsymbol{v}_{kl}^{(1)}} = \sum_m \sum_{i \in S^{(1)}} \sum_{j \in \mathcal{N}_i} \Big(\delta(x_i^{(1)m} = k) \delta(x_j^{(1)m} = l) - \Big\langle \delta(x_i^{(1)} = k) \delta(x_j^{(1)} = l) \Big\rangle \Big) \boldsymbol{\mu}_{ij}(\boldsymbol{y}^m), \tag{6}$$

where $\langle . \rangle$ denotes expectation with respect to the distribution $P(\boldsymbol{x}^{(1)} | \boldsymbol{y}^m, \theta^{(1)})$. Generally the expectation in (5) and (6) cannot be computed exactly due to the exponential number of configurations of $\boldsymbol{x}^{(1)}$. In this work, we estimate expectations using the pseudo-marginals returned by loopy Belief Propagation (BP) [3].

The transition probability matrices were assumed to be the same for all the directed links in the graph to avoid overfitting. The entries in this matrix were estimated using the normalized expected counts of transition from $\hat{x}_i^{(1)}$ to $x_r^{(2)}$, which are known at the training time. Note that the counts are computed using the refined label estimates $\hat{x}_i^{(1)}$ obtained directly from $\boldsymbol{b}(\boldsymbol{x}^{(1)})$.

Given $\boldsymbol{b}(\boldsymbol{x}^{(1)})$ and $P(x_r^{(2)} | \hat{x}_i^{(1)})$, the field parameters of layer 2 i.e., $\theta^{(2)}$ were obtained by maximizing the lower bound on the log likelihood of (2),

$$l'(\theta^{(2)}) \geq \sum_m \sum_h \Big\{ P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)m})) \log P(\boldsymbol{x}^{(2)m} | h, \boldsymbol{b}(\boldsymbol{x}^{(1)m}), \theta^{(2)}) \Big\} \tag{7}$$

The derivatives of the above lower bound also have similar forms as in (5) and (6) except that the gradients are now the expectations with respect to $P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)})$. In addition, the gradient for the unary parameters $w_k^{(2)}$ at a site r will have the features scaled by the product of transition probabilities for all the nodes in $S_r^{(1)}$. To deal with the problem of summing over h, in principle, one can use full MCMC sampling. However, by using a data-driven heuristic described in Section 4, samples from high probability regions of $P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)})$ can be obtained using local search. Usually, the resulting partitions will not be restricted to the valid space $\mathcal{H}_v$. In that case, the training label at node r in layer 2 is obtained by using a majority vote of labels at the nodes in $S_r^{(1)}$.

For inference, in this work we used the sum-product version of loopy BP to find the maximum marginal estimates of the labels on the image sites. The desired label estimates for each node i in set S can be obtained as,

$$\hat{x}_i = \arg\max_k \sum_{h,r:i\in S_r^{(1)}} \Big\{ P_r(x_r^{(2)} = k|h, \boldsymbol{b}(\boldsymbol{x}^{(1)})) \\ P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)})) \Big\}, \quad (8)$$

where the sum is carried out over all h by picking the site $r : i \in S_r^{(1)}$ for each h, and $P_r(.)$ is the marginal for site r in layer 2 estimated using loopy BP.

4. Experiments and Discussion

We conducted experiments to test the capability of the proposed hierarchical approach to incorporate three different types of contextual interactions i.e., *region-region*, *object-region* and *object-object*, as described in Section 1. Four datasets for two different applications (image labeling and contextual object detection) were used for testing. For the object detection experiments, the aim was to investigate if the performance of the existing classifiers could be improved by feeding their outputs in the hierarchical model.

4.1. Region-Region Interactions

The first set of experiments was conducted on the 'Beach' dataset from [9], which contains a collection of consumer photographs. The goal was to assign each image pixel one of the 6 class labels: {*sky*, *water*, *sand*, *skin*, *grass*, *other*}. This dataset is particularly challenging due to wide within-class variance in the appearance of the data (see Figure 5 or [9] for more images). The dataset contained 123 images, each of size 124×218 pixels. This set was randomly split into a training set of 48 images and a test set of 75 images.

