Unsupervised Classification and Part Localization by Consistency Amplification

Leonid Karlinsky, Michael Dinerstein, Dan Levi, and Shimon Ullman

Weizmann Institute of Science, Rehovot 76100, Israel
{leonid.karlinsky,michael.dinerstein,dan.levi,
shimon.ullman}@weizmann.ac.il

Abstract. We present a novel method for unsupervised classification, including the discovery of a new category and precise object and part localization. Given a set of unlabelled images, some of which contain an object of an unknown category, with unknown location and unknown size relative to the background, the method automatically identifies the images that contain the objects, localizes them and their parts, and reliably learns their appearance and geometry for subsequent classification. Current unsupervised methods construct classifiers based on a fixed set of initial features. Instead, we propose a new approach which iteratively extracts new features and re-learns the induced classifier, improving class vs. non-class separation at each iteration. We develop two main tools that allow this iterative combined search. The first is a novel star-like model capable of learning a geometric class representation in the unsupervised setting. The second is learning of "part specific features" that are optimized for parts detection, and which optimally combine different part appearances discovered in the training examples. These novel aspects lead to precise part localization and to improvement in overall classification performance compared with previous methods. We applied our method to multiple object classes from Caltech-101, UIUC and a sub-classification problem from PASCAL. The obtained results are comparable to state-of-the-art supervised classification techniques and superior to state-of-the-art unsupervised approaches previously applied to the same image sets.

1 Introduction

The goal of this paper is unsupervised classification, including discovery of a new category, learning a model of geometric arrangement of object parts and their appearance, and obtaining object and part localization, from a set of unlabeled images, which contains non-class images mixed with some unknown (usually small) percent of class images. The class instances may be uncropped, unaligned and of small size relative to the background.

The problem of unsupervised object classification has gained considerable recent interest [1,2,3,4,5,6,7,8,9,10,11,12], however, this task is still far from being completely solved. In this study we present a novel methodology to approach the problem. A common approach is to start from some limited, manageable set of

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part II, LNCS 5303, pp. 321–335, 2008.

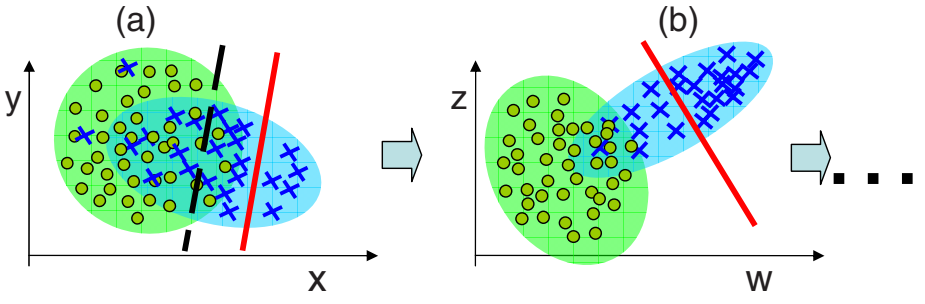


Fig. 1. Illustration of the feature re-extraction approach. (a) In the initial feature space (x-y) it is difficult to separate class (blue crosses) and non class (green circles) examples. In this feature space, the best separating hyperplane (which the unsupervised classification seeks to determine) is marked by the dashed line. Instead, our method identifies a subset of sure class examples separated from the rest (red solid line). (b) Using these examples, the method extracts a new feature set (z-w), in which a larger set of class examples can be identified. The process then continues iteratively. Each iteration uses new features, rather than previous features (or their combinations).

initial features $\mathcal{F}$, for example, a set of local descriptors extracted around image interest points or clusters extracted from such descriptors [1, 2, 11, 12, 13, 14, 15, 16, 5, 6, 7, 8, 9]. The set of features can be optimized by selecting a subset of the most useful features $\mathcal{F}_1 \subset \mathcal{F}$, or sometimes combinations of features in $\mathcal{F}_1$ are used as new features [2, 8, 14]. However, there is no guarantee that the choice of initial features will in general be sufficient for complete separation. In contrast, we approach the problem as a combined iterative search for features and a classifier. We do not use the initial feature set to obtain the final class separation, but only for identifying a subset of sure class examples which can be reliably separated from the rest (Fig. 1a). This goal is achieved by unsupervised training of a classifier that combines both appearance and part-geometry information. The extracted class examples are used to guide the subsequent extraction of new features, which were not a part of the initial feature set. It is not a-priori clear that this iterative approach will continue to improve classification: if the intermediate classification results are partly incorrect, their use could lead the process astray and cause deteriorating performance. In this work, we demonstrate that in the proposed algorithm, the constructed features become more class-specific as the computation evolves, and the class vs. non-class separation continuously improves (Fig. 1b), reaching a final high level of performance even compared with recent supervised methods.

We develop two main tools that allow the iterative combined search. One is the incremental discovery of part specific features, which combine different part appearances discovered in the training examples. The other is a novel star-like class-geometry model of object parts, which differs from the similar past models [3, 4, 5, 15, 16, 17, 13] and which can be learned efficiently without supervision in very noisy conditions. These two aspects are described briefly below, and explained in more detail in Sections 2.1 and 2.2.

