Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2008, Article ID 184618, 16 pages
doi:10.1155/2008/184618

Research Article

Unsupervised Modeling of Objects and Their Hierarchical Contextual Interactions

Devi Parikh and Tsuhan Chen

Department of Electrical and Computer Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Correspondence should be addressed to Devi Parikh, dparikh@andrew.cmu.edu

Received 11 June 2008; Accepted 2 September 2008

Recommended by Simon Lucey

A successful representation of objects in literature is as a collection of patches, or parts, with a certain appearance and position. The relative locations of the different parts of an object are constrained by the geometry of the object. Going beyond a single object, consider a collection of images of a particular scene category containing multiple (recurring) objects. The parts belonging to different objects are not constrained by such a geometry. However, the objects themselves, arguably due to their semantic relationships, demonstrate a pattern in their relative locations. Hence, analyzing the interactions among the parts across the collection of images can allow for extraction of the foreground objects, and analyzing the interactions among these objects can allow for a semantically meaningful grouping of these objects, which characterizes the entire scene. These groupings are typically hierarchical. We introduce hierarchical semantics of objects (hSO) that captures this hierarchical grouping. We propose an approach for the unsupervised learning of the hSO from a collection of images of a particular scene. We also demonstrate the use of the hSO in providing context for enhanced object localization in the presence of significant occlusions, and show its superior performance over a fully connected graphical model for the same task.

1. INTRODUCTION

Objects that tend to cooccur in scenes are often semantically related. Hence, they demonstrate a characteristic grouping behavior according to their relative positions in the scene. Some groupings are tighter than others, and thus a hierarchy of these groupings among these objects can be observed in a collection of images of similar scenes. It is this hierarchy that we refer to as the hierarchical semantics of objects (hSO). This can be better understood with an example.

Consider an office scene. Most offices, as seen in Figure 1, are likely to have, for instance, a chair, a phone, a monitor, and a keyboard. If we analyze a collection of images taken from such office settings, we would observe that across images, the monitor and keyboard are more or less in the same position with respect to each other, and hence can be considered to be part of the same super object at a lower level in the hSO structure, say a computer. Similarly, the computer may usually be somewhere in the vicinity of the phone, and so the computer and the phone belong to the same super object at a higher level, say the desk area. But the chair and the desk area may be placed relatively arbitrarily in the scene with respect to each other, more so than any of the other objects, and hence belong to a common super object only at the highest level in the hierarchy, that is, the scene itself. A possible hSO that would describe such an office scene is shown in Figure 1. Along with the structure, the hSO may also store other information such as the relative position of the objects and their cooccurrence counts as parameters.

The hSO is motivated from an interesting thought exercise: at what scale is an object defined? Are the individual keys on a keyboard objects, or the entire keyboard, or is the entire computer an object? The definition of an object is blurry, and the hSO exploits this to allow incorporation of semantic information of the scene layout. The leaves of the hSO are a collection of parts and represent the objects, while the various levels in the hSO represent the super objects at different levels of abstractness, with the entire scene at the highest level. Hence, hSOs span the spectrum between specific objects, modeled as a collection of parts, at the lower level and scene categories at the higher level. This provides a

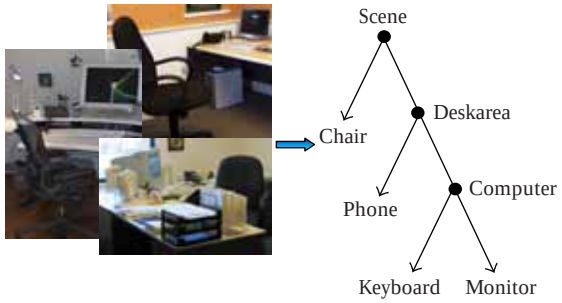


FIGURE 1: Images for "office" scene from Google image search. There are four commonly occurring objects: chair, phone, monitor, and keyboard. The monitor and keyboard occur at similar relative locations across images and hence belong to a common superobject, computer, at a lower level in the hierarchy. The phone is seen within the vicinity of the monitor and keyboard. However, the chair is arbitrarily placed, and hence belongs to a common super object with other objects only at the highest level in the hierarchy, the entire scene. This pattern in relative locations, often stemming from semantic relationships among the objects, provides contextual information about the scene "office" and is captured by an hSO: hierarchical semantics of objects (hSOs). A possible corresponding hSO is shown on the right.

rich amount of information at various semantic levels that can be potentially exploited for a variety of applications, ranging from establishing correspondences between parts for object matching and providing context for robust object detection, all the way to scene category classification.

Scenes may contain several objects of interest, and hand labeling these objects would be quite tedious. To avoid this, as well as the bias introduced by the subjectiveness of a human in identifying the objects of interest in a scene, unsupervised learning of hSO is preferred so that it truly captures the characteristics of the data.

In this paper, we introduce hierarchical semantics of objects (hSOs). We propose an approach for unsupervised learning of hSO from a collection of images. This algorithm is able to identify the foreground parts in the images, cluster them into objects, and further cluster the objects into a hierarchical structure that captures semantic relationships among these objects—all in an unsupervised (or semisupervised, considering that the images are all from a particular scene) manner from a collection of unlabeled images. We demonstrate the superiority of our approach for extracting multiple foreground objects as compared to some benchmarks. Furthermore, we also demonstrate the use of the learnt hSO in providing object models for object localization, as well as context to significantly aid localization in the presence of occlusion. We show that an hSO is more effective for this task than a fully connected network.

The rest of the paper is organized as follows. Section 2 describes related work in literature. Section 3 describes some applications that motivate the need for hSO and discusses prior works for these applications as well. Section 4 describes our approach for the unsupervised learning of hSO from a collection of images. Section 5 presents our experimental results in identifying the foreground objects and learning the hSO. Section 6 presents our approach for utilizing the information in the learnt hSO as context for object localization, followed by experimental results for the same. Section 7 concludes the paper.

2. RELATED WORK

Different aspects of this work have appeared in [1, 2]. We modify the approach presented in [1] by adopting techniques presented in [2]. Moreover, we propose a formal approach for utilizing the information in the learnt hSO as a context for object localization. We present thorough experimental results for this task including quantitative analysis and compare the accuracies of our proposed hierarchy (tree-structure) among objects to a flat fully connected model/structure over the objects.

2.1. Foreground identification

The first step in learning the hSO is to first extract the foreground objects from the collection of images of a scene. In our approach, we focus on rigid objects. We exploit two intuitive notions to extract the objects. First, the parts of the images that occur frequently across images are likely to belong to the foreground. And second, only those parts of the foreground that are found at geometrically consistent relative locations are likely to belong to the same rigid object.

Several approaches in literature address the problem of foreground identification. First of all, we differentiate our approach for this task from image segmentation approaches. These approaches are based on low-level cues and aim to separate a given image into several regions with pixel level accuracies. Our goal is a higher-level task, where using cues from multiple images, we wish to separate the local parts of the images that belong to the objects of interest from those that lie on the background. To reiterate, several image segmentation approaches aim at finding regions that are consistent within a single image in color, texture, and so forth. We are however interested in finding objects in the scene that are consistent across multiple images in occurrence and geometry.

Several approaches for discovering the *topic* of interest have been proposed such as discovering main characters [3] or objects and scenes [4] in movies or celebrities in collections of news clippings [5]. Recently, statistical text analysis tools such as probabilistic latent semantic analysis (pLSA) [6] and latent semantic analysis (LSA) [7] have been applied to images for discovering object and scene categories [8–10]. These use unordered *bag-of-words* [11] representation of documents to automatically (unsupervised) discover topics in a large corpus of documents/images. However, these approaches, which we loosely refer to as *popularity*-based approaches, do not incorporate any spatial information. Hence, while they can identify the foreground from the background, they cannot further separate the foreground into multiple objects. Hence, these methods have been applied to images that contain only one foreground object. We further illustrate this point in our results. These popularity-based approaches can separate the multiple objects of interest

only if the provided images contain different number of these objects. For the office setting, in order to discover the monitor and keyboard separately, pLSA, for instance, would require several images with just the monitor, and just the keyboard (and also a specified number of topics of interest). This is not a natural setting for images of office scenes. Leordeanu and Collins [12] propose an approach for the unsupervised learning of the object model from its low resolution video. However, this approach is also based on co-occurrence and hence cannot separate out multiple objects in the foreground.

Several approaches have been proposed to incorporate spatial information in the popularity-based approaches [13–16], however, only with the purpose of robustly identifying the single foreground object in the image, and not for separation of the foreground into multiple objects. Russell et al. [17], through their approach of breaking an image down into multiple segments and treating each segment individually, can deal with multiple objects as a byproduct. However, they rely on consistent segmentations of the foreground objects, and attempt to obtain those through multiple segmentations.

