Unsupervised Learning of Hierarchical Spatial Structures In Images

Devi Parikh
Carnegie Mellon University
Pittsburgh, PA, USA
dparikh@cmu.edu

C. Lawrence Zitnick
Microsoft Research
Redmond, WA, USA
larryz@microsoft.com

Tsuhan Chen
Cornell University
Ithaca, NY, USA
tsuhan@cornell.edu

Abstract

The visual world demonstrates organized spatial patterns, among objects or regions in a scene, object-parts in an object, and low-level features in object-parts. These classes of spatial structures are inherently hierarchical in nature. Although seemingly quite different these spatial patterns are simply manifestations of different levels in a hierarchy. In this work, we present a unified approach to unsupervised learning of hierarchical spatial structures from a collection of images. Ours is a hierarchical rule-based model capturing spatial patterns, where each rule is represented by a star-graph. We propose an unsupervised EM-style algorithm to learn our model from a collection of images. We show that the inference problem of determining the set of learnt rules instantiated in an image is equivalent to finding the minimum-cost Steiner tree in a directed acyclic graph. We evaluate our approach on a diverse set of data sets of object categories, natural outdoor scenes and images from complex street scenes with multiple objects.

1. Introduction

Our visual world is far from random, and demonstrates highly predictable spatial patterns. These patterns may be among high-level entities such as objects in a scene (keyboards are usually below monitors), regions in a scene (sky is usually above grass), parts within an object (the engine is usually in between the two wheels of a motorcycle), or among low-level features within object-parts. These classes of spatial structures are inherently hierarchical in nature, as shown in Figure 1.

Previous work has used each of these levels for various tasks. For instance, patterns among object parts are used to form compositional models to aid in object recognition [1–5]. The relationship of objects are used to capture semantic contextual information for robust object detection/localization or image labeling [6–10]. Clusters of low-level features have been shown to be more discriminative than single features for object recognition [11, 12].

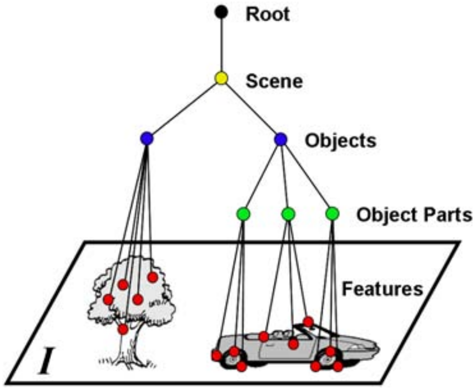

Figure 1: An illustration of the hierarchical spatial patterns present in an image.

Although seemingly quite different, these various forms of spatial patterns can simply be viewed as manifestations of different levels in a hierarchy [13–20]. It is clear that extracting this hierarchy of spatial structures could provide rich information to facilitate several vision tasks such as image classification, localization, object recognition, and others. However, learning such a hierarchy would be prohibitive if it required extensive supervision and laborious labeling of images. In this paper, we propose a unified approach to unsupervised learning of hierarchical spatial structures [16] from a generic collection of images. We describe each spatial pattern in the hierarchy as a rule. Each rule is represented by a star-graph [3], where a child of the star-graph may be a low-level feature or another star-graph (i.e. rule), thus forming a hierarchy.

The inference problem is to determine the subset (hierarchy) of learnt rules that best explains the observed features in a given image. We impose that the set of rules that can be used to explain the image forms a tree. That is, each feature or rule can only be explained by a single parent rule. We show that determining the optimal tree that maximizes the likelihood of the image is equivalent to finding the minimum cost Steiner tree [21] in a directed acyclic graph (DAG). This being an NP hard problem to solve, we use the

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

approximation algorithm proposed by Charikar *et al.* [22]. It should be noted that the structure of the optimal tree (as well as the underlying DAG) may be different for different images, and is determined automatically during inference. For computational feasibility, we reduce the number rules considered for inclusion in the tree using a voting scheme.

The learning task is to infer a set of rules from a given collection of images in an unsupervised manner. The number of rules, the structure and parameters of each rule, and number of children of each rule are learnt automatically. Ours is an EM-style algorithm where we initialize our model (a set of rules), infer instances of them across an image collection, and update the rule parameters.

We evaluate our approach on a diverse collection of datasets ranging from a subset of the Caltech101 object categories [23], outdoor natural scene categories [24], as well as complex street scenes from the LabelMe dataset [25]. We present qualitative results through visualizations of the rules learnt and the hierarchies inferred in images. To demonstrate the behavior of the rules in the hierarchy, we quantify at each level the localization and categorization abilities of the rules. We find that higher level rules are often specific to object categories, while lower-level rules can be shared between categories. To demonstrate the utility of the learnt spatial hierarchies, we perform unsupervised clustering of the images into object categories. We report comparable accuracies to the state-of-the-art techniques.