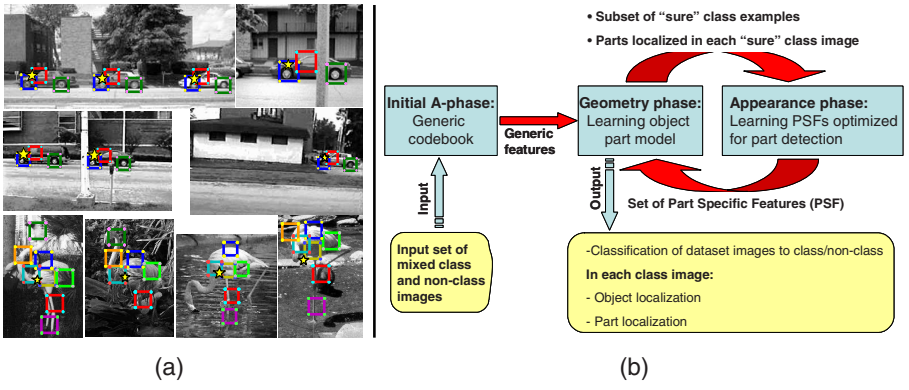


Fig. 2. (a) Example results of unsupervised object and part localization on two datasets (UIUC cars, flamingo). The yellow star is the detected model center location (see text), color coded rectangles are examples of detected object parts (for each object several out of about 150 modeled parts are shown). (b) Schematic diagram of the UCA algorithm.

Feature learning. most unsupervised approaches [1, 2, 5, 6, 7, 8, 9, 11, 12], including ours, start from some generic set of features $\mathcal{F}$. During learning, when a particular class is considered, the approaches select a subset $\mathcal{F}_1 \subset \mathcal{F}$ of so-called Class Specific Features (CSF), which coincide better with the class compared with the background or other classes. In contrast, our method extracts and learns a new set of features, termed Part Specific Features (PSF). The PSF are optimized to have higher detection scores at specific locations on the class objects, and at the same time to have lower scores at incorrect locations on the same objects and in non-object detections. Different part specific features have been used successfully in a number of supervised approaches, such as k-fan [17] and semantic hierarchy [18], and were shown to be useful for both object and part localization. Constructing such features in an unsupervised manner is challenging; our method is the first unsupervised method that learns and uses such features, resulting in improved object and part detection and localization.

Geometry learning. Past supervised and unsupervised classification methods can be categorized by their modeling of object geometry. In bag-of-feature methods [2, 9], geometry is ignored. Methods, such as [2, 8, 1, 14], extend the bag-of feature approach by using feature combinations. In [3, 4, 5, 11, 13, 15, 16, 17] object geometry is modeled by the spatial distribution of each feature in the object reference frame. A geometric part model is useful for classification, but it is challenging to construct such a model in an unsupervised setting. Most previous unsupervised methods therefore do not use a full geometric model [2,6,7,8,9,11,12]. Our method uses star-like geometry. It has several differences compared with similar past models. The method is not restricted by a small number of parts as in [3,4], unlike [13,15,16,17] it does not require any supervision, unlike [3,5,15,11] it models distribution of feature locations on the background, unlike [5,12] it does

not rely on non-geometric pLSA [2] for internal supervision, and unlike [10,6,7], it is not based on prior image segmentation. These differences are explained in more detail in Sections 2 and 2.1.

In terms of class vs. background classification performance, our method outperforms the state-of-the-art unsupervised methods [2,5,7,10,11,12] on 18 classes from the Caltech 101, Weizmann horses and UIUC cars datasets. Surprisingly, the method is also comparable in performance to existing state-of-the-art supervised (and weakly supervised) methods applied to the same datasets. We further demonstrate how our method can be used to separate different object views on the cars class from the PASCAL challenge 2007 dataset. As the method achieves precise object and part localization, it provides a basis for top-down segmentation, as illustrated in supplementary material.

The rest of the paper is organized as follows. Section 2 presents an overview followed by a detailed description of each of the method stages. Section 3 presents results obtained on various datasets, together with an analysis and comparison with previously reported results. Conclusions are discussed in Section 4.

2 The Consistency Amplification Method

Our approach alternates between model learning and data partitioning. Given an image set S, an initial model (learned using initial features) is used to induce an initial partitioning by identifying highly likely class members. The initial partitioning is then used to improve both the appearance and geometrical aspects of the model, and the process is iterated. In this manner the process exploits intermediate classification results at a given stage to guide the next stage. Each stage leads to an improved consistency between the detected features and the model, which is why the process is termed Unsupervised Consistency Amplification (UCA). Each UCA iteration consists of two phases of learning: the feature learning Appearance-phase (A-phase) followed by the part model learning Geometry-phase (G-phase), explained in detail in sections 2.2 and 2.1 respectively. The approach and the order of the phases are summarized in Fig. 2b.

