# An Unsupervised Hierarchical Feature Learning Framework for One-Shot Image Recognition

Zhenyu Guo, and Z. Jane Wang, *Senior Member, IEEE*

***Abstract*—One-shot recognition has attracted increasing attention recently, inspired by the fact that human cognitive systems could perform recognition tasks well provided only one or a few labeled training samples, in contrast to the conventional object recognition systems that require a large number of labeled training images. One-shot recognition is a visual classification task, where only one training sample is available for each object category in the target test domain, with the help of prior-knowledge data from the source domain. In this paper, we tackle this challenging one-shot recognition problem under a more exciting setting by using only unlabeled images as prior-knowledge, which requires less labeling efforts than previous works which adopt fully labeled data and/or a sophisticated attribute table designed by human experts. We propose a novel unsupervised hierarchical feature learning framework to learn a feature pyramid from the prior-knowledge domain. The proposed feature learning method also could be applied across multiple feature spaces. Furthermore, we propose using pyramid matching kernels to combine multi-level features. Examining the "Animals with Attributes" and Caltech-4 data sets in our one-shot recognition setting, we show that the proposed unsupervised feature learning approach with very limited information could achieve comparable performance with that of supervised ones.**

***Index Terms*—object recognition, deep structure, hierarchical feature learning, Dirichlet Process, feature combination, pyramid matching**

## I. Introduction

Object recognition using computer vision methods has gone through considerable progress during the last decade, including methods based on low level features (e.g., Scale-Invariant Feature Transform(SIFT) [1], Speeded Up Robust Feature (SURF) [2], pyramid Histogram of Oriented Gradients (pHOG) [3], and Self Similarity [4]) and specific designed machine learning techniques. Numerous papers [5] have shown that recognition accuracy generally increases as the number of training samples per category increases. However, a large number of labeled training samples might not be feasible in practice.

However, there are significant evidences that human beings can perform category-level object recognition in a more efficient way, by learning novel concepts from only one or a few exemplars. Motivated by such recognition ability of human cognitive systems, "one-shot" recognition [6], where the system is given only one training sample for each object category, has attracted increasing research attention very recently. In the computer vision community, the **one-shot recognition** task consists of data from two domains: the target domain and the prior-knowledge domain. The target domain, where we actually perform the one-shot classification, consists of data from *target categories* with only one training sample in each category. It is assumed that the prior-knowledge domain consists of data from categories that are *different* from the target categories. Based on previous works [6]–[9], as shown in Figure 1, the one-shot recognition procedure generally contains two major components: 1) feature learning in the prior-knowledge domain, and 2) supervised classification in the target domain with only one labeled training sample per category.

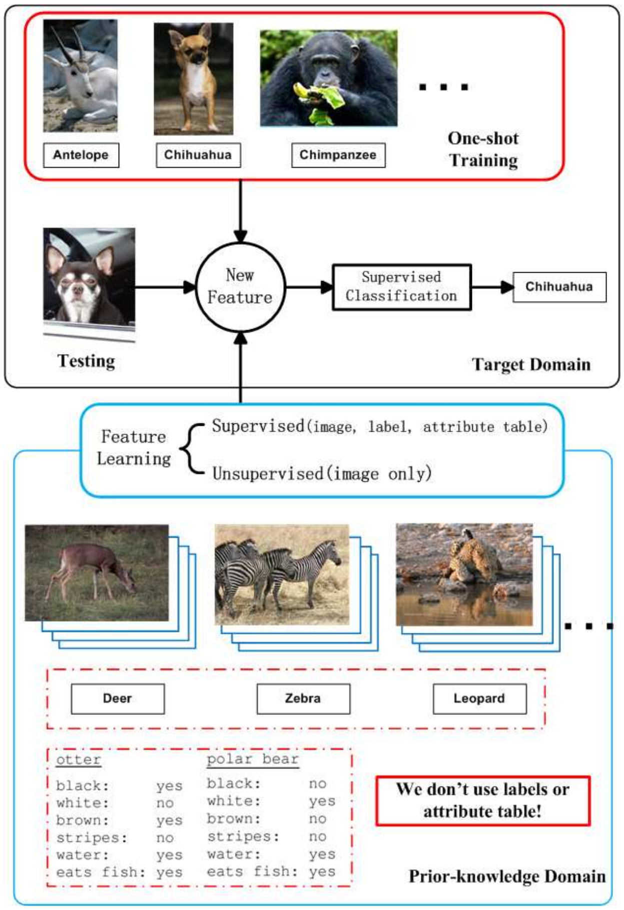


Fig. 1. One-shot recognition framework: Only one training image with label is provided for each category in the target domain. The prior-knowledge domain contains images from categories that are different from the target categories. Unlike previous works which usually use labeled images and the designed attribute-table, we only use unlabeled images in the prior-knowledge domain.

Zhenyu Guo and Z. Jane Wang are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada.
E-mail: {zhenyug,zjanew}@ece.ubc.ca

Since there is only one exemplar for each category in the target domain, a standard classifier (e.g. Nearest Neighbor [8], Naive Bayesian [7] and Support Vector Machine (SVM) [10]) is usually chosen for the supervised classification step. Previous methods for one-shot recognition [6]–[9] are different mainly in the settings of the feature learning step in the prior-knowledge domain. [7] uses label information for all images from the prior-knowledge domain, along with a sophisticated attribute table designed by human experts. [8] doesn't use any manually designed attributes, but it still relies on the fully labeled images. This similar setting was also adopted by a recent work [9]. Although considerable progress has been achieved by the methods mentioned above, the required side information (e.g., the large number of class labels and the manually designed attribute table) in the prior-knowledge domain can be difficult to acquire in practice. Recalling the original motivation mentioned earlier that human cognitive systems are able to adapt useful information from prior-knowledge without the help of such side information, we propose using only the **unlabeled** images as prior-knowledge, as illustrated in Fig 1.

To intuitively justify that the proposed setting of using unlabeled images as prior-knowledge for one-shot learning is feasible, in Figure 2, we take a "zebra-horse" problem as an example. It is a simple task of classifying a test image into category "zebra" or "horse". If we know the concept of "striped" texture pattern, we can easily distinguish zebras from horses by generating this classification rule. If we could learn this "striped" pattern from the " zebra-horse" data set, we can also use this simple classification rule to distinguish tigers from leopards. From this example, we believe that such meaningful features are shared among relevant categories and can significantly benefit category-level recognition. There are evidences [11] showing that human beings can learn from *unlabeled data* through manifold regularization, indicating that human beings can learn more meaningful features (for category-level recognition) from daily experiences. In the zebra-horse example, it is desirable to learn the "striped" feature (and other abstract attributes) from unlabeled images of zebra and horse. Therefore, how to learn the meaningful features is a challenging but valuable task.

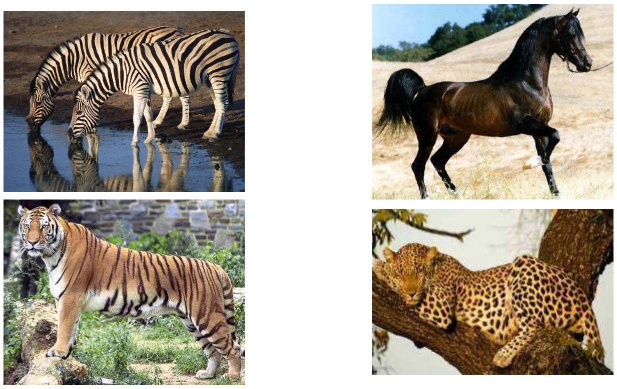


Fig. 2. The animal pairs, zebra and horse, and tiger and leopard, can be classified by the "striped" pattern.

Since many low-level visual descriptors (e.g., SIFT) have already showed good discriminative power in visual recognition tasks, we plan to incorporate the advantages of these handcrafted low-level features. In this paper, we formulate feature learning from prior-knowledge as a latent mixture modeling problem, which is to learn mixture components over the low-level descriptors as more meaningful features from unlabeled images. We propose using Hierarchical Dirichlet Process (HDP) [12] for this purpose. Since the feature learning process is actually a clustering operation on the histograms of base features, we call it the HDP-encoder which can encode low-level features' histograms into higher level feature representations automatically, as explained later in Section III-D and Figure 7.

