HIERARCHICAL OBJECT GROUPS FOR SCENE CLASSIFICATION

Amir Sadovnik and Tsuhan Chen

Department of Electrical and Computer Engineering, Cornell University

ABSTRACT

The hierarchical structures that exist in natural scenes have been utilized for many tasks in computer vision. The basic idea is that instead of using strictly low level features it is possible to combine them into higher level hierarchical structures. These higher level structures provide a more specific feature and can thus lead to better results in classification or detection. Although most previous work has focused on hierarchical combinations of low level features, hierarchical structures exist on higher levels as well. In this work we attempt to automatically discover these higher level structures by finding meaningful object groups using the Minimum Description Length (MDL) principle. We then use these structures for scene classification and show that we can achieve a higher accuracy rate using them.

Index Terms— Image Classification, Scene Classification, Object Detection, Object Groups

1. INTRODUCTION

Our visual world can be viewed as built from hierarchical structures. For example, a person can first be thought of as a combination of a face and a body. We can then say a face is a combination of eyes, a nose and a mouth, and further say the eye is a combination of the iris, pupil and eyelids. In fact, this idea is emphasized by recent research suggesting that the human visual system is constructed in a hierarchical manner, and has structured itself in this way in order to adapt to the hierarchical structure of the visual world [1]. Relying on this fact we attempt to exploit this structure for scene classification.

Our work is inspired by two main approaches for classification. One involves building hierarchical features for object recognition [2, 3]. These approaches usually describe objects as constructed of parts, which in turn can be described as constructed of smaller parts, and finally low level features, thus creating a hierarchical structure. These works show that by using features on different levels of the hierarchy they can achieve a higher rate of object recognition.

The other inspiration for our work is the use of object detectors as higher order attributes for scene classification [4]. In this work the responses from object detectors are pooled to construct an object bank which is used as a feature vector for scene classification. Li et al. [4] show that by using these features they are able to perform better at scene classification versus using only low level features.

In this work we combine these two ideas and stipulate that the relationship between objects and scenes is similar to the relationship between low level features and objects. That is, just as low level features can be combined to construct object parts and then entire objects, so can objects be combined together to create object groups and ultimately entire scenes. Therefore, just as using these object parts helps in object detection, so can these object groups help in scene classification.

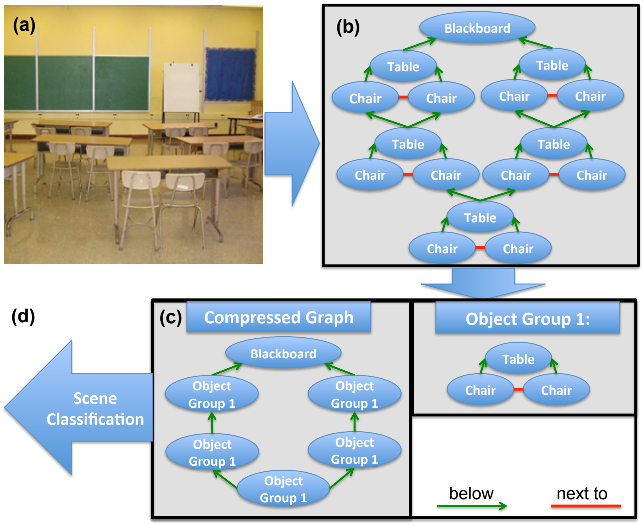


Fig. 1. Given a labeled image (a) we can construct a graph which represents the different objects and their spatial relationships in the image (b). We then use the MDL principle to discover groups of objects which are able to compress the graph by replacing them with a single node (c). These object groups represent higher order concepts in the image, and therefore we predict they will be useful for different tasks. In this paper we show their usefulness for the task of scene classification (d).

In order to discover these object groups we use a graph-based knowledge discovery system based on the MDL principle. This system looks for substructures in a graph by finding groups of nodes which, when replaced by a single node, can compress the data (see Fig. 1). Since the MDL principle states that the best hypothesis for given data is the one which compresses it the most, these node groups are expected to represent higher level concepts. More specifically we use the SUBDUE system which has been used previously to discover concepts in different relational databases such as chemical structures and activity databases [5]. To the best of our knowledge this is the first time such an algorithm is used with objects in natural scenes.