We discuss related work next in Section 2. Sections 3, 4 and 5 describe our model, our method for inference given an image and a learnt model, and our unsupervised approach to learning the proposed model. Section 6 describes our experiments and presents results. Section 7 raises some points of discussion and future work, followed by a conclusion in Section 8.

2. Related Work

Modeling image hierarchies and spatial structures has a long history in computer vision [13, 19, 26]. The works vary both in their representations used to encode the spatial information and their approaches for learning. We discuss both the *representations* and *learning* algorithms in turn.

Representation: Different representations based on global histograms [12], graphs [1–3] and hierarchies [13–17, 19, 20] have been explored in previous works. The bag-of-words model [12] uses a histogram representation which is efficient to compute and match. Graph-based methods have been proposed for recognizing individual objects using Constellation models [2] and star-graphs [1] where pairwise spatial location statistics are captured. Graphs have also been used for context modeling in street scenes by Hoiem *et al.* [6] and segment labeling [8, 9]. Numerous hierarchical methods have been proposed. Several approaches use a fixed number of levels such as Kumar and Hebert [7] that use a two level hierarchy to model context in classification. Sudderth *et al.* [18] use hierarchies for part sharing and modeling scenes, while Murphy *et al.* [15] model the spatial relationship of objects in scenes. Other models use hierarchies of arbitrary depth. These methods can be used to model individual objects, e.g. the segment tree approach of Todorovic and Ahuja [27, 28], the method of Zhu *et al.* [29] for deformable objects and the object part discovery approach of Fidler *et al.* [30]. Other approaches attempt to model relationships between objects as well as object parts within a hierarchy. These include the stochastic grammar approach of Zhu *et al.* [20] using And-Or graphs and the hierarchical representation of Parikh *et al.* [16] used to describe semantic relationships among objects. Finally, some approaches attempt to create hierarchies of object categories based on object appearances [31].

Learning: The level of supervision varies among the various approaches proposed in the literature. Supervised techniques [1, 20] require objects to be labeled in the images for learning. A less restrictive class of techniques called weakly-supervised [2, 5] only requires the knowledge of whether an object is present in the image or not. Several of the existing hierarchical representations are learnt in a supervised [28, 32, 33] or semi-supervised way [29, 34], or learn only part of the model from training data. For instance a structure of the hierarchy may be given and only the parameters are learnt from data [35], or the entire model is given and the task is to only infer the model in images [36]. Finally, unsupervised techniques require only a set of unlabelled images for learning. Unsupervised techniques have been proposed for bag-of-words models [12] and models that learn spatial structure [4, 37, 38].

3. Model

Our model is a hierarchy of rules. Each rule describes a spatial pattern, and is represented as a star-graph. Just as in language modeling, a sentence is modeled as a parse tree, we consider an image to have an associated tree formed by the subset of rules that best explains the observed features in the image. The leaves of the tree are the observed features, and the intermediate nodes are the higher-order spatial-patterns (instantiations of the rules), which we call image-parts. An image-part could correspond to higher order features, object-parts, objects, groups of objects or a scene. The inference task is to find the set of image-parts that best explain the features in an image, given a set of rules. We first introduce some notation.

Each feature $f \in F$ is an instantiation of a codeword at a certain location, denoted as a pair (c_f, l_f), where $c_f \in C$ and l_f is the location of feature f. C is the dictionary or vocabulary of all possible discrete appearances of the low-level features (codewords). Each rule r, as shown in Figure 4, is defined by a certain structure and associated parameters denoted by θ_r. A rule defines a star graph with associated children $Ch(r)$. A child $x \in Ch(r)$ may be either a codeword c, or a another rule r, i.e. $x \in C \cup R$. Allowing rules to be children of rules enables the formation of hierarchies. Not all children in a rule may be instantiated in an image. The parent of x is denoted as $Pa(x)$. The rule parameters θ_r contain both the occurrence probability for a child $\Pr(x|r)$ and the location probability $\Pr(l_x|r)$, where l_x is the location of the child relative to the parent. We model the location probability using a Gaussian with an associated mean and covariance.

Finally, we define a *background* or *prior* image-level rule, indicated by r_0, whose definition encompasses all codewords and rules i.e. $Ch(r_0) = C \cup R$. The parameters for this rule are the prior probabilities (for instance, the marginal probability of observing a certain codeword or rule at a certain location in an image). From here on, we include r_0 in the set of all rules R. r_0 acts as the root node, similar to the node corresponding to a sentence in language modeling.

We define the set of instantiated image-parts as H. A tree $T = \{V, E\}$ for image I consists of a set of vertices V and edges E. The vertices are the union of the image-parts and features, i.e. $V = H \cup F$. The edges E indicate the set of children $Ch(v)$ for each vertex $v \in V$. If v corresponds to a feature then $Ch(v) = \emptyset$. If v corresponds to a rule r_v, the rule's children $Ch(r_v)$ may or may not be instantiated. A child $x \in Ch(r_v)$ is instantiated if $x \in Ch(v)$, i.e. x is instantiated if there exists a vertex $v' \in Ch(v)$ corresponding to x. The parent of a vertex v is defined as $Pa(v) \in H$, and its location in the image by l_v.