Initial Appearance-phase. In all our experiments, we use a generic codebook of SIFT descriptors of 40×40 patches for the initial (appearance) features. This codebook, denoted by $\mathcal{F}_0$, is computed by a standard technique [19] from all the images in given set S. The codebook descriptors are compared to the descriptors at all points of all the images in S and storing the points of maximal similarity (either one or several, see below) in each image.

Geometry-phase. The detection of parts using the generic features is usually noisy, due to detections in non-class images, and at some incorrect locations in the class images. The goal of the geometric part model learning is to distinguish between the correct and incorrect detections, based on consistent geometric relations between features. This is accomplished by the G-phase of the algorithm, which is also used for the selection of the most useful features and the automatic assignment of each of their detections in every image in S to either object or

background model. In contrast with [3, 4, 5, 15, 11] that use uniform distribution of features on the background, we model the background by a distribution of the same family as the class object distribution, which allows to prevent the spurious geometric background consistency from being accounted for by the learned class model. In our experiments we found that modeling the background distribution is better than assuming uniformity with mean performance gain of $12 \pm 7\%$ EER in the first iteration of the UCA that uses initial generic appearance features. The learned background model is then discarded after the learning and is not used for classifying new images. Thus, the model used in the G-phase is a mixture of two stars, one for object and the other for background. It is learned without supervision from all the images in S using a novel graphical model formulation explained in detail in Section 2.1. After the geometric structure has been learned, a subset $H \subset S$ of images which contain class objects with high confidence is selected. In these images the object centers and parts are localized. Unlike [5,12] that learn the geometric constraints using only a set of objects identified by the non-geometric pLSA [2], our method identifies and localizes objects, and learns their part geometry, jointly and explicitly from the entire data.

Appearance-phase. Each part-specific feature constructed in the A-phase represents an object part by extracting several typical appearance patches of the part, from different images. Part-patches can be extracted, because the locations of the parts in the images of the subset H are already estimated from the previous G-phase. An optimal subset of these part patches is learned by a discriminative model described in section 2.2. The set of all part specific features extracted during the A-phase is denoted by $\mathcal{F}$.

Computing the output. After the G-phase at each iteration, the learned model is applied to produce classification, as well as object and part localization results for either the given dataset or an unseen test set. This is done without introducing any supervision to the system. The way we apply the learned model to test images is described in detail in section 2.3.

2.1 The Geometry Phase

We first describe the G-phase model, and then explain how it is learned from the data. The main goal of the G-phase is to identify the most likely locations of objects and their parts in all images of the given set S and to estimate a subset $H \subset S$ of images which contain class objects with high confidence. The G-phase models the data by a generative probabilistic graphical model depicted in Fig. 3a. Let the image set S have N unlabelled images: $S = \{I_1, I_2, \ldots I_N\}$ and the current feature set $\mathcal{F}$ consist of M features: $\mathcal{F} = \{F_1, F_2, \ldots F_M\}$. In the G-phase of the first UCA iteration these features are a codebook of generic SIFT descriptors $\mathcal{F}_0$, and in the following UCA iterations these are the learned PSFs. During the G-phase each feature is associated with an object part or the background. Denote the detected location of feature F_m in image I_n by X_m^n (the G-phase uses a single (maximal) detected location per feature in each image, see extension below.) The G-phase model independently generates observed samples:

$Data = \{ (F_m, I_n, X_m^n) \,|1 \le n \le N, 1 \le m \le M \}$ The probability of observing a specific image $\Pr(I = I_n)$ is taken to be uniform. The overall observed data likelihood under the G-phase model can be written as:

$$\Pr(Data) \propto \prod_{n=1}^{N} \prod_{m=1}^{M} \sum_{C_m^n=1}^{2} \int_{L_m^n} \Pr(C_m^n \,|I_n\,) \Pr(F = F_m \,|C_m^n\,) \\ \Pr(L_m^n \,|I_n, C_m^n\,) \Pr(L_F = X_m^n \,|F_m, C_m^n, L_m^n\,) dL_m^n \tag{1}$$

The meaning of the product inside the integral in eq. 1 is that each data sample (F_m, I_n, X_m^n) observed in image I_n for the feature F_m is independently generated as follows. First, the latent discrete binary "class" variable C_m^n is drawn with probability $\Pr(C_m^n = k\,|I_n\,) = \alpha_k^n$, independent of the feature F_m. $C_m^n = 1$ means that I_n contains a class object and F_m is generated from the class model. $C_m^n = 2$ means that F_m is generated from the background model, because either I_n does not contain an object or F_m was not detected consistently with the class model. After learning, the value α_k^n is the likelihood of class k (either object or background) in image I_n. Next, the latent location variable L_m^n is drawn from a Gaussian distribution $\Pr(L_m^n \,|I_n, C_m^n = k\,) = N(\mu_k^n, \Sigma_k^n)$. L_m^n represents the image position of the center of the star model (chosen by C_m^n), which generates the feature F_m in image I_n. Note that for every feature detected in image I_n that has chosen the class k, there is a separate variable L_m^n, but all of these variables are generated from the same distribution specific to I_n. Next, the observed feature variable F draws its value F_m from the distribution $\Pr(F = F_m \,|C_m^n = k) = \beta_k^m$ which depends on the chosen class k, but is independent of the image I_n. After learning, the value β_k^m is the likelihood of feature F_m to be consistent with the geometric model of class k. Finally, the observed feature location variable L_F draws its value X_m^n from a linear Gaussian distribution $\Pr(L_F = X_m^n \,|F_m, C_m^n = k, L_m^n\,) = N(L_m^n + \rho_k^m, \Lambda_k^m)$. This distribution models the uncertainty of the offset ρ_k^m of the feature F_m from the L_m^n - center of the star model chosen by C_m^n. It is specific to the feature F_m and the chosen class k and is independent of the specific image I_n.

