Unsupervised Object Class Discovery via Saliency-Guided Multiple Class Learning

Jun-Yan Zhu, Jiajun Wu, Yan Xu, Eric Chang, and Zhuowen Tu,

Abstract—In this paper, we tackle the problem of common object (multiple classes) discovery from a set of input images, where we assume the presence of one object class in each image. This problem is, loosely speaking, unsupervised since we do not know a priori about the object type, location, and scale in each image. We observe that the general task of object class discovery in a fully unsupervised manner is intrinsically ambiguous; here we adopt saliency detection to propose candidate image windows/patches to turn an unsupervised learning problem into a weakly-supervised learning problem. In the paper, we propose an algorithm for simultaneously localizing objects and discovering object classes via bottom-up (saliency-guided) multiple class learning (bMCL). Our contributions are three-fold: (1) we adopt saliency detection to convert unsupervised learning into multiple instance learning, formulated as bottom-up multiple class learning (bMCL); (2) we propose an integrated framework that simultaneously performs object localization, object class discovery, and object detector training; (3) we demonstrate that our framework yields significant improvements over existing methods for multi-class object discovery and possess evident advantages over competing methods in computer vision. In addition, although saliency detection has recently attracted much attention, its practical usage for high-level vision tasks has yet to be justified. Our method validates the usefulness of saliency detection to output "noisy input" for a top-down method to extract common patterns.

Index Terms—Unsupervised object discovery, object detection, multiple instance learning, weakly supervised learning, saliency.

1 Introduction

The computer vision field has witnessed milestone achievements in building real-world object detection systems [20], [47], [53]. However, these methods all require a large amount of labeled training data to build practically applicable systems. Recently, many unsupervised approaches have been proposed to perform object localization and categorization [27], [30], [43], [50], [61]. While many of these approaches report encouraging results on datasets like Caltech-101 [19], most of these existing approaches work under restrictive conditions such as large and centered foreground objects with clean backgrounds. However, in practice foreground objects often have large scale differences and are not centered; the background is also frequently cluttered, as indicated by unsupervised scene discovery research [35].

In this paper, we design a system for the discovery of unknown but common objects of multiple classes from a given set of images. This problem is known as *unsupervised object class discovery* [50] in which the input includes a set of unlabeled images. Due to differences in respective final goals and forms of outputs, the specific approaches to the unsupervised object discovery task can be very different. Figure 1 gives an illustration of some possible alternative paths for algorithm design. In particular, we consider three (nested) approaches for the object discovery task in which the input is a set of unlabeled images.

Approach I. *Output: image-level cluster labels.* The goal in this approach is to cluster the input images, with the desire that all the images of the same object class would be placed in a pure cluster corresponding to that class. The pro is that existing unsupervised clustering algorithms can be utilized. The con is that the object in each image is not localized, which can contribute to clustering error and limit the subsequent usefulness of the output. See [50] for a review of work along this line.

Approach II. *Output: localized objects.* The goal in this approach is to localize the object (of unknown class) in each image; image-level cluster label is then naturally determined from that image's localized object. The pro is that objects are detected and identified, outcomes that can then be used as the input for subsequent tasks. The con is that object class models are not explicitly learned within the framework, which reduces the scope of application of, for example, the object detection, due to the lack of a corresponding integrated object class model. A typical example of this approach is [42].

Approach III. *Output: learned object class models.* The goal in this approach is to automatically learn object models, which can then be naturally used to detect the object in each image. The pro is that object localization, object class discovery, and object detector training are all

- *J.-Y. Zhu is with the Computer Science Division, University of California at Berkeley, Berkeley, CA 94720, U.S.A. E-mail: junyanz@eecs.berkeley.edu*
- *J. Wu is with the Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, P.R. China, 100084. E-mail: jiajunwu.cs@gmail.com*
- *Y. Xu is with Beihang University, No. 37 Xueyuan Road, Beijing, P.R. China, 100191. E-mail: xuyan04@gmail.com*
- *E. Chang is with Microsoft Research Asia, No. 5 Danling Street, Beijing, P.R. China, 100080. E-mail: echang@microsoft.com*
- *Z. Tu is with the Department of Cognitive Science, University of California, San Diego, La Jolla, CA 92093, U.S.A. E-mail: ztu@ucsd.edu*

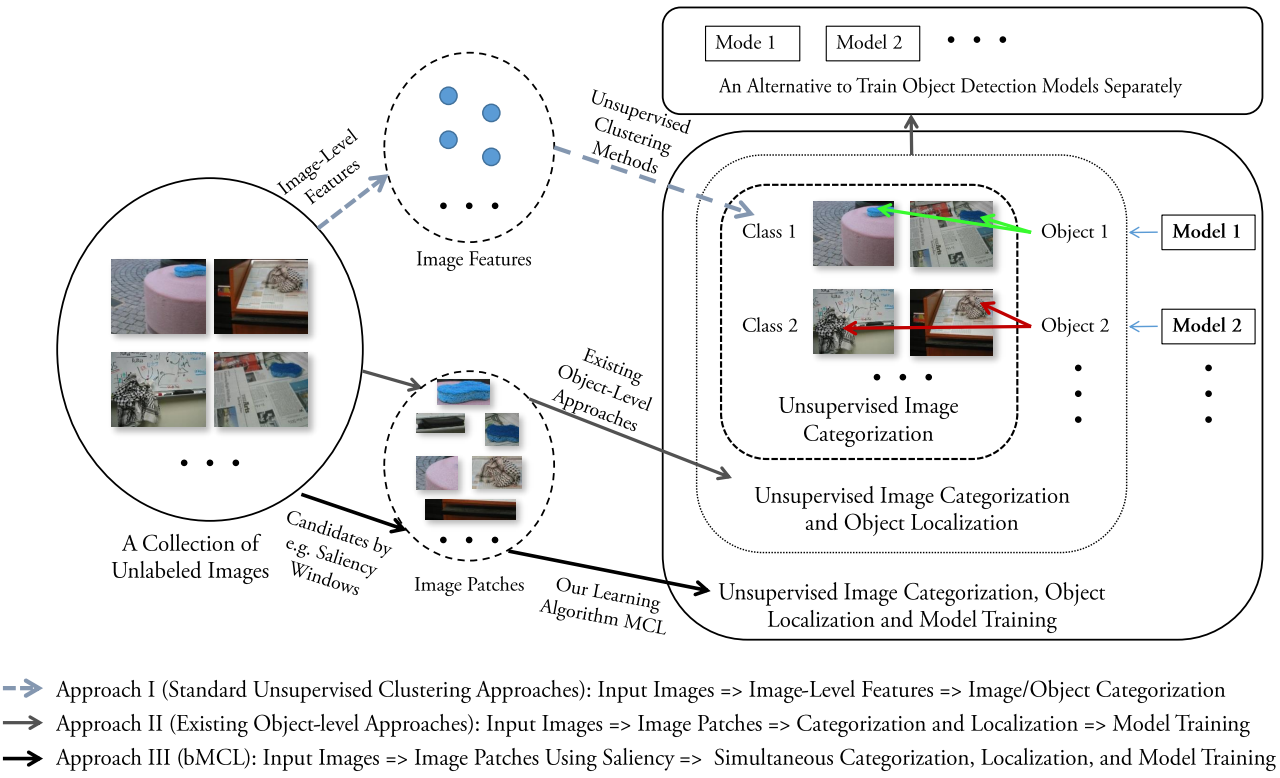


Fig. 1: An overview of the *unsupervised object class discovery* problem. The input is the same for different types of algorithms: a set of unlabeled images. On the other hand, since different algorithms have different purposes, the outputs of the algorithms will vary according to those purposes.

performed in an integrated framework. The con is that the complexity of the system might be high. We take this approach here.

Approach I, II, and III belong to a nested family shown in Figure 1. As we can see, if we can successfully localize and differentiate the objects (approach II), the image-level cluster label (approach I) can be obtained easily; if we can learn the explicit object models (approach III), then object localization can be directly performed by applying the object models to the images (approach II). Figure 1 gives an overview of the unsupervised object class discovery problem and Figure 2 illustrates the specific strategy of the method proposed in the present paper, discussed in much more detail later.

Before continuing, we observe that the general task of fully unsupervised object class discovery is intrinsically ambiguous. This is due to large variations, corruptions, foreground object outliers, as well as to the inherent ambiguity between complex objects and background clutter. Despite this ambiguity, it is nevertheless desirable to build an unsupervised object discovery system with relatively loose constraints due to its much lighter human labeling requirements and its general adaptability. With the assumption that the common objects across multiple images live in an intrinsically lower-dimensional space, we extract many local image region windows from each image, pick "correct" image windows that contain the objects of interest, and then naturally perform clustering. As described above, this problem is evidently highly combinatorial and high-dimensional. Here we show how to tackle this daunting task using our proposed bottom-up Multiple Class Learning (bMCL) approach.

In this paper, we adopt saliency detection to place the original unsupervised problem into a multiple instance learning context [12]. Our framework has the following new aspects: (1) Unlike the direct top-down discovery of object classes [57], [58] or the use of specifically trained classifiers [11], we utilize bottom-up saliency detection to guide top-down learning in unsupervised object discovery. We create negative training examples (bags) containing the least salient windows for each particular image, which is a unique property of our method. (2) Object localization, object class discovery, and object detector training are performed simultaneously in an integrated framework, named bottom-up Multiple Class Learning (bMCL). (3) Our algorithm demonstrates significant improvements over existing systems on challenging benchmark datasets. Figure 2 illustrates our bMCL approach.

We now briefly discuss the general concepts underlying our learning framework. Multiple instance learning (MIL) [12] occupies a middle ground between completely unsupervised learning and completely supervised learning; in MIL, we are provided with weak supervision in the form of image (bag) level labels rather than the full supervision of detailed annotation of object locations. MIL thus significantly reduces the manual effort in order to build object detection systems [3], [14], [54]. Furthermore, when multiple object classes are

present, it is desirable to automatically discover them simultaneously in the MIL scheme.