On the object detection/recognition front, approaches such as applying object localization classifiers through a sliding window approach could be considered, with a stretch of argument, to provide rough foreground/background separation. However, these are supervised methods. Part-based approaches, like ours, however towards this goal of object localization, have been proposed such as [18, 19] which use spatial statistics of parts to obtain objects masks. These are supervised approaches as well, and for single objects. Unsupervised part-based approaches for learning the object models for recognition have also been proposed, such as [20, 21]. These also deal with single objects.

2.2. Modeling dependencies among parts

Several approaches in text data mining represent the words in a lower-dimensional space where words with supposedly similar semantic meanings collapse into the same cluster. This representation is based simply on their occurrence counts in documents. pLSA [6] is one such approach that has also been applied to images [8, 10, 22] for unsupervised clustering of images based on their *topic* and identifying the part of the images that are foreground. Our goal however is a step beyond this towards a higher-level understanding of the scene. Apart from simply identifying the *existence* of potential semantic relationships between the parts, we attempt to characterize these semantic relationships, and accordingly cluster the parts into (super) objects at various levels in the hSO. Several works [23, 24] model dependencies among parts of a single object for improved object recognition/detection. Our goal however is to model correlations among multiple objects and their parts. We define dependencies based on relative location as opposed to co-occurrence.

It is important to note that, our approach being entirely unsupervised, the presence of multiple objects as well as background clutter makes the task of clustering the foreground parts into hierarchial clusters, while still maintaining the integrity of objects yet capturing the interrelationships among them, challenging. The information coded in the learnt hSO is hence quite rich. It entails more than a mere extension of the above works to multiple objects.

2.3. Hierarchies

Using hierarchies or dependencies among parts of objects for object recognition has been promoted for decades [23–31]. However, we differentiate our work from these, as our goal is not object recognition, but is to characterize the scene by modeling the interactions between multiple objects in a scene. More so, although these works deal with hierarchies per se, they capture philosophically very different phenomena through the hierarchy. For instance, Marr and Nishihara [25] and Levinshtein et al. [28] capture the shape of articulated objects such as the human body through a hierarchy, whereas Fidler et al. [31] capture varying levels of complexity of features. Bienenstock et al. [27] and Siskind et al. [32] learn a hierarchical structure among different parts/regions of an image based on rules on absolute locations of the regions in the images, similar to those that govern the grammar or syntax of a language. These various notions of hierarchy are strikingly different from the interobject, potentially semantic, relationships that we wish to capture through a hierarchical structure.

3. APPLICATIONS OF hSO

Before we describe the details of the learning algorithm, we first motivate hSOs through a couple of interesting potential areas for their application.

3.1. Context

Learning the hSO of scene categories could provide contextual information for tasks such as object recognition, detection, or localization. The accuracy of individual detectors can be enhanced as the hSO provides a prior over the likely position of an object, given the position of another object in the scene.

Consider the example shown in Figure 1. Suppose we have independent detectors for monitors and keyboards. Consider a particular test image in which a monitor is detected. However, there is little evidence indicating the presence of a keyboard due to occlusion, severe pose change, and so forth. The learnt hSO (with parameters) for office settings would provide the contextual information indicating the presence of a keyboard and also an estimate of its likely position in the image. If the observed bit of evidence in that region of the image supports this hypothesis, a keyboard may be detected. However, if the observed evidence is to the contrary, not only the keyboard is not detected, but also the confidence in the detection of the monitor is reduced as well. The hSO thus allows for propagation of such information among the independent detectors.

Several works use context for better image understanding. One class of approaches involves analyzing individual

images for characteristics of the surroundings of the object such as geometric consistency of object hypotheses [33], viewpoint and mean scene depth estimation [34, 35], and surface orientations [36]. These provide useful information to enhance object detection/recognition. However, our goal is not to extract information about the surroundings of the object of interest from a single image. Instead, we aim to learn a characteristic representation of the scene category and a more higher-level understanding from a collection of images by capturing the semantic interplay among the objects in the scene as demonstrated across the images.

The other class of approaches models dependencies among different parts of an image [37–43] from a collection of images. However, these approaches require hand-annotated or labeled images. Also, the authors of [37–39, 41] are interested in pixel labels (image segmentation) and hence do not deal with the notion of *objects*. Torralba et al. [44] use the global statistics of the image to predict the type of scene which provides context for the location of the object, however their approach is also supervised. Torralba et al. [45] learn interactions among the objects in a scene for context, however their approach is supervised and the different objects in the images need to be annotated. Marszałek and Schmid [46] also learn relationships among multiple classes of objects, however indirectly through a lexical model learnt on the labels given to images, and hence is a supervised approach. Our approach is entirely unsupervised—the relevant parts of the images, and their relationships are automatically *discovered* from a corpus of unlabeled images.

3.2. Compact scene category representation

hSOs provide a compact representation that characterizes the scene category of the images from which it has been learnt. Hence, hSOs can be used for scene category classification. Singhal et al. [47] learn a set of relationships between different regions in a large collection of images with a goal to characterize the scene category. However, these images are hand segmented, and a set of possible relationships between the different regions are predefined (above, below, etc.). Other works [48, 49] also categorize scenes but require extensive human labeling. Fei-Fei and Perona [8] group the low-level features into *themes* and *themes* into scene categories. However, the *themes* need not corresponding to semantically meaningful entities. Also, they do not include any location information, and hence cannot capture the interactions between different parts of the image. They are able to learn a hierarchy that relates the different scenes according to their similarity, however, our goal is to learn a hierarchy for a particular scene that characterizes the interactions among the entities in the scene, arguably according to the underlying semantics.

3.3. Anomaly detection

As stated earlier, the hSO characterizes a particular scene. It goes beyond an occurrence-based description, and explicitly models the interactions among the different objects through their relative locations. Hence, it is capable of distinguishing between scenes that contain the same objects, however in different configurations. This can be useful for anomaly detection. For instance, consider the office scene in Figure 1. In an office input image, if we find the objects at locations in very unlikely configurations given the learnt hSO, we can detect a possible intrusion in the office or some such anomaly.

These examples of possible applications for the hSO demonstrate its use for object level tasks such as object localization, scene level tasks such as scene categorization and one that is somewhere in between the two: anomaly detection. Later in this paper we demonstrate the use of hSO for the task of robust object localization in the presence of occlusions.

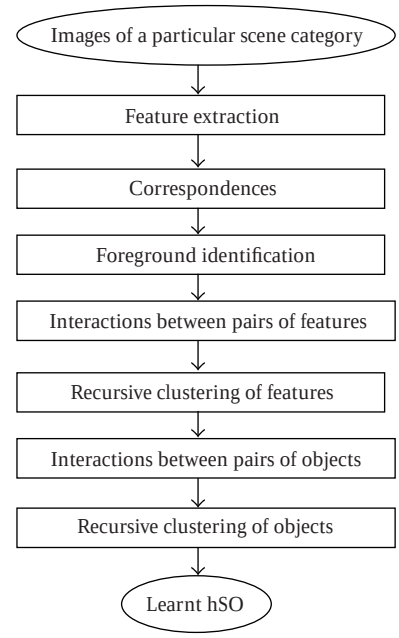


Figure 2: Flow of the proposed algorithm for the unsupervised learning of hSOs.

4. UNSUPERVISED LEARNING OF hSO

Our approach for the unsupervised learning of hSOs is summarized in Figure 2. The input is a collection of images taken in a particular scene, and the desired output is the hSO. The general approach is to first separate the features in the input images into foreground and background features, followed by clustering of the foreground features into the multiple foreground objects, and finally extracting the hSO characterizing the interactions among these objects. Each of the processing stages is explained in detail in Section 4.1.

4.1. Feature extraction

Given the collection of images taken from a particular scene, local features describing interest points/parts are extracted in all the images. These features may be appearance-

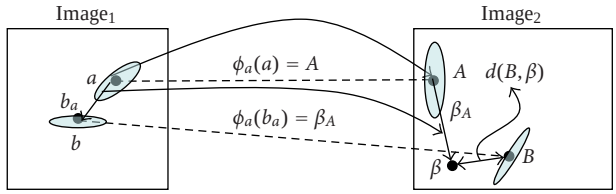


FIGURE 3: An illustration of the geometric consistency metric used to retain *good* correspondences.

based features such as SIFT [50], shape-based features such as shape context [51], geometric blur [52], or any such discriminative local descriptors as may be suitable for the objects under consideration. In our current implementation, we use the derivative of Gaussian interest point detector, and SIFT features as our local descriptors.

