Learning Subcategory Relevances for Category Recognition

Sinisa Todorovic and Narendra Ahuja
Beckman Institute, University of Illinois at Urbana-Champaign
{sintod, n-ahuja}@uiuc.edu

Abstract

A real-world object category can be viewed as a characteristic configuration of its parts, that are themselves simpler, smaller (sub)categories. Recognition of a category can therefore be made easier by detecting its constituent subcategories and combing these detection results. Given a set of training images, each labeled by an object category contained in it, we present an approach to learning: (1) Taxonomy defined by recursive sharing of subcategories by multiple image categories; (2) Subcategory relevance as the degree of evidence a subcategory offers for the presence of its parent; (3) Likelihood that the image contains a subcategory; and (4) Prior that a subcategory occurs. The images are represented as points in a feature space spanned by confidences in the occurrences of the subcategories. The subcategory relevances are estimated as weights, necessary to rescale the corresponding axes of the feature space so that the images with the same label are closer to each other than to those with different labels. When a new image is encountered, the learned taxonomy, relevances, likelihoods, and priors are used by a linear classifier to categorize the image. On the challenging Caltech-256 dataset, the proposed approach significantly outperforms the best categorizations reported. This result is significant in that it not only demonstrates the advantages of exploiting subcategory taxonomy for recognition, but also suggests that a feature space spanned by part properties, instead of direct object properties, allows for linear separation of image classes.

1. Introduction

Suppose an image contains objects belonging to multiple categories. The image is labeled by one of these category names. Given a set of images, containing repeated occurrences of each label, this paper is aimed at estimating the models of the category corresponding to each label. By finding the subimages shared by the images carrying the same label, and contrasting them with the remaining images, carrying other labels, we wish to identify the subimages occupied by the category and obtain an image model for the category. The final result is a model for each of the label categories which is well represented in the image set.

Our approach follows from the well recognized notion that objects consist of parts. The intrinsic nature of the parts and their spatial configurations define the category model. These parts themselves define their own object categories. Thus, larger categories, in general, are hierarchical configurations of smaller and simpler categories. In the sequel, we refer to the constituent categories also as subcategories. A category model captures the observed variations in the category instances. These variations are due to their natural diversity, as well as differences in imaging parameters, e.g., illumination, viewing direction, camera characteristics, and occlusion. The larger the extent of an object, the more diverse is the range of such effects, and more complex the model. Therefore, recognition of the (smaller and simpler) object parts is more reliable than the recognition of the entire object. This makes recognition of a category from its subcategory detections an efficient strategy, but requires that these detections are efficiently combined.

The approach of combining part detections for object recognition has been pursued in [10, 2]. They build a taxonomy of categories found in an image set. The taxonomy is represented by a directed acyclic graph (DAG) in which each category node is connected to all those representing its subcategories. The taxonomy encodes sharing of (simpler, smaller) subcategories by multiple, more complex categories (e.g., “buses” and “cars” share “wheels”). The sharing of a subcategory results in a node being connected to multiple parent nodes. Recognition of the category of an image with unknown label is achieved by maximizing its match with the taxonomy, i.e., with subgraphs of the taxonomy such that the structurally matching nodes have the most similar image properties (e.g., shape, size, and color of image regions). The part of the taxonomy that yields the maximal match identifies the detected category. However, this process has limitations. First, the taxonomy contains only the likelihoods of a category match; no prior probabilities of the occurrence of different categories are estimated. This amounts to using the maximum likelihood criterion for detection, which is not optimal in the Bayesian sense. The second limitation is that matches of all parts are given equal weight, because the taxonomy does not encode the relevance of a subcategory to the definition of a specific parent category. A subcategory may be very common and appear in the hierarchical definitions of many parent categories (e.g., “windows” are often shared by “buildings,” “houses,” “recreational-vehicles,” etc.). Detection of such a subcategory provides poor evidence that any specific parent occurs. On the other hand, a subcategory may appear in the definitions of only a few parents, or, in the extreme case, a unique parent (e.g., “two-humps-on-the-back” of “camels”). Detection of such a subcategory unambiguously suggests the presence of its unique parent. Since detections of different subcategories provide different degrees of evidence for category recognition, estimation of the subcategory relevances is important.

This paper is aimed at improved category modeling and recognition by addressing the aforementioned limitations. We pursue five interrelated objectives. To achieve the first two objectives, we extend and use the approach of [10, 2] which is a prerequisite for the remaining three objectives that are completely new and represent the major contributions of this paper. Given a set of labeled images, each showing a category, our approach is aimed at: (1) Unsupervised extraction of region-based, hierarchical definitions of image categories in terms of their shared subcategories; (2) Estimation of the likelihoods that a given image contains any occurrences of the discovered subcategories; (3) Estimation of the subcategory relevances to the recognition of each image category, relative to the other category labels present; (4) Estimation of the subcategory priors; and (5) Specification of a linear classifier that will categorize a new image by fusing the subcategory recognition results and their learned relevances, as illustrated in Fig. 1.

Objectives (1)-(4) are aimed at overcoming the limitations discussed earlier. Objective (5) is aimed at using the learned, extended taxonomy for object recognition, while still retaining the simplicity of the linear classifier used in [10, 2]. Although the recognition process in [10, 2] is intended to be hierarchical, as one of the contributions of this paper, we show that their approach actually uses a linear recognizer. They flatten the taxonomy so all parts are represented as separate object features, thus discarding their hierarchical interdependencies. This yields a subcategory feature space in which images are represented as points whose coordinates are the confidences of detecting the subcategories. To recognize objects, they use a linear classifier in this feature space that equally weights all the axes. Despite the flattening and subsequent use of a simple, linear classifier, they achieve successful recognition on challenging benchmark datasets. This suggests that a feature space spanned by part properties, instead of direct image properties of entire objects, allows for linear separation of image classes. In particular, since each subcategory is characterized by a certain combination of photometric, geometric, and structural properties of regions it contains, the subcategory feature space is obtained by a transformation of the original feature space whose axes are these region properties. This transformation may be viewed as serving the same goal as intended for the kernel transforms to define higher-dimensional spaces sought by SVM’s. Although complex, this transform is obtained by a “natural” (subcategory based) definition of categories, so one may expect it to be a reasonable direction to pursue. We do so in this paper. Specifically, towards objective (5), we extend the work of [10, 2] by incorporating the learned subcategory relevances and priors in their linear classifier.