Now suppose we have the HDP-encoder and its output is also a histogram vector based on the higher level features. We assign the low-level descriptor as $level-0$ (L0) and the new features as $level-1$ (L1). From WordNet [13] in the natural language processing community, a large lexical database of English, we note that human language has a hierarchical structure to describe objects and events from holistic aspects to details. Also, in the object recognition community, spatial pyramid matching [14] shows the matching power of "coarse to fine" in the 2-D image spatial domain. Inspired by the hierarchy observed in human language and in image spatial pyramid subdivision, we propose constructing a deep structure of the feature pyramid by stacking the HDP-encoders layer by layer, where each layer has a unique "describing resolution". The details will be explained in Section III-D (Figure 7) and Section IV-C (Figure 10). From top to down, the pyramid provides image representations from "coarse to fine", where a higher level captures more "macro" information while a lower level captures more "detail" information. It is worth noting that HDP-encoders can be stacked across feature spaces (e.g., the texture-L2 feature may be learned from SIFT-L1 and SURF-L1 features). The joint feature vectors could be viewed as histograms generated from a large joint dictionary.

Based on the obtained feature-pyramid, how to transfer such rich descriptions into classification power is our next concern. Similar to spatial pyramid matching [14], we choose to use weighted summation of intersection kernels to combine the features at different levels. In addition, since the proposed HDP-encoder could also model the latent components across different feature spaces, we can learn a single feature that can capture information from multiple lower levels of features and is sufficiently compact for real world applications. Therefore, the proposed feature learning algorithm can also be viewed as a novel feature combination method. In summary, the main contributions of this paper are as follows:

1) We propose a novel feature learning algorithm based on HDP modeling, which can encode low-level image descriptors into a high level feature vector from unlabeled images in the prior-knowledge domain, as illustrated in Section III-D (Figure 7).
2) We propose a deep structure for feature learning by stacking the HDP encoders to learn a feature pyramid with multiple "describing resolutions", as illustrated in SectionIII-D (Figure 7) and Section IV-C (Figure 10).
3) We evaluate the proposed unsupervised feature learning

framework on one-shot image recognition tasks and show comparable performances to that of previous supervised feature learning methods [7]–[9], [15].

4) We evaluate the proposed framework on conventional multi-shot tasks and show the recognition improvements.

In the remainder of this paper, Section II provides a brief review of previous work on one-shot recognition and feature combination. Section III describes the proposed unsupervised hierarchical feature learning framework. Section IV demonstrates the performances of the proposed method on real data sets. Section V concludes the paper.

This paper expends upon our conference paper [10], in which we investigated learning a high level feature from a single type of local descriptor. In this paper, we propose a novel deep structure for feature learning based on multiple types of local descriptors and propose a new feature combination scheme using feature pyramid and averaging kernels. We also conduct more experiments.

## II. Related Works

Compared with the conventional multi-shot visual recognition, there has been relatively less work in the area of one-shot recognition. The concept "one-shot learning" was first introduced in [6]. [6] propose a Bayesian framework with a class prior learned from labeled prior-knowledge. [7] tackles the challenge by incorporating human specified high level attributes, where a number of supervised classifiers are used to associate bag-of-feature representations with the binary attributes. Their proposed cascade recognition system provides the state-of-art recognition accuracy on the "Animal with Attributes" data set. To learn the semantic attributes automatically from the prior-knowledge data, a nonlinear mapping based method is used [8] to learn a mapping function by optimizing the discriminative power of the intermediate representation, where the mapping function can be viewed as a projection from the original feature basis onto higher level latent attributes. Their results on multi-class one-shot recognition are better than the simple naive Bayesian method in [7]. [8] still requires a large number of labeled images as prior-knowledge. Later, similar to the setting in [7], the work in [9] focuses on the attributes used in the attribute table and designs a better knowledge transfer scheme by modeling the attribute priors. [9] can be considered as a better way of associating image features with the manually designed binary attributes.

There are several research works in other areas which emphasize weak supervision, not specific to one-shot learning. [16] exploits side information from pairs of object images labeled as "same" or "different" to learn a metric for measuring similarity between unseen object images. Metric learning approaches like [17] require at least weak supervision on the prior-knowledge data, which cannot be obtained in our problem. So far, the closest work to our paper is self-taught learning [18], which uses sparse coding to learn a set of bases for the linear combination to approximate the image data in the prior-knowledge domain. The weights of the bases are used as features to represent images. In [18], sparse coding is performed on image pixel patches and cannot be applied directly to the discrete space of the histogram of low-level features (in the bag-of-feature representation). By contrast, our proposed method is based on the handcrafted image descriptors and thus it is suitable to learn a higher level feature without losing the nice property of low-level features. Another area in machine learning that seems be related to our one-shot problem is semi-supervised learning (SSL) [19]–[21], where unlabeled data are used to improve the weakly supervised classification tasks. As we stated in Section I, one-shot recognition generally contains two separated components: 1) feature learning in the prior-knowledge domain, and 2) supervised classification in the target domain with only one labeled training sample per category. Since there are no labeled images in the prior-knowledge domain in our setting, there is no room for semi-supervision in the prior-knowledge data. In addition, it is not feasible to pool the unlabeled prior-knowledge data and target one-shot training data together to run a SSL method during the classification. Usually SSL methods [19], [20] assume that the unlabeled training data come from the same categories as the labeled training data, while it is not true in our one-shot recognition setting where the images in the prior-knowledge domain generally come from categories that are different from the categories in the target domain.

To our knowledge, this paper is the first work that attempts to learn a deep structure of the feature pyramid based on low-level image descriptors in the area of one-shot recognition. Note that the idea of describing an object image in a "coarse to fine" way has a long history, also reflected in the design of low-level descriptors. [14] extended the pyramid matching idea [22] to the spatial domain by matching the feature points in different spatial subdivisions. Since the proposed feature pyramid provides multiple "describing resolutions" which are similar to the "spatial resolutions" in [14], we will adopt the weighted intersection kernels to combine different features learned in the feature-learning step.

## III. The Proposed Method

In this section, we will first formulate our unsupervised feature learning problem for unlabeled prior-knowledge data, then describe the proposed feature combination method and the supervised classifier for one-shot recognition. For the unsupervised feature learning step, we propose using a Hierarchical Dirichlet Process. By wrapping up the feature learning process into the HDP-encoder, we propose a deep structure to construct the feature pyramid in two ways: One emphasizes the recognition performance, and the other emphasizes the efficiency and compactness of the learned features. For the supervised classification in the target domain, we present how to incorporate intersection kernels and the average kernel to combine all features in the pyramid, and then a standard Support Vector Kernel Machine is used for classification.

### A. Problem Formulation

We denote the data set in the target domain $\mathcal{T}$ as $X_{\mathcal{T}}$, with data points $\{x_1, x_2, \dots, x_{n_{\mathcal{T}}}\}$, and the data set in the prior-knowledge domain $\mathcal{P}$ as $X_{\mathcal{P}}$, with data points $\{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{n_{\mathcal{P}}}\}$. The data in $X_{\mathcal{T}}$ comes from $M$ categories $C_{\mathcal{T}} = \{c_1, c_2, \ldots, c_M\}$, and the data in $X_{\mathcal{P}}$ comes from categories $C_{\mathcal{P}} = \{c_{M+1}, c_{M+2}, \ldots, c_S\}$. It is worth noting that $C_{\mathcal{T}}$ and $C_{\mathcal{P}}$ are disjoint, which makes one-shot learning a problem different from semi-supervised learning. In one-shot recognition tasks, the training data from $X_{\mathcal{T}}$ is $X_{train_{\mathcal{T}}}$ with category labels $Y_{train_{\mathcal{T}}}$. The prior-knowledge data is denoted as $X_{train_{\mathcal{P}}}$ without corresponding category labels. We want to learn latent features based on $X_{train_{\mathcal{P}}}$, and to project the target training data $X_{train_{\mathcal{T}}}$ into the learned feature space, which is denoted as $\hat{X}_{train_{\mathcal{T}}}$. At last, a supervised classifier is trained on $\{\hat{X}_{train_{\mathcal{T}}}, Y_{train_{\mathcal{T}}}\}$. In the testing phase, the trained classifier is used to predict the labels of projected test data $\hat{X}_{test_{\mathcal{T}}}$ to get $Y_{predict}$. The framework can be described by major steps in Table I, where step 1 is described in Section III-B and Section III-D, and steps 2 and 4 are described in Section III-C and Section III-E, and steps 3 and 5 are described in Section III-F.