4.2. Correspondences

Having extracted features from all images, correspondences between these local parts are identified across images. For a given pair of images, potential correspondences are identified by finding k nearest neighbors of each feature point from one image in the other image. We use Euclidean distance between the SIFT descriptors to determine the nearest neighbors. The geometric consistency between every pair of correspondences is computed to build a geometric consistency adjacency matrix.

Suppose that we wish to compute the geometric consistency between a pair of correspondences shown in Figure 3 involving interest regions a and b in image$_1$ and A and B in image$_2$. All interest regions have a scale and orientation associated with them. Let ϕ_a be the similarity transform that transforms a to A. β_A is the result of the transformation of b_a (the relative location of b with respect to a in image$_1$) under ϕ_a. β is thus the estimated location of B in the image$_2$ based on ϕ_a. If a and A as well as b and B are geometrically consistent, the distance between β and B, $d(B, \beta)$, would be small. A score that decreases exponentially with increasing $d(B, \beta)$ is used to quantify the geometric consistency of the pair of correspondences. To make the score symmetric, a is similarly mapped to α under the transform ϕ_b that maps b to B, and the score is based on $\max(d(B, \beta), d(A, \alpha))$. This metric provides us with invariance only to scale and rotation, the assumption being that the distortion due to affine transformation in realistic scenarios is minimal among local features that are closely located on the same object.

Having computed the geometric consistency score between all possible pairs of correspondences, a spectral technique is applied to the geometric consistency adjacency matrix to retain only the geometrically consistent correspondences [53]. This helps eliminating most of the background clutter. This also enables us to deal with incorrect low-level correspondences among the SIFT features that cannot be reliably matched, for instance, at various corners and edges found in an office setting. To deal with multiple objects in the scene, an iterative form of [53] is used. However, it should be noted that due to noise, affine and perspective transformations of objects, and so forth, correspondences of all parts even on a single object do not always form one strong cluster and hence are not entirely obtained in a single iteration, instead they are obtained over several iterations.

4.3. Foreground identification

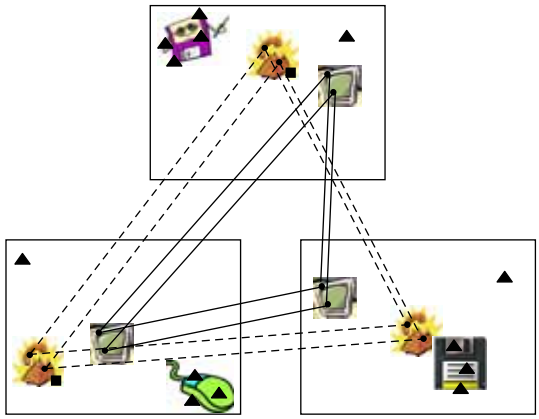
Only the feature points that find geometrically consistent correspondences in most other images are retained. This is in accordance with our perception that the objects of interest occur frequently across the image collection. Also, this post-processing step helps to eliminate the remaining background features that may have found geometrically consistent correspondences in another image by chance. Using multiple images gives us the power to be able to eliminate these random errors which would not be consistent across images. However, we do not require features to be present in all images in order to be retained. This allows us to handle occlusions, severe view point changes, and so forth. Since these affect different parts of the objects across images, it is unlikely that a significant portion of the object will not be matched in many images, and hence be eliminated by this step. Also, this enables us to deal with different number of objects in the scene across images, the assumption being that the objects that are present in most images are the objects of interest (foreground), while those that are present in a few images are part of the background clutter. This proportion can be varied to suit the scenario at hand.

We now have a reliable set of *foreground* feature points and a set of correspondences among all images. An illustration can be seen in Figure 4, where only a subset of the detected features and their correspondences is retained. It should be noted that by the approach being unsupervised, there is no notion of an object yet. We only have a cloud of features in each image which have all been identified as foreground and correspondences among them. The goal now is to separate these features into different groups, where each group corresponds to a foreground object in the scene, and further learn the hierarchy among these objects that will be represented as an hSO that will characterize the entire collection of images and hence the scene.

4.4. Interaction between pairs of features

In order to separate the cloud of retained feature points into clusters, a graph is built over the feature points, where the weights on the edge between the nodes represent the interaction between the pair of features across the images. The metric used to capture the interaction between the pairs of features is the same geometric consistency as computed in Section 4.2, averaged across all pairs of images that contain these features. While the geometric consistency could contain errors for a particular pair of images due to errors in correspondences, and so forth, averaging across all pairs suppresses the contribution of these erroneous matchings and amplifies the true interaction among the pairs of features.

If the geometric consistency between two feature points is high, they are likely to belong to the same rigid object. On the



▲ Features discarded as no geometrically consistent correspondences in any image (background)
■ Features discarded as geometrically consistent correspondences not found across enough images (occlusions, etc.)
• Features retained

Figure 4: An illustration of the correspondences and features retained. For clarity, the images contain only two of the four foreground objects we have been considering in the office scene example from Figure 1, and some background.

other hand, features that belong to different objects would be geometrically inconsistent because the different objects are likely to be found in different configurations across images. An illustration of the geometric consistency and adjacency matrix can be seen in Figure 4 and 5 respectively. Again, there is no concept of an object yet. The features in Figure 4 are arranged in an order that corresponds to the objects, and each object is shown to have only two features, only for illustration purposes.

4.5. Recursive clustering of features

Having built the graph capturing the interaction between all pairs of features across images, recursive clustering is performed on this graph. At each step, the graph is clustered into two clusters. The properties of each cluster are analyzed, and one or both of the clusters are further separated into two clusters, and so on. If the variance in the adjacency matrix corresponding to a certain cluster (subgraph) is very low but with a high mean, it is assumed to contain parts from a single object, and is hence not divided further. The approach is fairly insensitive to the thresholds used on the mean and variance of the (sub) adjacency matrix. It can be verified, for the example shown in Figure 4, that the foreground features would be clustered into four clusters, each cluster corresponding to a foreground object. Since the statistics of each of the clusters formed are analyzed to determine if it should be further clustered or not, the number of foreground objects needs not to be known a priori. This is an advantage as compared to pLSA or parametric methods such as fitting a mixture of Gaussians to the foreground features spatial distribution. Our approach is nonparametric.

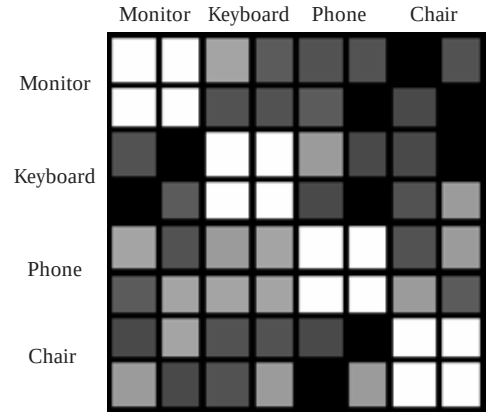


Figure 5: An illustration of the geometric consistency adjacency matrix of the graph that would be built on all retained foreground features for the office scene example as in Figure 1.

We use normalized cuts [54] to perform the clustering. The code provided at [55] was used.

4.6. Interaction between pairs of objects

Having extracted the foreground objects, the next step is to cluster these objects in a (semantically) meaningful way and extract the underlying hierarchy. In order to do so, a fully connected graph is built over the objects, where the weights on the edges between the nodes represent the interaction between the pairs of objects across the images. The metric used to capture the interaction between the pairs of objects is the predictability of the location of one object if the location of the other object was known. This is computed as the negative entropy of the distribution of the location of one object conditioned on the location of the other object, or the relative location of one object with respect to the other. The higher the entropy is, the less predictable the relative locations are. Let O be the number of foreground objects in our image collection. Suppose that $\mathbf{M}$ is the $O \times O$ interaction adjacency matrix we wish to create, then $\mathbf{M}(i, j)$ holds the interaction between the ith and jth objects as

$$\mathbf{M}(i, j) = -E[P(l_i - l_j)], \tag{1}$$

where $E[P(x)]$ is the entropy in a distribution $P(x)$, and $P(l_i - l_j)$ is the distribution of the relative location of the ith object with respect to the jth object. In order to compute $P(l_i - l_j)$, we divide the image into a $G \times G$ grid. G was typically set to 10. This can be varied based on the amounts of relative movements the objects demonstrate across images. Across all input images, the relative locations of the ith object with respect to the jth object are recorded as indexed by one of bins in the grid. We use MLE counts (an histogram like operation) on these relative locations to estimate $P(l_i - l_j)$. If appropriate, the relative locations of objects can be modeled using a Gaussian distribution in which case the covariance matrix would be a direct indicator of the entropy of the distribution. The proposed nonparametric approach is more general. An illustration of the $\mathbf{M}$ matrix is shown in Figure 6.