In the machine learning literature, several multiple instance clustering (MIC) algorithms [57], [58] have been designed to perform localized content-based image clustering. These methods introduce the multiple instance concept into standard clustering methods such as K-means or maximum margin clustering [55]. However, most of the existing MIC solutions report discouraging cluster purity results (e.g., 37.1%) [57], [58] in the SIVAL benchmark dataset [41]; just as discouraging is the fact that they do not perform simultaneous object localization. In comparison, while state-of-the-art unsupervised object discovery methods [27], [30] perform well in Caltech-101 (98% in purity), when applied to SIVAL, their cluster purity [30] declines to 28.3%.

The use of saliency scoring to generate positive and negative bags is an important aspect of our method. Saliency detection has become an active research area [8], [21], [25] where objects of interest are assumed to be "salient" in images. Recently, a related idea called "objectness" has appeared [1], [16]; "objectness" is similar to the concept of saliency but is more specific to high-level knowledge. Another related method [11] uses a classifier trained on several classes of objects as "meta information" that is then used to learn other object types. Such learned "objectness" detectors have been adopted in systems, e.g. in [6] for the PASCAL object segmentation task. Although saliency detection is an active research area, there has remained uncertainty about the effectiveness of saliency detection in high-level vision tasks; we demonstrate that in the unsupervised object discovery task, the notion of saliency guidance can indeed be of great help.

2 Related Work

For unsupervised object class discovery, there are several alternative approaches one can take (see Figure 1). Indeed, even within the same pipeline, one can select from several different component choices. Related work can thus be viewed from several angles: one immediate view would be based on the overall approach for the task; alternately, if one takes Approach II or III as discussed in the previous section, related work could then be discussed with respect to the choice of core learning method and the candidate window extraction method.

Related work in unsupervised object discovery

For recent unsupervised object learning methods, Tuytelaars et al. [50] give a comprehensive survey, albeit with a focus on probabilistic latent models. Earlier references on unsupervised object learning are mostly clustering-based approaches in which the concept of object is rather weak (Approach I). For example, [23] adopts the EM algorithm to cluster faces under translations and small variations. Although the unsupervised learning concept in [23] is insightful, it is unclear how to generalize methods like [23], [29] to deal with challenging real-world images [17]. Several unsupervised approaches have recently been proposed for object localization and categorization [23], [30], [32], [43], [50], [61]. Zhu et al. [61] learn a probabilistic grammar for object classes but report their results on a restricted subset of the Caltech dataset [19] — namely, images in which the foreground objects are mostly centered and often occupy a significant portion of the image. Lee and Grauman [30] group edge/contour fragments into objects without supervision, but require the objects to have well-defined strong shape cues. In contrast to work such as [24], [31] in which researchers use known categories as the context information, Deselaers et al. [11] encourage the new objects to fit the "meta information" learned on other objects.

The recent literature on cosegmentation [5], [26], [38], [42], [51] is also related to our method. However, most work on cosegmentation proceeds via "Approach II", in which the goal is not to automatically learn an object model for detection. These cosegmentation algorithms typically focus on large objects without significant scale differences and, moreover, are only applied to modest numbers of images, e.g. $2 \sim 40$ images. Recently, Vicente et al. [51] learn a category-independent pairwise regression model between two segmentation proposals extracted by [6]. Foreground regions containing multiple objects are represented as sparse subspace structure in [38]. Neither of these methods demonstrate object detection in unseen images due to lack of explicitly trained category model. In Section 6, we apply a scalable cosegmentation method [28] to multi-class object discovery but its results are not fully satisfactory.

Related work in multiple instance learning

There have also been previous attempts to use multiple instance learning (MIL) for unsupervised object discovery [57], [58]. However, these existing MIL approaches are mostly used as alternatives to the unsupervised clustering method in "Approach I". That is, [57], [58] provide image-level cluster labels but provide no localization of the specific objects. We focus on the scenario in which true positive objects of the same but unknown class exist within each bag, which separates our method from most of the existing clustering-based MIL approaches. In addition, one particular novel aspect of our paper is the use of an effective bottom-up process to convert unsupervised learning into multiple instance learning, which leads to significant performance gain. Next, we provide more background discussion.

In the machine learning literature, Dietterich et al. [12] introduced multiple instance learning (MIL) for drug activity prediction. Since then, researchers have proposed a large number of algorithms for tasks of MIL type. For example, Andrews et al. [2] developed mi-SVM and MI-SVM for instance-level and bag-level classification, respectively. There are also numerous computer vision applications that naturally fit into the MIL framework. Examples include object and face detection [3], [14], [54], visual categorization [52], and robust object tracking [4].

Multiple instance clustering (MIC) algorithms [57], [58] perform clustering in an MIL setting, and can also

be used to learn multiple object classes in unsupervised object discovery. Zhou et al. [58] view bags as atomic items with respect to which they define three types of inter-bag distance; they then apply K-means to cluster the bags in question. Zhang et al. [57] introduce the concept of maximum margin clustering (MMC) [55] into MIC, and then propose M^{3}IC. Because there are no "negative" images nor any specific prior information about the foreground objects in these formulations, they both reported discouraging results on challenging datasets like SIVAL [41]. Some other semi-supervised learning approaches have also appeared [60], but with an emphasis on clustering rather than object discovery. MIForests [33] also works on the multi-label case but it requires the image-level class labels to be given while in our case, these cluster labels are unknown since we are dealing with an unsupervised learning problem. Due to their assumption of the presence of one positive cluster within each positive bag, the previous multi-class/multi-label MIL methods [14], [33], [57], [58] do not directly apply to our case here. Our multiple class learning (MCL) algorithm is motivated by the multiple pose learning and multiple instance learning (MPL-MIL) idea [3], and can be viewed as a general and formal formulation to MPL-MIL. Here we explicitly study the hidden class variable and instance label, and provide a general learning strategy under an EM-like framework.

Existing work in saliency detection

The pipeline of "Approach III" utilizes windows extracted from each image. Our approach uses saliency detection for two purposes: (1) reducing the search space by extracting candidate windows of highest salience, and (2) differentiating object from background by creating negative bags of the least salient windows. Next, we discuss some recent work in saliency detection.

Impressive results have been reported using mostly data-driven bottom-up processes [8], [21], [25]. In addition to measuring the saliency of individual pixels [8], [25], Feng et al. propose and compute window saliency [21]. Chang et al. [7] utilize multiple images to perform co-saliency, but they primarily focus on single-class unsupervised cosegmentation rather than object localization or multiple object model learning . Despite notable interest in computer vision, saliency detection has received relatively less attention in the object discovery community. A recently proposed concept "objectness" [1], [16] is similar to the saliency concept, but more specific to objects.

Other related work

We view saliency as generic prior knowledge that may be of use in various high-level vision tasks. We adopt the bottom-up saliency process into an integrated learning framework for simultaneous object localization, object class discovery, and discriminative object model training, which differs from previous approaches [11], [32], [57].

Other references using bottom-up cues focus on multiple segmentations [43] or on self-paced discovery, which refers to progressive model accumulation [32]. In [44], the most "salient" regions are selected to update the models based on a fixed matching threshold. Moosmann et al. [37] redefine the concept of "saliency" in their supervised object classification scheme. However, their work differs from ours in problem setting, goal, and algorithm design. Recently, Deselaers et al. [11] attempt to use "meta information" to aid object detection in the weakly supervised setting. The "meta information" in question comes in the form of a classifier trained on several selected object classes. Our work differs from [11] because we perform simultaneous localization, clustering, and object detector training, whereas [11] train object detectors separate from their main formulation. Also, we extract least-salient regions in each image to compose negative bags, a particular advantage of our method over [11], [57], [58] that leads to significant performance gains. In addition, our work sheds light onto an emerging line of approaches [13], [34] which exploit visual concepts or object models from a large-scale of unlabeled/weakly-labeled image data.

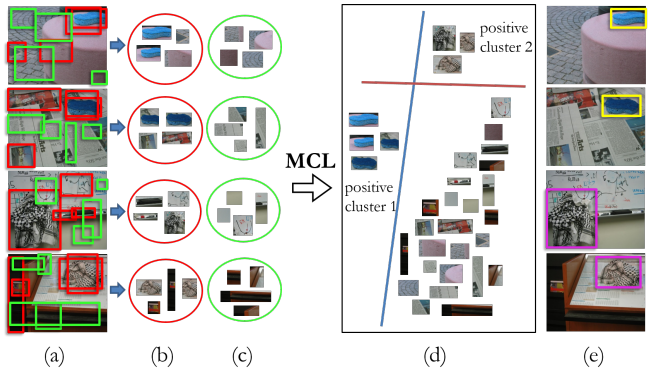


Fig. 2: The pipeline of the proposed bMCL algorithm: (a) saliency-scored windows, (b) high-salience "probably positive" bags (in which we expect the object to be present), (c) low-salience "probably negative" bags (in which we expect only background to be present), (d) Bottom-up Multiple Cluster Learning algorithm: different colors represent positive bags that belong to different classes, and (e) object clustering and detection results.

3 Bottom-up Multiple Class Learning

In this section we provide a brief overview of our framework, bottom-up Multiple Class Learning (bMCL).

As shown in Figure 2 and Algorithm 1, the pipeline of our method consists of three steps. First, we use saliency scores to construct "probably positive" and "probably negative" bags from input images, thereby converting our original unsupervised learning problem into a weakly supervised learning problem — namely, a multiple instance learning problem. We discuss the details of this saliency scoring step in Section 4. Second, we collect the S most salient windows from each image and derive initial class labels using K-means. We then formulate the problem as a weakly supervised multiple class learning task with two kinds of hidden labels: H_K containing unobserved bag-level cluster labels and

Algorithm 1 Bottom-up Multiple Class Learning (bMCL)

Input: N input images. K classes.
Output: K object models: $g^1, \ldots, g^K$. Predicted class labels for images: $\{\hat{k}_1, \ldots, \hat{k}_n\}$. Bounding boxes of detected objects $\{\hat{b}_1, \ldots, \hat{b}_n\}$.