With this notation, we can now introduce our model. Given an image with a set of observed features F, our goal is to find a tree T such that each feature f corresponds to a leaf in the tree. Each feature may only be explained once, i.e. it may only have one parent, and each feature must be directly or indirectly attached to the root node corresponding to rule r_0. The intermediate nodes in the tree are image-parts corresponding to instantiated rules r_v. An image I may have numerous feasible trees, and the likelihood of the image under any such tree T is given by:

$$\Pr(I|T,R) = \prod_{v}^{V} \prod_{x}^{Ch(r_v)} \rho(x,v) \tag{1}$$

where the value of $\rho(x, v)$ depends on whether the child x of r_v is instantiated in the tree T.

$$\rho(x,v) = \left\{ \begin{array}{ll} \Pr(x|r_v)\Pr(l_x|r_v) & x \in Ch(v) \\ 1 - \Pr(x|r_v) & \text{otherwise} \end{array} \right\} \tag{2}$$

Before we present our approach to unsupervised learning of our model R from a collection of images, we describe our approach to the inference problem.

4. Inference

The inference problem entails determining the tree T^* that best explains the observed set of features F in a particular image I, given our learnt model R. This can be formulated as

$$T^* = \arg\max \Pr(I|T,R) \tag{3}$$

As stated earlier, a tree contains image-parts (hidden) as intermediate nodes and features (observed) as leaves. The task is to determine which and at what location rules from our model should be instantiated in the image, such that the observed features are best explained. Considering a dense sampling of potential locations for every rule in the model would result in a very large number of potential image-parts to be considered, making this task computationally infeasible. Instead, we select a sparse set of likely locations for each rule. While this greatly increases the computational efficiency, the optimally of the tree cannot be guaranteed.

We first present our approach for determining the subset of optimal image-parts from a pool of potential image-parts such that the resulting tree best explains the image. This is followed by a section describing how the initial set of potential image-parts is found.

4.1. Inferring the tree

Having computed a set of potential image parts $\tilde{H}$, we need to determine the subset of parts $H \subset \tilde{H}$ that best explains the image in the form of a tree. An image is considered to be explained if all the observed features in the image are assigned to some image-part. All image-parts that are retained must be directly or indirected connected to the root node corresponding to r_0.

The set of all possible assignments of features to image-parts and image-parts to image-parts forms a weighted directed acyclic graph (DAG) G. Our goal is to find a tree $T \subset G$ such that Equation (1) is maximized. To achieve this goal we map our problem to that of a Steiner tree [21]. A minimum cost Steiner tree is the same as a Minimum Spanning Tree (MST) except some vertices in the graph do not need to be in the final tree. For our task, all image-parts not corresponding to the root node are considered optional. To map our problem to a the Steiner tree, we need to define a set of edge weights for every edge in G. Since Equation (1) is dependent on uninstantiated parts that may not exist in T, it cannot be directly applied for computing edge weights. Instead we perform the following manipulations on equation (1) to find our set of edge weights. First, we define two helper functions $\alpha(x, v) = \Pr(x|r_v)\Pr(l_x|r_v)$ and $\beta(x, v) = 1 - \Pr(x|r_v)$ corresponding to the two parts of Equation (2). If x_v corresponds to the rule or codeword at vertex v, we find:

$$\Pr(I|T,R) = \left(\prod_{v}^{V}\prod_{v'}^{Ch(v)} \alpha(x_{v'},v)\right) \times \left(\prod_{v}^{V}\prod_{x}^{Ch(r_v)\setminus Ch(v)} \beta(x,v)\right) \tag{4}$$

$$= \left(\prod_{v}^{V}\prod_{v'}^{Ch(v)} \frac{\alpha(x_{v'},v)}{\beta(x_{v'},v)}\right) \times \left(\prod_{v}^{V}\prod_{x}^{Ch(r_v)} \beta(x,v)\right) \tag{5}$$

Since the value of $\prod_{x}^{Ch(r_0)} \beta(x,v_0)$ for the root node is constant for all trees, we can rewrite the second part of Equation (4) as:

$$\prod_{v}^{V}\prod_{x}^{Ch(r_v)} \beta(x,v) \propto \prod_{v}^{V}\prod_{v'}^{Ch(v)}\prod_{x'}^{Ch(r_{v'})} \beta(x',v') \tag{6}$$

As a result:

$$\Pr(I|T,R) \propto \prod_{v\in V}\prod_{v'}^{Ch(v)} \left(\frac{\alpha(x_{v'},v)}{\beta(x_{v'},v)}\prod_{x'}^{Ch(v')} \beta(x',v')\right) \tag{7}$$