The layer 1 of the proposed hierarchical model implemented the smoothness of pixel labels as the local context. Hence, the sites in layer 1 were the image pixels and the neighborhood was defined to be the 4-nearest neighbors on a grid. Similar to [9], three HSV color features and two texture features, based on the eigenvalues of the second moment matrix, gave a 5 dim unary feature vector. Further, we used a quadratic kernel to obtain a 21 dim feature vector $\boldsymbol{h}_i$. To implement label smoothing, the pairwise feature vector $\boldsymbol{\mu}_{ij}$ was set to 1, resulting in a Potts model i.e., $\boldsymbol{v}_{kl} = 0$ if $k \neq l$. The parameters of layer 1 i.e., $\theta^{(1)} = \{\boldsymbol{w}_k^{(1)}, \boldsymbol{v}_{kk}^{(1)}\}_{\forall k}$ were all learned simultaneously using the maximum likelihood procedure described Section 3. The training time was about 10 min on a 2.8 GHz Pentium class processor.

Before proceeding to layer 2, we describe how we do local sampling of partition h in a high probability region of $P(h|\boldsymbol{b}(\boldsymbol{x}^{(1)}))$. As explained in Section 2.4, good partitions are those that promote homogeneous labeling within a region. So, given the beliefs from layer 1, first a binary map is generated for each class by thresholding the pixel-wise beliefs at a small value. Then, a partition is obtained by simply intersecting these binary maps for all the classes, i.e., by dividing bigger regions into smaller ones whenever there is an overlap between regions from any two maps. By varying the threshold for generating the binary maps, one can have the desired number of samples. We observed that even less than 5 samples were sufficient to give good results. This was because the beliefs from layer 1 are smoothed due to message passing between the nodes in this layer while implementing the local context.

The layer 2 encodes interactions among different regions given the beliefs at layer 1 and a partition. Each region of the partition is a site in layer 2. Note that the sites are not placed in a regular grid as in layer 1. For this dataset, the number of sites at layer 2 varied from 13 to 49 for different images. Since we want every region in the scene to influence every other region, each node in the graph was connected to every other node. The computations over these complete graphs are still efficient because of the small number of nodes in the graph. The unary feature vector for each node r consists of normalized product of beliefs from all the sites i in $S_r^{(1)}$ and the normalized centroid location of the region r. This gives an 8 dim feature vector. Further, quadratic transforms were used to obtain a 44 dim vector $\boldsymbol{h}_i$. Similar to [13], we use pairwise features between regions to be binary indicator attributes. These were: a region is *above*, *beside* or *enclosed* within another region. The maximum likelihood learning took about 5 minutes.

Two example results from the test set are shown in Figure 5. The top row shows that good accuracy is obtained even for the pixels from the *other* class which has traditionally been hard to model because of large within class variations. Table 1 gives a quantitative comparison of the results on the test set. The use of the local context (label smoothing) improves the accuracy slightly ('Layer 1' in Table 1) over

Table 1. Pixelwise classification accuracy (%) for image labeling on two different datasets. Final results of the hierarchical approach are shown in bold. The column 'Others' gives the results for the techniques proposed by other researchers.

Datasets	Softmax	Layer1	Full	MRF	Others
Beach	62.3	63.8	**74.0**	61.5	64.0 [9]
Sowerby	85.4	85.8	**89.3**	81.8	89.5 [5]

the softmax which uses no context. However, the main use of the local context is to propagate improved beliefs and partitions to layer 2. The full hierarchical model ('Full' in Table 1) performs significantly better than the others. The time taken for inference was about 6 sec for each image. For the MRF, results were obtained using the Potts model.

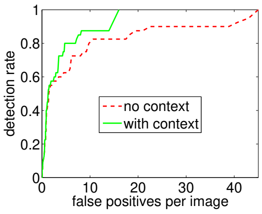
Next, the hierarchical model was applied to the standard Sowerby dataset. The dataset contained 104 images (64×96 pixels). The training and the test set contained 60 and 44 images respectively. As used by [5], the CIE Lab color features and oriented DoG filters based texture features gave a 30 dim feature vector that was used as input to layer 1. The rest of the features, parameter learning and inference were the same as for our implementation on the Beach dataset. Figure 5 shows two typical test results. Note the road marking in the bottom image, which is preserved in the final result even though layer 1 tends to smooth it out. The quantitative comparisons are given in Table 1. Note that we achieve almost the same accuracy as reported in [5] even though their technique is specifically tuned for the image labeling problems, while our approach is more general, integrating different applications in a single framework.