Saliency-Guided Bag Construction

for $i = 1 \rightarrow N$ **do**
Extract the most salient windows to construct a positive bag $(x_i, y_i = 1)$.
Extract the least salient windows from random samples to construct a negative bag $(x_{i+N}, y_{i+N} = -1)$.
end for

Multiple Class learning

Initialization
Apply K-means to the S most salient windows of each image to obtain K centroids $\{c_1, \ldots, c_K\}$.
Compute evaluation score $q_{ij}^k = [1 + \exp(-\sigma||x_i - c_k||^2)]^{-1}$.
Initialize class labels $H_K^0 \propto \Pr(y_i = 1, k_i = k|x_i, \theta)$.
MCL Algorithm
Assign $X = \{x_1, ..., x_N, ..., x_{2N}\}$, $Y = \{y_1, ..., y_N, ..., y_{2N}\}$.
Call Algorithm 2: $[g^1, \ldots, g^K] = \text{MCL}(X, Y, K, H_K^0)$

Object Clustering and Detection

for $i = 1 \rightarrow N$ **do**
Compute $\hat{k}_i = \arg\max_k \Pr(k_i = k|y_i = 1, x_i; \theta_r) \propto \Pr(y_i = 1, k_i = k|x_i; \theta_r)$, where $\theta_r = \{g^1, \ldots, g^K\}$.
Detect $\hat{b}_i = x_{i\hat{j}}$, where $\hat{j} = \arg\max_j \Pr(k_{ij} = \hat{k}_i|x_{ij}; g_{\hat{k}_i})$.
end for

(a) (b)

Fig. 3: (a) Localized objects from SIVAL [41]. (b) Original images from SIVAL [41].

H_Y as unobserved instance-level cluster labels. We then solve the problem using an EM-like optimization approach that we refer to as Discriminative EM (DiscEM). We discuss the details of our formulation in Section 5. The third step is to use the K learned object models to perform object detection and to assign class labels. Note that because our framework learns multiple discriminative object models, it is natural to apply them to detect objects in novel images. Please see Section 6 for detailed discussions of our experiments.

4 Saliency Guidance

In this section, we demonstrate that the problem of unsupervised object discovery is, in general, ambiguous. Utilizing bottom-up saliency detection helps to guide the learning process by turning unsupervised learning into weakly supervised learning.

4.1 General unsupervised object discovery is ambiguous

In an empirical study, we asked ten human participants to divide two groups of images from the SIVAL dataset [41] into three categories. While all the participants divided the first group of object-centered images (Figure 3a) into three object classes spontaneously, they felt confused and spent much more time on the second group of images in which the object was not emphasized by centering (Figure 3b). In addition, while seven of the participants divided the non-centered group of images into object classes (apples, toys, and books), the three remaining participants categorized the images by scene type (indicating that their attention was on the background of chair, newspaper, or room). That even human performance does not always immediately focus on the object in the images indicates the strong ambiguity in unsupervised object discovery, especially when localizing objects in complex backgrounds. Direct clustering based algorithms [15], [55], [57], [58] may fail to separate the objects from the background clutter.

4.2 Window-based saliency detection

Saliency detection, usually considered as a bottom-up process, can guide the object discovery task because objects of interest often appear to be salient in many real-world images. Feng et al. [21] show that the windows with the highest saliency scores have a large overlap with the windows that contain objects in popular datasets such as the PASCAL dataset [17]. This observation also holds for images retrieved from Internet image search engines. In the light of the previous observation, Feng et al. propose window composition [21] to measure how likely it is that a given image window contains a salient object. This window composition method computes saliency scores for windows of different scales and at different locations. Although complex backgrounds sometimes lead to high saliency scores even for background windows, essentially almost every object in the SIVAL dataset is covered by some window with a high saliency score; cf. Section 4.3 and Figure 4.

4.3 From unsupervised object discovery to weakly supervised learning

To validate the object saliency property in the SIVAL dataset [41], we conducted an experiment and found that the 70 most salient windows extracted by [21] cover 98% of objects. This allows us to define positive and negative bags based on window saliency scores; these bags can then be used in a multiple instance learning formulation. Specifically, for each image we define a "probably positive" bag (in which we expect the object to be present) consisting of the most salient windows and a "probably negative" bag (in which we expect only background to be present) consisting of the least salient windows from a large set of randomly sampled windows, as illustrated in Figure 4. In this way, we convert unsupervised object

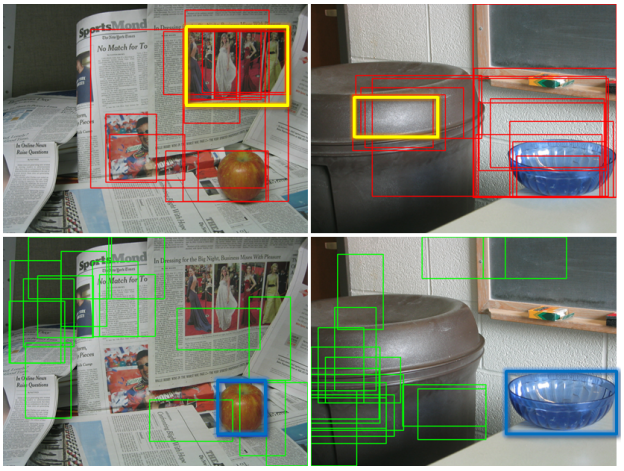


Fig. 4: Example of bags and instances. On the first row, red rectangles: the most salient windows as instances in the positive bag; yellow rectangles: the most salient window obtained by [21]. On the second row, green rectangles: the least salient windows from a large set of randomly sampled windows as instances in the negative bag; blue rectangles: the desired object window.

discovery into a weakly supervised learning problem. Note that using the least salient windows to construct the negative bags is a particularly interesting step that has never appeared in existing methods.

5 Formulation

Standard MIL solutions [2], [54] cannot be directly applied to unsupervised object discovery due to the lack of the concept of multiple classes. Recent multi-class multi-instance methods like MIForests [33] need image-level class labels for training, which also does not fit our setting. While multiple instance clustering (MIC) approaches [57], [58] are designed to explore hidden patterns in multiple classes, their performance is unsatisfactory because they treat every image as a positive bag without incorporating any notion of negative bags. In the present paper we propose a model named Multiple Class Learning (MCL) that tackles multiple classes among positive bags, explores unknown class labels, learns instance labels, and utilizes negative bags. The learning process is performed by a process we refer to as Discriminative EM (DiscEM), in which MIL-Boost is a component for learning instance-level labels.

In what follows, we first review the MIL-Boost algorithm, then introduce our formulation for Multiple Class Learning (MCL) problem and a derivation of our optimization algorithm for the learning process.

5.1 MIL-Boost

Multiple instance learning (MIL) is a major topic in weakly supervised learning. Here we give a brief overview with a focus on boosting-based MIL approaches [3], [54]. In MIL, each bag $x_i \in \mathcal{X}^{n_i}$ consists of a set of instances $\{x_{i1}, \ldots, x_{in_i}\}(x_{ij} \in \mathcal{X})$. Each bag x_i has a given class label $y_i \in \mathcal{Y} = \{-1, 1\}$, and instance labels $y_{ij} \in \mathcal{Y}$ are unknown and treated as hidden variables. A bag is regarded as positive if at least one of its instances is positive and a bag is regarded as negative when all of its instances are negative, *i.e.* $y_i = \max_j (y_{ij})$. For notational simplicity, we assume that each bag contains the same number of instances, *i.e.* $n_i = m$ $(i = 1, \ldots, n)$.

Standard boosting methods [22], [36] assume an additive model of instance-level decisions: $g_{ij} = g(x_{ij})$ where $g(x_{ij}) = \sum_t \lambda_t g_t(x_{ij})$ is a weighted vote of weak classifiers $g_t : \mathcal{X} \to \mathcal{Y}$. Assuming that $y_{ij} \in \mathcal{Y}$ is the hidden instance-level label, the associated probability of being positive is given by

$$p_{ij} = \Pr(y_{ij} = 1 | x_{ij}; g) = \frac{1}{1 + \exp(-g_{ij})}. \quad (1)$$

The bag-level probability is computed via a Noisy-OR (NOR) model, which gives

$$p_i = \Pr(y_i = 1 | x_i; g) = 1 - \prod_{j=1}^{m} (1 - p_{ij}). \quad (2)$$

Because the bag labels are given in the training set, we can optimize the negative log-likelihood function:

$$\mathcal{L}_{\text{MIL}} = -\sum_{i=1}^{n} [\mathbf{1}(y_i = 1) \log p_i + \mathbf{1}(y_i = -1) \log (1 - p_i)],$$

where $\mathbf{1}(\cdot)$ is an indicator function. The algorithm greedily searches for g^t over a weak classifier candidate pool, followed by a line search for λ_t. According to the AnyBoost [36] framework, the weight w_{ij} on each instance x_{ij} is updated as

$$w_{ij} = -\frac{\partial \mathcal{L}_{\text{MIL}}}{\partial g_{ij}} = \begin{cases} -\dfrac{1}{1 - p_{ij}} \dfrac{\partial p_{ij}}{\partial g_{ij}} & \text{if } y_i = -1; \\ \dfrac{1 - p_i}{p_i (1 - p_{ij})} \dfrac{\partial p_{ij}}{\partial g_{ij}} & \text{if } y_i = 1. \end{cases} \quad (3)$$

5.2 Multiple Class Learning

We now introduce our formulation, multiple class learning (MCL), for the task of learning multiple discriminative models with weak labels and hidden variables. The overall formulation of MCL tries to (1) discriminate the positive instances (in which an object is present) from the negative instances (in which only background is present); (2) learn the differences between different object classes in the positive bags.