Overview of our approach: Suppose we are given a set of labeled images, $\mathbb{D}$, where each label c means that the image contains at least one occurrence of category c from the user-specified set of label categories $\mathbb{C}$. Each image may also contain occurrences of other categories from $\mathbb{C}$, even though it does not carry their labels. Where all in an image the category occurs is unknown. Together, the images with a given label are assumed to capture all representative occurrences of the category. Our approach consists of the following four steps. **Step 1:** We begin with obtaining a segmentation-tree representation of the images, to derive subcategory based definitions of the label categories in $\mathbb{C}$. Segmentation trees capture the recursive embedding of image regions, obtained through a multiscale segmentation. In the sequel, we will refer to (sub-)trees and (sub-)images interchangeably. **Step 2:** The subimages corresponding to recurrences of a subcategory in $\mathbb{D}$ are found by identifying subtrees across $\mathbb{D}$ that match. Each transitive closure of the best matching subtrees is identified as the set of occurrences of the corresponding subcategory. These transitive closures are used to learn the subcategory likelihoods and priors. The discovery of subcategories $\mathbb{I}$ present in $\mathbb{D}$ allows us to learn the definitions of label categories $\mathbb{C}$ in terms of $\mathbb{I}$ in the following step. **Step 3:** We define the flattened feature space in which the images are represented as points with coordinates that measure the posterior probabilities of occurrence of the subcategories from $\mathbb{I}$. This subcategory feature space is suitable for estimating the subcategory relevances. This is done so the axes corresponding to the different subcategories are rescaled such that all image points with the same category label are closer to each other than to the points with other labels. This rescaling increases the weights of those subcategories that are more discriminative for a given image category, as was the goal. Let $\mathbb{D}_c$ denote all training images with the same label c. Then, the learned subcategory relevances represent such weights of the subcategories that jointly maximize the smallest margins between each image point in $\mathbb{D}_c$ to its nearest neighbor in $\mathbb{D}_c$ and to its nearest neighbor in the remaining image set. By the large-margin

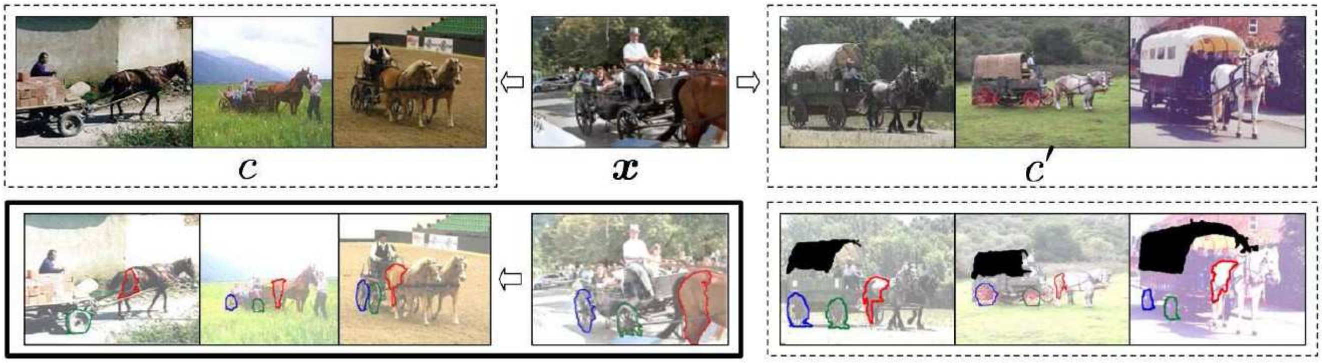

Figure 1. Samples from Caltech-256: (top row) Two labeled image sets for categories c and c', which share subcategories horses, wheels, trees. The subcategory wagon-top frequently occurs in c', but not in c. Recognition of c' is particularly aided by observing wagon-top, whereas the shared subcategories do not provide strong evidence for either category. Therefore, the learned relevance of the wagon-top for recognizing category c' is larger than those of shared subcategories. (bottom row) The absence of the wagon-top and the presence of shared subcategories in $\boldsymbol{x}$ leads to the decision that $\boldsymbol{x}$ belongs to category c.

theory, the proposed learning algorithm will have good generalization over new images. **Step 4:** Once the subcategory relevances, priors, and likelihoods are learned, they are used to define a linear classifier for categorizing new images.

Sec. 2 reviews prior work, Sec. 3 describes Steps 1-2, Sec. 5 presents Step 3, Sec. 6 presents experimental evaluation, and two Appendices provide details of our algorithms.

2. Relationships to Prior Work

Image categorization can be formulated as scene matching that uses a hierarchical structure of local image features (e.g., SIFTs) [5, 8]. However, the scene-based categorization usually requires volumes of training images to robustly learn a combinatorially large number of object configurations in the scene. The generalizabilty of these approaches to images containing occlusion and large variations in scale is still unclear. The advantages of using object-specific features for the purposes of image categorization have been well argued in prior work [12, 1, 11, 10, 2, 3]. We advance these approaches by using the region-based taxonomy of categories, which yields significant improvements in categorization performance.

Our approach to learning the subcategory relevances is related to general feature weighting algorithms. The computationally intensive wrapper methods evaluate the performance of a classifier to select relevant features, whereas in the filter methods features are weighted by their information content [12]. Our learning algorithm belongs to the group of embedded methods that incorporate feature weighting into the learning process of a classifier [1, 11, 4, 7, 9]. Specifically, we modify and use the algorithm presented in [9], which is based on the well known RELIEF algorithm [7] that estimates feature weights by maximizing the margins of the 1-NN classifier over data. A major problem with RELIEF is that the in-class and out-of-class nearest neighbors of a sample are computed prior to learning, and thus are very unlikely to remain so in the weighted feature space, once the feature weights are estimated. This problem has been addressed by RELIEF-F and I-RELIEF [9]. They learn one global set of feature weights for all classes. We modify and extend their work to learn category-specific weights. The weights thus locally learned per category, in our case, are expected to more robustly handle large intra-category variations from image to image than globally estimated weights over the entire training set. In [4], learning category-specific feature weights is pushed to the extreme of estimating the weights for each image separately. However, they formulate this estimation as maximizing the difference in distances between the image and all other in-class and out-of-class images. Since for large training sets this is computationally infeasible, they select, prior to learning, a fixed number of in-class and out-of-class closest neighbors, and thus encounter the same aforementioned problem as RELIEF.

3. Discovering Subcategories

This section presents Steps 1–2 of our approach that concern deriving segmentation-tree representations of the images in training set $\mathbb{D}$, and discovering subcategories present in $\mathbb{D}$. To this end, we use the approach of [10, 2], briefly reviewed below.