We then assign our edge weights $\omega(v',v)$ for all $v, v' \in V$ such that $v = Pa(v')$ as:

$$\omega(v',v) = -\log\left(\frac{\alpha(x_{v'},v)}{\beta(x_{v'},v)}\prod_{x'}^{Ch(v')} \beta(x',v')\right) \tag{8}$$

Using Equation (8) we can assign edge weights to every edge in G and solve for the minimum cost Steiner tree. That is, the tree with minimum edge weights that connects each feature to the root node, using any subset of image-parts. Since this has been shown to be a NP-hard problem, we use the approximation algorithm proposed by Charikar *et al.* [22]. For a graph G, the minimum cost Steiner tree is the optimal solution for Equation (3) except in special cases when multiple instantiations of a rule's child are found. In these cases, we simply choose the most likely instantiation of the child and add the rest to the root node.

4.2. Determining candidate locations

In the previous section we discussed how to find the optimal tree given a candidate set of image-part locations. In this section we describe how the candidate set is found. The candidate locations of rules are determined through a voting mechanism. A map is thus created over the entire image, indicating the likelihood of the rule occurring at that location. The peaks in this distribution are then computed using non-local-maxima-suppression, which form candidate part locations. These distributions for the rules are computed in order of their associated levels, where the lowest level parts (codewords) vote for the first level parts, which in turn vote for the second level parts, and so on. The level of a rule is recursively defined as one more than the maximum level of all its children. The level of codewords is arbitrarily defined to be 0.

The cumulative votes $\xi(v)$ of all children of potential vertex v are computed as:

$$\xi(v) = \sum_{x}^{Ch(r_v)} \alpha(x,v) \tag{9}$$

This additive form allows for our framework to be robust to missing children and occlusions. This provides an advantage over other methods such as pictorial structures [3] that are not robust to occlusions. By using a subset of image-part locations, the globally optimal tree for an image may not be found. However, it allows for the computational feasibility of the algorithm.

5. Learning

We use an EM-style approach for unsupervised learning of rules for image parts. A set of rules is first initialized. Then we iteratively infer the rules in our image data set using the Steiner tree formulation described above, update the rule parameters given their found instantiations and repeat. In addition, we add and remove rules during each iteration. Example rules are illustrated for the face and motorbikes data sets in Figures 3 and 4.

We initialize each rule by randomly selecting an image and location. Children are assigned to the rule based on the codewords that exist in a certain spatial neighborhood. In all our experiments, we randomly selected 10 codewords in a spatial neighborhood equal to a quarter of the image size. The mean relative location of the children is set according to their location in the image, the covariance matrix is set to a diagonal matrix with entries equal to one third the image size and the probability of occurrence is set to 0.25. This gives us our initial model R. Initially all rules belong to the first level. As the learning proceeds, higher level rules are added in a similar manner.

Given a set of rules, a new set of instantiated image-parts are inferred, and the rule parameters are updated. First, every non-root vertex v in the inferred tree T is assigned to an image-part. If a vertex was assigned to the root node by the

Steiner tree, then it is reassigned to the nearest image-part of higher ranking if one exists. Next, the vertices corresponding to the same rules or features are clustered using mean-shift. Each cluster is then assigned as a child to the parent rule with the appropriate occurrence probability, mean and covariance. Clusters with fewer than ten members are removed. It is worth noting that multiple instances of the same codewords or rules can be added to a parent rule. This allows a rule to have multiple children with similar appearances, such as the two wheels of a motorbike.

Rules may also be removed and added during each iteration. If a rule was not inferred in at least 10 images, it is removed from the set of rules. New rules are added in a similar manner as in initialization. However, image-parts are now inferred, so a hierarchy of parts may form. It is possible to also limit rules to only have image-parts as children. This explicitly encourages higher level rules to be formed.

In all our experiments, we used 30 iterations for each level, and computed a total of two levels. Rules were added only once every three iterations to allow the existing rules to stabilize before new rules were added. The number of added rules varied from 4 to 18 depending on the database size.

6. Experiments and Results

We present results of our unsupervised learning algorithm on a variety of datasets containing object categories, natural outdoor scene categories as well as complex street scenes with multiple objects. We present quantitative results on various tasks such as object categorization and localization, and qualitatively explore the behavior of rules at different levels in the hierarchy.

6.1. Faces vs Motorbikes: SIFT

For illustration and intuition-building purposes, we first present results on a dataset composed of 100 random images each from the Face and Motorbike categories of the Caltech101 data set [23]. We use the SIFT [39] descriptor on interest points as our low-level descriptor, along with a dictionary of 200 visual words. Our learning procedure learnt 15 first level rules, and 2 second-level rules.