Given K object classes and N unlabeled images, we obtain N positive bags and N negative bags based on bottom-up saliency detection. We denote the total number of all the bags by $n = 2N$. There are two kinds of hidden variables in MCL: 1) the instance-level label $y_{ij} \in \{-1, 1\}$ for each instance x_{ij} in bag x_i and 2) the class latent label $k_{ij} \in \mathcal{K} = \{0, 1, \ldots, K\}$ for the instance x_{ij} that belongs to the k^{th} class. Note that we use $k_{ij} = 0$ and $k_i = 0$ to represent a negative instance and a negative bag, respectively. Here, we assume the existence of only one foreground object class in each positive bag; that is, we regard each image as containing only one class of object. Thus, the class label k_i for each

positive bag of class k is defined based on the class labels of its instances as

$$k_i = k \iff \forall j,\ k_{ij} \in \{0, k\} \text{ and } \exists j_0, k_{ij_0} = k, \quad (4)$$

where $k \in \{1, ..., K\}$. Throughout the paper, we denote $H = (H_K, H_Y)$ as hidden variables where $H_K = \{k_i \mid i = 1, .., n\}$ and $H_Y = \{y_{ij} \mid i = 1, .., n, j = 1, .., m\}$. Note that we purposely define $y_{ij} \in \{-1, 1\}$ so that $k_{ij} = \frac{(y_{ij}+1)}{2} \cdot k_i$.

For bags $X = \{x_1, \ldots, x_n\}$ with labels $Y = \{y_1, \ldots, y_n\}$ ($y_i \in \{-1, 1\}$), we define the overall negative log-likelihood function $\mathcal{L}(\theta; Y, X)$ as

$$\begin{aligned}\mathcal{L}(\theta; Y, X) &= -\log \Pr(Y|X;\theta) = -\log \sum_{H_K} \Pr(Y, H_K|X;\theta) \\ &= -\log \sum_{H_K} \sum_{H_Y} \Pr(Y, H|X;\theta), \end{aligned} \quad (5)$$

where the model parameter $\theta = \{g^1, \ldots, g^k, \ldots, g^K\}$ and g^k is the appearance model for the k^{th} object class. The evaluation score for x_{ij} to the k-th class is computed as $q_{ij}^k \equiv q^k(x_{ij}) = [1 + \exp(-g_{ij}^k)]^{-1}$ where $g_{ij}^k \equiv g^k(x_{ij})$. The instance-level probability is thus

$$p_{ij}^k = \Pr(k_{ij} = k|x_{ij};\theta) \propto \prod_{t=1}^{K} (q_{ij}^t)^{\mathbf{1}(t=k)} (1-q_{ij}^t)^{\mathbf{1}(t\neq k)}. \quad (6)$$

Next, we derive the probability $\Pr(Y, H_K|X;\theta)$. Assuming that all bags are conditionally independent, we have

$$\Pr(Y, H_K|X;\theta) = \prod_{i=1}^{n} \Pr(y_i, k_i|x_i;\theta) = \prod_{i=1}^{n} [\Pr(k_i|x_i;\theta) \cdot s_i], \quad (7)$$

where $s_i = \mathbf{1}((y_i = -1 \wedge k_i = 0) \vee (y_i = 1 \wedge k_i \neq 0))$.

For each positive or negative bag, we denote the probability $p_i^k = Pr(k_i = k|x_i;\theta)$. Because the full derivation is combinatorial, we approximate the probability as

$$p_i^k \approx \prod_{t=1}^{K} \left[(q_i^t)^{\mathbf{1}(t=k)} (1 - q_i^t)^{\mathbf{1}(t\neq k)}\right] \quad (8)$$

where $q_i^t = \Pr(\exists j, k_{ij} = t|x_i;\theta) = 1 - \prod_{j=1}^{m} (1 - p_{ij}^t)$ denotes the measure for at least one instance x_{ij} in bag x_i belonging to the t^{th} class. Details of the above approximation are discussed in Section 7.

Then $\Pr(Y, H_K|X;\theta)$ can be written in a class-wise manner as

$$\Pr(Y, H_K|X;\theta) \propto \prod_{t=1}^{K} \prod_{i=1}^{n} \left[(q_i^t)^{\mathbf{1}(t=k_i)} (1 - q_i^t)^{\mathbf{1}(t\neq k_i)} \cdot s_i\right]. \quad (9)$$

We could further explicitly use the instance-level hidden variables H_Y to expand $\Pr(Y, H|X;\theta)$. Similar to the overall loss function $\mathcal{L}(\theta; Y, X)$, we also define the bag-level loss function $\mathcal{L}(\theta; Y, X, H_K) = -\log \Pr(Y, H_K|X;\theta)$ and the instance-level loss function $\mathcal{L}(\theta; Y, X, H) = -\log \Pr(Y, H|X;\theta)$, which will be later used in our Discriminative EM (DiscEM) algorithm (See Section 5.3).

In MCL, if the expectation of $H = \{H_K, H_Y\}$ is estimated, we could subsequently decompose the minimization of the overall loss function $\frac{d}{d\theta}\mathcal{L}(\theta; Y, X)$ into $\frac{d}{d\theta}\mathcal{L}(\theta; Y, X, H)$ and optimize K standard boosting additive models on instance-level decisions: $g_{ij}^k = g^k(x_{ij})$, where $g^k(x_{ij}) = \sum_t \lambda_t g_t^k(x_{ij})$ is a weighted vote of weak classifiers $g_t^k : \mathcal{X} \to \mathcal{Y}$. To this end, in the following subsection we derive an EM-style optimization method to estimate the collected hidden variables H.

5.3 Optimization

The optimization of Eqn. (5) involves the collected hidden variables H. To solve this problem, we employ a general formulation of Discriminative EM (DiscEM) that performs discriminative learning in the presence of hidden variables. We directly apply DiscEM to explore the hidden variables H in MCL. We also observe that under the MIL assumption, MIL-Boost [54] is equivalent to DiscEM in tackling instance-level hidden labels, as shown in Section 7. Based on this observation, a standard MIL-Boost approach is naturally able to handle the instance-level hidden variables H_Y and we only need to tackle the class labels H_K explicitly. Further, in contrast to other multi-class MIL formulations like MIForests [33], DiscEM can be applied to other situations with various forms of hidden variables.

Our DiscEM approach is similar in spirit to standard EM [10]. The primary difference is that in our model, labels $Y = \{y_1, \ldots, y_n\}$ are given in addition to observations $X = \{x_1, \ldots, x_n\}$, and the goal is to estimate the parameter θ that minimizes the negative log-likelihood function $\mathcal{L}(\theta; Y, X)$.

We iteratively update an initial estimate θ_0 with successively better estimates $\theta_1, \theta_2, \ldots,$ until convergence. The update from θ_r to θ_{r+1} consists of two steps:

E step: Compute $\Pr(H|Y, X;\theta)$ via previous estimate θ_r.
M step: Update θ_{r+1} by minimizing $\mathcal{L}(\theta; Y, X)$.

Note that in the above formulation, the parameter θ can be purely discriminative, *i.e.*, it can be a parameter of classifiers. In this way, DiscEM takes advantage of discriminative learning algorithms. This distinguishes DiscEM from other conditional EM frameworks [45], in which the task is to learn a generative parameter through a discriminative objective. Compared with standard supervised algorithms, DiscEM is thus better able to handle hidden variables and to embrace the weakly supervised learning setting.

DiscEM is particularly suitable for MCL because MCL forms an optimization problem with a discriminative cost function $\mathcal{L}(\theta; Y, X)$ and complex hidden variables $H = (H_K, H_Y)$ in Eqn. (5), which makes the other MIL approaches not directly applicable. Because of the equivalence between DiscEM and MIL-Boost in dealing with instance-level hidden labels (See section 7 for details), we could further replace the **EM** steps for the instance labels H_Y with standard MIL-Boost [54], or in other words, it is only necessary to integrate H_K out.

Algorithm 2 Multiple Class Learning (MCL)

Input: Bags $\{x_1, \ldots, x_n\}, \{y_1, \ldots, y_n\}$. Number of classes K. Initial labels H_K^0.
Output: K discriminative classifiers: $g^1, \ldots, g^K$.

$r \leftarrow 0$
repeat
  $r \leftarrow r + 1$
  for $k = 1 \rightarrow K$ **do** {**M** Step}
    Given class variables H_K^{r-1}, group terms $\mathcal{L}^k(g_r^k; Y, X, H_K^{r-1})$ by class indices.
    Train a strong MIL classifier g_r^k to minimize $\mathcal{L}^k(g_r^k; Y, X, H_K^{r-1})$ via MIL-Boost. T is the number of weak classifiers in MIL-Boost.
  end for
  for $i = 1 \rightarrow n$ **do** {**E** Step}
    Compute $\Pr(y_i = 1, k_i = k | x_i; \theta_r)$ using the estimated model $\theta_r = \{g_r^1, \ldots, g_r^K\}$. Sample k_i via $\Pr(k_i = k | y_i = 1, x_i; \theta_r) \propto \Pr(y_i = 1, k_i = k | x_i; \theta_r)$.
  end for
until $H_K^r = H_K^{r-1}$

We rewrite $\frac{d}{d\theta}\mathcal{L}(\theta; Y, X)$ as

$$\frac{d}{d\theta}\mathcal{L}(\theta; Y, X) = \mathbb{E}_{H_K \sim \Pr(H_K | Y, X, \theta)}\left[\frac{d}{d\theta}\mathcal{L}(\theta; Y, X, H_K)\right]. \tag{10}$$

The loss function can be decomposed in a class-wise manner, *i.e.*, $\mathcal{L}(\theta; Y, X, H_K) = \sum_{k=1}^{K} \mathcal{L}^k(g^k; Y, X, H_K)$. Using Eqn. (9), $\mathcal{L}^k(g^k; Y, X, H_K)$ can be computed as

$$\begin{aligned}\mathcal{L}^k(g^k; Y, X, H_K) = -\sum_{i=1}^{n} [&\mathbf{1}(k = k_i) \log q_i^k \\ &+ \mathbf{1}(k \neq k_i) \log (1 - q_i^k)],\end{aligned} \tag{11}$$

which is valid when all the (y_i, k_i) in (Y, H_k) satisfy the condition $s_i = \mathbf{1}[(y_i = -1 \wedge k_i = 0) \vee (y_i = 1 \wedge k_i \neq 0)]$, as shown in Eqn. (9). Note that the normalization term in Eqn. (9) is ignored here for computational simplicity as it is close to 1.