Images are represented by segmentation trees (Fig. 2) whose nodes correspond to regions obtained via a multi-scale segmentation, and edges capture region embedding. A vector $\boldsymbol{\psi}_v$ of region properties, such as brightness, area, perimeter, centroid location, etc., is associated with each node v, defined relative to v's parent, to allow scale and rotation-in-plane invariance of recognition. In this paper, we illustrate our approach for the case when each image has exactly one category label. In the training set

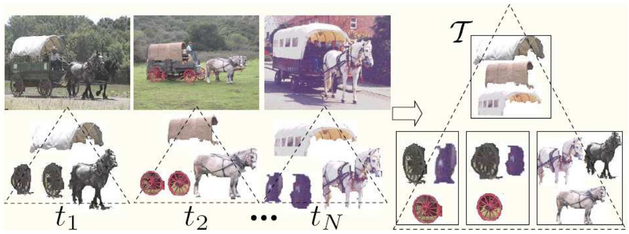

Figure 2. Segmentation trees t_n capture the recursive embedding of regions in images. The tree-union T is the minimum-size graph that contains all segmentation trees.

$\mathbb{D}=\{(t_n, y_n)\}_{n=1}^{N}$, where $y\in\mathbb{C}$ denotes the label of segmentation tree t, subcategories appear as similar subtrees within the image trees, having similar node properties $\boldsymbol{\psi}$ and structure. They can be discovered by matching the trees, and then finding their maximum-similarity, common subtrees. Given two trees t and t', the goal of matching is to find the subtree isomorphism, $f=\{(v,v')\}\subset t\times t'$, which preserves the original structure of t and t', and maximizes their similarity measure, $S_{tt'}$, defined as

$$S_{tt'}\triangleq \sum_{(v,v')\in f} [2\min(\|\boldsymbol{\psi}_v\|,\|\boldsymbol{\psi}_{v'}\|)-\max(\|\boldsymbol{\psi}_v\|,\|\boldsymbol{\psi}_{v'}\|)]. \tag{1}$$

After matching the trees, we find the transitive closures of maximally matching subtrees, to discover subcategories present in $\mathbb{D}$. Note that a subtree is uniquely defined by a node at which the subtree is rooted. We will refer to a node and the subtree under it interchangeably. As explained in greater detail in [10, 2], finding the transitive closures of maximally matching nodes is equivalent to constructing a tree-union, T, from all the trees in $\mathbb{D}$ (Fig. 2). As in [10], we construct the tree-union sequentially. In each iteration τ, $T^{(\tau)}$ is matched with a new tree $t\in\mathbb{D}$, and then the unmatched nodes of t are added and appropriately connected to $T^{(\tau)}$ to form $T^{(\tau+1)}$. Each tree-union node $i\in T$ records a collection of all regions across the images in $\mathbb{D}$ that matched with i (Fig. 2). That is, tree-union node i represents a transitive closure of matching regions, defining subcategory i. Consequently, the set of nodes in T, denoted as $\mathbb{I}$, represents the discovered set of subcategories. Each tree-union node i is characterized by a Gaussian pdf, and thus by the mean vector $\boldsymbol{\mu}_i$=mean$\{\boldsymbol{\psi}_v\}$, and covariance $\boldsymbol{\Sigma}_i$=cov$\{\boldsymbol{\psi}_v\}$ of all matching regions v across $\mathbb{D}$ transitively grouped under i. These parameters specify the Gaussian likelihoods of the corresponding subcategories, $P(\boldsymbol{\psi}|i)\triangleq\mathcal{N}(\boldsymbol{\psi};\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i)$, $\forall i\in\mathbb{I}$. We also compute the frequency of occurrence of each discovered subcategory i in the training set $\mathbb{D}$. For large $\mathbb{D}$, as is the case in this paper, this frequency represents the estimate of the prior probability that i occurs, $P(i)$.

4. Categorization Using a Linear Classifier

In this section, we show that the approach of [10, 2] categorizes images using a linear classifier. Recall that every image from the training set is uniquely registered within tree-union T. Therefore, all training images with the same label c uniquely define subgraph T_c within T that represents the model of category $c\in\mathbb{C}$. The subset of tree-union nodes that belong to T_c represent the subset of subcategories, denoted as $\mathbb{I}_c$, defining category c. Thus, a previously unseen image is categorized in [10, 2] by matching its segmentation tree t with every T_c, $\forall c\in\mathbb{C}$, which produces their similarity measures S_{tT_c}, given by (1), and then by finding the maximum similarity $y=\arg\max_{c\in\mathbb{C}} S_{tT_c}$. From (1), matching node $v\in t$ with node $i\in T_c$ amounts to identifying subcategory i in image t, with confidence measured heuristically as x_i=$[2\min(\|\boldsymbol{\psi}_v\|,\|\boldsymbol{\mu}_i\|)-\max(\|\boldsymbol{\psi}_v\|,\|\boldsymbol{\mu}_i\|)]$ if node pair (v,i) is included in subtree isomorphism $f\subset t\times T_c$, otherwise x_i=0. Therefore, we can conveniently rewrite the expression of (1) to highlight that S_{tT_c} is computed as a sum of confidences x_i that the subcategories $i\in\mathbb{I}_c$ occur in the image, $S_{tT_c}=\sum_{i\in\mathbb{I}_c} w_i x_i$, where all weighting coefficients w_i=1. This proves that [10, 2] perform linear separation of image categories in the feature space spanned by equally weighted confidences in subcategory detections.

5. Learning the Subcategory Weights

This section presents Step 3 of our approach (Sec. 1) that estimates the relevance of discovered subcategories for recognition of each individual image category. Given a distance function between two images, defined in terms of confidences in subcategory detections in the images, the relevances are computed so that each image from the training set is closer to its nearest in-class image pair than to its nearest out-of-class pair. To this end, we modify and use the algorithm presented [9]. For completeness, we review this algorithm, and point our major differences. We begin with describing our vector image representation. Notation is summarized in Tab. 1.