Looking at relationships between objects has been done many times before. Perhaps the most common use of object cooccurrence statistics is to provide context for object detection [6, 7, 8, 9]. However, these methods do not attempt to discover higher order structures as one entity or utilize them for scene classification. Sadeghi et al. [10] use visual phrases, which represent groups of objects which co-occur often in a specific spatial setup. However, there are two main differences between this work and ours. First, they use a supervised method to discover these groups. For example, the ”person riding a horse” phrase would have had to be manually labeled in many images, and not automatically discovered by the co-occurence of the human and the horse. This limits the amount of phrases that can be found, and is also subjective. In addition, they do not attempt to use these phrases for higher level classification such as scene classification. In our work we automatically discover these phrases, and thus are able to find many of them in our dataset, and use them to perform scene categorization.

The work done by Parikh et al. [11] is similar in spirit to ours in that it also attempts to discover a hierarchy of object groups. However, our work differs from this one in a few ways. First, this previous work is limited to images of a particular scene in different arrangements with objects that are consistent throughout all the images. In our work we find object groups from many different scenes with many object categories which can vary in appearance as well as arrangement. In addition, our group detection method is different. While [11] uses the correlation between feature positions which limits objects to belong to only one group, we use the MDL principle which allows the same object category to be part of different groups. Finally, although Parikh et al. do mention the possibility of using these groups for scene classification, they do not define how to do this or attempt to do so in the paper.

2. METHOD

Our algorithm can be split into three main stages. First, we construct a graph which represents the spatial relationships between objects in a given image database. We then use the MDL principle to discover substructures of the graph which represent higher level concepts. Finally, we train detectors for each of the object groups and use them to extract a feature vector for scene classification. We now describe the three stages in more detail.

2.1. Graph Construction

Our goal is to construct a graph in which nodes represent objects and edges represent the different possible spatial relationships between them. We use images in which objects are manually marked so we can localize the different object polygons. We then use simple rules to define the relationships between them.

We focus on three types of relationships between objects: “below”, “overlapping”, and “next-to”. The first two relationships are defined as directed edges in the graph, while the ”next-to” relationship is an undirected edge. To detect these, we simply calculate the relative position (Δx, Δy) and overlap (O) between the polygon markings of all pairs of objects that are less than a certain number of pixels away from each other. We then use the following criteria to define the relationship:

1. A “overlaps” C if $\frac{O_{AC}}{P_A} > 0.8$ where O_{AC} is the overlap area and P_A is the polygon area of A.
2. A is “below” C if $\Delta y_{CA} > 0$ and $0.375\pi < \arctan(\frac{\Delta y_{CA}}{|\Delta x_{CA}|}) < 0.625\pi$
3. A is “next-to” C for all other object pairs whose distance is less than the threshold number of pixels.

Given these rules we can construct a graph for each labeled image in our training set. An example of a graph constructed for a kitchen scene image is shown in Fig. 2

2.2. Group Discovery

In order to find the object groups we rely on the MDL principle which states that the best model to describe a set of data is the one which compresses it the best. Thus, the groups which are found

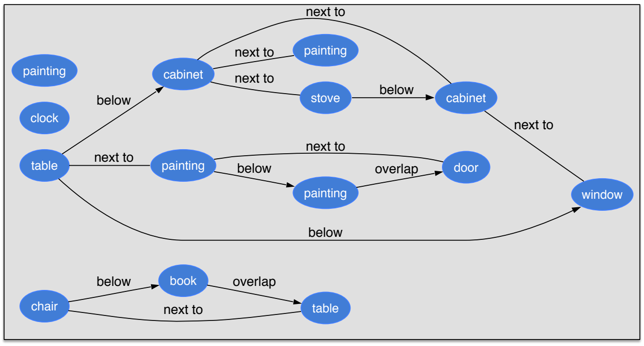


Fig. 2. An example of a graph constructed from objects in a kitchen.

by following this principle should represent regularities in the data which correspond to important object groups.