Eqn. (11) essentially builds K classifiers, among which each classifier g^k takes bags with class label k as positive ones and all the rest as negative ones, and minimizes $\mathcal{L}^k(g^k; Y, X, H_K)$ separately.

For each $\mathcal{L}^k(g^k; Y, X, H_K)$, hidden instance variables H_Y could be further integrated out as

$$\begin{aligned}&\frac{d}{d\theta}\mathcal{L}^k(g^k; Y, X, H_K) = \\ &\mathbb{E}_{H_Y \sim \Pr(H_Y | Y, H_K, X; \theta)}\left[\frac{d}{d\theta}\mathcal{L}^k(g^k; Y, X, H)\right].\end{aligned} \tag{12}$$

Rather than integrating H_Y out, we use standard MIL-Boost [54] to minimize the function based on the equivalence between DiscEM and MIL-Boost for the instance-level labels (section 7). Algorithm 2 summarizes the DiscEM optimization. To initialize H_K^0 in Algorithm 2, we perform K-means on top S salient windows. Details of K-means initialization could be found in Algorithm 1.

6 Experiments

Datasets: Our goal is to build effective systems that can perform unsupervised object discovery in practice. To compare our algorithm with previous approaches, we use a number of challenging benchmarks from machine learning and computer vision, briefly described below.

The SIVAL dataset [41] is frequently used in MIL, semi-supervised learning, and image retrieval. It is a difficult dataset because the scenes are highly diverse and often complex; moreover the objects may occur anywhere spatially in the image and may also be photographed with different orientations. We follow the same setting as [57], [58] and randomly partition the twenty-five object classes into five groups, named `SIVAL1` to `SIVAL5`.

The CMU-Cornell iCoseg dataset [5] is designed for cosegmentation with 38 object classes. We construct a subset, named `CC`, containing five classes with certain similarities in object appearances and backgrounds: helicopter, kite, hot balloon, and two kinds of planes.

The 3D object category dataset [46] contains ten object classes, where each class contains ten different object instances imaged under different viewpoints and distances. We randomly select one object instance from each class and partition the ten set of images of selected instances into two datasets, named `3D1` and `3D2`. To increase the difficulty, only images of the smallest object scale are included.

Parameters and features: In this paper, each positive bag contains the 70 most salient windows returned by [21], and each negative bag contains the 40 least salient windows from a large set of randomly sampled windows. Note that our algorithm is not sensitive to the numbers of windows in the bags. The other parameters are fixed at $K = 5, S = 3, \sigma = 0.1$. We represent our appearance model as a Boosting classifier [36] trained on Color Moment [49], Edge Histogram, and GIST [39] extracted from image windows. For MIL-Boost, a decision stump is used as a weak classifier and we set the number of weak classifiers to be 100 throughout our experiment.

Measures and metrics: We adopt following metrics for a fair comparison of all the methods.

Purity has been widely used in previous clustering and unsupervised object discovery work [30], [55], [59] as an evaluation metric that measures the extent to which a cluster contains images of a single class. Specifically, let $\Omega = \{\omega_1, \ldots, \omega_K\}$ be the set of K discovered clusters, and $C = \{c_1, \ldots, c_K\}$ be the set of ground truth classes. Purity is then computed as $P = \frac{1}{N}\sum_{i=1}^{K} \max_j |\omega_i \cap c_j|$, where N is the number of images.

Clustering accuracy has been used in previous multiple instance clustering methods [57], [58] to evaluate clustering algorithms. Specifically, we first take a set of labeled bags, remove the labels of these bags and run the clustering algorithms, then we relabel these bags using the clustering assignment returned by algorithms. Finally, the accuracy is measured as $\text{Acc} = \frac{1}{N}\sum_{i=1}^{K} |\omega_i \cap$

	bMCL	SD	M^{3}IC	BAMIC	UnSL	MFC
SIVAL1	**95.3**	80.4	39.3	38.0	27.0	45.0
SIVAL2	**84.0**	71.7	40.0	33.3	35.3	33.3
SIVAL3	**74.7**	62.7	37.3	38.7	26.7	41.3
SIVAL4	**94.0**	86.0	33.0	37.7	27.3	53.0
SIVAL5	**75.3**	70.3	35.3	37.7	25.0	48.3
CC	**80.0**	73.9	46.1	47.8	60.0	50.4
3D1	**81.1**	64.0	46.9	43.2	37.3	51.4
3D2	**85.6**	82.9	52.3	51.4	37.5	48.7

(a) measured in terms of purity

	bMCL	SD	M^{3}IC	BAMIC	UnSL	MFC
SIVAL1	**95.3**	78.7	39.3	37.7	25.3	45.0
SIVAL2	**84.0**	65.7	38.7	33.3	34.0	33.3
SIVAL3	**74.7**	62.7	37.0	38.7	26.0	39.0
SIVAL4	**94.0**	86.0	33.0	37.7	26.3	53.0
SIVAL5	**75.3**	70.3	35.3	36.7	23.3	48.3
CC	**73.9**	63.5	38.2	46.1	53.3	42.6
3D1	**81.1**	64.0	46.0	43.2	34.7	51.4
3D2	**78.4**	76.6	52.3	51.4	35.0	48.7

(b) measured in terms of clustering accuracy

	bMCL	SD	M^{3}IC	BAMIC	UnSL	MFC
SIVAL1	**89.9**	72.7	11.4	12.4	10.8	19.2
SIVAL2	**73.2**	57.3	10.1	5.8	19.1	7.3
SIVAL3	**64.9**	42.4	8.7	11.3	6.1	17.0
SIVAL4	**87.2**	75.4	7.4	13.3	10.6	26.0
SIVAL5	**61.4**	52.3	8.3	9.1	11.1	17.2
CC	**77.3**	59.7	15.8	23.0	59.7	32.3
3D1	**69.7**	52.3	20.3	15.4	23.6	32.9
3D2	**87.9**	75.8	22.4	25.9	29.4	29.6

(c) measured in terms of NMI

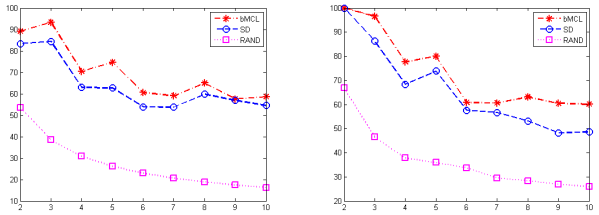
TABLE 1: Object categorization performance is measured in terms of (a) purity, (b) clustering accuracy, and (c) NMI. We compare bMCL with recent MIC approaches (M^{3}IC [56], BAMIC [58]), a state-of-the-art unsupervised discovery method (UnSL [27]), the multiple foreground cosegmentation algorithm (MFC [28]), and the saliency detection baseline (SD).

$map(c_i)|$, where map$(\cdot)$ is the function that maps each cluster to a class, given by the Hungarian algorithm, and $\mathbf{1}(\cdot)$ is an indicator function. We note that Acc considers the one-to-one relationship between clusters.

Normalized Mutual Information (NMI) is a symmetric metric to quantify statistical information shared between two distributions [48]. It has been previously used in [56], [58] to evaluate the clustering performance of multiple instance clustering methods. To calculate NMI, we use $\text{NMI}(\Omega, C) = \frac{I(\Omega,C)}{[H(\Omega)+H(C)]/2}$ where I and H refer to mutual information and entropy.

6.1 Simultaneous Categorization and Localization

We demonstrate bMCL's superior performance relative to two recent multiple instance clustering (MIC) approaches BAMIC [58] (with the best distance metric) and M^{3}IC [56], one state-of-the-art unsupervised object discovery approach [27] (UnSL), that achieves top performance (about 98% measured in purity) on a subset of Caltech-101 [19], and one foreground cosegmentation



(a) experiment results of SIVAL (b) experiment results of CC

Fig. 5: Object categorization results with varying number of clusters K are measured by purity. We compare bMCL with saliency detection baseline (SD) and random guess (RAND).

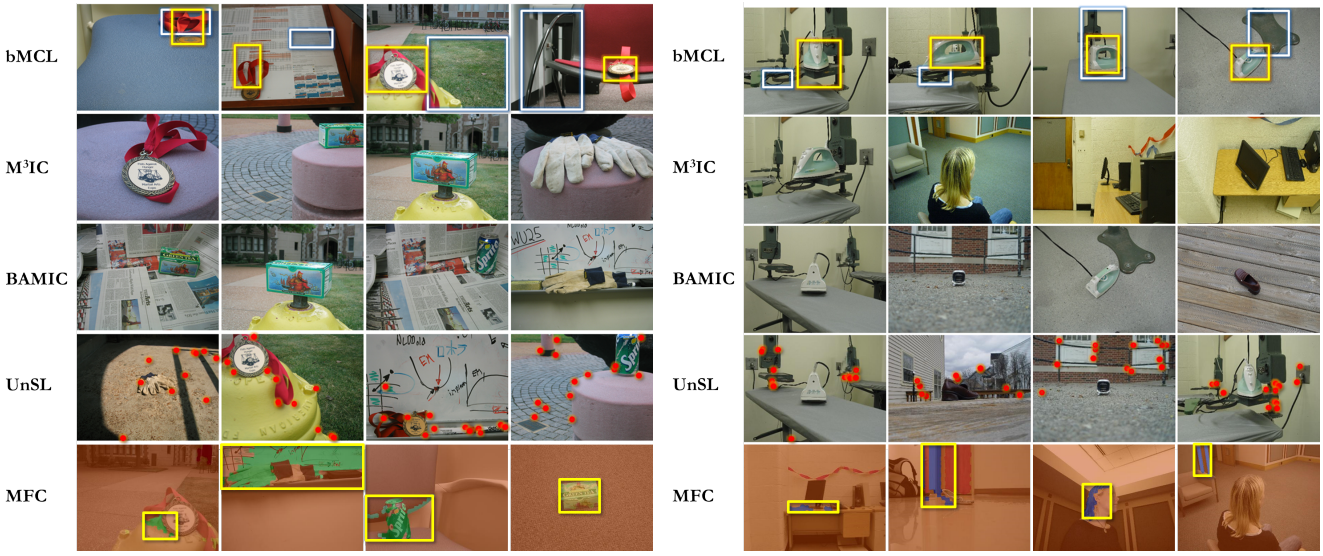
method [28] (MFC). We use their implementations and the same parameter settings used in the original work. The same feature space for bMCL is provided to BAMIC and to M^{3}IC. Note that for MFC [28], we assign each image the class label of segments whose area is the largest in the image.