The vector of relevances of all subcategories $i\in\mathbb{I}$ for a given image category $c\in\mathbb{C}$ is defined as $\boldsymbol{w}_c\in\mathbb{W}=\{\boldsymbol{w}: \boldsymbol{w}\in\mathbb{R}^{|\mathbb{I}|}, \|\boldsymbol{w}\|=1, \boldsymbol{w}\geq\boldsymbol{0}\}$, where $\|\cdot\|$ is the two-norm, and w_{ic}=0 if subcategory i does not appear in the definition of c. In general, the vectors $\boldsymbol{w}_c$ differ for distinct categories in $\mathbb{C}$. Since in this section we consider only the subcategory relevances to one parent category, we will drop the category indication in subscript, $\boldsymbol{w}_c\rightarrow\boldsymbol{w}$, for simplicity. To estimate $\boldsymbol{w}$, as in [10, 2], we flatten the segmentation trees of the training images, and specify their vector representations as $\boldsymbol{x}\in\mathbb{X}=\{\boldsymbol{x}: \boldsymbol{x}\in\mathbb{R}^{|\mathbb{I}|}, \boldsymbol{x}\geq\boldsymbol{0}\}$, where x_i is a confidence measure that subcategory $i\in\mathbb{I}$ occurs in image $\boldsymbol{x}$. Since we use segmentation tree t and vector $\boldsymbol{x}$ as two alternative representations of the same image, in the sequel, we will refer to image trees and image vectors interchangeably. Analogous to the heuristic confidence measures x_i used in [10, 2] for recognition (Sec. 4), we here specify x_i using the estimates of subcategory likelihoods $\mathcal{N}(\boldsymbol{\psi}_v; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ and priors $P(i)$, obtained in Step 2 of our approach. Recall that subcategory detections in the training images are automatically achieved by constructing the tree-union (Sec. 3). That is, all those nodes v from the image trees that are transitively clustered under tree-union node i indicate the presence of subcategory i in the corresponding images. For subcategories $i{\in}\mathbb{I}$ whose occurrences v are identified in image $\boldsymbol{x}$, we compute x_i as: $x_i{\triangleq}\mathcal{N}(\boldsymbol{\psi}_v; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)P(i){\propto}P(i|\boldsymbol{\psi}_v)$, where the prior over region properties, $P(\boldsymbol{\psi}_v)$, is assumed uniform. For the rest of subcategories $i{\in}\mathbb{I}$ that are not identified in the image, we set $x_i{\triangleq}0$. By using this vector image representation, similar to the approach of [9], we define a distance function $d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w})$ between images $\boldsymbol{x}$ and $\boldsymbol{x}'$ as

$$d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}) \triangleq \boldsymbol{w}^{\mathrm{T}}|\boldsymbol{x}{-}\boldsymbol{x}'|\ . \tag{2}$$

Given the training set, $\mathbb{D}{=}\{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N}$, and $d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w})$, the relevances $\boldsymbol{w}$ are estimated so as to transform the original space of images $\mathbb{X}$ into a weighted space $\tilde{\mathbb{X}}$ by maximizing intra- vs. inter-category homogeneity of the image vectors with the same label, as measured by $d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w})$. This is done by maximizing the margins of 1-NN classifiers associated with every image, defined using the image's nearest miss and hit as follows. For each $\boldsymbol{x}$ with label y, the other training images can be grouped into two sets, referred to as hits $\mathbb{H}_{\boldsymbol{x}}$ and misses $\mathbb{M}_{\boldsymbol{x}}$, where $\mathbb{H}_{\boldsymbol{x}} \triangleq \{\boldsymbol{x}'{:}(\boldsymbol{x}', y'){\in}\mathbb{D}, y'{=}y, \boldsymbol{x}'{\neq}\boldsymbol{x}\}$, and $\mathbb{M}_{\boldsymbol{x}} \triangleq \{\boldsymbol{x}'{:}(\boldsymbol{x}', y'){\in}\mathbb{D}, y'{\neq}y\}$. The nearest miss $\boldsymbol{m}_{\boldsymbol{x}}(\boldsymbol{w}) \triangleq \arg\min_{\boldsymbol{x}'\in\mathbb{M}_{\boldsymbol{x}}} d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w})$ and hit $\boldsymbol{h}_{\boldsymbol{x}}(\boldsymbol{w}) \triangleq \arg\min_{\boldsymbol{x}'\in\mathbb{H}_{\boldsymbol{x}}} d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w})$ of $\boldsymbol{x}$ are used to define the margin of the image's 1-NN classifier as

$$r_{\boldsymbol{x}}(\boldsymbol{w}){\triangleq}d_{\boldsymbol{x}\boldsymbol{m}_{\boldsymbol{x}}}{-}d_{\boldsymbol{x}\boldsymbol{h}_{\boldsymbol{x}}}{=}\boldsymbol{w}^{\mathrm{T}}(|\boldsymbol{x} - \boldsymbol{m}_{\boldsymbol{x}}(\boldsymbol{w})| - |\boldsymbol{x}{-}\boldsymbol{h}_{\boldsymbol{x}}(\boldsymbol{w})|). \tag{3}$$

By maximizing the margins $r_{\boldsymbol{x}}(\boldsymbol{w})$ of all images $\boldsymbol{x}$ with the same label, say, $y = c \in \mathbb{C}$, we compute the subcategory relevances $\boldsymbol{w}_c$ for recognition of image category c:

$$\boldsymbol{w}_c{=}\max_{\boldsymbol{w}\in\mathbb{W}} \boldsymbol{w}^{\mathrm{T}} \textstyle\sum_{\boldsymbol{x}\in\mathbb{D}_c}(|\boldsymbol{x}{-}\boldsymbol{m}_{\boldsymbol{x}}(\boldsymbol{w})|{-}|\boldsymbol{x}{-}\boldsymbol{h}_{\boldsymbol{x}}(\boldsymbol{w})|). \tag{4}$$

Table 1. Frequently Used Notation

$\mathbb{C}$ - user-specified set of object categories, i.e., image labels;
$\mathbb{D}, \mathbb{D}_c$ - training set, and its subset with images of the same label c;
$\mathbb{I}$ - set of subcategories discovered in $\mathbb{D}$;
t - segmentation tree representation of the image;
T - tree-union of all the image trees in the training set;
v - node in the segmentation tree;
i - node in tree-union T = discovered subcategory;
$\boldsymbol{x} = [\ldots x_i \ldots]^{\mathrm{T}}$ - confidence vector representation of the image;
x_i - confidence in detecting subcategory i in image $\boldsymbol{x}$;
y - category label assigned to image $\boldsymbol{x}$, i.e., to tree t;
$\boldsymbol{w} = [\ldots w_i \ldots]^{\mathrm{T}}$ - vector of relevances w_i of subcategories $i{\in}\mathbb{I}$;
$\boldsymbol{\psi}$ - vector of region properties;
$\boldsymbol{\mu}_i, \Sigma_i$ - Gaussian parameters associated with subcategory i;
$d_{\boldsymbol{x}\boldsymbol{x}'}$ - distance between images $\boldsymbol{x}$ and $\boldsymbol{x}'$;
$\mathbb{M}_{\boldsymbol{x}}$ ($\mathbb{H}_{\boldsymbol{x}}$) - set of misses (hits) of image $\boldsymbol{x}$;
$m_{\boldsymbol{x}}$ ($h_{\boldsymbol{x}}$) - nearest miss (hit) of image $\boldsymbol{x}$;
σ - kernel width for probability density estimation.