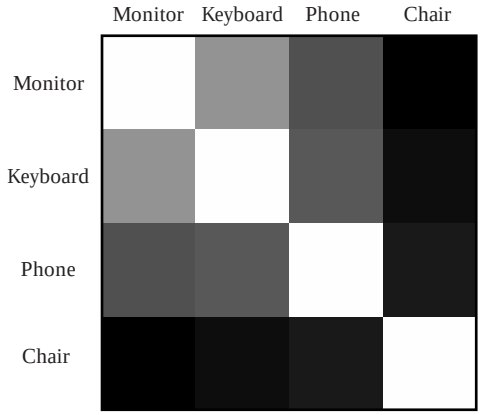


FIGURE 6: An illustration of the entropy-based adjacency matrix of the graph that would be built on the foreground objects in the office scene example as in Figure 1.

4.7. Recursive clustering of objects

Having computed the interaction among the pairs of objects, we use recursive clustering on the graph represented by **M** using normalized cuts. We further cluster every subgraph containing more than one object in it. The objects, whose relative locations are most predictable, stay in a common cluster till the end, whereas those objects whose locations are not well predicted by most other objects in the scene are separated out early on. The iteration of clustering at which an object is separated gives us the location of that object in the final hSO. The clustering pattern thus directly maps to the hSO structure. It can be verified for the example shown in Figure 6 that the first object to be separated is the chair, followed by the phone, and finally the monitor and keyboard, which reflects the hSO shown in Figure 1. With this approach, each node in the hierarchy that is not a leaf has exactly two children. Learning a more general structure of the hierarchy is part of future work.

In addition to learning the structure of the hSO, we also learn the parameters of the hSO. The structure of the hSO indicates that the *siblings*, that is, the objects/super objects (we refer to them as entities form here on) sharing the same parent node in the hSO structure, are the most informative for each other to predict their location. Hence, during learning, we learn the parameters of the relative location of an entity with respect to its sibling in the hSO only, as compared to learning the interaction among all objects (a flat fully connected network structure instead of hierarchy) where all possible combinations of objects would need to be considered. This would entail learning a larger number of parameters, which for a large number of objects could be prohibitive. Moreover, with limited training images, the relative locations of unrelated objects cannot be learnt reliably. This is clearly demonstrated in our experiments in Section 6.

The location of an object is considered to be the centroid of the locations of the features that lie on the object. The relative locations are captured nonparametrically as described previously in Section 4.6 (parametric estimations could be easily incorporated in our approach). The relative locations of entities in the hSO that are connected by edges are stored (we store the joint distribution of the location of the two entities and not just the conditional distribution) as MLE counts. The location of a super object is considered to be the centroid of the locations of the objects composing the super object. Thus, by storing the relative location of a child with respect to the parent node in the hierarchy, the relative locations of the siblings are indirectly captured. In addition to the relative location statistics, we could also store the co-occurrence statistics.

5. EXPERIMENTS

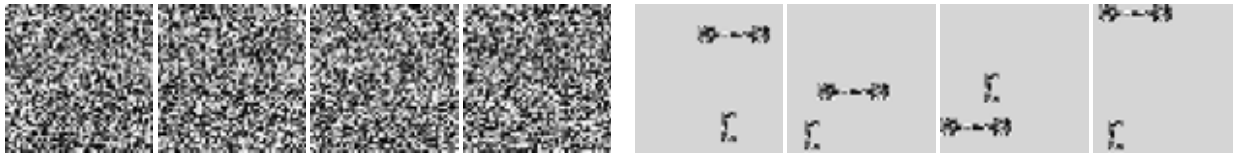
We first present experiments with synthetic images to demonstrate the capabilities of our approach for the subgoal of extracting the multiple foreground objects. The next set of experiments demonstrates the effectiveness of our entire approach for the unsupervised learning of hSO.

5.1. Extracting objects

Our approach for extracting the foreground objects of interest uses two aspects: popularity and geometric consistency. These can be loosely thought of as first-order as well as second-order statistics. In the first set of experiments, we use synthetic images to demonstrate the inadequacy of either of these alone.

To illustrate our point, we consider 50×50 synthetic images as shown in Figure 7(a). The images that contain 2500 distinct intensity values, of which 128, randomly selected from the 2500, always lie on the foreground objects and the rest is background. We consider each pixel in the image to be an interest point, and the descriptor of each pixel is the intensity value of the pixel. To make visualization clearer, we display only the foreground pixels of these images in Figure 7(b). It is evident from these that there are two foreground objects of interest. We assume that the objects undergo pure translation only.

We now demonstrate the use of pLSA, as an example of an unsupervised popularity-based foreground identification algorithm, on 50 such images. Since pLSA requires negative images without the foreground objects, we also provide 50 random negative images to pLSA, which our approach does not need. If we specify pLSA to discover 2 topics, the result obtained is shown in Figure 8. It can be seen that it can identify the foreground from the background, but is unable to further separate the foreground into multiple objects. One may argue that we could further process these results and fit a mixture of Gaussians (for instance) to further separate the foreground into multiple objects. However, this would require us to know the number of foreground objects a priori and also the distribution of features on the objects that need not to be Gaussian as in these images. If we specify pLSA to discover 3 topics instead, with the hope that it might separate the foreground into 2 objects, we find that it arbitrarily splits the background into 2 topics, while still maintaining a single foreground topic, as seen in Figure 8. This is because pLSA simply incorporates occurrence (popularity) and no



(a) (b)

Figure 7: (a) A subset of the synthetic images used as input to our approach for the unsupervised extraction of foreground objects. (b) Background suppressed for visualization purposes.

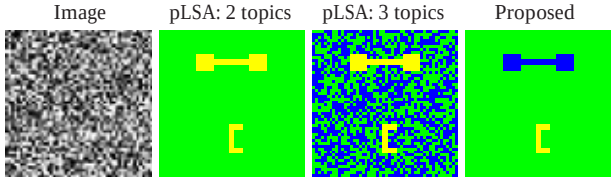


Figure 8: Comparison of results obtained using pLSA with those obtained using our proposed approach for the unsupervised extraction of foreground objects.

spatial information. Hence, pLSA is inherently missing the information required to perceive the features on one of the foreground objects any different than those on the second object, which is required to separate them.

On the other hand, our approach does incorporate this spatial/geometric information and hence can separate the foreground objects. Since the input images are assumed to allow only translation of the foreground objects, and the descriptor is simply the intensity value, we alter the notion of geometric consistency than that described in Section 4.2. In order to compute the geometric consistency between a pair of correspondences, we compute the distance between the pairs of features in both images. The geometric consistency decreases exponentially as the discrepancy in the distances increases. The result obtained by our approach is shown in Figure 8. We successfully identify the foreground from the background and further separate the foreground into multiple objects. Also, our approach does not require any parameters to be specified, such as number of topics or foreground objects in the images. The inability of a popularity-based approach for obtaining the desired results illustrates the need for geometric consistency in addition to popularity.

In order to illustrate the need for considering popularity and not just geometric consistency, let us consider the following analysis. If we consider all pairs of images such as those shown in Figure 7 and keep all features that find correspondences that are geometrically consistent with at least one other feature in at least one other image, we would retain approximately 2300 of the background features. This is because even for background, it is possible to find at least some geometrically consistent correspondences. However, by the background being random, this would not be consistent across several images. Hence, instead of retaining features that have geometrically consistent correspondences in one other image, if we now retain only those that have geometrically consistent correspondences in at least two other images, only about 50 of the background features are retained. As we use more images, we can eliminate the background features entirely. By our approach being unsupervised, the use of multiple images to prune out background clutter is crucial. Hence, this demonstrates the need for considering popularity in addition to geometric consistency.

5.2. Learning hSO

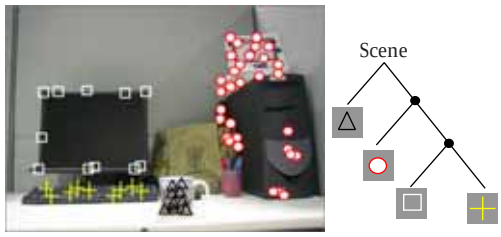
We now present experimental results on the unsupervised learning of hSO from a collection of images. It should be noted that the goal of this work is not to improve object recognition through better feature extraction or matching. We focus our efforts on learning the hSO that codes the different interactions among objects in the scene by using well-matched parts of objects, and not on the actual matching of parts. This work is complementary to the recent advances in object recognition that enable us to deal with object categories and not just specific objects. These advances indicate the feasibility to learn hSO even among objects categories. However, in our experiments we use specific objects with SIFT features to demonstrate our proposed algorithm. SIFT is not an integral part of our approach. This can easily be replaced with patches, shape features, and so forth, with appropriate matching techniques as may be appropriate for the scenario at hand—specific objects or object categories. Future work includes experiments in such varied scenarios. Several different experimental scenarios were used to learn the hSOs. Due to lack of standard datasets where interactions between multiple objects can be modeled, we use our own collection of images. The rest of the experiments use the descriptors as well as geometric consistency notions as described in our approach in Section 4.