There has been little work on exploiting saliency for the task, except [44]. We implement a clustering algorithm by selecting the most "salient" window obtained by [21] in each image and cluster those windows by K-means. This algorithm serves as a more reasonable saliency detection baseline (SD) than the straightforward greedy method in [44]. The SD algorithm is different from the initialization used in bMCL because it considers only one salient window for each image, and gives hard assignments of labels to windows without sampling.

In bMCL, we use learned object detectors to evaluate the densely sampled (multi-scale, multi-size) image windows and output the class label k_i and the instance (window) x_{ij} with the highest probability p_{ij}^k for each bag (image) x_i.

As stated above, purity, clustering accuracy and NMI are used as the evaluation metrics for the categorization problem. Table 1 reports the average results from ten runs for each method. We see that using SD only already provides a significant performance increase because the saliency information helps to distinguish foreground objects from background clutter. Further, bMCL outperforms all the other methods under all criteria by a large margin $(50\% \sim 200\%)$.

The performance gap can be well explained by the illustrative results shown in Figure 6. Without using negative bags, the MIC approaches (BAMIC [58] and M^{3}IC [56]) cannot explicitly differentiate objects from background or distinguish between similar backgrounds, nor can they perform object localization. The keypoint-based UnSL [27] approach lacks a spatial constraint on key points, which causes the found object key points to be scattered over the entire image. The cosegmentation method MFC [28], while taking the foreground knowledge into account, cannot effectively maximize the distance between multiple clusters, nor can it perform object detection on unseen images. In contrast, bMCL finds object classes (top-down models) across different images in noisy input from bottom-up



(a) experiment results of SIVAL2 (b) experiment results of 3D1

Fig. 6: Illustrative categorization results of four methods in the experiments, the left from SIVAL2 [41] and the right from 3D1 [46]. From top to bottom: bMCL, M^{3}IC [56], BAMIC [58] and UnSL [27]. In bMCL, the yellow rectangle is the localized object and the white rectangle is the most salient window given by [21]. In UnSL, the learned object key points are overlayed (red points). See Section 6.1 for detailed discussion.

methods. Note that as shown in the first row in Figure 6, merely using saliency detection is not enough to get convincing levels of performance, which demonstrates the power and necessity of the overall bMCL framework.

Number of clusters: We further perform experiments with different number of clusters ($K = 2 \sim 10$) on SIVAL and CC datasets. We measure results by purity; for comparison, we also run $500,000$ Monte Carlo experiments to estimate the clustering purity of randomly assigned class labels (RAND). Figure 5 demonstrates that bMCL repeatedly outperforms the saliency detection baseline (SD) for each K; moreover, as K increases, the performance of bMCL degrades more slowly than the performance of RAND. Also, please note that even with $K = 10$ (58.7%), the performance of bMCL is still higher than that of competing systems with $K = 5$, among which the highest value is 53.0% from MFC.

Running time: After saliency detection and feature extraction, our MCL algorithm takes less than half an hour to discover five object classes in 300 images, on a PC with Intel Core 2 Duo P7450 @ 2.13GHz.

6.2 Object Detection

Previous unsupervised object discovery methods do not obtain discriminative object models in an integrated manner. Some are restricted to object categorization and therefore cannot perform object localization [27], [61]; some have to resort to a separate detector training process with localization results [32]; others obtain specialized detectors such as Chamfer-distance-based shape templates [30]. By contrast, bMCL integrates the detector training into the framework for generic object classes.

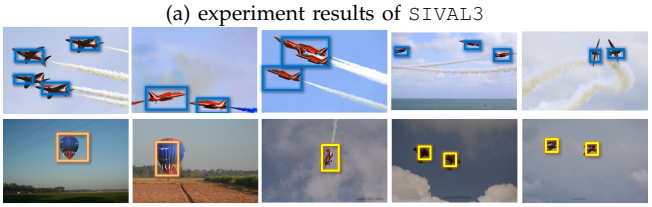
To validate the generalization ability of detectors learned by bMCL, we randomly pick five images from each object class, train bMCL models using the remaining images, and detect the objects in the selected images under the same conditions as previously described. We further use non-maximum suppression to eliminate overlapping detections for each object. An object is considered to be correctly detected if we have both a correctly reported class label and an intersection between the algorithm-derived bounding box and the ground-truth area that is larger than 50% of the union. The detection accuracies for SIVAL (averaged over five SIVAL datasets), CC and 3D (averaged over 3D1 and 3D2) are 74.4%, 72.0% and 76.0%, respectively. Note that as shown in Table 1, the average clustering purities on the three datasets are 84.7%, 80.0% and 83.4%, respectively. The detection accuracies are satisfactory because the detectors are trained on a smaller training set. Figure 7 shows some detection results. Notice that in each image, bMCL can detect multiple objects of the same class, *e.g.* planes, as shown in Figure 7b.

6.3 Weakly Supervised Single Class Recognition

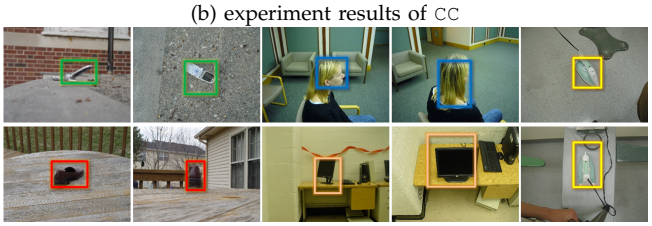
Previous work [9], [11], [40], [43] addresses the problem of localizing objects of a single class and learning a corresponding detector. Because bMCL performs multi-class object discovery in an integrated framework, our framework can naturally handle single class recognition by setting $K = 1$. Also, there are no hidden class labels in this single class recognition task, and therefore we employ MIForests [33], in addition to MIL-Boost, as the discriminative classifier in bMCL.

We follow the same experimental setting as [11] and directly cite the results reported in [9], [11], [43] and [40]. Please refer to [11] for datasets, features, measurements, and other specific aspects of the experiment. We also tested the performance of one state-of-the-art cosegmentation method [26]. For [26], we set the number of classes $K = 4$ for each image category, and the

(a) experiment results of SIVAL3



(b) experiment results of CC



(c) experiment results of 3D1

Fig. 7: Object detection results for novel images: the top from SIVAL3 [41], the middle from CC [5], and the lower from 3D1 [46]. The rectangles are the localization results given by bMCL. Different colors represent the class labels returned by the algorithm.

	bMCL	[33]	[11]	[9]	[43]	[40]	[26]
PASCAL 06	45	36	**49**	34	27	N/A	43
PASCAL 07	**31**	25	28	19	14	30	30

TABLE 2: Results for the single class recognition experiment. We compare our approach bMCL to MIForests [33], previous weakly supervised learning methods [9], [11], [40], [43] and one cosegmentation approach [26], measured in CorLoc [11]. To be consistent, the datasets used are identical to those in [11], which, although referred to by the names PASCAL 06 and PASCAL 07, are actually subsets of the classes in the PASCAL VOC datasets. For details, please refer to [11].

segments whose bounding boxes have the highest score in CorLoc are considered to be detected objects. Pandey and Lazebnik [40] applied the deformable part-based models (DPM's) with latent SVM training [20] to weakly supervised single class learning and recognition. They reported their results only on PASCAL VOC 2007 [17].

Table 2 shows that bMCL outperforms [9], [26], [40], [43] and is comparable with [11] on the challenging PASCAL datasets [17], [18]. MIForests [33] performs slightly worse than bMCL as it may require some special tuning to produce good results. Note that the method in [11] trains varying meta-information classifiers for different datasets whereas bMCL adopts bottom-up saliency detection to discover multi-class objects; this is more general, more efficient, and more convenient in practice. In [40], the method begins by learning root filter weights from the features of the entire training images. Our notion of saliency guidance is complementary to DPM because, when comparing with the entire image, we find it more reasonable to use salient windows as the initialization for LSVM training [20]. Figure 8 illustrates exemplar object localization results on PASCAL 07.

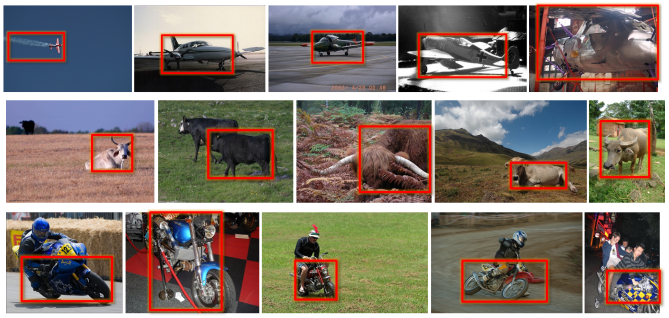


Fig. 8: Red rectangles: bMCL object localization results with a single object class (from top to bottom: aeroplane, cow, and motorbike) on the challenging PASCAL VOC 07.

	bMCL	SD	M^3IC	BAMIC	UnSL	MFC
Purity	**96.25**	88.75	66.25	75.00	56.25	54.43

TABLE 3: Clustering results of images returned by image search engines.

	apple	bean	bolt	bow	football
Purity	95.00	88.33	85.00	85.00	91.67

TABLE 4: Clustering results of Internet images associated with double meaning queries.

6.4 Learning Object Classes from Internet Images

6.4.1 *Clustering Internet images*

To further demonstrate the effectiveness of bMCL on images with significant variability, we apply bMCL on Internet images retrieved from Google and Bing image search engines. We crawl 40 images from image search engines for each of the keywords "monkey" and "train". The images retrieved are highly diverse, differing in illuminations, poses, backgrounds, and types (photograph, line drawing, clip art, etc.). We then test all clustering methods described in Section 6.1 under the same setting. We can see from Table 3 that, again, bMCL consistently outperforms all other methods by a large margin. This proves that even when input images are complex and somewhat noisy, bMCL still have good performance.