An illustration of the rules learnt can be seen in Figure 2. We see that the second level rules correspond to the object category, while their children (first level rules) correspond to object parts (chin, cheek, wheel, etc.), as also seen in Figure 3. Some of these parts are shared across both categories, while some are specific to each category.

As seen in Figure 2, we see that at higher levels, the objects are better localized. A similar trend is seen for categorization as seen in Figure 5. To quantify this behavior, we use the occurrence of each part individually to categorize the image as well as localize the foreground. For the purpose of evaluation, we considered faces to be the positive class for categorization; and hand-labeled bounding boxes around faces for localization (and the rest of the image as the negative class). For localization, we find that the sensitivity of the different levels of parts shown in Figure 2 is 0.44, 0.56, 0.61, 0.69 and 0.94, while the specificity is 0.61, 0.76, 0.82, 0.95 and 0.99. The higher level spatial patterns provide more accurate categorization and localization. It should be noted that we only penalize the firing of a part on background, and not the assignment of a foreground codeword to background.

For the task of object categorization, we use the bag-of-words model followed by k-means clustering, which gives

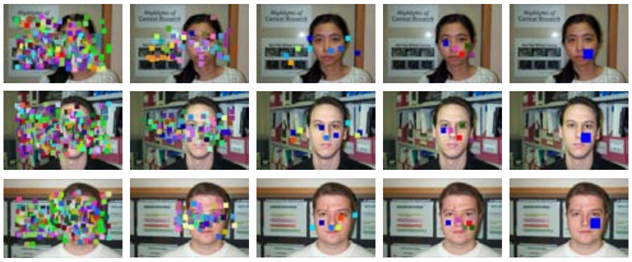

Figure 2: The first column illustrates all the visual words observed in the image. The second column depicts the subset of codewords that were assigned to a higher level part. The third column depicts the location of the first level parts, a subset of which (fourth column) support a second level part which are shown in the last column.

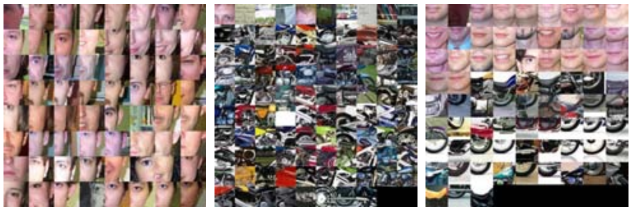

Figure 3: Patches extracted around instantiation of three first level rules for the faces and motorbikes data set. The first rule is specific to faces, the second one is specific to motorbikes, while the third one is shared across categories.

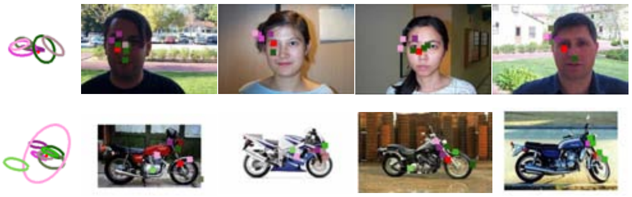

Figure 4: Example rules learnt by our algorithm from an unlabeled collection of face and motorbike images. The first column illustrates the structure of these first level rules and the relative spatial locations of its children. The last four columns show instantiations of the rules in example images.

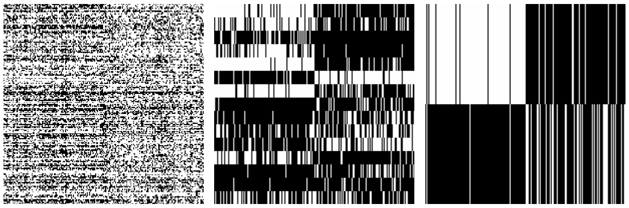

Figure 5: On the left is the occurrence matrix of the codewords (rows) in the face (left half of the matrix) and motorbike images (right half of the matrix). It is evident that codewords are not specific to either category. The middle plot is the occurrence matrix of the first level rules, where the distinction between the two categories improves, followed by the occurrence matrix of the second level rule.

us an accuracy of 93.5%. A bag-of-rules descriptor with k-means can classify each image correctly. Given the simplicity of the dataset, SIFT features alone can separate faces from motorbikes accurately. We experiment with less descriptive edge features (at 4 orientations and 6 scales, forming a dictionary of 24 codewords) and find that using our learnt rules the categorization accuracy increases from 55% using bag-of-words to 81.7%.

6.2. Six object categories

Unsupervised clustering of images into object categories is one potential application of the proposed model. To this end, we evaluate our approach on 100 random images from 6 object categories (faces, motorbikes, airplanes, car-rear, watches and ketches) from Caltech101 [23], similar to the recent work of Kim *et al.* [37]. Our accuracies are reported in Table 1, and are comparable. We compare the bag-of-words descriptor ("Words") based on codewords to one based on the inferred rules ("Rules"). While a bag-of-rules descriptor captures which rules were instantiated in an image, it does not capture which children of the rule were instantiated to support the rule. We use the parse tree inferred for the image as a descriptor ("Tree") to capture this information. The length of the descriptor is the same as the number of parent-child relationships in the learnt rules, where an element is set to 1 if the corresponding parent child relationship was instantiated in the image.