5.2.1. Scene semantic analysis

Consider a surveillance type scenario where a camera is monitoring, say an office desk. The camera takes a picture of the desk every few hours. The hSO characterizing this desk, learnt from this collection of images, could be used for robust object detection in this scene, in the presence of occlusion due to a person present, or other extraneous objects on the desk. Also, if the objects on the desk are later found in an arrangement that cannot be explained by the hSO, that can be detected as an anomaly. Thirty images simulating such a scenario were taken. Examples of these can be seen in

(a) (b) (c) (d)

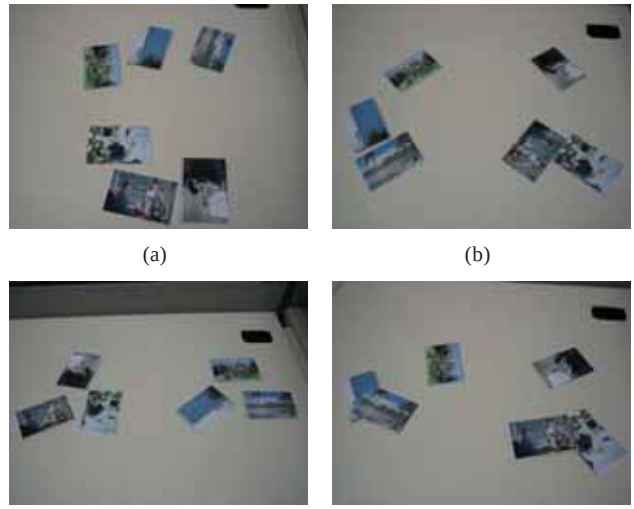
FIGURE 9: A subset of images provided as input to learn the corresponding hSO.



(a) (b)

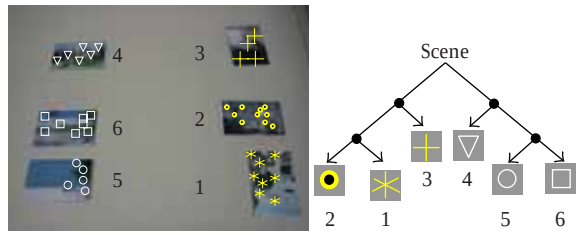
FIGURE 10: Results of the hSO learning algorithm. (a) The cloud of features clustered into groups. Each group corresponds to an object in the foreground. (b) The corresponding learnt hSO which captures meaningful relationships between the objects.

FIGURE 11: The six photos that users arranged.



(a) (b) (c) (d)

FIGURE 12: A subset of images of the arrangements of photos that users provided for which the corresponding hSO was learnt.



(a) (b)

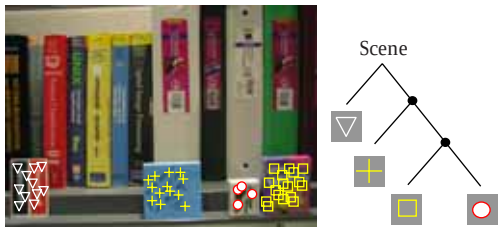
FIGURE 13: Results of the hSO learning algorithm. (a) The cloud of features clustered into groups. Each group corresponds to a photograph. (b) The corresponding learnt hSO which captures the appropriate semantic relationships among the photos. Each cluster and photograph is tagged with a number that matches those shown in Figure 11 for clarity.

Figure 9. Note the occlusions, background clutter, change in scale and viewpoint, and so forth. The corresponding hSO as learnt from these images is depicted in Figure 10.

Several different interesting observations can be made. First, the background features are mostly eliminated. The features on the right side of the bag next to the CPU are retained while the rest of the bag is not. This is because, due to several occlusions in the images, most of the bag is occluded in images. However, the right side of the bag resting on the CPU is present in most images, and hence is interpreted to be foreground. The monitor, keyboard, CPU, and mug are selected to be the objects of interest (although the mug is absent in some images). The hSO indicates that the mug is found at the most unpredictable locations in the image, while the monitor and the keyboard are clustered together till the very last stage in the hSO. This matches our semantic understanding of the scene. Also, since the photo frame, the right side of the bag, and the CPU are always found at the same location with respect to each other across images (they are stationary), they are clustered together as

(a) (b) (c) (d)

Figure 14: A subset of images of staged objects provided as input to learn the corresponding hSO.



(a) (b)

Figure 15: Results of the hSO learning algorithm. (a) The cloud of features clustered into groups. Each group corresponds to an object in the foreground. (b) The corresponding learnt hSO which matches the ground truth hSO.

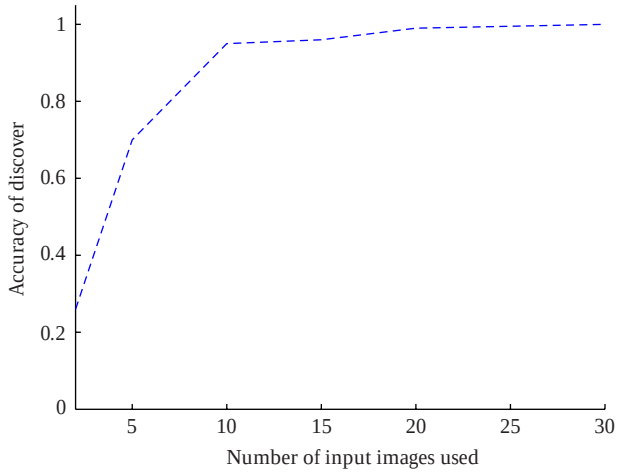


Figure 16: The accuracy of the learnt hSO as more input images are provided.

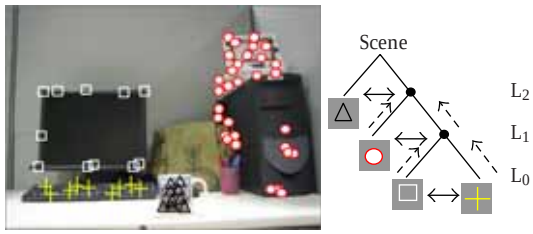


Figure 17: The simple information flow used within hSO for context for proof-of-concept. Solid bi-directional arrows indicate exchange of context. Dotted directional arrows indicate flow of (refined) detection information. The image on the left is shown for reference for what objects the symbols correspond to.

Figure 18: Test image in which the four objects of interest are to be detected. Significant occlusions are present.

the same object. By ours being an unsupervised approach, this artifact is expected, even natural, since there is in fact no evidence indicating these entities to be separate objects.

5.2.2. *Photo grouping*

We consider an example application where the goal is to learn the semantic hierarchy among photographs. This experiment is to demonstrate the capability of the proposed algorithm to truly capture the semantic relationships, by bringing users in the loop, since semantic relationships are not a very tangible notion. We present users with 6 photos: 3 outdoor (2 beaches, 1 garden) and 3 indoor (2 with a person in an office, 1 empty office). These photos can be seen in Figure 11. The users were instructed to group these photos such that the ones that are similar are close by. The number of groups to be formed was not specified. Some users made two groups (indoor versus outdoor), while some made four groups by further separating these two groups into two each. We took pictures that capture 20 such arrangements. Example images are shown in Figure 12. We use these images to learn the hSO. The results obtained are shown in Figure 13.

We can see that the hSO can capture the semantic relationships among the images, the general (indoor versus outdoor) as well as more specific ones (beaches versus garden) through the hierarchical structure. It should be noted that the content of the images was not utilized to compute the similarity between images—this is based purely on the user arrangement. In fact, it may be argued that although this grouping seems very intuitive to us, it may be very challenging to obtain this grouping through low-level features extracted from the photos. Such an hSO on a larger number of images can hence be used to empower a content-

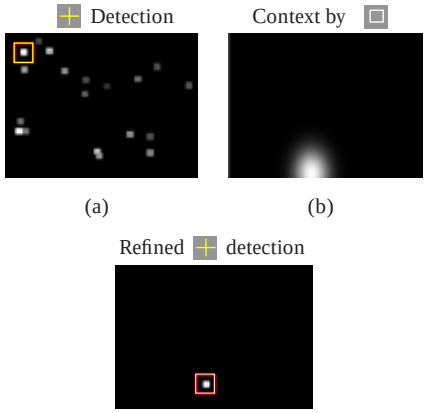


FIGURE 19: (a) Candidate detections of keyboard, along with the max score (incorrect) detection. (b) Context prior provided by detected monitor. (C) Detections of keyboard after applying context from monitor along with the max score (correct) detection. The centers of the candidate detections are shown.