To summarize, the parameters of the model are α, β, μ, Σ, ρ and Λ, all of them are learned by soft EM as described further below. A schematic drawing illustrating the data generation process and the meaning of the main model parameters is shown in Fig. 3b. The model uses a star-like geometry, but an important difference between the current model and past star model formulations is worth noting. In contrast with [17, 16, 15], that have a single reference point or k-fan per image, in our model there exists a separate reference point (center) random variable for each part, drawn, however, from the same distribution specific to the given image. This allows the features detected in the same image to be updated individually: features assigned to the class update the class star and features assigned to the background update the background star, both the assignments and the updates are soft. Although it may sound technical, it has fundamental importance, since, as we saw in our experiments, in different class images, different subsets of features are geometrically consistent with the object model. It is interesting to note that the transition from the standard star-model

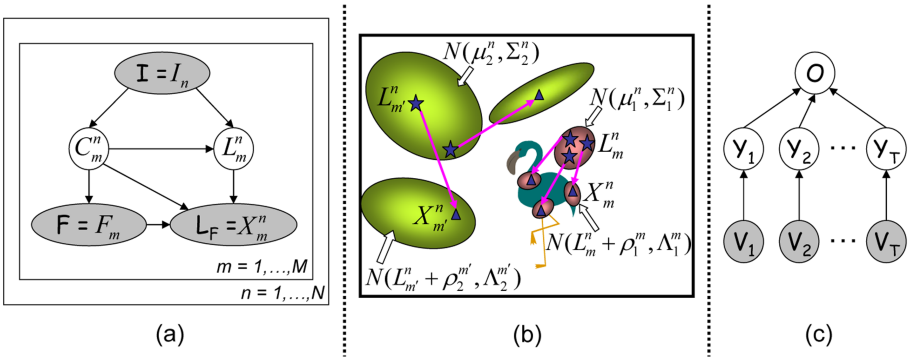


Fig. 3. The probabilistic models used by UCA. Shaded ellipses are observed variables, unfilled are hidden (latent). (a) Graphical representation of the G-phase generative model. (b) Generating the object and background. The object model illustrated in red, the background in green. Each generates centers, denoted by ★ (star) and features denoted by ▲ (triangle). The ellipses denote the uncertainty in position. X_m^n denotes the detected location of feature F_m. Every feature F_m detected on the object is generated using its own star center point L_m^n, but all these L_m^n are generated from the same distribution $N(\mu_1^n, \Sigma_1^n)$ specific to the image. As illustrated, the learned object distributions are tighter than the background distributions. (c) Graphical representation of the "Continuous Noisy OR" discriminative model.

to our version is entirely analogous to the transition from Naive Bayes (NB) to pLSA. In the NB there is only a single class node generating the entire feature vector of an image, while in pLSA each feature has a separate topic node generated from an image specific distribution. The pLSA is more flexible than NB and was found useful for unsupervised classification [2, 9]. Similarly we found that the modified star is useful in modeling feature geometry in the unsupervised setting.

Learning: The model is learned from the data using the soft EM algorithm. The EM update equations are provided in the supplementary material. As mentioned above, the data samples fitted by our model are of the form (F_m, I_n, X_m^n). In order to incorporate the features' detection scores into the learning process, we weight each sample by its score. Namely, the sample (F_m, I_n, X_m^n) is weighted by R_m^n - the similarity of F_m with I_n at location X_m^n. The parameters of the model: α, β, μ, ρ and Λ, are initialized at random and EM is run until convergence. In order to have the object model learned with respect to $C_m^n = 1$, in all our experiments, we initialize $|\Sigma_1^n| << |\Sigma_2^n|$ for every n. During EM iterations, this initialization causes the object feature detections to tend to update the $C_m^n = 1$ model, since they usually appear in more tight and repeatable configurations (i.e. fit a star with smaller center uncertainty Σ_1^n). At the same time, background feature detections, that are usually loosely scattered all over the image, will tend to coincide with the $C_m^n = 2$ model. This method of the initialization of all the parameters (including Σ) was identical throughout all our experiments.