4.2. Object-Region Interactions

We conducted the next set of experiments on a building/road/car dataset from [15].[1] The dataset contained 31 images, each of size less than 100×100 pixels. The size and pose of the object (car) was roughly the same in all the images. As shown in Figure 6, the local appearance of cars is impoverished due to low resolution, making the car detection hard using stand-alone detectors. In addition, high variability in the appearance of the building data also makes it difficult to disambiguate them from roads just on the basis of intensity and texture features. However, the relationships among the object (car) and the two regions (building and road) provide strong context to improve the detection of all the three entities simultaneously.

For object detection, layer 1 models the relationship among parts of an object. Ideally, in layer 1 one can implement a CRF on object parts similar to [12][8]. However, to investigate if our framework can be used for improving

[1] Only a partial dataset was available in the public domain.



		R	B
No Context	R	14.12	22.86
	B	6.48	56.54
With Context	R	36.02	0.96
	B	0.99	62.03

Figure 3. Left: The ROC curves for contextual car detection compared to a boosting based detector. Right: Confusion matrices (as % of overall pixels) for building and road detection. Rows contain the ground truth. No context implies the output of the Softmax classifier.

the performance of a standard boosting-based detector, we use the detector output in layer 1. Rectangular patches centered at the locations that have a score above a threshold are designated as sites for both layer 1 and 2. The threshold is chosen to be small enough to make the false negatives relatively rare. Of course, it increases the false positives considerably. So, the question is: can our framework handle a large number of false positives?

In the hierarchical model, the set of sites $S^{(1)}$ in layer 1 contains all the image pixels and the object patches. The neighborhood structure for the pixels was 4 nearest neighbors. Since each object patch represents a possible hypothesis about the full object, there is no interaction among these patches in layer 1. The set of sites in layer 2, $S^{(2)}$, consists of image regions and the same object patches as in layer 1. Note that the sites in $S^{(2)}$ induce a partition on the nodes in $S^{(1)}$. The label sets $\mathcal{L}^{(1)}$ and $\mathcal{L}^{(2)}$ for the sites in the two layers were the same as $\{building, road\}$ for pixels and regions, and $\{car, background\}$ for the patches.

The features used by layers 1 and 2 for image pixels and regions were the same as described for the Sowerby dataset in the previous section. The output of the object detector was used as a feature for a patch in layer 2. All the nodes in layer 2 were connected with each other inducing a complete graph. The pairwise features between the object patches and the regions in layer 2 were simply the difference in the coordinates of the centroids of a region and a patch.

In all the experiments we used a detector trained by gentle boosting as the base detector [15]. The classification results for two typical examples from the test set are given in Figure 6. The classification accuracy of building and road detection goes up from 70.66% to 98.05% as shown in Figure 3. Also, the ROC curve for the car detection shows that the number of false positives is reduced considerably compared to the base detector.

4.3. Object-Object Interactions

The final set of experiments was conducted on the monitor/keyboar/mouse dataset from [15], which contained 164 images of size less than 100×100 pixels each. The dataset

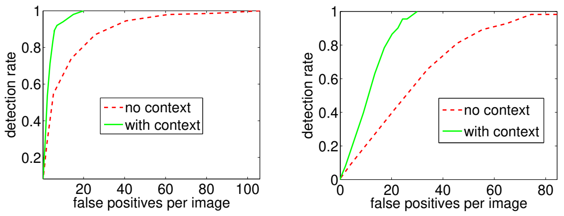


Figure 4. The ROC curves for the detection of keyboard (left) and mouse (right). Relatively high false alarm rates for mouse were due to very small size of mouse (about 8×5 pixels) in the input images.

was randomly split in half to generate the training and the test sets. The main challenge in the dataset was the detection of the keyboard and the mouse, which spanned only a few pixels in the images. In this section we show that by taking interactions among the three objects, one can decrease the false alarms in detection significantly.

For each object, we use a detector which was also trained using gentle boosting as the base detector. Since the size of the mouse in the input images was very small (on average about 8×5 pixels), the boosting based detector could not be trained for the mouse. Instead, we implemented a simple template matching detector by learning a correlation template from the training images. A patch at a location where the output of any of the three detectors is higher than a threshold, represents a site in $S^{(1)}$. The set of sites $S^{(2)}$ in layer 2, was the same as in layer 1 , indicating a trivial partition.. The label set for the sites in $S^{(1)}$ and $S^{(2)}$ was $\{monitor, keyboard, mouse, background\}$. Since layer 1 uses the output of a standard object detector, interactions among sites take place only at layer 2.