We use the implementation of a graph based knowledge discovery system called SUBDUE [5]. Besides the fact that the algorithm relies on the MDL principle it is suitable to our needs for 3 main reasons:

1. The algorithm works on relational graphs. These graphs, in which nodes represent entities and edges represent relationships between the entities, are compatible with our representation of scenes in which nodes represent objects and edges represent their spatial relationships (as described in Sec. 2.1).
2. The algorithm searches for the patterns in a hierarchical manner. That is, it finds smaller common substructures and allows them to be part of bigger substructures which will be discovered later on. This is a desired feature since we believe that natural scenes are built hierarchically.
3. This is a greedy algorithm, which at each iteration selects only the substructure which compresses the graph the most. Although this does not guarantee an optimal solution it does allow the algorithm to run fast on large graphs. Since we are searching through a large image database with tens of thousands of nodes, this is a crucial feature.

At each step of the algorithm, each object group S from the previous step is expanded in each possible direction available in graph G. It then keeps the top n substructures which have the highest score according to the following equation:

$$score(S, G) = \frac{size(G)}{size(S) + size(G|S)} \tag{1}$$

where the size function is simply defined as:

$$size(G) = \#vertices(G) + \#edges(G) \tag{2}$$

and $size(G|S)$ is defined as the size of G where each occurrence of S in G is replaced by a single node. The algorithm is initialized by having all node types be their own substructures, and then expanding them as previously defined.

The scoring function defined in Eq. 1 follows the crude MDL principle, in which we try to pick the model which minimizes the size of the model in addition to the size of data given the model [12]. Intuitively, basing the score on this principle makes sense since it takes into account both the size of the substructure in addition to the number of times it appears. We combine all the graphs created from our training dataset to one large graph of disconnected subgraphs and feed this into the SUBDUE algorithm to find the object groups. Examples of object groups discovered can be seen in Fig. 4

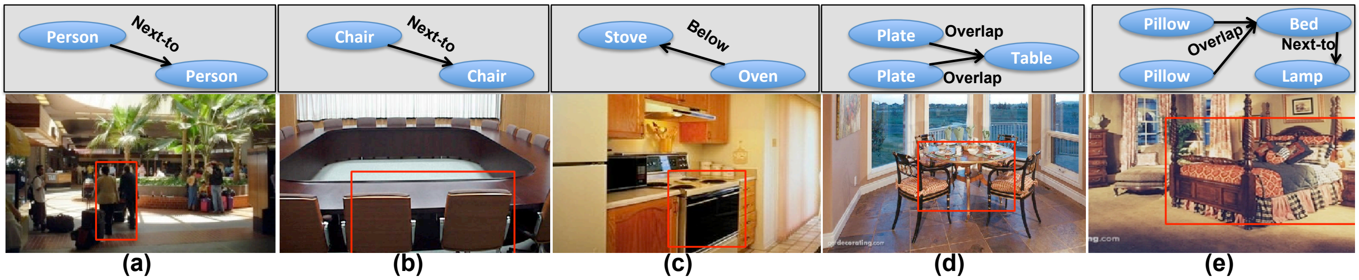


Fig. 3. Examples of different object groups and the bounding boxes returned by the detection algorithm [13] after training. Although this paper does not focus on object detection, we show that these detectors are useful in finding the existence of the groups in the test images.

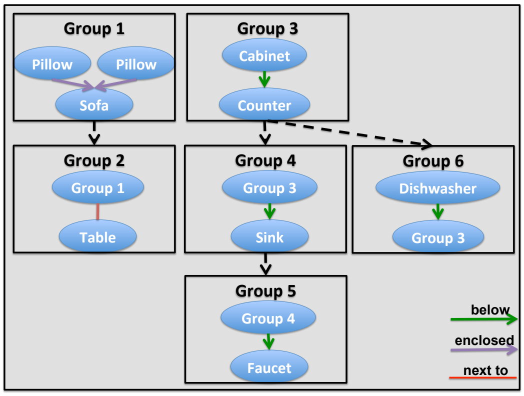


Fig. 4. Examples of object groups discovered by the SUBDUE algorithm. Each rectangle represents a group, where the nodes and edges in the rectangles represent the structure of the group. The dashed arrows between the groups simply imply that one object group (the tail) is part of another object group (the head), thus showing the hierarchical structure.

2.3. Scene Classifiction

After we extract all the substructures we train object detectors for each group. We use the object detection training algorithm as provided by [13]. However, instead of training these detectors on single objects, we use the bounding boxes of the entire object groups as positive examples for the detector. Thus the models learned during this training can be thought of as object group detectors. Examples of groups and their actual detection results can be seen in Fig. 3.