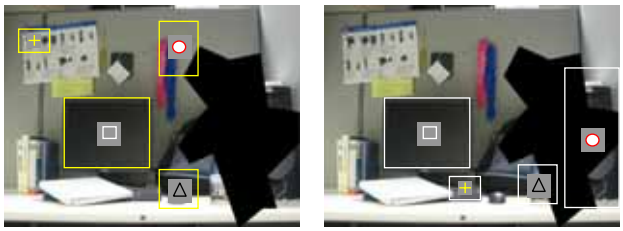


FIGURE 20: (a) Detections of the 4 objects without context—3 of 4 are incorrect due to significant occlusions. (b) Detections with context—all 4 are correct.

based digital image retrieval system with the users' semantic knowledge. In such a case, a user interface, similar to [56], may be provided to users and merely the position of each image can be noted to learn the underlying hSO without requiring feature extraction and image matching. In [56], although user preferences are incorporated, a hierarchial notion of interactions is not employed which provides much richer information.

5.2.3. *Quantitative results*

In order to better quantify the performance of the proposed learning algorithm, a hierarchy among objects was staged, that is, the ground truth hSO is known. As shown in the example images in Figure 14, two candy boxes are placed mostly next to each other, a post-it-note around them, and an entry card is tossed arbitrarily. Thirty such images were captured against varying cluttered backgrounds. Note the rotation and change in view point of the objects as well as varying lighting conditions. These were hand labeled so that the ground truth assignments of the feature points to different nodes in the hSO are known and accuracies can be computed. The corresponding hSO was learnt from the unlabeled images. The results obtained are as seen in Figure 15. The feature points have been clustered appropriately, and the learnt hSO matches the description of the ground truth hSO above. The clutter in the background has been successfully eliminated. Quantitative results reporting the accuracy of the learnt hSO, measured as the proportion of features assigned to the correct level in the hSO, with varying number of images used for learning, are shown in Figure 16. It can be seen that with significantly few images a meaningful hSO can be learnt. It should be noted that this accuracy simply reports the percentage of features detected as foreground that were assigned to the right levels in the accuracy. While it penalizes background features considered as foreground, it does not penalize dropping foreground features as background and hence not consider them in the hSO. Visual quality of results indicates that such a metric suffices. In less textured objects, the accuracy metric would need to be reconsidered.

6. hSO TO PROVIDE CONTEXT

Consider the hSO learnt for the office scene in Section 5.2.1 as shown in Figure 17. Consider an image of the same scene (not part of the learning data) as shown in Figure 18 which has significant occlusions (real on the keyboard, and synthetic on the CPU and mug). We wish to detect (we use *detection* and *localization* interchangeably) the four foreground objects.

The leaves of the hSO hold the clouds of features (along with their locations) for the corresponding objects. To detect the objects, these are matched with features in the test image through geometrically consistent correspondences similar to that in Section 4.2. Multiple candidate detections along with their corresponding scores are retained, as seen in Figure 19(a). The location of a detection is the centroid of the matched features in the test image. The detection with the highest score is determined to be the final localization. Due to significant occlusions, background may find candidate detections with higher scores and hence the object would be miss detected, as seen in Figure 20(a), where three of the four objects are incorrectly localized.

In the presence of occlusion, even if a background match has a higher score, it will most likely be pruned out if we consider some contextual information (prior). To develop some intuition, we present a simple greedy algorithm to apply hSO to provide this contextual information for object localization. The flow of information used to incorporate the context is shown in Figure 17. In the test image, candidate detections of the foreground objects at the lowest level (L_0) in the hSO structure are first determined. The context prior provided by each of these (two) objects is applied to the other object, and these detections are pruned/refined as shown in Figure 19. The distribution in Figure 19 (middle) is strongly peaked because it indicates the relative location of the keyboard with respect to the monitor, which is quite predictable. However, the distribution of the absolute location of the keyboard across the training images as shown in Figure 9 is significantly less peaked. The hSO allows us to condition on the appropriate objects and obtain such peaked contextual distributions. This refined detection information

is passed on to the next higher level (L_1) in the hSO, which constitutes the detection information of the super object containing these two objects, which in turn provides context for refining the detection of the other object at L_1, and so on.

The detection results obtained by using context with this greedy algorithm is shown in Figure 20(b) which correctly localizes all four objects. The objects, although significantly occluded, are easily recognizable to us. So the context is not hallucinating the objects entirely, but the detection algorithm is amplifying the available (little) evidence at hand, while enabling us not to be distracted by the false background matches.

We now describe a more formal approach for using the hSO for providing context for object localization, along with thorough experiments. We also compare the performance of hSO (tree-structure) to a fully connected structure.

6.1. Approach

Our model is a conditional random field where the structure of the graphical model is the same as the learnt hSO. Hence, we call our graphical model an hSO-CRF. The nodes of the hSO-CRF are the nodes of the hSO (the leaves being the objects and intermediate nodes being the super objects). The state of each node is one of the location grids in the image. Our model thus assumes that every object is present in the image exactly once. Future work involves generalizing this assumption and making use of the cooccurrence statistics of objects that can be learnt during the learning stage to aid this generalization.

Say we have N nodes (entities) in the hSO-CRF. The location of the ith entity is indicated by l_i. Since the image is divided into a $G \times G$ grid, $l_i \in \{1, \ldots, G^2\}$. We model the conditional probability of the locations of the objects $L = (l_1, \ldots, l_{G^2})$ given the image as

$$P(\mathbf{L}|\mathbf{I}) = \frac{1}{Z}\prod_{i=1}^{N}\Psi_i(l_i)\prod_{(i,j)\in E}\Phi_{ij}(l_i, l_j), \tag{2}$$

where Z is the partition function, and E is the set of all edges in the hSO-CRF. The data term $\Psi_i(l_i)$ computes the probability of location of the ith entity l_i across the entire image I. The pairwise potentials $\Phi_{ij}(l_i, l_j)$ capture the contextual information between entities using the learnt relative location statistics from the learning stage.

6.1.1. Appearance

To compute our data term $\Psi_i(l_i)$ for the leaves of the hSO-CRF, we first match the object models stored at the leaves of the hSO to the test image as explained earlier, to obtain a detection map as shown in Figure 19(a). For each bin in the grid, we compute the maximum matching score, which is then normalized to obtain a distribution $p(l_i|I)$. Our data term (node potential) is then $\Psi_i(l_i) = p(l_i|I)$, which is a vector of length G^2. The data term for the nodes corresponding to the super objects is set to a uniform distribution over all the bins.

(a) (b)

Figure 21: Illustrations of the two types of occlusions we experiment with: (a) uniform occlusion and (b) localized occlusion. In our experiments, the amount of occlusion is varied.

6.1.2. Context

The edge interactions $\Phi_{ij}(l_i, l_j)$ capture the contextual information between the ith and jth entities through relative location statistics. This is modeled as the empirical probability of the ith and jth entities occurring at locations l_i and l_j. This was learnt through MLE counts during the learning stage.

We use loopy belief propagation to perform inference on the hSO-CRF using a publicly available implementation [57]. After convergence, for each object, the bin with the highest belief is inferred to be the location of object. Generally, we are not interested in the location of the super objects, but those can be inferred similarly if required.

6.2. Experimental setup

To demonstrate the use of hSO in providing context for object localization, we wish to compare the performance of hSO-CRF in providing context, to that of a fully connected CRF (which we call f-CRF) over the objects. The f-CRF is modeled similar to (2), except in this case E which consists of all the edges in the fully connected graph, and N which is the number of objects and not the total number of entities, that is, the f-CRF is over the objects in the images, and hence there is no concept of super objects in an f-CRF. The node potentials and edge potentials of the f-CRF are computed in a similar manner as the hSO-CRF. We collect test images in a similar setting as those used to learn the hSO (since the learning is unsupervised, the same images could also be used). We collect images from the office scene (example images of which are in Figure 9). We test only on those images that contain all the foreground objects exactly once (which form a majority of images since the foreground objects by definition occur often). We hand labeled the locations of the foreground objects in these images so that localization accuracies can be computed using these labels as ground truth.

As demonstrated in [58], the use of context is beneficial in scenarios where the appearance information is not sufficient. We simulate such a scenario with occlusions. We consider two different forms of occlusions: a uniformly distributed occlusion and a localized occlusion. The uniformly distributed occlusion is obtained by randomly (uniformly across the image) removing detected features in the image.