The unary features at layer 2 consisted of the score from each detector yielding a 3 dim feature vector. The difference of coordinates of the patch centers resulted in a 2 dim pairwise feature vector. Each node was connected to every other node in this layer. Figure 7 shows a typical result from the test set. It is clear that the false alarms were reduced considerably in comparison to the base detector. The use of context did not change the results for the monitor, since the base detector itself was able to give good performance. This is reasonable because one hopes that context will be more useful when the local appearance of an object is more ambiguous. The ROC curves for the keyboard and the mouse detection are compared with the corresponding base detectors in Figure 4.

5. Conclusions and Future Work

We have presented a unified approach to modeling different types of contexts in images using a hierarchical field formulation. The benefits of the proposed approach, in spite of a few simplistic assumptions, were demonstrated on the problems of image labeling and contextual object detection. In the future, we will explore the use of variational approximations to relax some of the assumptions made in this work. We also plan to develop efficient ways of learning the parameters of the two layers simultaneously. Finally, it will be interesting to explore the possibility of adding other layers in the hierarchy, which could encode more complex relations between different scenes in a video, leading to event or activity recognition.

Acknowledgments

Our gratitude to BAE Systems and X. He for the Sowerby dataset and the features, and A. Torralba for the object detection database and the boosting code.

References

[1] X. Feng, C. K. I. Williams, and S. N. Felderhof. Combining belief networks and neural networks for scene segmentation. *IEEE Trans. PAMI*, 24(4):467–483, 2002.

[2] M. Fink and P. Perona. Mutual boosting for contextual inference. *Neural Information Processing Systems*, 2004.

[3] B. Frey and D. J. C. Mackay. A revolution: Belief proapagation in graphs with cycles. *NIPS*, 10, 1997.

[4] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution and the bayesian restoration of images. *IEEE Trans. on Patt. Anal. Mach. Intelli.*, 6:721–741, 1984.

[5] X. He, R. Zemel, and M. Carreira-Perpinan. Multiscale conditional random fields for image labelling. *CVPR*, 2004.

[6] G. E. Hinton, S. Osindero, and K. Bao. Learning causally linked markov random fields. *AI & Statistics*, 2005.

[7] S. Kumar and M. Hebert. Discriminative fields for modeling spatial dependencies in natural images. *NIPS*, Dec 2003.

[8] S. Kumar and M. Hebert. Multiclass discriminative fields for parts-based object detection. *Snowbird Workshop*, 2004.

[9] S. Kumar, A. C. loui, and M. Hebert. An observation-constrained generative approach for probabilistic classification of image regions. *Image and Vision Computing, Special Issue on Generative Models Based Vision*, 21:87–97, 2003.

[10] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *In Proc. ICML*, 2001.

[11] M. Mignotte, C. Collet, P. Prez, and P. Bouthemy. Sonar image segmentation using a hierarchical mrf model. *IEEE Trans. on Image Proc.*, 75(2):1216–1231, 2000.

[12] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. *NIPS*, December 2004.

[13] A. Singhal, J. Luo, and W. Zhu. Probabilistic spatial context models for scene content understanding. *CVPR*, 2003.

[14] M. Szummer and Y. Qi. Contextual recognition of hand-drawn diagrams with conditional random fields. *Workshop on Frontiers in Handwriting Recognition*, 2004.

[15] A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. *NIPS*, December 2004.