Identifying the set of high-likelihood images: After the EM converges, the value of α_1^n (the image probabilities to belong to the class) shows a strong separation between a subset of class images and all the non-class images, see figure 6a. As a result, by the end of G-phase it becomes possible to identify a subset of high-likelihood class candidate images H. In all our experiments, we marked an image I_n as high likelihood class candidate if $\alpha_1^n > \eta \cdot \max_n(\alpha_1^n)$, where $\eta = 0.85$ was chosen empirically and used throughout all the experiments. Examples of objects automatically identified and localized by the G-phase of the first UCA iteration are shown in Fig. 6b. These are examples of the first G-phase output, obtained using the initial generic features. As can be seen, the localized object model centers appear at similar locations within the object in the different images. In the A-phase, these points are used to extract stacks of corresponding fragments, which are used to construct the part specific features - the CNOR part-detectors, as explained in the next section.

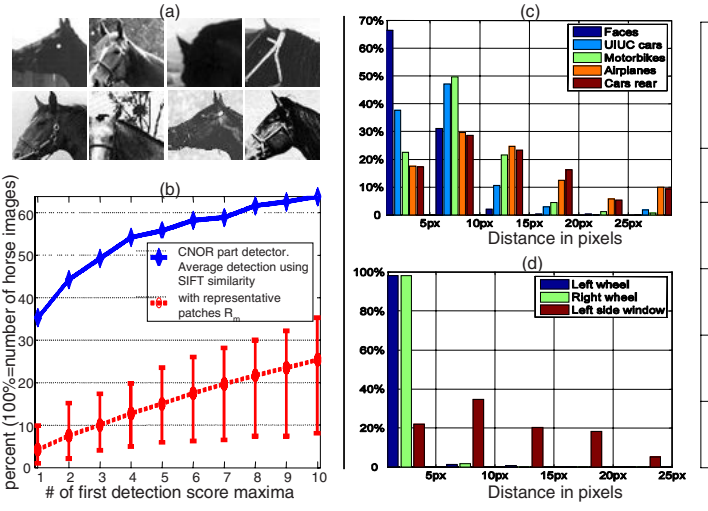
2.2 The Appearance Phase

In the G-phase we learned the position of each part relative to the object model center, and detected this center in the images belonging to H. We localize each part in these images by assuming it is located at the learned relative position ρ_1^m from the center located at μ_1^n. In the A-phase we learn for each part a detector trained to distinguish image patches in correct part locations from patches in incorrect ones. The detector is trained using the detected part locations as positive examples and all other locations on the images of H as negative examples. The constructed part detectors form the new feature set $\mathcal{F}$ for the G-phase of subsequent UCA iteration. We next describe the novel probabilistic discriminative model used by the part detector, the Continuous Noisy OR (CNOR), and how this model is trained.

For each part m, corresponding to F_m above, we extract a set of appearances in the following way. In each class candidate image $I_n \in H$, we take the 40x40 image patch at position $\mu_1^n + \rho_1^m$, where μ_1^n is the location of the learned object center in I_n and ρ_1^m is the learned offset of part m from the object center. The accumulated set of image patches is the candidate set of part appearances: $A_m = \{Z_1^m, \ldots, Z_T^m\}$. The next step is to select a subset of appearance representatives $R_m \subseteq A_m$, and learn to optimally combine their detection evidence in order to reliably detect the object part. Both tasks are achieved simultaneously by training the CNOR model, depicted in Fig.3c. Let P be an arbitrary image patch taken from an arbitrary location L in a new image. The binary variable O^P is set to $O^P = 1$ iff L and P are the location and appearance of part m respectively. The probability of O^P is discriminatively modeled as:

$$\Pr(O^P|V;\Theta) = \sum_Y \Pr(O|Y) \cdot \prod_{t=1}^{T} \Pr(Y_t|V_t;\theta_t,R_m) \tag{2}$$

The $\Theta = \{R_m, \theta_1, \ldots, \theta_T\}$ are the learned parameters of the model. $V = \{V_t\}$, where V_t is the output of a continuous SIFT similarity measure between P and



	Object localization error STD, % of object size
Faces	2.6
UIUC cars	4.6
Motor-bikes	5
Air-planes	8
Cars rear	8

Fig. 4. (I) Example of a CNOR part detector. (a) Selected representative appearances. (b) Cumulative histograms of detections within 15px from ground truth location relative to the number of detection score local maxima used. (II) Evaluation of object and part localizations obtained by the UCA method showing a distribution of localization error in pixels relative to manually marked ground truth. 10px is less then 7% of object size in all the sets in the figure. (c) Object localization, 100% = number of class images. (d) Part localization on UIUC cars, each part was detected in about 95% of objects. In the graph 100% = number of part detections.

$Z_t \in A_m$. Note that we do not explicitly model $\Pr(V)$, which can be a complex distribution. $Y = \{Y_t\}$, where Y_t is a latent binary variable representing the detection of appearance Z_t with:

$$\Pr(Y_t = 1 | V_t; \theta_t, R_m) = \begin{cases} \frac{1}{1+e^{-\alpha_t(V_t - \tau_t)}} & Z_t \in R_m \\ 0 & Z_t \in A_m \backslash R_m \end{cases} \quad (3)$$

Here $\theta_t = \{\tau_t, \alpha_t\}$ are the parameters of the sigmoid in 3. $Y_t = 1$ becomes likely if patch P exceeds a similarity threshold τ_t with the representative patch $Z_t \in R_m$, with α_t representing the uncertainty of τ_t. If $Z_t \in A_m \backslash R_m$ (meaning Z_t is not a chosen representative) then V_t and Y_t have no effect on $\Pr(O^P | V; \Theta)$. Finally, $\Pr(O^P | Y)$ is a deterministic "or" of Y:

$$\Pr(O^P = 1 | Y) = \begin{cases} 1 & \exists t. Y_t = 1 \\ 0 & otherwise \end{cases} \quad (4)$$

The entire model can intuitively be described as follows: the part is detected ($O_p = 1$) whenever P is "sufficiently" similar to at least one of the part m's representative patches in R_m.