The optimization problem of (4) uses $\boldsymbol{m}_{\boldsymbol{x}}(\boldsymbol{w})$ and $\boldsymbol{h}_{\boldsymbol{x}}(\boldsymbol{w})$ which require the subcategory relevances to have already been learned. Note that different values of $\boldsymbol{w}$ may yield completely different nearest misses and hits of $\boldsymbol{x}$ in the weighted image space $\tilde{\mathbb{X}}$ from those computed in the original space $\mathbb{X}$. To account for the uncertainty in estimating the nearest neighbors in $\tilde{\mathbb{X}}$, we reformulate (4) within a probabilistic framework. In particular, we specify $\boldsymbol{h}_{\boldsymbol{x}}$ and $\boldsymbol{m}_{\boldsymbol{x}}$ as hidden random variables characterized by certain probability density functions (pdf's). Given the pdf's of these hidden variables, $\boldsymbol{w}$ can be estimated from (4) by averaging out $\boldsymbol{h}_{\boldsymbol{x}}$ and $\boldsymbol{m}_{\boldsymbol{x}}$ over the entire training set. This is similar to the EM algorithm, where the incomplete (hidden) data are substituted by their mean values. To this end, we introduce the following probabilities: (1) $P_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}}$ – probability that image $\boldsymbol{x}'{\in}\mathbb{H}_{\boldsymbol{x}}$ is the nearest hit of $\boldsymbol{x}$; and (2) $P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}$ – probability that $\boldsymbol{x}'{\in}\mathbb{M}_{\boldsymbol{x}}$ is the nearest miss of $\boldsymbol{x}$. These probabilities can be computed using the standard kernel-based density estimation, with a kernel function $\kappa(\cdot)$, as

$$P_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}} \triangleq \frac{\kappa(d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}))}{\sum_{\boldsymbol{x}''\in\mathbb{H}_{\boldsymbol{x}}}\kappa(d_{\boldsymbol{x}\boldsymbol{x}''}(\boldsymbol{w}))},\ \forall\boldsymbol{x}'{\in}\mathbb{H}_{\boldsymbol{x}}, \tag{5}$$

$$P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}} \triangleq \frac{\kappa(d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}))}{\sum_{\boldsymbol{x}''\in\mathbb{M}_{\boldsymbol{x}}}\kappa(d_{\boldsymbol{x}\boldsymbol{x}''}(\boldsymbol{w}))},\ \forall\boldsymbol{x}'{\in}\mathbb{M}_{\boldsymbol{x}}. \tag{6}$$

We use the exponential kernel $\kappa(d){\triangleq}\exp(-d/\sigma)$, where kernel width σ is an input parameter. As shown in Sec. 6, our algorithm is largely insensitive to a specific ("meaningful") choice of σ. Other kernel functions can also be used.

The probabilities defined in (5) and (6) allow us to find the expectation of the margins in (4), and thus address the uncertainty of estimating the image's nearest neighbors in the weighted space $\tilde{\mathbb{X}}$. Similar to the EM, our learning algorithm consists of the E-step and M-step that are iterated alternatively until objective function Q reaches convergence. In the E-step, Q is computed as the expectation of $r_{\boldsymbol{x}}(\boldsymbol{w})$ with respect to the hidden variables $\boldsymbol{h}_{\boldsymbol{x}}$ and $\boldsymbol{m}_{\boldsymbol{x}}$, using the current estimate of $\boldsymbol{w}^{(\tau)}$. In the M-step, Q is maximized with respect to $\boldsymbol{w}$, resulting in new estimate $\boldsymbol{w}^{(\tau+1)}$. In the E-step, we compute:

$$Q{=}\boldsymbol{w}^{\mathrm{T}}\,\underbrace{[\sum_{\substack{\boldsymbol{x}\in\mathbb{D}_c\\ \boldsymbol{x}'\in\mathbb{M}_{\boldsymbol{x}}}}|\boldsymbol{x}{-}\boldsymbol{x}'|P^{(\tau)}_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}{-}\sum_{\substack{\boldsymbol{x}\in\mathbb{D}_c\\ \boldsymbol{x}''\in\mathbb{H}_{\boldsymbol{x}}}}|\boldsymbol{x}{-}\boldsymbol{x}''|P^{(\tau)}_{\boldsymbol{x}''=\boldsymbol{h}_{\boldsymbol{x}}}]}_{\boldsymbol{z}_c^{(\tau)}}, \tag{7}$$

where $P^{(\tau)}_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}}$ and $P^{(\tau)}_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}$ are obtained from (5) and (6) using $\boldsymbol{w}^{(\tau)}$. Then, in the M-step, we compute:

$$\boldsymbol{w}^{(\tau+1)} = \arg\max_{\boldsymbol{w}\in\mathbb{W}} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z}_c^{(\tau)} = \frac{[\boldsymbol{z}_c^{(\tau)}]_+}{\|[\boldsymbol{z}_c^{(\tau)}]_+\|} , \qquad (8)$$

where $[a]_+ \triangleq \max(a,0)$. The detailed derivation of the update rule (8) is given in [9] and in Appendix I.