**Algorithm 1** One-shot Recognition Framework

```
1: HDP-encoder        ←  HDP modeling on $X_{train_{\mathcal{P}}}$,
2: $\hat{X}_{train_{\mathcal{T}}}$  ←  Project $X_{train_{\mathcal{T}}}$ onto the latent feature space
                          using HDP-encoder,
3: $Classifier$       ←  Train a SVM on $\{\hat{X}_{train_{\mathcal{T}}}, Y_{train_{\mathcal{T}}}\}$ ,
4: $\hat{X}_{test_{\mathcal{T}}}$   ←  Project $X_{test_{\mathcal{T}}}$ onto the latent feature space
                          using HDP-encoder,
5: $Y_{predict}$      ←  Predict the labels of $\hat{X}_{test_{\mathcal{T}}}$ using $Classifier$.
```

TABLE I
MAJOR STEPS FOR THE PROPOSED ONE-SHOT RECOGNITION SYSTEM.

Furthermore, we briefly discuss data representation in general object recognition problems. A standard way to represent object images is to use the bag-of-feature model, which was originally borrowed from the document modeling area. In the bag-of-feature model, the low-level image descriptors serve as visual words, a codebook or dictionary is computed by clustering the total samples of visual words, then images are represented by the occurrence histograms of the visual words in the dictionary. We use $\mathbf{V} = \{w_1, w_2, \ldots, w_i, \ldots, w_d\}$ to denote the dictionary with vocabulary size $d$, where $w_i$ means the $i$th visual word. For an image $I$, we use its histograms of the dictionary visual words $\mathbf{h} = (h_1, h_2, \ldots, h_i, \ldots, h_d)$ to represent it in the bag-of-feature model, where $h_i$ means the occurrence frequency of $w_i$ in the image $I$. The histogram vector is based on the low-level features and can be used as training and testing data in classifiers. Considering that each image is represented by a histogram vector, our goal here is to learn a higher level feature by fitting a mixture model to the grouped feature data (i.e., histograms of the images).

Let us denote the L0 level feature dictionary as $\mathbf{V_0} = \{w_1, w_2, \ldots, w_i, \ldots, w_d\}$ and the L1 level feature dictionary as $\mathbf{V_1} = \{z_1, z_2, \ldots, z_i, \ldots, z_l\}$. Here a higher level visual word $z_i$ may be a mixture component of $w_i$'s with different proportions. We could see that a low level feature word may belong to different higher level features with different probabilities according to its group statistics. Since $w_i$'s are exchangeable across different groups, we also need $z_j$'s to be shared among all groups. Through the feature learning method, we could obtain $p(z_j|w_i)$, the probability that a low-level feature belongs to a higher level feature. We also can obtain the new histogram of the image $I$ based on the L1 dictionary $\mathbf{V_1}$ by normalizing the posterior $p(z_j|\mathbf{I})$.

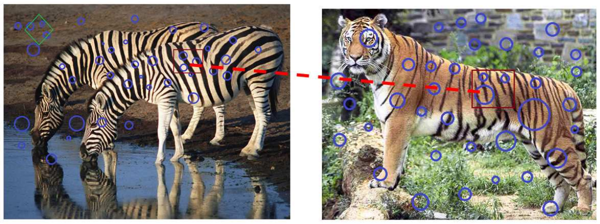


Fig. 3. The blue circles indicate local patches which the low-level descriptors are generated from, and the squares are local clusters of descriptors within each image. Those two red squares linked by a dotted line belong to the same global cluster across image groups.

We therefore need a method that could solve the mixture modeling problem from one lower level to the next higher level. Since the new features are homogeneous with the lower level ones they are learned from, it is desirable to be able to apply the feature learning method layer by layer to construct a feature pyramid. We propose using Hierarchical Dirichlet Process (HDP) [12] as the feature encoder in the proposed unsupervised feature learning approach. We will explain our choice of HDP and describe the details of HDP shortly.

### B. *Hierarchical Dirichlet Process Mixture Model*

Our idea is to assemble related descriptors (lower level features) into mixture components as higher level features. In Fig 3, the blue circles indicate local patches which the low-level descriptors are generated from, and the squares are clusters on the descriptors as higher level features. Since we use the bag-of-feature representation, we choose the latent topic model in the document modeling community to find the latent mixture components. After comparing with parametric latent topic models such as Latent Dirichlet Allocation (LDA) [23] and Probabilistic Latent Semantic Analysis (pLSA) [24], we adopt Hierarchical Dirichlet Process (HDP), a nonparametric generative model, for our feature learning task. As an infinite mixture model, HDP provides a way to sample an unbounded number of latent mixture components for grouped data, which means HDP can find the number of the mixture components and the data points related to each component automatically. This property is highly desirable for our feature learning task, since there is no easy way to predetermine the number of mixture components. Although Dirichlet Process (DP) is also a nonparametric infinite mixture model and is easier to sampling, we don't choose DP because DP is suitable for mixture modeling in non-grouped data (all the data in a single group), but can't be applied to grouped data. As illustrated in Fig 3, the squares connected by the dotted line indicate the same latent component ("striped pattern") shared by two individual groups. In contrast to DP, HDP has the clustering property to model the latent components shared among groups. Before describing the details of HDP, we first define some notations:

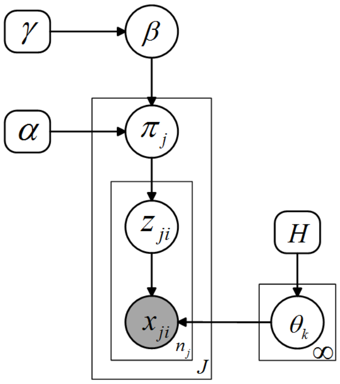


Fig. 4. The graphical model of HDP with auxiliary variables. $x_{ji}$ is the $i$th observation (visual word) in group $j$ ( image $j$), and $z_{ji}$ is the mixture component indicator associated with $x_{ji}$. $\pi_j$ is the prior distribution on mixture components, which follows a Beta distribution controlled by the concentrating parameter $\alpha$ and the stick-breaking random variable $\beta$. $\beta$ follows a Beta distribution controlled by the parameter $\gamma$. $\theta_k$ controls the distribution over the observation $x_{ji}$ and $H$ is the Dirichlet Prior distribution on $\theta_k$.

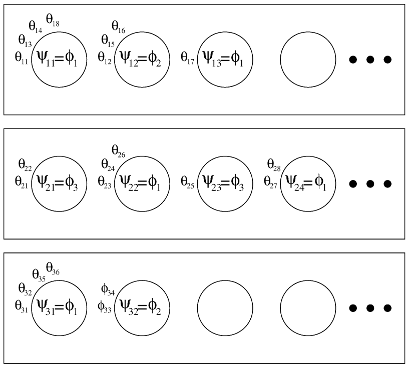


Fig. 5. Illustration of the clustering property of the Chinese Restaurant Franchise [12]. Tables $\psi$'s, as the local clusters of customers $\theta$'s, are linked by dishes $\phi$'s to form global clusters across the restaurants.

1) A feature dictionary $\mathbf{V}$ at a low-level. $\mathbf{V} = \{w_1, w_2, \ldots, w_d\}$, where each entry is a dictionary visual word.

2) An image is a group of visual feature data and represented by orderless visual words denoted as $\mathbf{x_j} = (x_{j1}, x_{j2}, \ldots, x_{jN})$, where $x_{ji}$ is corresponding to an instance of $i$th visual word in the $j$th image. Note that although we use bag-of-feature histogram to represent an image, the $x_{ji}$ here is the indicator for certain dictionary word in $\mathbf{V}$, not the frequency.

We represent an image as a group of orderless visual words, as defined in the bag-of-feature model. We assume that there exist latent mixture components corresponding to clusters of low level visual words with related attributes. We also assume that such latent mixture components are shared among different images. We therefore need to study the latent components and the component memberships of the visual words. In order to model the images with better describing ability, we construct a new visual dictionary based on the learned latent components, and encode the images with the new dictionary. To serve this learning purpose, we use HDP to model the unlabeled images in the prior-knowledge domain in an unsupervised way. The graphical model of HDP with auxiliary variables is showed in Fig 4. In HDP model, $x_{ji}$ means the $i$th visual word in image $j$, $z_{ji}$ is the indicator variable (index) associated with a mixture component and $z_{ji}$ has discrete values on $\{1, 2, \ldots\}$. $\theta$ is the factor associated with the distribution of $x_{ji}$ given each $z_{ji}$. Referring to Fig 4, we now show how to generate $x_{ji}$ for image $j$.