To learn the model parameters, a training set of image patches E is constructed by taking all 40×40 image patches (on a fixed step grid) from all the

images in the current class candidate set H. For each patch $P \in E$ the observed data vector is constructed as: $D^P = \langle V^P, O^P \rangle$ where $V^P = \{V_t^P\}$ is computed by measuring similarity between P and A_m patches and $O^P = 1$ iff $P \in A_m$ (and $O^P = 0$ otherwise). Finally the training data for the CNOR model of part m is: $D = \{D^P | P \in E\}$. By treating the correct part appearances (A_m) as positive examples and all other appearances (either other parts of the object or background patches) as negative examples, the object part detector is trained for correct localization of the part. To limit the number of representatives, the learning objective is to find the Minimum Description Length (MDL) parameters Θ, in other words, to find Θ that maximize a combined score of the model complexity (number of representatives) and model performance (data likelihood). We solve this learning problem using the Structural EM (SEM) algorithm optimizing the Bayesian Information Criterion (BIC) score [20]:

$$BIC = \sum_{P \in E} \log(\Pr(O^P | V^P; \Theta)) - \frac{\log T}{2} \cdot |R_m| \tag{5}$$

The SEM algorithm iterates between two stages. The first stage is, given a set of representatives R_m, to find the optimal values for the $\{\theta_t\}$ parameters. This stage is solved using the EM algorithm. It is computed efficiently, since in our model each EM iteration has linear time complexity. The second stage is, given the current assignment of $\{\theta_t\}$, to estimate an improved R_m. This is achieved by running several iterations of a greedy search over subsets of $R_m \subseteq A_m$, where at every step of the search a current subset R_m is modified by either adding or removing one element.

After learning, the set of part m detections is obtained by identifying first few local maxima of the probability $\Pr(O^P | V; \Theta)$ computed for all patches P in a given image. Selected representatives for an example part are shown in Fig. 4a. The resulting CNOR part detectors for all parts m, are a significantly more reliable set of object features then the initial generic set of features, as demonstrated in Fig. 4b and in section 3, and it provides a general method for reliable part detection for both supervised and unsupervised classification.

2.3 Applying the Learned Model to Classify and Localize Objects and Parts

To compute the classification score for an unseen test image, and to localize the class objects in it, we use the learned ρ_1^m (offsets from object star center) and Λ_1^m (STDs for these offsets) parameters in a voting scheme similar to [13] as follows. For each part detector, a number (five in our experiments) of highest-scoring locations at each image are marked. To identify the object star's center location, each detection X votes for a center location, by placing a Gaussian mask with STD Λ_1^m around the expected location $X - \rho_1^m$. After all the detectors voted, the point with the maximal accumulated vote determines the location of the object star's center in each image (in case there are multiple objects, several local maxima that exceed a global threshold are taken). The accumulated vote value at

the detected center point serves as the object detection score in the image. These scores are then used to create the ROC that tests the separation between the class and the non-class images in the results section 3. The object localization results of our method are evaluated in Fig. 4c. The parts are localized by "back-projection" as in [13]. Each part detector that voted into one of the selected object center locations (with one of its five detections) is declared as 'detected' and is marked in the image. The accuracy of our part localization is demonstrated in figures 2a and 5 and evaluated in Fig. 4d. Marking all the detected parts in the image can be used for a top-down segmentation of the detected object (see examples in supplementary material). The details of the top-down segmentation are outside the scope of the current discussion.

3 Results

To test the performance of the UCA method, it was applied to the task of fully unsupervised classification and object and part localization on 18 different object classes. The list of the classes, the parameters of the datasets and ROC EERs obtained by the UCA are summarized in Table 1. The results show that our method obtains superior performance over the existing unsupervised methods in challenging conditions such as small objects relative to the background (e.g. UIUC cars, Caltech101 cars, flamingo), small percent of class images in the set (e.g. schooner, guitars), significant inter-class variability due to non-rigid deformations (e.g. bonsai, horses, crab, flamingo, starfish) and significant lack of alignment (e.g. UIUC cars, faces, PASCAL car views). Examples of the classes and object and part localizations obtained by UCA are shown in figures 2a and 5. Fig. 4c,d shows quantitative evaluation of automatic object and part localization by UCA compared to hand generated ground truth on several dataset. The background images for each dataset were chosen randomly out of Caltech backgrounds set containing 900 images. To challenge our method, we tested it on different class vs. non-class mixes, namely 10%, 20%, 30% and 50%. This is compatible with experimenting with Google data, since manual validation done by [5] showed that on average, above 25% of images returned by Google image search are good examples. For every dataset, increasing percent of class images above the percent reported in the table gives even better results. The UIUC cars dataset contained only the 170 non-cropped and non-aligned test images of the original set (and equal amount of random background images), the training images of the original set are cropped, so to make the task harder they were not used. The Caltech-5 datasets (from [3]) were tested in order to compare with past unsupervised approaches that were tested on the same data, namely [2,12,5,10,7]. To ensure that the chosen Caltech101 classes are sufficiently hard, 9 of the 11 tested Caltech101 classes are the ones with lowest reported performance by [7] (average of entries for these classes on [7]'s confusion matrix diagonal is 49%). Note that unlike [7], we do not use color information in our scheme.