For unsupervised clustering of images into object categories, we use PLSA [12], k-means and normalized cuts [40]. For normalized cuts, we construct a graph ("NN-graph"), where each node corresponds to an image, and a node is connected to its five nearest neighbors computed using normalized dot-product of the image descriptors. On the other hand, for supervised classification of images, we use a linear SVM.

A total of 61 first level rules, and 12 second level rules were learnt from these 600 images. On average, the first level rules had 9 children, and the second level rules had 3. We also train our model using only 30 images per category, and obtain comparable accuracies.

Table 1: Categorization accuracy (%) using 100/30 images per category

	Kmeans	PLSA	NNgraph	SVM
Words	70.7/72.8	80.5/78.3	86.5/84.7	93.3/91.8
Rules	85.2/86.5	84.7/85.6	94.2/92.6	91.3/90.7
Both	73.9/74.3	82.6/84.7	90.1/88.8	95.8/93.2
Tree	88.1/89.5	85.1/88.2	95.0/93.5	91.3/89.8

6.3. Scene categories

We experiment with a dataset containing 150 images from the outdoor scene recognition dataset by Torralba *et al.* [24]. We segmented these images to obtain on average 10 segments per image using the segmentation algorithm of Felzenszwalb *et al.* [41]. Each segment was described with its average RBG color vector. These color descriptors from all the segments from all images were clustered to form a dictionary of 25 codewords. Using our learning algorithm on these images, we were able to find only first level rules, whose spatial extent was often the entire image. This is intuitive behavior for this dataset, where a deeper hierarchy is non-existent. A total of 17 rules were learnt. A visualization of a subset of rules learnt can be seen in Figure 6. We can see that images with consistent spatial layout of colors are grouped together. In the last two rows (first and last image respectively), we see that the color histogram of the images may be similar, however the spatial layout of the colors distinguish them from each other.

6.4. Street scenes

We select 66 images from the street scenes in the LabelMe dataset [25]. We use SIFT features with a dictionary of 200 codewords. 25 first level and 8 second level rules were learnt. Illustrations of these rules are shown in Figures 7 and 8. It can be seen that the features supporting the first level rules are consistently found on objects/regions of the image, and the second level rules correspond to objects (cars, trees, buildings), or combine contextually meaningful objects (cars and buildings).

To evaluate the specificity of the learnt rules to the data w.r.t. noise, we infer the rules learnt on the street scene images on 66 background images (from the Catech101 dataset). Figure 9 depicts the histogram of the number of rules instantiated in the background images, as compared to the "foreground" street scene images. We can see that the histograms are well separated, and simply by counting the number of rules instantiated in an image, it can be separated

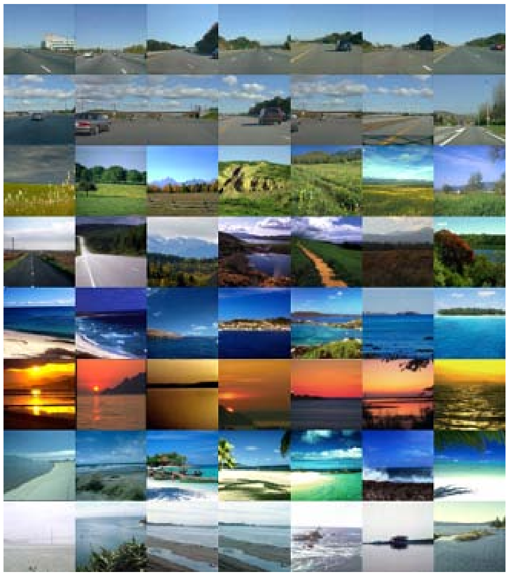

Figure 6: Each row corresponds to a rule learnt from an unstructured collection of outdoor scene category images. For each rule we show 7 random images that instantiated this rule. It can be seen that the images are consistent in the spatial distribution of their colors.

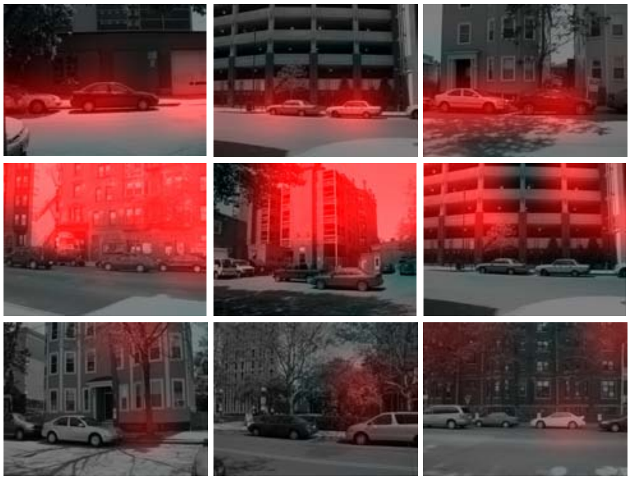

Figure 7: An illustration of three first level rules (rows) learnt from street scene images. We highlight the regions of the image with a high density of features that support each rule. In general, the first rule corresponds to buildings, the second one to cars and the third one to trees.

into the street scene vs. background.