1) Sample $\beta \sim GEM(\gamma)$, where GEM is a distribution designed from stick-breaking construction of Dirichlet Process:

$$\begin{aligned} \beta_k^{'} \sim beta(1,\gamma), \beta_k = \beta_k^{'} \prod_{l=1}^{k-1}(1-\beta_l^{'}), \\ \beta = (\beta_1, \beta_2, \ldots, \beta_\infty). \end{aligned} \tag{1}$$

2) Sample $\theta_k$ from the Dirichlet prior's base distribution $H$.
3) Generate the group $j$ (image $j$) by the following steps:
   a) Sample $\pi_j$ from
   $\pi_j|\alpha,\beta \sim DP(\alpha,\beta)$ by the construction:

$$\begin{aligned} \pi_{jk}^{'} \sim beta(\alpha\beta_k, \alpha(1-\prod_{l=1}^{k}\beta_l)), \\ \pi_{jk} = \pi_{jk}^{'} \prod_{l=1}^{k-1}(1-\pi_{jl}^{'}). \end{aligned} \tag{2}$$

   where $\pi_j = (\pi_{j1}, \pi_{j2}, \ldots, \pi_{j\infty})$.
4) Given $\pi_j$, generate $x_{ji}$ by the following steps:
   a) Sample component indicator $z_{ji}$ from a multinomial distribution $\pi_j$ : $z_{ji}|\pi_j \sim \pi_j$.
   b) Sample $x_{ji}$ given $z_{ji}$ and $\theta_k$ from a multinomial distribution $F(\theta_{z_{ji}})$: $x_{ji}|z_{ji},\theta_k \sim F(\theta_{z_{ji}})$.

To estimate the HDP model, among those Markov Chain Monte Carlo Sampling schemes, Chinese Restaurant Franchise (CRF) is probably the most intuitive one, which also can illustrate the clustering property of HDP over grouped data. Before going into details of CRF, we first introduce the CRF metaphor.

In CRF, there are multiple restaurants with an unbounded number of tables. A customer comes into a restaurant and chooses a table to sit at, and there is a shared menu across the restaurants and one dish is ordered from the menu by the first customer who sits at that table. Tables within each restaurant play the role of local clusters within one group, over the customers sitting at them. The dishes shared across the restaurants are used to link all the tables together to get mixture components over all data points in different groups. Table II shows the notations and the process of CRF.

As an analogy to the CRF model in Table II, in our one-shot recognition problem, we treat each image as one restaurant, a visual word in that image as customer $x_{ji}$ coming in and a local cluster (within the image) as table $t_{ji}$ where $x_{ji}$ sitting at. To link the tables in different restaurants, we use dish $k_{jt}$ serving at table $t$ in restaurant $j$ as the indicator of global mixture component shared across all images. The distributions of $t_{ji}$ and $k_{jt}$ given previous random variables are given here:

| Chinese Restaurant Franchise |
|---|
| $\theta_{ji}$ : the $i_{th}$ customers in restaurant $j$. $\phi_k$ : dishes in the global menu. $\psi_{jt}$ : the dish served at table $t$ in restaurant $j$.<br>$t_{ji}$ : the index of the $\psi_{jt}$ associated with $\theta_{ji}$. $k_{jt}$ : the index of $\phi_k$ associated with $\psi_{jt}$ |
| The metaphor will be:<br>customer $i$ in restaurant $j$ sits at table $t_{ji}$, whereas table $t$ in restaurant $j$ serves dish $k_{jt}$. |
| $n_{jtk}$ : the number of customers in restaurant $j$ at table $t$ eating dish $k$.<br>$n_{jt.}$ : the number of customers in restaurant $j$ at table $t$.<br>$n_{j.k}$ : the number of customers in restaurant $j$ eating dish $k$.<br>$n_{j..}$ : the number of customers in restaurant $j$.<br>$m_{jk}$ : the number of tables in restaurant $j$ serving dish $k$.<br>$m_{j.}$ : the number of table in restaurant $j$.<br>$m_{.k}$ : the number of tables serving dish $k$.<br>$m_{..}$ : the number of tables occupied. |
| Initialization: customer $i = 1$ enters the restaurant $j$ and sits at table 1, and orders dish 1.<br>$\theta_{j1} = \psi_{j1}, n_{j11} = 1, m_{j1} = 1$<br>For $i = 2, ...,$<br>customer $i$ sits at table $\begin{cases} t & \text{with probability} \frac{n_{jt.}}{i-1+\alpha_0} \quad \text{, for } \; i = 1, 2, ..., n_{j..} \\ n_{j..} + 1 & \text{with probability} \frac{\alpha_0}{i-1+\alpha_0} \quad \text{, for new table} \end{cases}$<br>table $t$ serves dish $\begin{cases} k & \text{with probability} \frac{m_{.k}}{m_{..}+\gamma} \quad \text{, for } k = 1, 2, ..., m_{..} \\ m_{..} + 1 & \text{with probability} \frac{\gamma}{m_{..}+\gamma} \quad \text{, for new dish} \end{cases}$ |

TABLE II
PARAMETER DETAILS AND THE SAMPLING PROCEDURE OF THE CHINESE RESTAURANT FRANCHISE SCHEME.

$$t_{ji}|t_{j1}, \ldots, t_{j(i-1)}, \alpha, G_0 \sim \sum_{t=1}^{T_j} n_{jt}\delta_{t_{ji}=t} + \alpha G_0, \qquad (3)$$

$$k_{jt}|k_{11}, k_{12}, \ldots, k_{21}, \ldots, k_{jt-1}, \gamma \sim \sum_{k=1}^{K} m_k \delta_{k_{jt}=k} + \gamma H. \qquad (4)$$

where $G_0 \sim DP(\gamma, H)$. Equations (3), (4) have the same meaning as Table II, which is also the guidance for us to do sampling for HDP. Note that the number of $k_{jt}$ variables is not fixed by the algorithm, which is an important property of infinite mixture model that the mixture component space is infinite. The CRF also illustrates the clustering property of HDP, as shown in Fig. 5. After modeling the local clusters (tables) in images (restaurants), HDP also models the global clusters across all groups using table specific dishes. Finally, the dish $k_{jt}$ indicates the cluster associated with customers $x_{ji}$'s sitting at the table $t_{jt}$, which is the higher level feature we want to model.

**Sampling t**: According to Equations 3 and 4, the likelihood due to $x_{ji}$ given $t_{ji} = t$ for some previously used $t$ is $f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}})$, where $f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}})$ is the conditional probability of $x_{ji}$ given all data points except itself. The likelihood for $t_{ji} = t^{new}$ can be calculated as :

$$\begin{aligned} & p(x_{ji}|t^{-ji}, t_{ji} = t^{new}, k) \\ &= \sum_{k=1}^{K} \frac{m_{.k}}{m_{..} + \gamma} f^{-x_{ji}}(x_{ji})|\theta_{k_{ji}} + \frac{\gamma}{m_{..} + \gamma} f^{-x_{ji}}(x_{ji}|\theta_{new}). \end{aligned} \qquad (5)$$

Thus the conditional distribution of $t_{ji}$ is:

$$\begin{aligned} & p(t_{ji} = t|x_{ji}, t^{-ji}, k) \\ & \propto \begin{cases} n_{jt.}^{-ji} f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}}) & \text{if } t \text{ previously used} \\ \alpha p(t_{ji} = t|x_{ji}, t^{-ji}, k) & \text{if } t = t^{new} \end{cases}. \end{aligned} \qquad (6)$$

If the sampled value of $t_{ji}$ is $t^{new}$, then we need to assign a global cluster to this $t^{new}$. The probability for $k_{jt^{new}}$ is:

$$\begin{aligned} & p(k_{jt^{new}} = k|t, k^{-jt^{new}}) \\ & \propto \begin{cases} m_{.k} f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}}) & \text{if } k \text{ previously used} \\ \gamma f^{-x_{ji}}(x_{ji}|\theta_{k_{new}}) & \text{if } k = k^{new} \end{cases}. \end{aligned} \qquad (7)$$

If some table $t$ becomes unoccupied during the updating of $t_{ji}$, we may delete the corresponding $k_{jt}$ from the data structure. If the result of deleting $k_{jt}$ some mixture component $k$ becomes unallocated, then we delete this mixture components as well.

**Sampling k:** Because $k_{jt}$ determines the component membership of all the data points in table $t$, the likelihood by setting $k_{jt} = k$ is given by $f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}})$, so the probability of $k_{jt}$ is:

$$\begin{aligned} & p(k_{jt} = k|t, k^{-jt}) \\ & \propto \begin{cases} m_{.k} f^{-x_{ji}}(x_{ji}|\theta_{k_{ji}}) & \text{if } k \text{ previously used} \\ \gamma f^{-x_{ji}}(x_{ji}|\theta_{k_{new}}) & \text{if } k = k^{new} \end{cases}. \end{aligned} \qquad (8)$$

Following the sampling scheme above, given $z_{ji} = k_{jt}$, we can update $F(\theta_{z_{ji}})$ in Fig. 4 for image $j$.