6.4.2 *Finding Visual Subcategories*

Here we further evaluate our method on the task of clustering Internet images associated with keywords with double meanings. We use five queries: apple (brand vs. fruit), bean (vegetable vs. actor), bolt (movie vs. athlete), bow (weapon vs. bow tie), and football (American football vs. soccer). We crawl 30 images for each meaning, resulting in 60 images per query. We then test whether bMCL can cluster these 60 images into two clusters with one cluster per meaning. As shown in Table 4 and Figure 9, bMCL performs consistently well in these highly diverse cases and effectively distinguishes different visual subcategories of images within each category.

Fig. 9: Illustrative clustering and localization results on Internet images with keywords "bean" and "bow"

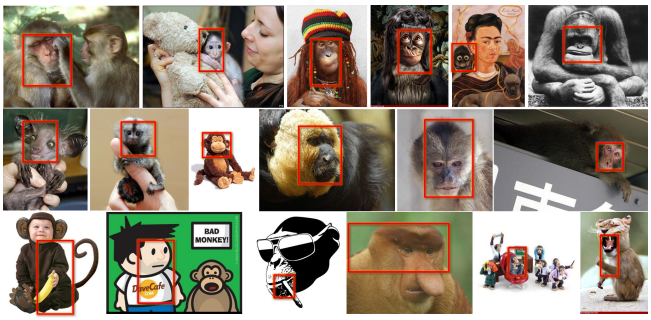


Fig. 10: Object detection results for novel images returned by image search engines. The first two rows illustrate exemplar successful detection results and the last row illustrates exemplar failure cases.

6.4.3 *Learning an object model under weak supervision*

We also apply bMCL in learning a generic single class concept under weak supervision. To do this, we crawl 500 images returned by image search engines with keyword "monkey". We use 450 images for training and 50 images for evaluation. With the same setting described in Section 6.3, bMCL achieves a detection rate of 37.4%. As shown in Figure 10, some of the failures are due to irrelevant images and/or largely occluded objects.

7 Verification for Remarks

Verification for Eqn. (8) Now we check the Eqn. (8) that the probability $p_i^k = Pr(k_i = k|x_i;\theta)$ can be approximated as

$$p_i^k \approx \prod_{t=1}^{K} \left[(q_i^t)^{\mathbf{1}(t=k)}(1-q_i^t)^{\mathbf{1}(t\neq k)}\right], \tag{13}$$

where $q_i^t = \Pr(\exists j, k_{ij} = t|x_i;\theta) = 1 - \prod_{j=1}^{m}(1-p_{ij}^t)$ and $p_{ij}^t = \Pr(k_{ij} = t|x_{ij};\theta)$ (Eqn. (6)).

For each positive bag, the probability p_{ij}^k $(k \neq 0)$ can be computed as

$$p_i^k = \Pr(k_i = k|x_i;\theta) \propto \prod_{j=1}^{m}(p_{ij}^0 + p_{ij}^k) - \prod_{j=1}^{m} p_{ij}^0, \tag{14}$$

and for each negative bag,

$$p_i^0 = \Pr(k_i = 0|x_i;\theta) \propto \prod_{j=1}^{m} p_{ij}^0. \tag{15}$$

Since $\Pr(k_i = k|x_i;\theta)$ is the form of combinational explosion, we use $1 - \sum_{k=1}^{K} p_{ij}^k \approx \prod_{k=1}^{K}(1-p_{ij}^k)$ to approximate the p_i^k as q_i^k. For each positive bag, we have

$$\begin{aligned}
p_i^k &\propto \prod_{j=1}^{m}(p_{ij}^0 + p_{ij}^k) - \prod_{j=1}^{m} p_{ij}^0 \\
&= \prod_{j=1}^{m}(1 - \sum_{t=1}^{K}(p_{ij}^t)^{\mathbf{1}(t\neq k)}) - \prod_{j=1}^{m}(1-\sum_{t=1}^{K} p_{ij}^t) \\
&\approx \prod_{j=1}^{m}\prod_{t=1}^{K}\left[(1-p_{ij}^t)^{\mathbf{1}(t\neq k)}\right] - \prod_{j=1}^{m}\prod_{t=1}^{K}(1-p_{ij}^t) \\
&= \prod_{t=1}^{K}(1-q_i^t)^{\mathbf{1}(t\neq k)} - \prod_{t=1}^{K}(1-q_i^t) = \prod_{t=1}^{K}\left[(q_i^t)^{\mathbf{1}(t=k)}(1-q_i^t)^{\mathbf{1}(t\neq k)}\right].
\end{aligned}$$

For each negative bag, we have

$$\begin{aligned}
p_i^0 &\propto \prod_{j=1}^{m} p_{ij}^0 = \prod_{j=1}^{m}(1-\sum_{t=1}^{K} p_{ij}^t) \approx \prod_{j=1}^{m}\prod_{t=1}^{K}(1-p_{ij}^t)^{\mathbf{1}(t\neq 0)} \\
&= \prod_{t=1}^{K}\left[(q_i^t)^{\mathbf{1}(t=0)}(1-q_i^t)^{\mathbf{1}(t\neq 0)}\right].
\end{aligned}$$

Thus we could model the p_i^k, $k \in \{0,1,\ldots,K\}$, as $\prod_{t=1}^{K}\left[(q_i^t)^{\mathbf{1}(t=k)}(1-q_i^t)^{\mathbf{1}(t\neq k)}\right]$.

The equivalence between DiscEM and MIL-Boost

Claim: Assuming all bags are conditionally independent *i.e.*, $\Pr(Y|X;\theta) = \prod_{i=1}^{n}\Pr(y_i|x_i;\theta)$, we have

$$\frac{d}{d\theta}\mathcal{L}(\theta;Y,X) = -\frac{d}{d\theta}\sum_{i=1}^{n}\log\Pr(y_i|x_i;\theta) \tag{16}$$

When the instance-level model Eqn. (1) and the bag-level model Eqn. (2) are used, MIL-Boost's update rule Eqn. (3) is equivalent to DiscEM, which reads as

$$\frac{d}{d\theta}\log\Pr(y_i|x_i;\theta) = \begin{cases} \sum_{j=1}^{m}\frac{-1}{1-p_{ij}}\frac{d}{d\theta}p_{ij} & \text{if } y_i = -1; \\ \sum_{j=1}^{m}\frac{1-p_i}{p_i(1-p_{ij})}\frac{d}{d\theta}p_{ij} & \text{if } y_i = 1. \end{cases} \tag{17}$$

Proof: Recall that the data is a set of bags $X = \{x_1,\ldots,x_n\}$, where each bag X_i contains a set of instances $\{x_{i1},\ldots,x_{im}\}$. Label y_i is given for bag x_i while y_{ij} is a hidden variable associated with instance x_{ij}. We denote $H_i = \{y_{i1},\ldots,y_{im}\}$ as the hidden variables for bag x_i and $H_Y = \{H_1,\ldots,H_n\}$ as the set of all the hidden variables. Under the MIL setting, each instance x_{ij} in a negative bag is known to be negative, and at least one instance in each positive bag is positive. In other words, given $y_i = -1$, we know $y_{ij} = -1$ for every j. Assuming that instances in each bag are independent, then for negative bags Eqn. (16) becomes

$$\begin{aligned}
\frac{d}{d\theta}\log\Pr(y_i = -1|x_i;\theta) &= \sum_j \frac{d}{d\theta}\log\Pr(y_{ij} = -1|x_{ij};\theta) \\
&= \sum_j \frac{d}{d\theta}\log(1-p_{ij}) = \sum_j \frac{-\frac{d}{d\theta}p_{ij}}{1-p_{ij}},
\end{aligned}$$

where $p(y_{ij}) = \Pr(y_{ij}|x_{ij};\theta)$ and $p_{ij} = p(y_{ij} = 1)$.

Next we derive the expression for each positive bag. The hidden variables H_i are conditionally dependent given y_i, but within each bag we assume they are independent, *i.e.*, $\Pr(H_i|x_i;\theta) = \prod_j \Pr(y_{ij}|x_i;\theta)$. We observe that $\Pr(H_i = -1, y_i = 1|x_i;\theta) = 0$ (the event is impossible) and $\Pr(H_i, y_i = 1|x_i;\theta) = \Pr(H_i|x_i;\theta)$ for all $H_i \neq -1$ (If $H_i \neq -1$ then $y_i = 1$). This leads to

$$\Pr(H_Y|y_i = 1, x_i;\theta) = \begin{cases} 0 & \text{if } H_i = -1; \\ \prod_j p(y_{ij})/p_i & \text{otherwise.} \end{cases} \quad (18)$$

In the above we use the Noisy-OR model shown in Eqn. (2), which gives $p_i = \Pr(y_i = 1|x_i;\theta) = 1 - \prod_j (1 - p_{ij})$. We now expand Eqn. (16) for positive bags as

$$\begin{aligned}
&\frac{d}{d\theta}\log\Pr(y_i = 1|x_i;\theta) \quad (19)\\
=&\sum_{H_i}\Pr(H_i|y_i = 1, x_i;\theta)\frac{d}{d\theta}\log\Pr(y_i = 1, H_i|x_i;\theta)\\
=&\sum_{H_i\neq -1}\prod_k \frac{p(y_{ik})}{p_i}\frac{d}{d\theta}\log\prod_j p(y_{ij})\\
=&\frac{1}{p_i}\sum_j\left[\sum_{H_i}\prod_k p(y_{ik})\frac{d}{d\theta}\log p(y_{ij}) - \sum_{H_i=-1}\prod_k p(y_{ik})\frac{d}{d\theta}\log p(y_{ij})\right]\\
=&\frac{1}{p_i}\sum_j\left[\sum_{y_{ij}} p(y_{ij})\frac{d}{d\theta}\log p(y_{ij}) - \prod_k (1 - p_{ik})\frac{d}{d\theta}\log(1 - p_{ij})\right]\\
=&\frac{1}{p_i}\sum_j\Big[p_{ij}\frac{d}{d\theta}\log(p_{ij}) + (1 - p_{ij})\frac{d}{d\theta}\log(1 - p_{ij})\\
&-(1 - p_i)\frac{d}{d\theta}\log 1 - p_{ij}\Big]\\
=&\frac{1}{p_i}\sum_j\Big[\frac{d}{d\theta}p_{ij} - \frac{d}{d\theta}p_{ij} - (1 - p_{ij})\frac{d}{d\theta}\log(1 - p_{ij})\Big]\\
=&\sum_j\frac{1 - p_i}{p_i(1 - p_{ij})}\frac{d}{d\theta}p_{ij}. \quad (20)
\end{aligned}$$