Learning the subcategory relevances for a given image category is summarized in Alg. 1. The E-step and M-step are iterated until $\|\boldsymbol{w}^{(\tau+1)} - \boldsymbol{w}^{(\tau)}\| < \epsilon = 10^{-3}$. Appendix II presents the convergence analysis of Alg. 1 – specifically, the proof of Theorem 1 that states the following attractive properties: (1) Alg. 1 converges if the chosen kernel width σ is sufficiently large; (2) Alg. 1 converges for a wide range of σ values, and thus is largely insensitive to a specific choice of σ; (3) The convergence rate of Alg. 1 increases for larger σ; (4) Unlike many machine learning algorithms (e.g., neural nets), Alg. 1 converges always to a unique solution that is not affected by the initialization value of $\boldsymbol{w}^{(0)}$. The range of σ values that is reasonable to select for image categorization is shown in Sec. 6. Complexity of Alg. 1 is $O(|\mathbb{I}|\tau)$.

```
Algorithm 1: Learning the Subcategory Relevances
Input   : Image category $c\in\mathbb{C}$, $\mathbb{D}=\{(\boldsymbol{x},y)\}$; $\mathbb{I}$; $\sigma$; $\epsilon$
Output: Optimal relevances $\boldsymbol{w}_c \leftarrow \boldsymbol{w}^{(\tau)}$

Set $\tau=0$; $\forall i\in\mathbb{I}$: set $w_i^{(0)}$ to a positive random value; $\forall\boldsymbol{x}\in\mathbb{D}_c$ find
    $\mathbb{M}_{\boldsymbol{x}}$ and $\mathbb{H}_{\boldsymbol{x}}$ ;
repeat
    E-step: $\forall\boldsymbol{x}\in\mathbb{D}_c$ and $\forall\boldsymbol{x}'\in\mathbb{M}_{\boldsymbol{x}}$ and $\forall\boldsymbol{x}''\in\mathbb{H}_{\boldsymbol{x}}$ compute
        $P^{(\tau)}_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}}$ and $P^{(\tau)}_{\boldsymbol{x}''=\boldsymbol{m}_{\boldsymbol{x}}}$, as in (5)-(6); Compute $Q$ as in (7);
    M-step: Update $\boldsymbol{w}^{(\tau+1)}$ as in (8); $\tau=\tau+1$;
until $||\boldsymbol{w}^{(\tau)} - \boldsymbol{w}^{(\tau-1)}|| < \epsilon$ ;
```

```
Algorithm 2: The Entire Learning
Input   : Training image trees $\mathbb{D}=\{(t,y)\}$; $\sigma$
Output: Optimal relevances $\boldsymbol{w}_c$, $\forall c\in\mathbb{C}$

Find tree-union $T$ of $\mathbb{D}$ (Sec. 3);
Interpret nodes $i\in T$ as discovered subcategories $i\in\mathbb{I}$;
Compute the vector image representations, $\boldsymbol{x}$, where $x_i$ is
    proportional to the posterior probability that subcategory $i\in\mathbb{I}$
    occurs in image $\boldsymbol{x}$;
Compute $\boldsymbol{w}_c$, $\forall c\in\mathbb{C}$, by using Alg. 1.
```

The entire learning algorithm including the discovery of subcategories and estimating the subcategory relevances of all image categories is summarized in Alg. 2. Its complexity is computed as follows. Complexity of image matching is $O(|t|^2)$, where $|t|$ is the number tree nodes. Matching two image trees with approximately 50 nodes each, takes about 20s on a 2.8GHz, 2GB RAM PC with code implemented in MATLAB. Also, complexity of Alg. 1 with τ iterations is $O(|\mathbb{I}|\tau)$. Thus, complexity of Alg. 2 is $O(|\mathbb{I}|^2 + |\mathbb{C}||\mathbb{I}|\tau)$.

6. Results

Categorization of a new image is conducted by matching the image's segmentation tree with tree-union T, to identify subcategory occurrences in the image. Then, the vector representation $\boldsymbol{x}$ of the image is computed. Finally, the image is categorized using the linear classifier which is parameterized by the learned subcategory relevances for each image category: $\hat{c} = \arg\max_{c\in\mathbb{C}} \boldsymbol{w}_c^{\mathrm{T}}\boldsymbol{x}$.

For evaluation, we use two datasets: Caltech-101 and Caltech-256 [6], which, at the time of conducting these experiments, contained more categories by an order of magnitude than any other publicly available dataset. The well-known drawbacks of Caltech-101, such as little variation in pose or scale within many categories, have been addressed in Caltech-256. Specifically, in Caltech-256, the categories are carefully selected so as to represent a broader variety of natural and artificial objects appearing in indoor and outdoor scenes, with larger inter- and intra-category variability. The images are acquired under challenging lighting conditions, at different scales, and from diverse viewpoints. The images also contain occlusion and background clutter. As a major challenge to categorization, Caltech-256 contains similar categories whose definitions share almost identical sets of subcategories, as illustrated in Fig. 3. This help evaluate the generality of our approach.

Performance is evaluated as a function of: (i) the number of training images per category N_c, and (ii) the specific choice of kernel width σ – the only input parameter. For training, we use $N_c = \{5{:}5{:}30\}$, while the remaining images are used for testing. The convergence rate of learning is tested for $\sigma = \{0.1{:}0.1{:}1\}$. The average categorization error of 10 experiments is reported.

Fig. 4 compares the best reported results on Caltech-101 and Caltech-256 with ours. For every value of N_c, our approach outperforms the existing work. For N_c=30, we outperform by 9.9% the best recorded result on Caltech-256 [3]. To quantify the accuracy gain of our approach, we compute the increase in the area under our recognition-rate curve vs. those of competing approaches. Let $RR_{\text{our}}(N_c)$ and $RR_{\text{old}}(N_c)$ denote the recognition rates of our approach and prior work obtained for N_c training images per category. Then, the accuracy gain is defined as $\alpha = \frac{\sum_{N_c}(RR_{\text{our}}(N_c) - RR_{\text{old}}(N_c))}{\sum_{N_c} RR_{\text{old}}(N_c)}$. Table 2 presents the α values. Note that the approach of [10] is not developed to address image categorization, and does not use any discriminative information between categories. When this information is incorporated, as in our approach, categorization performance substantially improves. Our performance on Caltech-256, for N_c=30, is lower by 7.3% if confidences in subcategory detections, $\boldsymbol{x}$, are defined using the likelihoods $x_i = P(\boldsymbol{\psi}_v|i)$ instead of the posterior of i. These results quantify the value of computing the subcategory priors and relevances for image categorization.

Fig. 5 shows that our approach is largely insensitive to a specific choice of kernel width σ. Also, Fig. 5 demonstrates that larger σ values yield faster convergence rates,

Figure 3. Samples from Caltech-256, labeled as (left-to-right): baseball-bat, computer-monitor, covered-wagon, ladder, and rainbow. These images contain subcategories that also appear in the definitions of categories: people, palm-tree, car-tire, dog, and waterfall. Subcategory sharing makes categorization challenging.

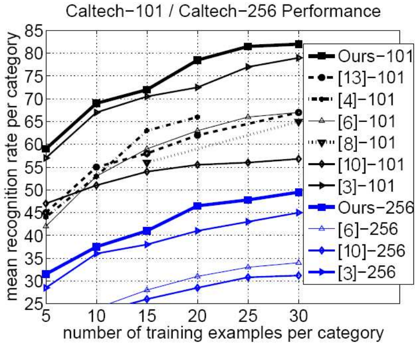

Table 2. Accuracy gain α in (%)

	Cal-101	Cal-256
[3]	4.5	9.6
[4]	23.2	NA
[13]	25.6	NA
[6]	26.2	51.1
[10]	37.9	57.9

Figure 4. Caltech-101 and Caltech-256: Our recognition rates and accuracy gain α vs. [3, 8, 4, 13, 6, 10].