### *C. New Feature Representation*

After modeling HDP over the prior-knowledge data, we now obtain the likelihood $p(w_i|z_k, D_{prior})$ and the probability $p(z_k|w_i)$ for connecting latent components with the dictionary visual words. To find the representations of images based on the latent components, we need to compute $p(z_k|\mathbf{I_j})$ for the $j$th image. Recall that the histogram for image $I_j$ based on visual dictionary $V$ is $\mathbf{h_j}$, and we have $h_{ji} = p(w_i|I_j)$. According to the Bayesian rule, we have

$$p(z_k|\mathbf{I_j}) = \sum_{w_i \in \mathbf{I_j}} p(z_k|w_i)p(w_i|\mathbf{I_j}), \tag{9}$$

where $p(w_i|\mathbf{I_j})$ corresponds to the $i$th dimension of the normalized bag-of-feature histogram, i.e. $\frac{h_{ji}}{\sum_{s=1}^{d_j} h_{js}}$. We define $f_k(\mathbf{h_j}) \equiv p(z_k|\mathbf{I_j})$ to map the raw feature vector $\mathbf{h_j} = (h_{j1}, h_{j2}, \ldots, h_{jd_j})$ to the higher level representation $\mathbf{F}(\mathbf{h_j})$, where $\mathbf{F}(\cdot) = (f_1(\cdot), f_2(\cdot), \ldots, f_k(\cdot), \ldots, f_{d_j}(\cdot))$, for $j = 1, 2, \ldots, N$. Suppose $\mathbf{h_j}$ is the bag-of-feature histogram representation based on visual dictionary $V_l$, the new representation $\mathbf{F}(\mathbf{h_j})$ is actually the normalized bag-of-feature histogram based on the next level dictionary $V_{l+1}$, where each entry is a latent mixture component learned from HDP modeling. We call $\mathbf{F}(\cdot)$ the HDP-encoder.

### D. Hierarchical Feature Learning

In this section, we will present the hierarchical feature learning structure based on the HDP unsupervised feature learning model described in previous sections. Fig 7 shows the construction process of multiple-level features based on a single type of low-level features.

To motivate the proposed structure, we first review the spatial pyramid matching scheme [14]. In spatial pyramid matching, as shown in Fig 6, the two-dimensional image space is divided into sub-images equally, then the sub-images are treated as separate channels to compute the feature matching at each level. Experimental results show that this spatial pyramid construction yields improvements in similarity measurement using intersection kernels. Since the bag-of-feature representation treats the visual vocabulary features orderless, the improvement introduced by such multiple spatial histogram resolutions is intuitive: It actually takes the location information of feature points into consideration. In this paper, we propose a similar solution in the discrete feature space: We construct a multi-layer feature space where each level consists of features with different “describing resolutions”. For example, in the “zebra-horse” problem, the “stripped” pattern has higher level of “describing resolution” than “black” and “white”, since we can describe certain areas as “black” or “white”, while only a structure with repeating lines and alternative color areas in between could be called “stripped”. A lower level feature captures local characteristics of an image while a higher level feature describes properties related to certain structure of the object or image background.

Incorporating the idea of multiple “describing resolutions”, we now present how to use HDP-encoders to build the feature-pyramid. Fig 7 reveals that the feature learning method is actually a clustering process on the previous lower level, where $L0$ denotes the bottom level and $L1$ denotes the next higher level, and so on. In contrast to spatial pyramid subdivisions where higher levels have more details, our method provides more informative descriptions at higher levels. In our feature pyramid, from bottom to top, the descriptions focus from local details to regions, then to objects. Similar to spatial pyramid, we believe that our multiple “describing resolutions” could provide a more comprehensive similarity metric, which can benefit the classification later. In details, Fig 7 shows how the HDP-encoders work under the pyramid structure based on a single type of features. In Fig 7, $HDP_{L0-L1}$ means the transformation function learned from $L0$ features using Equation (9) based on HDP modeling, and this $HDP_{L0-L1}$ is used to encode $L0$ features into $L1$ features. Recursively, we learn the function $HDP_{L1-L2}$ from $L1$ features and encode them into $L2$ features and so on. Note that with applying the HDP-encoder multiple times, we reduce the feature space to a low dimensional one. In practice, we stop the multiple HDP-encoder process once the dimensionality of the new level feature is below 100, to ensure the discriminative power of each level. It is worth noting that we estimate the stacked HDPs layer by layer in a greedy way, under the assumption that features at each layer follow a multinomial distribution.

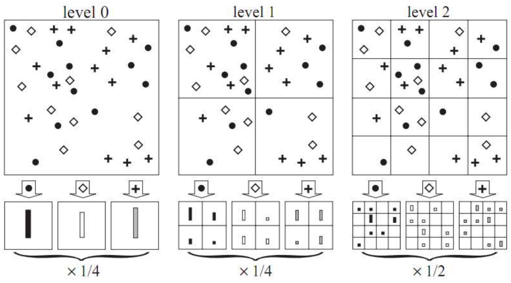


Fig. 6. Illustrations of spatial pyramid subdivisions [14] and the weights for each level of resolutions.

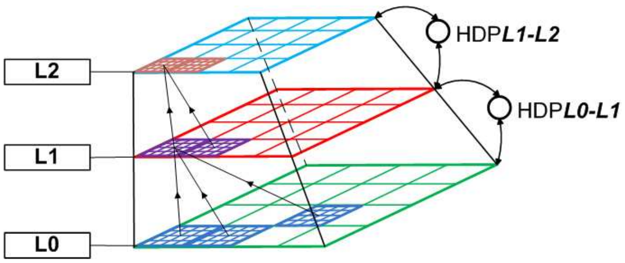


Fig. 7. Construction of the feature pyramid based on a single type of features using HDP-encoders. Each HDP-encoder is obtained by Equation (9) from HDP modeling on the lower level features.

So far the feature learning we described is based on a single type of image descriptor (e.g., SIFT). Is this enough? Empirically conducting feature learning on one individual image descriptor cannot capture all useful information. For instance, both color and texture information can be important for classification tasks and should be learned into joint higher level features. It is therefore important that we can jointly model different image descriptors (e.g., SIFT from gray scale images and Color Histogram from color images). We propose learning a higher level feature vector from multiple types of low-level descriptors: To couple the feature spaces together in HDP modeling, we concatenate different feature vectors into a long vector and then apply HDP, which is equivalent to encode images with a large joint dictionary. In practice, we may need to design a specific learning structure for a particular task, by analyzing the features provided. Fig 10 in Section IV-C shows two possible feature learning procedures.

### E. Feature Combination

In this section, we will discuss how to combine features learned from above sections into the classifier's input data. One reason why object recognition is a challenging task is that images within the same class usually have high intraclass variability. The low-level image descriptors are designed to be invariant to the variations within classes. At the same time, the descriptors are desired to have discriminative power for different classes. There is no single descriptor that can satisfy both requirements for all object classes, thus adaptively combining different types of features is preferred. Basically, we want to combine descriptors based on color, shape and texture information.

In this paper, we are facing two kinds of feature combination problems: 1) How to combine all types of features at the same level; and 2) how to combine features at different levels? For the first question, the described crossing-space HDP modeling can be a solution. The new higher level features learned from concatenated spaces capture information from all lower level features, yielding a much more compact feature space than the original ones. However, as we mentioned earlier, since every feature space has its unique advantages, we need to integrate useful information as much as we can, especially for one-shot tasks where only extremely limited information is provided.