Based on Eqn. (18) and Eqn. (19), we have proved the claim for both negative bags and positive bags. □

8 Conclusion

In this paper, we have introduced a new learning algorithm, bottom-up multiple class learning (bMCL), which performs object localization, object class discovery, and object detector training in an integrated framework. We show the great advantage of the proposed method on a variety of benchmark datasets. We also demonstrate that our method performs comparably to state-of-the-art weakly supervised single class object recognition systems. Moreover, our proposed method can handle diverse and noisy Internet images for both clustering and detection tasks. Because we have illustrated the use of saliency as a generic prior knowledge source in a variety of vision tasks, our notion of saliency guidance may spark further interest in utilizing saliency measures in the context of high-level vision applications in the future.

The limitations of our method include: (1) difficulty of dealing with objects undergoing non-rigid transformation or articulation; (2) inability to handle multiple object categories in a single image; (3) sensitivity to initialization associated with being an EM-like algorithm; and (4) general difficultly associated with the intrinsic ambiguity between objects and common object parts.

Acknowledgments

This work was supported by Microsoft Research Asia, NSF IIS-1216528 (IIS-1360566), NSF CAREER award IIS-0844566 (IIS-1360568), and ONR N000140910099. We thank Jiayan Jiang, Yichen Wei, Tao Chen, Patrick Gallagher, and Piotr Dollar for encouraging discussions.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2189–2202, 2012.
[2] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Proc. Neural Inf. Process. Syst.*, 2002.
[3] B. Babenko, P. Dollár, Z. Tu, and S. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. In *Proc. Workshop on Faces in Real-Life Images, European Conf. Computer Vision*, 2008.
[4] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, 2011.
[5] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. iCoseg: Interactive co-segmentation with intelligent scribble guidance. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2010.
[6] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2010.
[7] K. Chang, T. Liu, and S. Lai. From co-saliency to cosegmentation: An efficient and fully unsupervised energy minimization model. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2011.
[8] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2011.
[9] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2007.
[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. Series B*, 39(1):1–38, 1977.
[11] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *International journal of computer vision*, 100(3):275–293, 2012.
[12] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71, 1997.
[13] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *Proc. Neural Inf. Process. Syst.*, 2013.
[14] P. Dollár, B. Babenko, S. Belongie, P. Perona, and Z. Tu. Multiple component learning for object detection. In *Proc. European Conf. Comput. Vis.*, 2008.
[15] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, Nov. 2001.
[16] I. Endres and D. Hoiem. Category independent object proposals. In *Proc. European Conf. Comput. Vis.*, 2010.
[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.
[18] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf.
[19] L. Fei-Fei, R. Fergus, S. Member, and P. Perona. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):594–611, 2006.
[20] P. F. Felzenszwalb, R. B. Girshick, and D. R. David A. McAllester. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1627–1645, 2010.
[21] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun. Salient object detection by composition. In *Proc. Int'l Conf. Comput. Vis.*, 2011.
[22] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. of Comp. and Sys. Sci.*, 55(1):119–139, 1997.

[23] B. J. Frey and N. Jojic. Transformation-invariant clustering using the em algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(1):1–17, 2003.

[24] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Proc. European Conf. Comput. Vis.*, 2008.

[25] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2007.

[26] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2012.

[27] G. Kim, C. Faloutsos, and M. Hebert. Unsupervised modeling of object categories using link analysis techniques. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2008.

[28] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2012.

[29] F. D. la Torre and M. J. Black. A framework for robust subspace learning. *Int'l J. Comput. Vis.*, 54(1):117–142, 2003.

[30] Y. J. Lee and K. Grauman. Shape discovery from unlabeled image collections. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2009.

[31] Y. J. Lee and K. Grauman. Object-graphs for context-aware category discovery. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2010.

[32] Y. J. Lee and K. Grauman. Learning the easy things first: Self-paced visual category discovery. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2011.

[33] C. Leistner, A. Saffari, and H. Bischof. Miforests: multiple-instance learning with randomized trees. In *Proc. European Conf. Comput. Vis.*, 2010.

[34] Q. Li, J. Wu, and Z. Tu. Harvesting mid-level visual concepts from large-scale internet images. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2013.

[35] N. Loeff and A. Farhadi. Scene discovery by matrix factorization. In *Proc. European Conf. Comput. Vis.*, 2008.

[36] L. Mason, J. Baxter, P. Bartlett, and M. Frean. Boosting algorithms as gradient descent. In *Proc. Neural Inf. Process. Syst.*, 2000.

[37] F. Moosmann, D. Larlus, and F. Jurie. Learning saliency maps for object categorization. In *ECCV Workshop on the Representation and Use of Prior Knowledge in Vision*, 2006.

[38] L. Mukherjee, V. Singh, J. Xu, and M. D. Collins. Analyzing the subspace structure of related images: Concurrent segmentation of image sets. In *Proc. European Conf. Comput. Vis.*, pages 128–142. Springer, 2012.

[39] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int'l J. Comput. Vis.*, 42(3):145–175, 2001.

[40] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *Proc. Int'l Conf. Comput. Vis.*, 2011.

[41] R. Rahmani, S. A. Goldman, H. Zhang, J. Krettek, and J. E. Fritts. Localized content based image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(11):1902–1912, 2008.

[42] C. Rother, T. Minka, and A. Blake. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2006.

[43] 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 *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2006.

[44] U. Rutishauser, D. Walther, C. Koch, and P. Perona. Is bottom-up attention useful for object recognition? In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2004.

[45] J. Salojarvi, K. Puolamaki, and S. Kaski. Expectation maximization algorithms for conditional likelihoods. In *Proc. Neural Inf. Process. Syst.*, 2005.

[46] S. Savarese and L. Fei-Fei. 3D generic object categorization, localization and pose estimation. In *Proc. Int'l Conf. Comput. Vis.*, 2007.

[47] J. Shotton, A. W. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2011.

[48] A. Strehl and J. Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J. Mach. Learning Research*, 3(3):583–617, 2003.

[49] M. Stricker and M. Orengo. Similarity of color images. In *Proc. Storage and Retrieval for Image and Video Databases*, 1995.

[50] T. Tuytelaars, C. H. Lampert, M. B. Blaschko, and W. Buntine. Unsupervised object discovery: A comparison. *Int'l J. Comput. Vis.*, 88(2):284–302, 2009.

[51] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2011.

[52] S. Vijayanarasimhan and K. Grauman. Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In *Proc. Conf. Comput. Vis. and Pattern Recogn.*, 2008.

[53] P. A. Viola and M. J. Jones. Robust real-time face detection. *Int'l J. Comput. Vis.*, 57(2):137–154, 2004.

[54] P. A. Viola, J. C. Platt, and C. Zhang. Multiple instance boosting for object detection. In *Proc. Neural Inf. Process. Syst.*, 2006.

[55] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *Proc. Neural Inf. Process. Syst.*, 2005.

[56] D. Zhang, F. Wang, L. Si, and T. Li. M^3IC: Maximum margin multiple instance clustering. In *Proc. Int'l Joint Conf. Artif. Intell.*, 2009.

[57] D. Zhang, F. Wang, L. Si, and T. Li. Maximum margin multiple instance clustering with its applications to image and text clustering. *IEEE Trans. Neural Netw.*, 22(5):739–751, 2011.

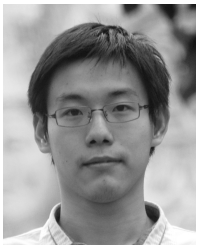
[58] M.-L. Zhang and Z.-H. Zhou. Multi-instance clustering with applications to multi-instance prediction. *Applied Intelligence*, 31:47–68, August 2009.

[59] B. Zhao, F. Wang, and C. Zhang. Efficient multiclass maximum margin clustering. In *Proc. Int'l Conf. Mach. Learning*, 2008.

[60] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li. Multi-instance multi-label learning. *Artificial Intelligence*, 176(1):2291–2320, 2012.

[61] L. Zhu, Y. Chen, and A. L. Yuille. Unsupervised learning of probabilistic grammar — markov models for object categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):114–128, 2009.

Jun-Yan Zhu received a BE degree with honors in computer science and technology from Tsinghua University in 2012. He is currently a PhD student at UC Berkeley in the Computer Science Division. His research interests include computer vision, computer graphics and computational photography.



Jiajun Wu is an undergraduate student at Institute for Interdisciplinary Information Sciences, Tsinghua University. His research interests include computer vision and machine learning.

Yan Xu is an assistant professor of school of biological science and medical engineering in Beihang University of China. She received a Ph.D. in biomedical engineering from Tsinghua university in China. Her area of interest includes biomedical image analysis, medical informatics, and big data.

Eric Chang received Bachelor, Master, and Ph.D. degress, all in electrical engineering and computer science from Massachusetts Institute of Technology. Eric Chang joined Microsoft Research Asia (MSRA) in 1999. Eric is currently the Senior Director of Technology Strategy at MSR Asia. His research interests include computer vision, bioinformatics, machine learning, and signal processing.

Zhuowen Tu received the ME degree from Tsinghua University and the PhD degree from Ohio State University. He is an assistant professor in the Department of Cognitive Science, University of California, San Diego (UCSD).