An important characteristic of the UCA is its ability to deal with a low percentage of class images in the dataset. Methods such as [5, 12] that apply the

Table 1. Summary of fully unsupervised classification results obtained by the UCA method. For all datasets, the EER STD for UCA was $\leq$ 2% (computed by cross-validation). For motorbikes, airplanes and cars-rear, the class images were randomly chosen from larger sets and remaining images were also used for testing the learned models obtaining 1.3%, 2.3% and 2.8% average EER respectively. The average EER of UCA on the Caltech-5 datasets was 2.65%. Results of other unsupervised methods reported for Caltech-5 were: average EER of 4.08% [12], 7.35% [5] and 11.38% [2] and average multiclass detection rate of 5.4% [7]. Results of leading supervised methods on Caltech-5 are comparable to our unsupervised result: average EER of 2.25% [15] and 1% [21] ([21] did not test on cars-rear class). The object size relative to the background for each dataset was approximated from several characteristic images.

Dataset	Origin	Total number of images	% class images in the set	Object size rel. to bgrnd.	UCA EER, %
Horses	Weizmann	646	50%	35%	2.7
Cars	UIUC	340	50%	3.5%	3.1
Car views	PASCAL	201	50%	40%	10.7
Motorbikes	Caltech-5	900	50%	30%	2.3
Airplanes	Caltech-5	900	50%	20%	2.7
Faces	Caltech-5	900	50%	20%	2.4
Cars-rear	Caltech-5	900	50%	20%	3.2
Bonsai	Caltech101	256	50%	35%	4.1
Ewer	Caltech101	283	30%	34%	2.3

Dataset	Origin	Total number of images	% class images in the set	Object size rel. to bgrnd.	UCA EER, %
Butterfly	Caltech101	303	30%	27%	6
Cars	Caltech101	600	20%	12%	1.1
Crab	Caltech101	365	20%	24%	10.9
Starfish	Caltech101	430	20%	11%	12.6
Laptop	Caltech101	405	20%	32%	4.3
Flamingo	Caltech101	335	20%	13%	7.3
Watch	Caltech101	1139	20%	40%	3.9
Guitars	Caltech101	650	10%	35%	8.2
Schooner	Caltech101	650	10%	27%	6.69

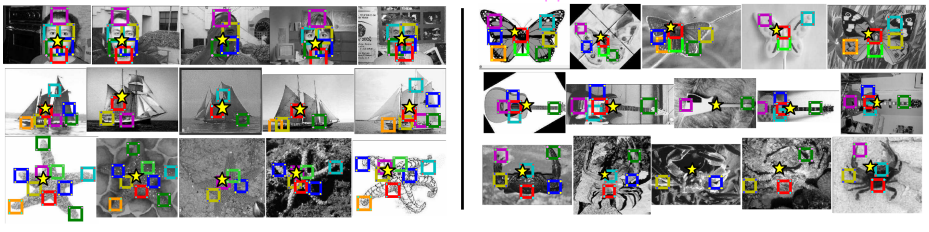
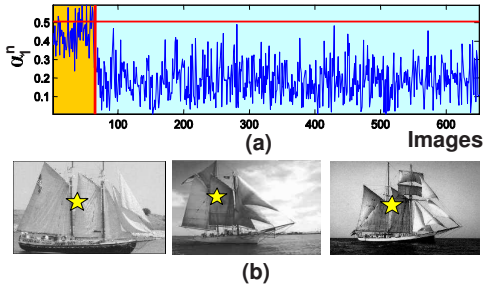


Fig. 5. More examples of unsupervised object and part localizations obtained by the UCA method. See explanation in fig. 2a.

pLSA method of [2] as a pre-processing step to identify and localize class examples may fail on such datasets. This was validated by testing the pLSA method with eight topics (optimal number proposed by [5]) on the schooner dataset that contains only 10% class images. From the N examples with maximal score in the class topic, less then 40% were class examples ($N = 10, 20, \ldots, 100$).