Grauman and Darrell [22] propose pyramid matching to find an approximate correspondence between two sets of features. Pyramid matching works by placing a sequence of increasingly coarser grids over the feature space and taking a weighted sum of matches that occur at each level of resolution. In our feature pyramid, at any level, the occurrence of the same feature in two images is considered as a match. More specifically, for the feature pyramid with levels $0, 1, 2, \ldots, L$, we use $H^l_{X_{(k)}}$ and $H^l_{Y_{(k)}}$ to denote the histograms of feature $k$ for images $X$ and $Y$ at level $l$. Then the intersection kernel for these two histogram vectors is:

$$\kappa^l_{(k)}(H_X, H_Y) = \sum_{i=1}^{D_k} min(H^l_{X_{(k)}}(i), H^l_{Y_{(k)}}(i)). \qquad (10)$$

It was shown that this histogram intersection is additive Mercer Kernel [22]. Now let us look closely at all features at level $l$ of the feature pyramid. [7], [8] simply concatenate all feature vectors into a long vector, which is equivalent to using the average kernel over intersection kernels calculated for all types of features. [5] shows that Multiple Kernel Learning (MKL) and its variants have the best performances on classification tasks. However, in our one-shot recognition problem, we don't have sufficient training data to optimize the linear combination of different kernels when optimizing the classifier's coefficients simultaneously. Thus, we choose the average kernel here for simplicity and good performance according to [5]. Then the kernel function for level $l$ is:

$$\kappa^l(H_X, H_Y) = \frac{1}{M_l} \sum_{k=1}^{M_l} \kappa^l_{(k)}(H_X, H_Y). \qquad (11)$$

After defining the matching kernel at each level, we put weights on the kernel scores according to the "describing resolutions" of different levels. Unlike the spatial pyramid, in which finer grid has higher weights, in our feature pyramid, intuitively higher level features have higher weights since we believe that the features at higher levels are more meaningful than the lower level ones with respect to objects' characteristics. More specifically, the low-level features have a large proportion of noisy information, which is an important concern in the one-shot recognition problem, and higher level features somehow 'filter' out unimportant information by clustering lower level features. Later in the Caltech 4-class experiments, we can see that the higher level feature SIFT-L1 yields better recognition performance than the lower SIFT-L0 feature. Therefore, intuitively it makes sense to assign higher weights for higher level features. The weight associated with level $l$ is heuristically set to be $\frac{1}{2^{L-l}}$, where $L$ means the highest level. The final kernel function for all levels in the pyramid is as:

$$\begin{aligned} &K(H_X, H_Y) = \textstyle\sum_{l=0}^{L} \frac{1}{2^{L-l}} \kappa^l(H_X, H_Y) \\ &= \textstyle\sum_{l=0}^{L} \frac{1}{2^{L-l}} \frac{1}{M_l} \sum_{k=1}^{M_l} \kappa^l_{(k)}(H_X, H_Y) \qquad . \\ &= \textstyle\sum_{l=0}^{L} \frac{1}{2^{L-l}} \frac{1}{M_l} \sum_{k=1}^{M_l} \sum_{i=1}^{D_k} min(H^l_{X_{(k)}}(i), H^l_{Y_{(k)}}(i)) \end{aligned} \qquad (12)$$

where the final intersection kernel is actually weighted summation of matching scores for all the dictionary words in all feature types from all levels.

### F. One-shot Recognition Decision

For one-shot recognition tasks, we first use the proposed unsupervised structural learning method to build a feature pyramid based on unlabeled prior-knowledge data, we then compute the kernel matrix for the testing data in the target domain according to Equation (12). Since [22] shows that the intersection kernel is additive Mercer Kernel, we can directly input our pre-computed final kernel $K$, as in Equation (12), into the popular Support Vector Machine classifier to make the classification decision.

## IV. Experimental Results

To evaluate the performance of the proposed method on one-shot image recognition, we examine two popular, publicly-available data sets in the area of one-shot recognition. We first evaluate the proposed unsupervised feature learning method on a 4-class data set which is a subset of Caltech-101, comparing with previous reported performances based on multi-shot training; We then report the results on the "Animals with Attributes" data set using two different feature learning procedures (as shown in Fig 10). All classifications are performed by LIBSVM [25] with using our pre-computed kernels or linear kernel (in the 4-class experiment) with the parameter $C = 10$.

### A. Data Sets

**4-Class** [6], [15], a subset of Caltech-101, is a data set consisting of images from Airplane, Faces, Leopard and Motorbikes 4 classes. Since no previous work using unlabeled data as prior-knowledge under one-shot recognition setting, we compare our one-shot recognition results with [15] which used 50 training samples in classifications. Airplane, Faces, Leopard

Fig. 8. Images from 4-class data set

| Classes | 50 training [15] | 50 training (SIFT-L1) | one-shot (SIFT-L1) | one-shot (SIFT-L0) |
|---|---|---|---|---|
| Airplanes | 94 % | 89% | 79 % | 46 % |
| Faces | 74 % | 93% | 82 % | 85 % |
| Leopard | 92 % | 89% | 77 % | 99 % |
| Motorbikes | 88 % | 94% | 93 % | 68% |
| mean | 87 % | **91**% | **83** % | 74% |

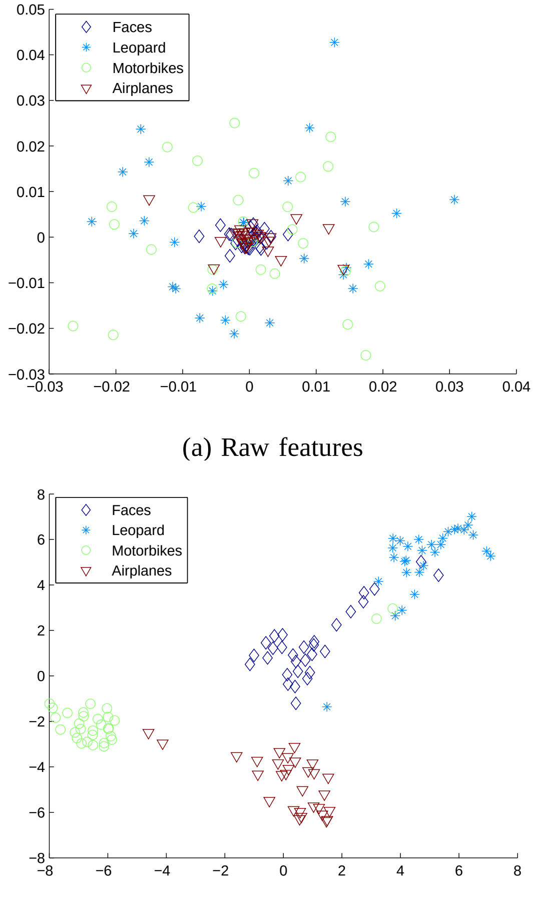
TABLE III
PERFORMANCES OF ONE-SHOT RECOGNITION ON THE 4-CLASS CALTECH DATA SET.

and Motorbikes are used as target categories, and 30 images from each of the remaining categories are used as prior-knowledge data. We compute SIFT descriptors on a dense grid of the images, every 8 pixels, to form the dictionary and vector data. In the testing phase, we randomly select 1 training sample from each target category, and use the remaining 29 samples as the testing data. The experiments are repeated 10,000 times to calculate the averaged classification accuracy.

**"Animals with Attributes"** data set [7] contains natural color images of 50 animal categories. There are 30,475 images in total and six types of pre-computed features for downloading, including RGB color histograms (CH), SIFT [1], rgSIFT [26], PHOG [3], SURF [2] and local self-similarity histograms (LSS) [27]. Lampert et al. extracted CH feature vectors for all 21 cells of a 3-level spatial pyramids(1 ×1, 2×2, 4×4). For each cell, 128-dimensional color histograms are extracted and concatenated to form a 2688-dimensional feature vector. Each of the other vectors, except PHOG, is 2000-bin bag-of-feature histogram. We didn't use PHOG because that its simple structure is not suitable for recursive feature learning. Among the descriptors, SIFT and SURF provide image gradient information, CH captures color information, LSS serves as texture descriptor and rgSIFT is a combination of color and local gradient information.

For **one-shot recognition**, we examine the same 10 target categories suggested by [7], and use the remaining as unlabeled prior-knowledge data. We only use 30 samples per prior-class as unlabeled training data, to achieve low computational cost and to show the generalization ability of the proposed feature learning method. In the testing phase, we randomly select one training sample from each target class and use the remaining as testing data. Therefore we have 10 training samples and 6170 testing samples for each independent experiment. We repeat the experiments 10,000 times to report the average classification accuracy. The 10 target testing categories are: chimpanzee, giant panda, leopard, Persian cat, pig, hippopotamus, humpback whale, raccoon, rat and seal (See Fig. 11).

For the conventional **multi-shot** recognition, we follow the protocol in [7] with using $50$ images in each target category for training and the rest for testing. We randomly select the training samples in each iteration, and repeat 10,000 iterations to report an averaged accuracy. Similar to the one-shot recognition experiments, for feature learning, we still only use the features learned from a small subset of prior-knowledge data, without using the training data in the target categories. The training data in the target categories are only used for classifier training.