In order to classify the scene we use the same method as used by Li et al. [4]. This method creates an object bank representation for each image. First, each detector is run on different scale levels of the image and produces a response map, which is the detector's response at each pixel. Then, for each response map we do spatial pyramid max-pooling to populate the feature vector. In [4] they simply use regular object detectors trained from web images. Our object bank representation is extended to include the object groups we discover, and thus gives better results.

3. EXPERIMENTS

For our experiments we use the indoor scene database [14]. We expect these types of scenes to benefit the most from our approach. Although gist features [15] have been shown to work well for outdoor scenes they do not seem to work as well for indoor scenes. We predict that for the task of separating indoor scenes from one another using actual object detectors might be much more useful. For example, the spatial envelope of an image (as given by gist) might not be enough to differentiate between a bedroom and a living room since they are both closed rooms with furniture. However, given that the room has a bed the task can be solved easily.

This database also has a large number of images labeled with objects. In our our experiments we use 12 scene categories which have at least 40 images labeled with objects, in order to train the object/object-group detection model, and have at least 100 images more to perform the scene classification training/testing as described in [14](80 for training, 20 for testing). First we clean the objects names from the indoor dataset by unifying synonyms and removing objects which do not appear at least 10 times. This provides us with a list of 86 objects.

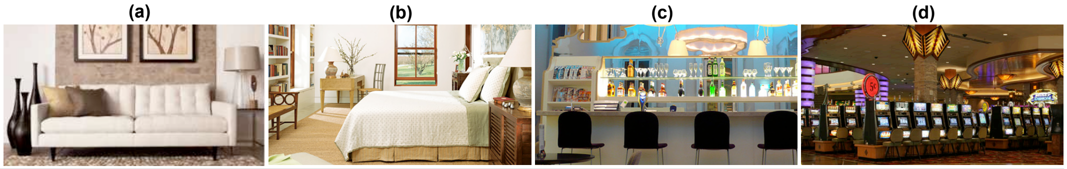
We then build graphs for all the labeled images using the method described in Sec. 2.1. We feed all these graphs (over 22,000 nodes) into the SUBDUE algorithm to discover common object groups, and reject the groups which appear less than 10 times. This leaves us with a list of 145 object groups. We then can train detectors for all the objects and object groups using the algorithm provided by [13]. Finally, these detectors are used for scene classification.

4. RESULTS AND DISCUSSION

Our classification results are shown in Fig. 5. We show the results for 4 different feature types. The gist feature vectors are extracted as described in [15] . Both object and object group feature vectors are of length 10836, since for each of the 86 object detectors we extract 126 features using spatial pooling over 6 levels. Although we discover 145 object groups, we truncate this list to the 86 which had the most appearances in the training data in order to make a fair comparison to the objects feature vector (we also attempted to use the full 145 object group detectors, but the improvement was insignificant). Finally, we also show the results for a feature vector which results from the concatenation of the objects and object groups.

As expected gist does not perform very well for indoor scenes. Gist has been shown to perform well for outdoor scenes in which the spatial envelope is significantly different between categories. This is obviously not the case for indoor scenes, in which the dominant spatial structures of the different categories are similar. Although solely using objects provides a significant increase over gist, using object groups we are able to achieve the highest results (56.0% accuracy).

Not only does using the object group feature vector achieve higher accuracy than the object feature vector, but it also does better than the concatenation of both as shown in Fig 5. This shows that by selecting to use only the object group features we manage to prevent over-fitting due to the noisier responses from the object-only detectors. For example, a pillow detector response might be very noisy, and fire for many other rectangular objects. However, once the pillow is put in a group with a bed or a couch (which is the



Category Name	Living-Room$_{(a)}$ vs. Bedroom$_{(b)}$		Bar$_{(c)}$ vs. Casino$_{(d)}$	
Feature Type	Objects	Object Groups	Objects	Object Groups
Accuracy	59.25%	67.75%	63.75%	75.50%
Most Important Features	lamp	2 pillows enclosed in bed	picture	door next to window
	bed	apple next to apple	chair	bottle next to glass
	coffee maker	2 pillows enclosed in bed next to lamp	coffee maker	bottle next to bottle next to bottle
	Candle	painting next to painting	switch	tray next to tray

Table 1. An example of the most important features selected by the SVM for two binary classification tasks (living room (a) vs. bedroom (b), bar (b) vs. casino (d)). For each task, the most important object and most important object groups are shown.

location of almost all pillows), the response becomes much more discriminative and hence more useful for scene classification.