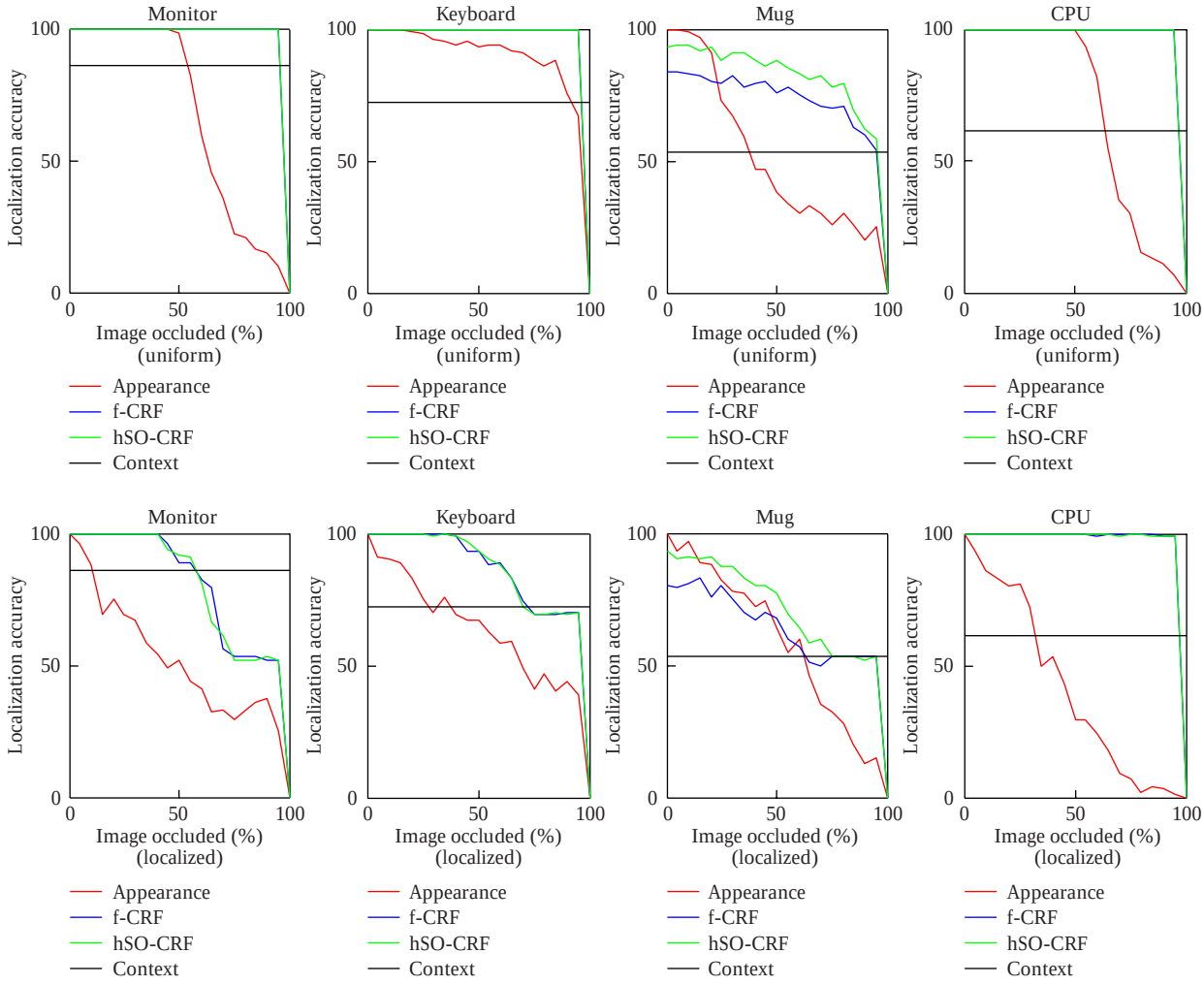


FIGURE 22: Localization results.

We show illustrations of this in Figure 21(a). It should be noted that we show blacked out pixels as an illustration, in reality, instead of blacking out pixels and then detecting features (which could cause several undesirable artifacts because of the nature of the SIFT detector and descriptor), we first detect features in the image and then randomly black out some of the features. This mimics a scenario where the images are of much lower resolution, and hence fewer features are detected in the image, making the localization task hard. The second type of occlusion is a more localized occlusion (perhaps closer to the conventional occlusions). In order to simulate this, we black out a square block of the image placed randomly in the image. An example of this is shown in Figure 21(b). In both types of occlusions, we vary the amounts of occlusions added to the test images.

The results obtained are shown in Figure 22. We show the localization accuracies for all four foreground objects: monitor, keyboard, mug, and CPU for the office scenario for which the hSO was learnt as shown in Section 5.2.1, for the two types of occlusions and for varying amounts of occlusions. We compare the accuracies of hSO-CRF to that of f-CRF. Recall that the learnt hSO as shown in Figure 10 indicates that the monitor and keyboard are most related, followed by the CPU, and the mug was the most unrelated/unpredictable in the scene. For more insight in the test scenario, we also report accuracies of using appearance information alone (edge potentials on the hSO-CRF were set to uniform) and using contextual information alone (node potentials in the hSO-CRF for all the objects were set to uniform). The accuracies of the hSO-CRF and f-CRF are similar for most objects. And since f-CRF is a fully connected network and hence much more complex to run inference on as opposed to hSO-CRF which has a tree structure, the advantage of hSO-CRF is clear. Moreover, the accuracy of hSO-CRF for the mug is much higher than that for f-CRF. This validates our claim that f-CRF is prone to over fitting because it explicitly models relationships among objects that may be unrelated, while the hSO-CRF models relationships only among entities that are related.

We find that in the presence of very little occlusion, appearance information alone has higher localization accuracy for the mug than both f-CRF and hSO-CRF (however, hSO-CRF has significantly higher accuracy than f-CRF). This is again because that the location of a mug is unpredictable,

and hence if available, the appearance information is most reliable. In general, we find that the localization accuracies for the uniform occlusion are higher than that for the localized occlusions. This makes intuitive sense. Also, similar to the findings of Parikh et al. [58], we find that context provides a boost in performance only when the appearance information is weak, and not otherwise. Another observation is that the monitor and keyboard localization accuracies using both hSO-CRF and f-CRF with significant amount of localized occlusions are lower than using context alone (no appearance information). This indicates that extremely poor appearance information can hurt the performance as compared to using no appearance information at all and relying only on learnt contextual statistics. This indicates that depending on the scenario (amount of occlusion), roles of appearance and contextual information vary. Overall, the performance of hSO-CRF is the most reliable.

7. CONCLUSION

We introduced hierarchical semantics of objects (hSOs) that capture potentially semantic relationships among objects in a scene as observed by their relative positions in a collection of images. The underlying entity is a patch, however the hSO goes beyond patches and represents the scene at various levels of abstractness—ranging from patches on individual objects to objects and groups of objects in a scene. An unsupervised hSO learning algorithm has been proposed. Given a collection of images of a scene, the algorithm can identify the foreground parts of the images, group the parts to form clusters corresponding to the foreground objects, learn the appearance models of these objects as well as relative locations of semantically related objects, and use these to provide context for robust object detection even with significant occlusions—all automatically and entirely unsupervised. This, we believe, takes us a step closer to true image understanding. We demonstrate the need for popularity as well as geometric consistency based cues for successful extraction of multiple foreground objects. We also demonstrate the effectiveness of a meaningful hierarchical structure to provide context for object localization as compared to a fully connected network that is prone to over fitting. Future work involves generalizing the proposed approach for learning hierarchical relationships among parts of categories of objects in addition to multiple objects through a unified treatment.

REFERENCES

[1] D. Parikh and T. Chen, "Hierarchical semantics of objects (hSOs)," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV '07)*, pp. 1–8, Rio de Janeiro, Brazil, October 2007.

[2] D. Parikh and T. Chen, "Unsupervised identification of multiple objects of interest from multiple images: dISCOVER," in *Proceedings of the 8th Asian Conference on Computer Vision (ACCV '07)*, vol. 4844 of *Lecture Notes in Computer Science*, pp. 487–496, Tokyo, Japan, November 2007.

[3] A. Fitzgibbon and A. Zisserman, "On affine invariant clustering and automatic cast listing in movies," in *Proceedings of the 7th European Conference on Computer Vision (ECCV '02)*, vol. 2352 of *Lecture Notes in Computer Science*, pp. 243–261, Copenhagen, Denmark, May 2002.

[4] J. Sivic and A. Zisserman, "Video data mining using configurations of viewpoint invariant regions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 1, pp. 488–495, Washington, DC, USA, June-July 2004.

[5] T. L. Berg, A. C. Berg, J. Edwards, et al., "Names and faces in the news," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 848–854, Washington, DC, USA, June-July 2004.