(a) Raw features

(b) Learned intermediate representations

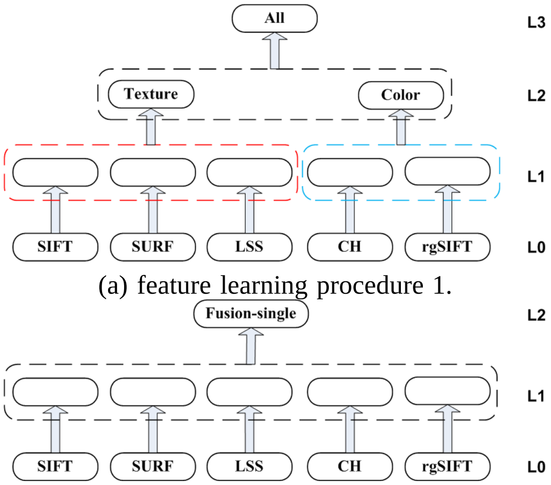
Fig. 9. 2D plots of the subsets of the 4-class data set. Each color/pattern represents a different class from the four categories.

### *B. Results on "4-class" data set*

Based on the SIFT-L1 features learned as in Section III, we employ the SVM classifier to perform one-shot recognition. For the 4-class data set, the classification accuracy results are reported in Table III, where an average accuracy of $83\%$ is observed for SIFT-L1. We clearly note a $9\%$ improvement from SIFT-L0. It is worth noting that the proposed one-shot learning method yields comparable performances to the method in [15] which is trained on 50 training samples per class and provides an average accuracy of $87\%$. In addition, to show the general applicability of the proposed method, we also report the results for SIFT-L1 with 50 training samples per category, and the average accuracy is $91\%$ which is better than that of [15]. It is worth mentioning that the 50 labeled samples per category are used in both feature learning and classification stages in [15], while they are only used in the classification stage in our proposed method.

To understand the feature learning process better, in Fig. 9, we visualize the raw SIFT-L0 features and the learned intermediate (SIFT-L1) features in 2D plots using the t-SNE technique [28]. In the plot of raw features, the testing data points from different classes are somehow mixed, though they seem to reveal a separable pattern by using nonlinear classifiers. However, it is important to recall that, since we only have one training sample per class in one-shot recognition, it is infeasible to discover such nonlinear distribution patterns based on one training sample without any prior assumption. In the plot of the learned intermediate representations, it is clear that testing data points from different classes are well separated in the feature space, therefore even with only one training sample available for each target class, the SVM classifier may still be able to make correct decisions for the testing data. The proposed feature learning based on disjoint prior-knowledge data may have the potential to achieve a comparable accuracy as that of the fully trained classification method in [15], although more investigations will be needed in the future to test on a larger number of categories.

### C. *One-shot Results on "Animals with Attributes"*



(a) feature learning procedure 1.

(b) feature learning procedure 2.

Fig. 10. Two feature learning procedures used for examining "Animals with Attributes". The dotted line boxes mean that we concatenate different feature spaces together and then apply the HDP-encoder.

We perform the unsupervised feature learning on a small subset (30 images each class and 1200 images in total) of the 40-class prior-knowledge data. The feature pyramids are constructed by two different procedures, as illustrated in Fig 10. In the first procedure in Fig 10(a), we use the HDP-encoder to learn $L1$ features from each of the 5 $L0$ descriptors. Here we don't do the cross-space learning yet mainly due to the concern of computational cost. The HDP learning from $L0$ to $L1$ actually filters the feature spaces. We then assign 5 types of $L1$ features into 2 categories, texture and color, and perform cross-space learning to obtain an $L2$ "texture" feature and an $L2$ "color" feature. Further, we combine the texture and color features together to learn an $L3$ "All" feature. Finally, we compute the final average kernel by Equation (12) and perform classifications using SVM.

Alternatively, we can construct a feature pyramid following Fig 10(b), to learn a compact $L2$ "Fusion-single" feature. We compute an interaction kernel based on this $L2$ feature by Equation (10) to do classifications. Since the "Fusion-single" feature is compact (e.g., 60 dimensions), it is possible to be used in real time mobile applications. Also [29] designs a fast approximate training approach to speed up the SVM training and testing over intersection kernels. We believe that the "Fusion-single" feature has practical advantages.

Table IV summarizes the results in one-shot recognition tasks when employing the two feature pyramid constructions showed in Fig 10. Note that a $14.1\%$ accuracy based on $L0$ features was reported by [7]–[9], where they simply concatenated all feature vectors into a long vector and used PCA to get a lower dimensional representation for classification. In our experiments, we use average intersection kernels and get a better accuracy as $17.8\%$. As we can see from the table, employing feature learning from L0 to L1 can provide a good performance improvement, i.e. from $17.8\%$ to $20.0\%$. By constructing the feature pyramid, we get accuracy gains gradually as the feature level increases. It seems that we can achieve the best performance by constructing a 4-level pyramid. From L0 to L0-L1-L2-L3 we observe an absolute $7.5\%$ accuracy improvement (i.e., representing a relative improvement over $50\%$). From L0-L1 to L0-L1-L2-L3 we only observe a $1.6\%$ absolute accuracy gain. However, it is worth emphasizing that even $1.6\%$ represents a significant progress in one-shot recognition where the provided information is limited and the average accuracy is generally quite low. In one-shot recognition, the standard deviation of recognition accuracy is usually large due to the randomness introduced by selecting only a single training sample in each category. However we observe a consistent improvement (more than 75% of the trials) from L0-L1 to L0-L1-L2-L3 during the repeated 10,000 independent experiments and the improvement represented by the pair-wise accuracy differences is statistically significant, judged by the t-test. In addition, it is worth mentioning that we used SIFT-L1 only and can achieve a $18.3\%$ accuracy in our conference paper [10], and here we can further improve the accuracy by $3.3\%$. From Table IV, we also note that the 60-dimensional $L2$ "Fusion-single" feature can achieve a $20.30\%$ accuracy. It suggests that the proposed feature learning method can learn a single compact feature with good discriminative power.

Since there are no specific algorithms designed for one-shot recognition with using only unlabeled prior-knowledge data, to have a feeling of the accuracy upper-bound, we compare the performance of the proposed method with the ones using much more prior information. Lampert et al. [7] used fully labeled images in the prior-knowledge domain and used a sophisticate animal attribute table designed by human experts for each of the class. They achieve $27.8\%$ in the IAP [7] setting and $40.5\%$ in the DAP [7] setting. Tang et al. [8] used fully labeled data in the prior-knowledge domain, which is probably the closest experimental setting to ours, and they reported an

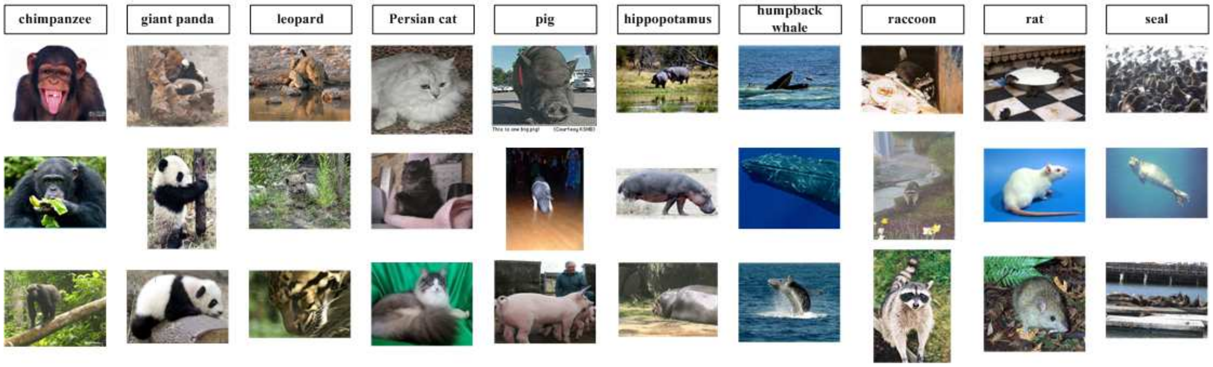


Fig. 11. Sample images from 10 target classes in “Animals with Attribute” data set

| Feature pyramid 1 | $L0$ | $L0 - L1$ | $L0 - L1 - L2$ | $L0 - L1 - L2 - L3$ | Feature pyramid 2 | $L2$ Fusion-single |
|---|---|---|---|---|---|---|
| Accuracy | 14.1% [7], [8]/17.8%(ours) | 20.0% | 20.3% | **21.6%** | Accuracy | **20.3%** |

TABLE IV
ONE-SHOT RECOGNITION ACCURACY ON “ANIMALS WITH ATTRIBUTES” DATA SET WITH FEATURE PYRAMID 1 AND 2. THE RESULTS SHOWN IN BOLDFACE ARE SIGNIFICANTLY BETTER THAN THE OTHERS, JUDGED BY A PAIRED-SAMPLE T-TEST.

average accuracy of $23.7\%$ for linear projection and $27.2\%$ for logistic projection. We are glad to notice that under our experimental setting which provides very limited information compared with previous works, the proposed method can achieve a comparable performance of $21.6\%$, which is close to the $23.7\%$ accuracy obtained by using fully labeled prior-knowledge [8].