7. Discussion and Future Work

This work describes a hierarchical representation of the image that inherently allows for the sharing of low-level image parts. Occlusions are also explicitly modeled. However, since we represent our rules using star-graphs we assume the children of every vertex in our tree are independent. This does not allow us to capture higher order relationships among parts.

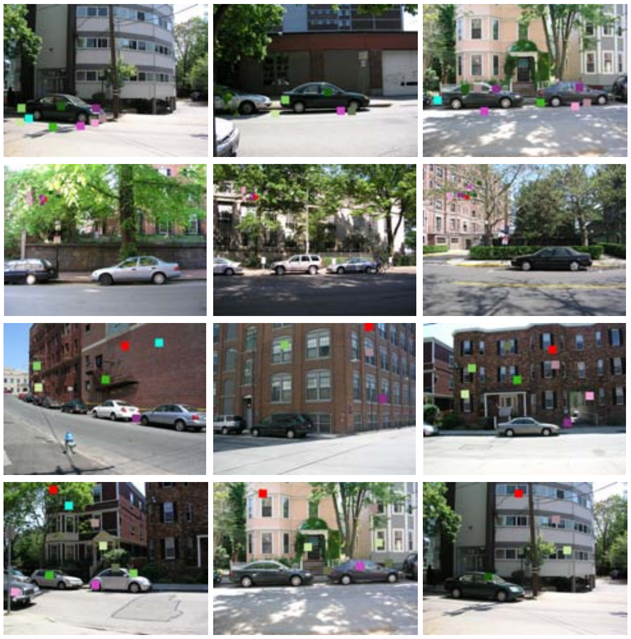

Figure 8: An illustration of four second level rules learnt from street scene images. The first level rules that support the second level rule are shown. The first rule (row) corresponds to cars (note the instantiation of the same rule twice for the two cars in the last column), the second rule corresponds to trees, the third to buildings and the fourth combines the cars and buildings in one rule.

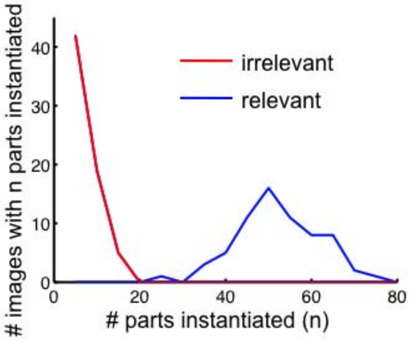

Figure 9: The number of parts learnt from street scenes (foreground) that were instantiated on background images. The rules learnt capture the spatial structures of the dataset, and not noise.

The efficiency of our approach is mainly limited by the choice of algorithm for solving the Steiner tree. Quasi-polynomial approximate algorithms have been developed to solve the general Steiner tree problem [22] that is known to be NP-Hard. Unfortunately, these algorithms are still too inefficient for large graph sizes. Given the specialized structure of our problem, it may be possible to create better approximate algorithms.

The accuracy of our approach is limited by the choice of low-level features. Features such as SIFT [39] already contain significant structural information. More primitive features such as edges may provide increased robustness to background clutter and shape ambiguity. These primitive features may also require more levels in the hierarchy to find coherent objects.

8. Conclusion

In this paper, we proposed an unsupervised method for learning hierarchical spatial structures in images. Our model consists of a set of rules modeled as star graphs, in which the children of each rule may be another rule or a low-level feature. The structure and parameters of the rules are learnt automatically. Given an image, a set of rules is inferred that best predicts the occurrence of the low-level features in the image. This subset of rules form a tree, and inference is accomplished by mapping the problem to that of finding the minimum cost Steiner tree in a directed acyclic graph, for which approximate algorithms exist.

We provide several results on various data sets including six Caltech 101 object categories, an outdoor scene data set, and a real-world street scene image collection from the LabelMe data set. Quantitative and qualitative results are provided. The unsupervised approach is shown to discover categories in images containing just one object, as well as multiple objects.

References

[1] D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical models. *CVPR*, 2005.

[2] R. Fergus, P. Perona and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. *CVPR*, 2003.

[3] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 2005.

[4] L. Karlinsky, M. Dinerstein, D. Levi and S. Ullman. Unsupervised Classification and Part Localization by Consistency Amplification. *ECCV*, 2008.