In the PASCAL car views experiment, we tested the ability of UCA to separate related sub-classes. In particular, out of the PASCAL 2007 training images, images depicting frontal and side views of cars were extracted. The scale of the images was normalized by vertical size and large background areas around each car was taken to make the set un-cropped and un-aligned. The UCA was then applied to this set in order to separate the views. Furthermore, when applied



	Iter. 2 EER %	Iter. 3 EER %	Iter. 4 EER %	Iter. 5 EER %
Starfish	16.3	15.1	12.7	~
Crab	17.8	14.9	12.3	10.9
Flamingo	11.9	7.4	~	~
Guitars	17.4	8.2	~	~
Bonsai	8.6	3.9	~	~

(c)

Fig. 6. Improvements due to consistency amplification. (a) Example of class vs. background separation obtained by the first iteration for the schooners class. Yellow part are the class images and the rest are backgrounds (the ordering is only for illustration purposes). The horizontal line shows the adaptive threshold $\eta \cdot Max$ used to select the set H of high likelihood class examples for the next UCA iteration. (b) Examples of the objects identified and localized. (c) Table of EER improvement with UCA iterations. Only the classes that ran for more then two iterations are shown. The iterations continue until the set H stops growing. The average EER of the first iteration (that used the generic features and not PSFs) was 30% for these classes. This illustrates the low (relative to the PSFs) consistency between the generic features and the class objects.

on a set of about 700 images containing all the car views, UCA successfully learned the "frontal cars" subclass with similar EER to the two view experiment. Applying pLSA to the same set has yielded high error (32% EER). The ability of our method to separate similar sub-classes and specifically different views of the same class can also be useful in supervised learning applications. If a given training set of images of the same class can be automatically separated into a meaningful set of (inherently similar) subclasses, then it can greatly facilitate the learning task, by allowing the modeling of each subclass separately.

4 Conclusions

The UCA method has a number of basic advantages compared with previous unsupervised classification methods. First, the overall classification results are higher than obtained previously, and remain high even when class examples are sparsely distributed within the dataset. Surprisingly, on the tested classes, results of the unsupervised method are as good as leading supervised methods. Second, the method obtains precise object localization, indicated by a repeatable reference point on each detected object. Third, precise locations of the parts participating in the model are also made available. Fourth, the method is capable of separating similar classes and sub-classes, such as different views of the same class. The main novel aspects of the UCA method are the following. The model is iteratively improved by exploiting intermediate classification results, consistently improving the performance. A novel geometric model is used, which can be

efficiently learned from the entire dataset, and therefore improve the methods ability to capture geometric consistencies, even when consistent configurations are sparse. The model uses a part detection scheme, which is trained to detect object parts with diverse appearances in their correct position. The resulting detections are therefore more reliable, providing precise part localization and improved overall performance.

Acknowledgments. This work was supported by EU IST Grant FP6-2005-015803 and ISF Grant 7-0369.

References

1. Fritz, M., Schiele, B.: Towards unsupervised discovery of visual categories. In: Franke, K., Müller, K.-R., Nickolay, B., Schäfer, R. (eds.) DAGM 2006. LNCS, vol. 4174, pp. 232–241. Springer, Heidelberg (2006)
2. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their localization in images. In: ICCV, pp. 370–377 (2005)
3. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. CVPR (2), 264–271 (2003)
4. Fergus, R., Perona, P., Zisserman, A.: A visual category filter for google images. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3021, pp. 242–256. Springer, Heidelberg (2004)
5. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google's image search. In: ICCV, pp. 1816–1823 (2005)
6. Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. CVPR (2006)
7. Cao, L., Fei-Fei, L.: Spatially coherent latent topic model for concurrent object segmentation and classification. In: ICCV (2007)
8. Nowozin, S., Tsuda, K., Uno, T., Kudo, T., Bakir, G.H.: Weighted substructure mining for image analysis. CVPR (2007)
9. Li, L.J., Wang, G., Fei-Fei, L.: Optimol: automatic object picture collection via incremental model learning. CVPR (2007)
10. Ahuja, N., Todorovic, S.: Discovering hierarchical taxonomy of categories and shared subcategories in images. In: ICCV (2007)
11. Liu, D., Chen, T.: Semantic-shift for unsupervised object detection. In: CVPR Workshop (2006)
12. Liu, D., Chen, T.: Unsupervised image categorization and object localization using topic models and correspondences between images. In: ICCV (2007)
13. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: ECCV (2004)
14. Quack, T., Ferrari, V., Leibe, B., Gool, L.V.: Efficient mining of frequent and distinctive feature configurations. In: ICCV (2007)
15. Loeff, N., Arora, H., Sorokin, A., Forsyth, D.: Efficient unsupervised learning for localization and detection in object categories. In: NIPS (2005)
16. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Learning hierarchical models of scenes, objects, and parts. In: ICCV (2005)

17. Crandall, D.J., Huttenlocher, D.P.: Weakly supervised learning of part-based spatial models for visual object recognition. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 16–29. Springer, Heidelberg (2006)
18. Epshtein, B., Ullman, S.: Semantic hierarchies for recognizing objects and parts. CVPR (2007)
19. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: ICCV (2005)
20. Friedman, N.: The bayesian structural em algorithm. UAI, 129–138 (1998)
21. Dorko, G., Schmid, C.: Object class recognition using discriminative local features. INRIA (2005)