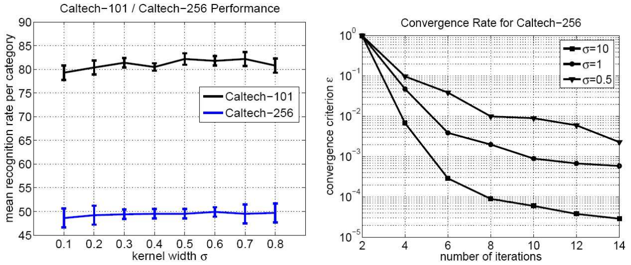

Figure 5. Recognition and convergence rates for varying kernel widths σ. $\epsilon = \|\boldsymbol{w}^{(t+1)} - \boldsymbol{w}^{(t)}\|$. If the algorithm converges, the classification accuracy is insensitive to a specific choice of σ. The convergence rate is faster for larger kernel widths. N_c=30.

as stated by Theorem 1 (Appendix II). Fig. 6 shows examples of regions occupied by subcategories that are found to be the most and least significant to recognition. As can be seen, our approach is capable of segmenting the most relevant regions responsible for the category label of the image.

7. Conclusion

In this paper, we have demonstrated that using subcategories as features of image categories allows efficient learning of a margin-based, linear classifier, which yields superior image categorization performance than seen in prior work. Recognition using the proposed linear classifier simultaneously achieves category segmentation, by virtue of identification of all the subcategories in the image that contribute to category recognition. While this paper considers only one label per image, our problem statement allows that a training image may contain occurrences of multiple label categories. Thus, the generalization of our approach to multiple labels per image is reasonably straightforward.

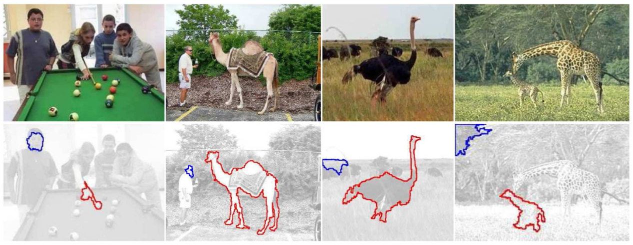

Figure 6. Samples from Caltech-256: The use of regions as features allows us to segment the occurrences of the (relevant/irrelevant) subcategories in the image. Regions with the largest weight (red) and the smallest weight (blue) for the images in categories: billiards, camel, ostrich, and giraffe.

Acknowledgment

The support of the Office of Naval Research under grant N00014-06-1-0101 is gratefully acknowledged.

Appendix I: M-step of the Learning Algorithm

The detailed proof that expression (8) has a closed-form solution is presented in [9]. We here briefly review this theoretical result, for completeness. The M-step of Alg. 1 reads $\max_{\boldsymbol{w}} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{z}$, s.t. $||\boldsymbol{w}||^2 = 1$, $\boldsymbol{w} \geq \mathbf{0}$. Its Lagrangian is $L = -\boldsymbol{w}^{\mathrm{T}}\boldsymbol{z} + \lambda(||\boldsymbol{w}||^2 - 1) + \sum_{i=1}^{|T|} \zeta_i(-w(i))$, where λ and $\boldsymbol{\zeta} \geq 0$ are the Lagrangian multipliers. Taking the derivative of L with respect to $\boldsymbol{w}$, and setting it to zero gives

$$\partial L / \partial \boldsymbol{w} = -\boldsymbol{z} + 2\lambda \boldsymbol{w} - \boldsymbol{\zeta} = 0 \;\Rightarrow\; \boldsymbol{w} = \frac{\boldsymbol{z} + \boldsymbol{\zeta}}{2\lambda}. \qquad (9)$$

To derive a closed-form solution, we make the assumption that there exists one node $i \in T$ for which $z(i) > 0$. From (7), this assumption is very weak, since the converse (i.e., $\boldsymbol{z} < 0$) would mean that there are more trees in $\mathbb{D}$ closer to their misses than trees that are closer to their hits. This case is very unlikely to occur in real applications with large datasets as we here use. This assumption has also been used for a number of machine learning algorithms that make decisions based on the distances of patterns and their neighbors, such as RBF, and SVM with RBF kernel. Given this assumption, we prove that $\lambda > 0$. Suppose the converse is true, i.e., $\lambda < 0$. Since there exists $z(i) > 0$, then $z(i) + \zeta(i) > 0$. It follows from (9) that $w(i) < 0$, which contradicts the constraint $\boldsymbol{w} \geq 0$. From the Karush-Kuhn-Tucker condition, namely $\sum_i \zeta(i) w(i) = 0$, we have the following three cases: (1) $z(i) = 0 \Rightarrow \zeta(i) = 0 \Rightarrow w(i) = 0$; (2) $z(i) < 0 \Rightarrow \zeta(i) > 0 \Rightarrow w(i) = 0$; and (3) $z(i) > 0 \Rightarrow z(i) + \zeta_v > 0 \Rightarrow w(i) > 0 \Rightarrow \zeta(i) = 0 \Rightarrow w(i) = \frac{z(i)}{2\lambda}$. It immediately follows that the expression for computing $\boldsymbol{w}$ has a closed form given by (8).

Appendix II: Convergence Analysis

This section briefly reviews the convergence analysis of Alg. 1, which is presented in [9]. We begin by studying its asymptotic behavior:

$$\forall \boldsymbol{x}, \lim_{\sigma\to+\infty} P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}} = 1/|\mathbb{M}_{\boldsymbol{x}}| \ , \forall \boldsymbol{x}'{\in}\mathbb{M}_{\boldsymbol{x}} \ , \tag{10}$$

since $\lim\limits_{\sigma\to+\infty} k(d){=}1$. Also, assuming that for every $\boldsymbol{x}$, $d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}){\neq}d_{\boldsymbol{x}\boldsymbol{x}''}(\boldsymbol{w})$ if $\boldsymbol{x}'{\neq}\boldsymbol{x}''$, we have

$$\lim_{\sigma\to 0} P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}(\boldsymbol{w}){=}\begin{cases} 1, \text{ if } d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}){=}\min\limits_{\boldsymbol{x}''\in\mathbb{M}_{\boldsymbol{x}}} d_{\boldsymbol{x}\boldsymbol{x}''}(\boldsymbol{w}) \\ 0, \text{ if } d_{\boldsymbol{x}\boldsymbol{x}'}(\boldsymbol{w}){>}\min\limits_{\boldsymbol{x}''\in\mathbb{M}_{\boldsymbol{x}}} d_{\boldsymbol{x}\boldsymbol{x}''}(\boldsymbol{w}). \end{cases} \tag{11}$$

Similar asymptotic behavior holds for $P_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}}$. Thus, for $\sigma{\to}{+}\infty$ the algorithm converges to a unique solution in one iteration, and for $\sigma \to 0$ rarely do we empirically observe that the algorithm converges. This suggests that the algorithm's convergence is fully controlled by the kernel width, which is formally stated in the following theorem.