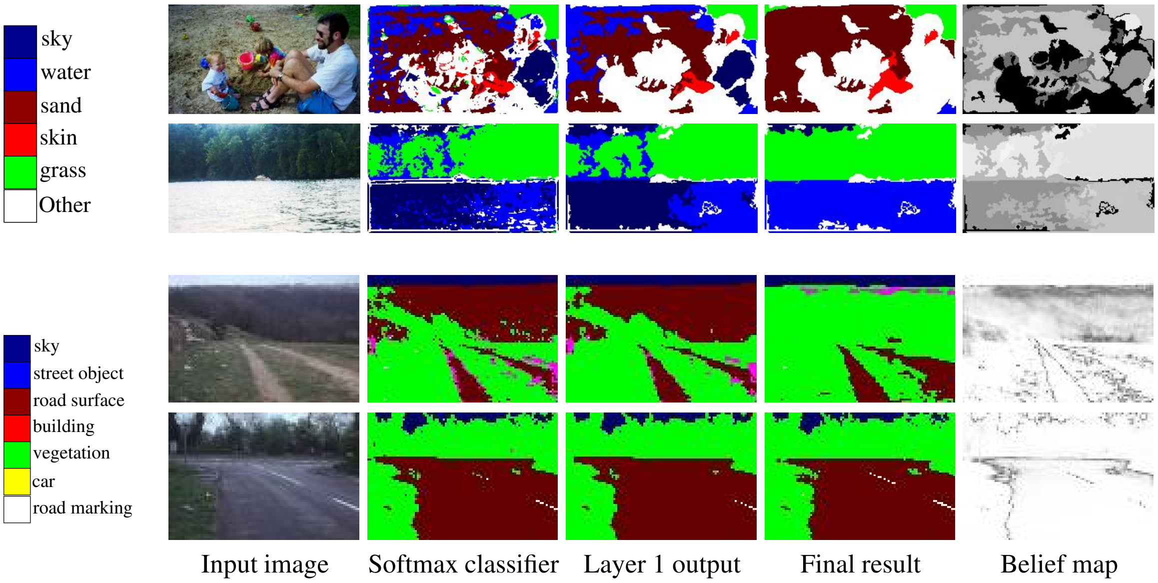


Figure 5. Results on the Beach dataset (top two rows) and the Sowerby dataset (bottom two rows) using context based on *region-region* **interactions. Note the correct classification of 'other' class in top row. In the bottom row, road markings are preserved in the final result. In a belief map, higher intensity indicates higher confidence.**

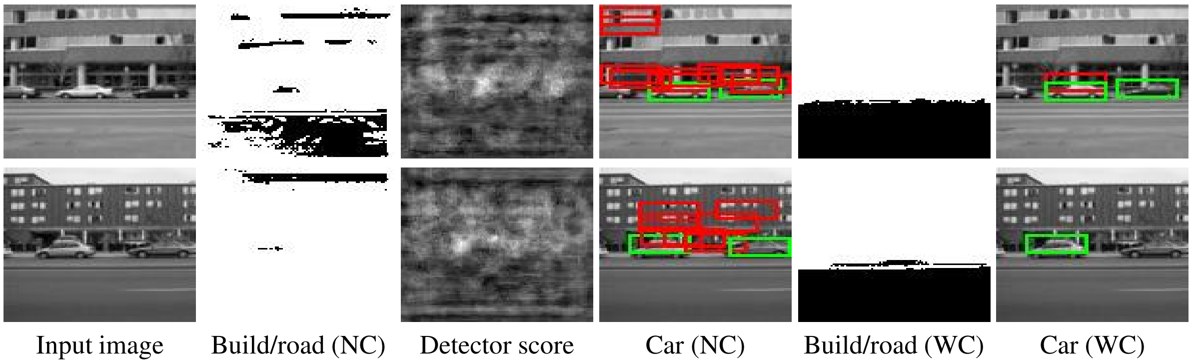


Figure 6. Detection results for buildings, road and car using context based on *object-region* **interactions. 'Build' - Building, NC - No Context, WC - With Context. Detector score shows the output of the base detector. Black indicates 'road' and white 'buildings'. Green and red indicate true detections and false alarms respectively.**

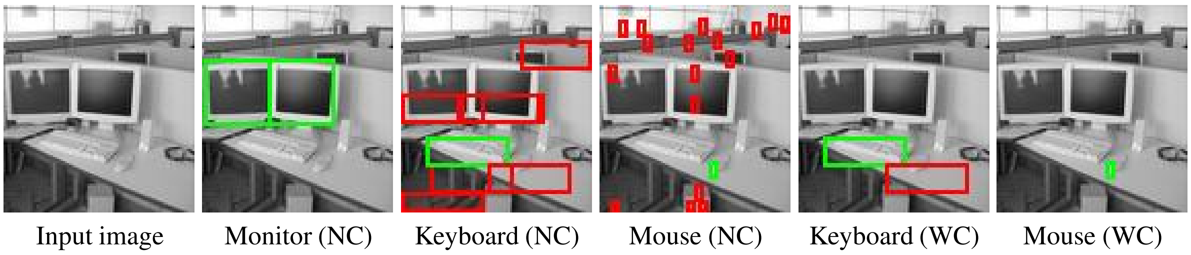


Figure 7. Detection results for monitor, keyboard and mouse using context based on *object-object* **interactions. NC - No Context, WC - With Context. Monitor detection was good with the base detector itself due to less appearance ambiguity. Note the impoverished appearances of the keyboard and the mouse. The detection color coding is the same as above.**