[6] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, no. 4-5, pp. 993–1022, 2003.

[8] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 524–531, San Diego, Calif, USA, June 2005.

[9] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van Gool, "Modeling scenes with local descriptors and latent aspects," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '05)*, vol. 1, pp. 883–890, Beijing, China, October 2005.

[10] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their location in images," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '05)*, vol. 1, pp. 370–377, Beijing, China, October 2005.

[11] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," in *Proceedings of the International Workshop on Statistical Learning in Computer Vision (ECCV '04)*, pp. 1–22, Prague, Czech Republic, May 2004.

[12] M. Leordeanu and M. Collins, "Unsupervised learning of object models from video sequences," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 1142–1149, San Diego, Calif, USA, June 2005.

[13] D. Liu and T. Chen, "Semantic-shift for unsupervised object detection," in *Proceedings of Conference on Computer Vision and Pattern Recognition Workshop (CVPR '06)*, p. 16, New York, NY, USA, June 2006.

[14] Y. Li, W. Wang, and W. Gao, "A robust approach for object recognition," in *Proceedings of the 7th Pacific Rim Conference on Advances in Multimedia Information Processing (PCM '06)*, vol. 4261 of *Lecture Notes in Computer Science*, pp. 262–269, Hangzhou, China, November 2006.

[15] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from Google's image search," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1816–1823, Beijing, China, October 2005.

[16] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2169–2178, New York, NY, USA, June 2006.

[17] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman, "Using multiple segmentations to discover objects

and their extent in image collections," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 1605–1612, New York, NY, USA, June 2006.

[18] M. Marszałek and C. Schmid, "Spatial weighting for bag-of-features," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2118–2125, New York, NY, USA, June 2006.

[19] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *Proceedings of the 8th European Conference on Computer Vision (ECCV '04)*, vol. 3022 of *Lecture Notes in Computer Science*, pp. 17–32, Prague, Czech Republic, May 2004.

[20] R. Fergus, P. Perona, and A. Zisserman, "Object class recognition by unsupervised scale-invariant learning," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, vol. 2, pp. 264–271, Madison, Wis, USA, June 2003.

[21] M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," in *Proceedings of the 6th European Conference on Computer Vision (ECCV '00)*, vol. 1842 of *Lecture Notes in Computer Science*, pp. 18–32, Dublin, Ireland, June-July 2000.

[22] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van Gool, "Modeling scenes with local descriptors and latent aspects," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 1, pp. 883–890, Beijing, China, October 2005.

[23] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky, "Learning hierarchical models of scenes, objects, and parts," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1331–1338, Beijing, China, October 2005.

[24] G. Wang, Y. Zhang, and L. Fei-Fei, "Using dependent regions for object categorization in a generative framework," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 1597–1604, New York, NY, USA, June 2006.

[25] D. Marr and H. K. Nishihara, "Representation and recognition of the spatial organization of three-dimensional shapes," *Proceedings of the Royal Society of London. Series B*, vol. 200, no. 1140, pp. 269–294, 1978.

[26] I. Biederman, "Human image understanding: recent research and a theory," *Computer Vision, Graphics, and Image Processing*, vol. 32, no. 1, pp. 29–73, 1985.

[27] E. Bienenstock, S. Geman, and D. Potter, "Compositionality, MDL priors, and object recognition," in *Advances in Neural Information Processing Systems (NIPS '97)*, pp. 838–844, Denver, Colo, USA, December 1997.

[28] A. Levinshtein, C. Sminchisescu, and S. Dickinson, "Learning hierarchical shape models from examples," in *Proceedings of the 5th International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR '05)*, vol. 3757 of *Lecture Notes in Computer Science*, pp. 251–267, St. Augustine, Fla, USA, 2005.

[29] G. Bouchard and B. Triggs, "Hierarchical part-based visual object categorization," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 710–715, San Diego, Calif, USA, June 2005.

[30] Y. Jin and S. Geman, "Context and hierarchy in a probabilistic image model," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2145–2152, New York, NY, USA, 2006.

[31] S. Fidler, G. Berginc, and A. Leonardis, "Hierarchical statistical learning of generic parts of object structure," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 1, pp. 182–189, New York, NY, USA, 2006.

[32] J. M. Siskind, J. J. Sherman Jr., I. Pollak, M. P. Harper, and C. A. Bouman, "Spatial random tree grammars for modeling hierarchical structure in images with regions of arbitrary shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1504–1519, 2007.

[33] D. A. Forsyth, J. L. Mundy, A. Zisserman, and C. A. Rothwell, "Using global consistency to recognise euclidean objects with an uncalibrated camera," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '94)*, pp. 502–507, Seattle, Wash, USA, June 1994.

[34] A. Torralba and P. Sinha, "Statistical context priming for object detection," in *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV '01)*, vol. 1, pp. 763–770, Vancouver, Canada, July 2001.

[35] A. Torralba and A. Oliva, "Depth estimation from image structure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1226–1238, 2002.

[36] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 2137–2144, New York, NY, USA, June 2006.

[37] C. K. I. Williams and N. J. Adams, "DTs: dynamic trees," in *Advances in Neural Information Processing Systems (NIPS '99)*, pp. 634–640, Denver, Colo, USA, November-December 1999.

[38] G. E. Hinton, Z. Ghahramani, and Y. W. Teh, "Learning to parse images," in *Advances in Neural Information Processing Systems (NIPS '00)*, pp. 463–469, Denver, Colo, USA, November-December 2000.

[39] A. J. Storkey and C. K. I. Williams, "Image modeling with position-encoding dynamic trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 859–871, 2003.

[40] K. Murphy, A. Torralba, and W. Freeman, "Using the forest to see the trees: a graphical model relating features, objects, and scenes," in *Advances in Neural Information Processing Systems (NIPS '03)*, pp. 1499–1506, Vancouver, Canada, December 2003.

[41] X. He, R. S. Zemel, and M. Á. Carreira-Perpiñán, "Multiscale conditional random fields for image labeling," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 695–702, Washington, DC, USA, June-July 2004.

[42] S. Kumar and M. Hebert, "A hierarchical field framework for unified context-based classification," in *Proceedings of the 10th IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1284–1291, Beijing, China, October 2005.

[43] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu, "Image parsing: unifying segmentation, detection, and recognition," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 113–140, 2005.

[44] A. Torralba, K. Murphy, W. Freeman, and M. Rubin, "Context-based vision system for place and object recognition," in *Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03)*, vol. 1, pp. 273–280, Nice, France, October 2003.

[45] A. Torralba, K. Murphy, and W. Freeman, "Contextual models for object detection using boosted random fields," in *Advances in Neural Information Processing Systems (NIPS '05)*, pp. 1401–1408, Vancouver, Canada, December 2005.

[46] M. Marszałek and C. Schmid, "Semantic hierarchies for visual object recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 1–7, Minneapolis, Minn, USA, June 2007.

[47] A. Singhal, J. Luo, and W. Zhu, "Probabilistic spatial context models for scene content understanding," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '03)*, vol. 1, pp. 235–241, Madison, Wis, USA, June 2003.

[48] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[49] J. Vogel and B. Schiele, "A semantic typicality measure for natural scene categorization," in *Proceedings of the 26th DAGM Symposium on Pattern Recognition*, vol. 3175 of *Lecture Notes in Computer Science*, pp. 195–203, Tübingen, Germany, August-September 2004.

[50] D. G. Lowe, "Distinctive image features from scaleinvariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–100, 2004.

[51] S. Belongie, J. Malik, and J. Puzicha, "Shape context: a new descriptor for shape matching and object recognition," in *Advances in Neural Information Processing Systems (NIPS '00)*, pp. 831–837, Denver, Colo, USA, November-December 2000.

[52] A. C. Berg and J. Malik, "Geometric blur for template matching," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 607–614, Kauai, Hawaii, USA, December 2001.

[53] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '05)*, vol. 2, pp. 1482–1489, Beijing, China, October 2005.

[54] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[55] J. Shi, http://www.cis.upenn.edu/~jshi/software.

[56] M. Nakazato, L. Manola, and T. Huang, "ImageGrouper: search, annotate and organize image by groups," in *Proceedings of the 5th International Conference on Recent Advances in Visual Information (VISual '02)*, Hsin Chu, Taiwan, March 2002.

[57] T. Meltzer, "Inference package for undirected graphical models," http://www.cs.huji.ac.il/~talyam/inference.html.

[58] D. Parikh, C. L. Zitnick, and T. Chen, "From appearance to context-based recognition: dense labeling in small images," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, Anchorage, Alaska, USA, June 2008.