[5] M. Leordeanu, M. Hebert, R. Sukthankar. Beyond local appearance: Category recognition from pairwise interactions of simple features. *CVPR*, 2007.

[6] D. Hoiem, A. Efros and M. Hebert. Putting objects in perspective. *CVPR*, 2006.

[7] S. Kumar and M. Hebert. A hierarchical field framework for unified context-based classification. *ICCV*, 2005.

[8] D. Parikh, C.L. Zitnick and T. Chen. From Appearance to Context-Based Recognition: Dense Labeling in Small Images. *CVPR*, 2008.

[9] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora and S. Belongie. Objects in Context. *ICCV*, 2007.

[10] A. Torralba, K. Murphy and W. Freeman. Contextual models for object detection using boosted random fields. *NIPS*, 2005.

[11] D. Liu, G. Hua, P. Viola and T. Chen, Integrated Feature Selection and Higher-order Spatial Feature Extraction for Object Categorization. *CVPR*, 2008.

[12] J. Sivic, B. Russell, A. Efros , A. Zisserman, and W. Freeman. Discovering Objects and their Location in Images. *ICCV*, 2005.

[13] K.S. Fu. *Syntactic Pattern Recognition and Applications*, Prentice-Hall, 1982.

[14] Y. Jin and S. Geman. Context and hierarchy in a probabilistic image model. *CVPR*, 2006.

[15] K. Murphy, A. Torralba and W. Freeman. Using the forest to see the trees: a graphical model relating features, objects, and scenes. *NIPS*, 2003.

[16] D. Parikh and T. Chen. Hierarchical semantics of objects (hSOs). *ICCV*, 2007.

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

[18] E. Sudderth, A. Torralba, W. Freeman and A. Willsky. Learning hierarchical models of scenes, objects, and parts. *ICCV*, 2005.

[19] S. Ullman. visual Routine, *Cognition*, 18:97-157, 1984.

[20] S. Zhu and D. Mumford. Quest for a Stochastic Grammar of Images. *Foundations and Trends in Computer Graphics and Vision*, 2007.

[21] F. Hwang, D. Richards and P. Winter. *The Steiner Tree Problem.* 1992

[22] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha. Approximation Algorithms for Directed Steiner Problems. *Symposium on Discrete Algorithms*, 1998.

[23] L. Fei-Fei, R. Fergus and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. *CVPR*, Workshop on Generative-Model Based Vision, 2004.

[24] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 2001.

[25] B. Russell, A. Torralba, K. Murphy and W. Freeman. Labelme: a database and web-based tool for image annotation. *MIT AI Lab Memo*, 2005.

[26] A. Hanson and E. Riseman. Visions: a computer system for interpreting scenes. *Computer Vision Systems*, 1978.

[27] S. Todorovic and N. Ahuja. Unsupervised category modeling, recognition, and segmentation in images. *PAMI*, 2007.

[28] S. Todorovic and N. Ahuja. Learning subcategory relevances to category recognition.*CVPR*, 2008.

[29] L. Zhu, C. Lin, H. Huang, Y. Chen and A. Yuille. Unsupervised Structure Learning: Hierarchical Recursive Composition, Suspicious Coincidence and Competitive Exclusion. *ECCV* 2008.

[30] S. Fidler, G. Berginc and A. Leonardis. Hierarchical Statistical Learning of Generic Parts of Object Structure. *CVPR*, 2006.

[31] J. Sivic, B. Russell, A. Zisserman, W. Freeman and A. Efros. Unsupervised Discovery of Visual Object Class Hierarchies, *CVPR*, 2008.

[32] E. Sudderth, A. Torralba, W. Freeman and A. Willsky. *IJCV*, 2008.

[33] B. Epshtein and S. Ullman. Feature hierarchies for object classification. *ICCV*, 2005.

[34] S. Fidler and A. Leonardis. Towards Scalable Representations of Object Categories: Learning a Hierarchy of Parts. *CVPR*, 2007.

[35] L. Zhu, Y. Chen, Y. Lu, C. Lin and A. Yuille. Max Margin AND/OR Graph Learning for Parsing the Human Body. *CVPR* 2008.

[36] F. Han and S.C. Zhu. Bottom-up/Top-Down Image Parsing by Attribute Graph Grammar. *ICCV*, 2005.

[37] G. Kim, C. Faloutsos and M. Hebert. Unsupervised Modeling of Object Categories Using Link Analysis Techniques. *CVPR*, 2008.

[38] D. Liu, T. Chen. Unsupervised image categorization and object localization using topic models and correspondences between images. *ICCV*, 2007.

[39] D. Lowe. Distinctive image features from scaleinvariant keypoints. *IJCV*, 60(2):91110, 2004.

[40] J. Shi, J. Malik. Normalized cuts and image segmentation. *PAMI)*, 2000

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