In order to try and pinpoint the cause of the improvement we examine the classes which were confused often when using the object feature vector, and were dramatically improved when using the object groups. For example, using the object-only feature vector, living rooms were classified as bedrooms 29% of the time, while casinos were classified as bars 18% of the time. These numbers were reduced to 20% and 11% respectively when using object groups. For each of these cases we train a binary classifier, and examine the features which effect classification the most by selecting the ones with the highest coefficients. The results of the binary classifiers and the list of the most important features are shown in Table 1. As can be seen from the table, the important features from the object groups feature vector are more discriminating than the ones from the object-only feature vector.

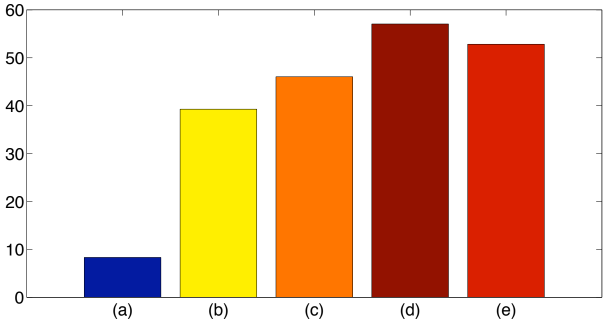


Fig. 5. Scene classification accuracy results for 12 scene categories using different features: (a) chance (b) gist [15] (c) objects [4] (d) object groups (our method) (e) objects + object groups.

5. CONCLUSION

Hierarchical structures exist on all levels of natural scenes. In this paper we propose a novel way to hierarchically group objects based on the MDL principle. These groups convey higher order concepts which can be viewed as the building blocks of a scene. We show that using detections of these object groups as feature vectors provide a significant increase (10%) in scene classification accuracy, thus proving that groups discovered in this manner can be highly beneficial for computer vision tasks. Although we have used these groups strictly for scene classification, further applications can be examined such as object detection and anomaly detection.

6. REFERENCES

[1] Shahbazi R., Field D., and Edelman S., “The role of hierarchy in learning to categorize images,” in *COGSCI*, 2011.

[2] Sanja Fidler, “Towards scalable representations of object categories: Learning a hierarchy of parts,” in *CVPR*, 2007.

[3] B. Epshtein and S. Ullman, “Semantic hierarchies for recognizing objects and parts,” in *CVPR*, 2007.

[4] L.J. Li, H. Su, Y. Lim, and L. Fei-Fei, “Objects as attributes for scene classification,” in *ECCV First International Workshop on Parts and Attributes*, 2010.

[5] D.J. Cook, L.B. Holder, and S. Djoko, “Knowledge discovery from structural data,” *Journal of Intelligent Information Systems*, 1995.

[6] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *PAMI*, 2010.

[7] C. Galleguillos, A. Rabinovich, and S. Belongie, “Object categorization using co-occurrence, location and appearance,” in *CVPR*, 2008.

[8] A. Torralba, Murphy K. P., and Freeman W. R, “Contextual models for object detection using boosted random fields.,” in *NIPS*, 2004.

[9] D. Parikh, C.L. Zitnick, and T. Chen, “Unsupervised learning of hierarchical spatial structures in images,” in *CVPR*, 2009.

[10] Mohammad Amin Sadeghi and Ali Farhadi, “Recognition using visual phrases,” *CVPR*, 2011.

[11] D. Parikh and Tsuhan Chen, “Unsupervised learning of hierarchical semantics of objects (hsos),” in *CVPR*, 2007.

[12] P.D. Grünwald, *The Minimum Description Length Principle*, The MIT Press, 2007.

[13] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, “Discriminatively trained deformable part models, release 3,” http://people.cs.uchicago.edu/ pff/latent-release3/.

[14] A. Quattoni and A. Torralba, “Recognizing indoor scenes,” in *CVPR*, 2009.

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