Theorem 1 *There exists $\sigma^*{>}0$ such that for any kernel width $\sigma{>}\sigma^*$ Alg. 1 converges, $\lim\limits_{\tau\to+\infty} ||\boldsymbol{w}^{(\tau+1)}{-}\boldsymbol{w}^{(\tau)}||{=}0$, where $\boldsymbol{w}^{(\tau)} \in \mathbb{W} = \{\boldsymbol{w} : \boldsymbol{w}{\in}\mathbb{R}^{|\mathbb{I}|}, \|\boldsymbol{w}\|{=}1, \boldsymbol{w}{\geq}0\}$. Moreover, for a fixed $\sigma{>}\sigma^*$, Alg. 1 converges to a unique solution for any initial $\boldsymbol{w}^{(0)}{\in}\mathbb{W}$.*

Theorem 1 can be proved by using the Banach theorem, stated below without proof. While there exist simpler ways of proving Theorem 1, the main advantage of using the Banach theorem is that it also allows establishing the conditions for achieving faster convergence rate.

Definitions: Let $\mathcal{U}$ be a subset of a normed space $\mathcal{Z}$, with norm $\|\cdot\|$. Operator $T : \mathcal{U}{\to}\mathcal{Z}$ is called a contraction operator if there exists a constant $q{\in}[0,1)$ such that $\|T(x){-}T(y)\|{\leq}q\|x{-}y\|, \forall x, y{\in}\mathcal{U}$. q is called the contraction number of T. An element of a normed space $\mathcal{Z}$ is called a fixed point of $T : \mathcal{U}{\to}\mathcal{Z}$ if $T(x){=}x$.

Banach Theorem: Let T be a contraction operator mapping a complete subset $\mathcal{U}$ of a normed space $\mathcal{Z}$ into itself. Then the sequence generated as $x^{(\tau+1)} = T(x^{(\tau)})$, with arbitrary $x^{(0)} \in \mathcal{U}$, converges to the unique fixed point x^* of T. Moreover, the following estimation error bounds hold: $\|x^{(t)} - x^*\| \leq \frac{q^t}{1-q}\|x^{(1)} - x^{(0)}\|$, and $\|x^{(t)} - x^*\| \leq \frac{q}{1-q}\|x^{(t)} - x^{(t-1)}\|$.

Proof of Theorem 1: The proof identifies a contraction operator of Alg. 1, and makes sure that the conditions of the Banach theorem are met. Let us define $\mathbb{P}{\triangleq}\{(P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}, P_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}})\}$, $\forall \boldsymbol{x}{\neq}\boldsymbol{x}'$ from the training set $\mathbb{D}$. Then E-step and M-step of Alg. 1 can be specified in a functional form as E:$\mathbb{W}{\to}\mathbb{P}$, and M:$\mathbb{P}{\to}\mathbb{W}$, i.e., Alg. 1 can be expressed as $\boldsymbol{w}^{(\tau+1)}{=}(\text{E}{\circ}\text{M})(\boldsymbol{w}^{(\tau)}){=}T(\boldsymbol{w}^{(\tau)})$, where $(\circ)$ denotes functional composition, and $T{:}\mathbb{W}{\to}\mathbb{W}$. Since $\mathbb{W}$ is a closed subset of normed space $\mathbb{R}^{|\mathbb{I}|}$ and complete, T is an operator mapping complete subset $\mathbb{W}$ into itself.

Recall that $\lim_{\sigma\to+\infty} P_{\boldsymbol{x}'=\boldsymbol{m}_{\boldsymbol{x}}}{=}|\mathbb{M}_{\boldsymbol{x}}|^{-1}$, $\forall \boldsymbol{w}{\in}\mathbb{W}$, and that similar asymptotic behavior holds for $P_{\boldsymbol{x}'=\boldsymbol{h}_{\boldsymbol{x}}}$. Then, $\lim_{\sigma\to+\infty}[T(\boldsymbol{w}', \sigma){-}T(\boldsymbol{w}'', \sigma)]{=}\boldsymbol{0}$, $\forall \boldsymbol{w}', \boldsymbol{w}''{\in}\mathbb{W}$. Since the 2-norm is a continuous function, it follows that $\lim_{\sigma\to+\infty} ||T(\boldsymbol{w}', \sigma){-}T(\boldsymbol{w}'', \sigma)||{=}0$, $\forall \boldsymbol{w}', \boldsymbol{w}''{\in}\mathbb{W}$. Therefore, in the limit, T is a contraction operator with contraction constant $\lim\limits_{\sigma\to+\infty} q(\sigma){=}0$. Consequently, for every $\varepsilon{>}0$, there exists σ^* such that $q(\sigma){\leq}\varepsilon$, whenever $\sigma{>}\sigma^*$. By setting $\varepsilon{<}1$, the resulting T is a contraction operator. From the Banach theorem, it immediately follows that Alg. 1 converges to a unique fixed point, provided σ is sufficiently large. Also, the error bound in the Banach theorem guarantees that a larger σ yields a faster convergence rate. □

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE TPAMI*, 26(11):1475–1490, 2004.

[2] N. Ahuja and S. Todorovic. Learning the taxonomy and models of categories present in arbitrary images. In *ICCV*, 2007.

[3] A. Bosch, A. Zisserman, and X. Munoz. Image classification using random forests and ferns. In *ICCV*, 2007.

[4] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *ICCV*, 2007.

[5] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *ICCV*, volume 2, pages 1458–1465, 2005.

[6] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007.

[7] K. Kira and L. A. Rendell. A practical approach to feature selection. In *ICML*, pages 249256, vol.1, 1992.

[8] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[9] Y. Sun. Iterative RELIEF for feature weighting: Algorithms, theories, and applications. *IEEE TPAMI*, 29(6), 2007.

[10] S. Todorovic and N. Ahuja. Extracting subimages of an unknown category from a set of images. In *CVPR*, 2006.

[11] A. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *CVPR*, volume 2, pages 762–769, 2004.

[12] M. Vidal-Naquet and S. Ullman. Object recognition with informative features and linear classification. In *ICCV*, pages 281–288, vol.1, 2003.

[13] H. Zhang, A. Berg, M. Maire, and J. Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *CVPR*, pages 2126–2136, 2006.