To show the scalability of the proposed one-shot recognition method, we also conduct a 50-class recognition task as described in [8]. Here all the testing images are still drawn from the 10 target categories, and the training images from the rest 40 prior-knowledge categories serve as distractors [8]. For convenience and saving computational cost, we use a subset of distractors in the feature learning, and the results are shown in Table V. For the proposed method, the L0 features achieve an accuracy of $4.68\%$, and the combination of 4 levels of features achieve an accuracy of $5.27\%$. [8] reported an accuracy of $5.38\%$ for the raw features and an accuracy of $7.5\%$ for the logistic projection method. The accuracy difference between the proposed method and [8] when using the raw features (L0) could be due to the following major setting differences: We use a subset of 40 prior-knowledge categories as distractors for simplicity and [8] uses the entire data set; we only use the unlabeled prior-knowledge images and [8] uses the labeled ones. However the proposed method still shows an improvement from $4.68\%$ to $5.27\%$ when combining different levels of features.

| Methods | Proposed | Method in [8] |
|---|---|---|
| Accuracy(raw/learned) | 4.68%/5.27% | 5.38%/7.5% |

TABLE V
RECOGNITION ACCURACY RESULTS OF USING RAW FEATURES AND LEARNED FEATURES IN DIFFERENT METHODS.

| Methods | Raw Features [7] | Feature Pyramid 1 | Feature Pyramid 2 |
|---|---|---|---|
| Accuracy | 65.9% | **71.4%** | 70.0% |

TABLE VI
MULTI-SHOT RECOGNITION ACCURACY RESULTS ON THE “ANIMALS WITH ATTRIBUTES” DATA SET WHEN USING RAW FEATURES IN [7] AND FEATURE PYRAMIDS 1 AND 2 IN THE PROPOSED METHOD.

### D. Multiple Shots Results on “Animals with Attributes”

To evaluate the general applicability of the proposed method, we conduct the conventional multi-shot recognition experiments on the “Animals with Attributes” data set, and the results are shown in Table VI. The proposed method based on feature pyramid 1 in Fig. 10(a) achieves a $71.4\%$ accuracy, and the proposed “Fusion-single” feature also yields an accuracy as high as $70.0\%$. [7] reported a $65.9\%$ accuracy in multiple shots experiments based on 6 types of $L0$ raw features. It is noted that our feature learning process performed only on the prior-knowledge data can improve the absolute accuracy by $5.5\%$ in multi-shot recognition tasks. It indicates that the proposed feature learning method can learn a general representation and provides better discriminative power by transferring information between the prior-knowledge and target domains. This example shows that the usage of the proposed method is not limited to one-shot recognition tasks. It is expected that we could get further improvement if we perform feature learning also on the multiple training samples in the target categories.

## V. CONCLUSIONS AND FUTURE WORK

In this paper, we tackle the problem of one-shot image recognition and we propose a novel unsupervised hierarchical feature learning framework to learn higher level features based on low-level image descriptors. To construct the hierarchical feature pyramid, we propose using Hierarchical Dirichlet Process to perform feature learning from a lower level to a higher level. We also show that the HDP encoder can be applied recursively, which makes the feature learning procedure flexible and can be customized depending on particular tasks. Furthermore, we propose using the summation of weighted intersection kernels and the average kernel to transfer our feature pyramid into discriminative power. The proposed feature pyramid construction procedure is capable of learning a single compact feature for recognition. Our experimental results show that the proposed feature learning framework could benefit both one-shot recognition and conventional multi-shot recognition tasks.

Since we could perform the HDP modeling across feature spaces, in the future, we plan to incorporate multiple media sources into the proposed one-shot recognition system. For instance, we would like to add the image-associated text descriptions into the feature learning phase.

## ACKNOWLEDGMENT

This work was supported by the Canadian Natural Sciences and Engineering Research Council (NSERC) under Grant STPGP 365164-08 and Grant 11R82396.

## REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, 2004.

[2] H. Bay, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," in *Europeam Conference on Computer Vision (ECCV)*, 2006.

[3] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *ACM International Conference on Image and Video Retrieval (CIVR)*, 2007.

[4] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07)*, June 2007.

[5] P. Gehler and S. Nowozin, "On feature combination for multiclass object classification," in *International Conference on Computer Vision (ICCV)*, 2009.

[6] L. Fei-fei, R. Fergus, S. Member, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intellegence*, 2006.

[7] C. H. Lampert, H. Nickisch, and S. Hareling, "Learning to detect unseen object classes by between-class attribute transfer," in *IEEE Conference on Computer Vision and Pattern Recogntion (CVPR)*, 2009.

[8] K. D. Tang, M. F. Tappen, R. Sukthankar, and C. H. Lampert, "Optimizing one-shot recognition with micro-set learning," in *IEEE Conference on Computer Vision and Pattern Recogntion (CVPR)*, 2010.

[9] X. Yu and Y. Aloimonos, "Attribute-based transfer learning for object categorization with zero/one training example," in *European Conference on Computer Vision (ECCV)*, 2010.

[10] Z. Guo and Z. J. Wang, "One-shot recognition using unsupervised attribute-learning," in *Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, 2010.

[11] B. R. Gibson, X. Zhu, T. T. Rogers, C. W. Kalish, and J. Harrison, "Humans learn using manifolds, reluctantly," in *Neural Information Processing Systems (NIPS)*, 2010.

[12] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, 2006.

[13] C. Fellbaum, Ed., *WordNet An Electronic Lexical Database*. Cambridge, MA ; London: The MIT Press, 1998.

[14] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognition natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[15] Y. Ji, K. Idrissi, and A. Baskurt, "Object categorization using boosing within hierarchical bayesian model," in *IEEE International Conference on Image Processing*, 2009.

[16] E. Nowak and F. Jurie, "Learning visual similarity measures for comparing never seen objects," in *IEEE Conference on Computer Vision and Pattern Recogntion (CVPR)*, 2007.

[17] J. Davis, B. Kulis, S. Sra, and I. Dhillon, "Information-theoretic metric learning," in *International Conference on Machine Learning (ICML)*, 2007.

[18] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *International Conference on Machine Learning (ICML)*, 2007.

[19] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *International Conference on Machine Learning (ICML)*, 2003.

[20] K. P. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Proceedings of Neural Information Processing Systems (NIPS)*, 1998.

[21] A. Gammerman, V. Vovk, and V. Vapnik, "Learning by transduction," in *In Uncertainty in Artificial Intelligence (UAI)*. Morgan Kaufmann, 1998, pp. 148–155.

[22] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *IEEE International Conference on Computer Vision (ICCV)*, 2005.

[23] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, p. 2003, 2003.

[24] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of Neural Information Processing Systems (NIPS)*, 1998.

[25] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, 2011.

[26] T. Gevers and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intellegence*, 2008.

[27] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recogntion (CVPR)*, 2007.

[28] L. der Maaten and G. Hinton, "Visuaizing data using t-sne," *Journal of Machine Learning Research*, Sep. 2008.

[29] S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

**Zhenyu Guo** received the B.E. degree from Zhejiang University, China, in 2009. He is currently a Ph.D. candidate with department of Electrical and Computer Engineering at University of British Columbia, Canada. His research interests are in machine learning and statistical signal processing with application in multimedia, computer vision, and smart grid data.

**Z. Jane Wang** Z. Jane Wang (M02,SM12) received the B.Sc. degree from Tsinghua University, China, in 1996 and the M.Sc. and Ph.D. degrees from the University of Connecticut in 2000 and 2002, respectively, all in electrical engineering. While at the University of Connecticut, Dr. Wang received the Outstanding Engineering Doctoral Student Award. She has been Research Associate of Electrical and Computer Engineering Department at the University of Maryland, College Park. Since Aug. 1, 2004 she has been with the Department Electrical and Computer Engineering at the University of British Columbia, Canada, and she is currently Associate Professor. Her research interests are in the broad areas of statistical signal processing theory and applications, with focus on multimedia security and biomedical signal processing and modeling. She co-received the EURASIP Journal on Applied Signal Processing (JASP) Best Paper Award 2004, and the IEEE Signal Processing Society Best Paper Award 2005. She is serving as Associate Editor for the IEEE TSP, IEEE TIFS and IEEE TBME. She is the Chair and founder of the IEEE Signal Processing Chapter at Vancouver.