A Probabilistic Model for Recursive Factorized Image Features

Sergey Karayev [1] Mario Fritz [1,2] Sanja Fidler [1,3] Trevor Darrell [1]

[1] UC Berkeley and ICSI, Berkeley, CA
[2]MPI Informatics, Saarbrücken, Germany
[3]University of Toronto, Toronto, ON

{sergeyk, mfritz, sanja, trevor}@icsi.berkeley.edu

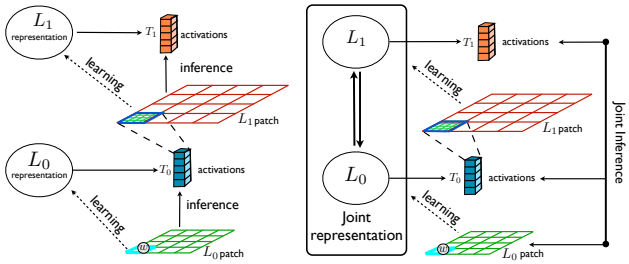
Abstract

Layered representations for object recognition are important due to their increased invariance, biological plausibility, and computational benefits. However, most of existing approaches to hierarchical representations are strictly feedforward, and thus not well able to resolve local ambiguities. We propose a probabilistic model that learns and infers all layers of the hierarchy jointly. Specifically, we suggest a process of recursive probabilistic factorization, and present a novel generative model based on Latent Dirichlet Allocation to this end. The approach is tested on a standard recognition dataset, outperforming existing hierarchical approaches and demonstrating performance on par with current single-feature state-of-the-art models. We demonstrate two important properties of our proposed model: 1) adding an additional layer to the representation increases performance over the flat model; 2) a full Bayesian approach outperforms a feedforward implementation of the model.

1. Introduction

One of the most successful and widely used developments in computer vision has been the rise of low-level local feature descriptors such as SIFT [21]. The basic idea of such local feature descriptors is to compactly yet discriminatively code the gradient orientations in small patches of an image. These features have been successfully used for scene and object recognition by representing densely extracted descriptors in terms of learned *visual words*—cluster centers in descriptor space [29]. On top of this quantized representation, more global image representations such as bags of words or spatial pyramids [17] can be assembled.

Recent publications in the field have started re-evaluating the hard clustering approach of visual words in favor of "softer" representations that allow a single descriptor to be represented as a mixture of multiple components [32]. The increased robustness of such distributed representations is appealing, and may exist in biological systems [23]. It has also been shown that such factorization leads to state-of-the-art performance on existing object recognition datasets [5], and allows good performance on novel datasets [10].



(a) Feedforward visual hierarchies (b) Our full probabilistic model

Figure 1: (a) Traditional visual hierarchies are feedforward with known disadvantages [19]. In contrast, our probabilistic model (b) learns and infers all layers jointly.

Despite increased representative power, these local features are still input directly to global object representations. While this approach has yielded some of the best recognition performance to date [32] , some recent works have shown that multi-layer intermediate visual representations could improve recognition performance by increased robustness to variance [22, 9, 33].

This is also in line with current theories of hierarchical, or layered, processing in the mammalian visual cortex [26]. Indeed, these theories give strong support to the importance of feedback, which both improves features that are being learned and disambiguates local information during inference. However, most existing hierarchical models are strictly feedforward (with a few notable exceptions, such as [13]), with each layer of the hierarchy operating over the fixed output of the previous layer [27, 25, 18].

The aim of this paper is to develop a fully probabilis-

tic hierarchical model to learn and represent visual features at different levels of complexity. The layers of the hierarchy are learned jointly, which, we show, has crucial advantage in performance over its feedforward counterpart. Our model is based on Latent Dirichlet Allocation (LDA) [4], a probabilistic factorization method originally formulated in the field of text information retrieval. It has been successfully applied to the modeling of visual words [28], and, more recently, to modeling of descriptors in an object detection task [10]. Here, we extend the latter representation to a recursively defined hierarchical probabilistic model, and show its advantage over the flat approach and the feedforward implementation of the model.

The approach is tested on a standard recognition dataset, showing performance that is higher than previous hierarchical models and is on par with current single-descriptor-type state of the art. We demonstrate two important properties of our proposed model: 1) adding an additional layer to the LDA representation increases performance over the flat model; 2) a full Bayesian approach outperforms a feedforward implementation of the model. This highlights the importance of feedback in hierarchical processing, which is currently missing from most existing hierarchical models. Our probabilistic model is a step toward nonparametric approaches to distributed coding for visual representations.

2. Related work

There is strong biological evidence for the presence of object-specific cells in higher visual processing areas. It is believed that the complexity of represented shape gradually increases as visual signal travels from the retina and LGN through V1 and higher cortical areas [26]. While the connectivity between different visual areas is far from being understood, there is evidence for a hierarchical organization of the visual pathway, in which units at each level respond to the output of the units at the level below, aggregated within a larger spatial area [6].

We will refer to computational modeling of a hierarchy of stacked layers of increasing complexity as *recursive models*. The main idea behind these approaches is to achieve high-level representation through recursive combination of low-level units, gradually encoding larger and larger spatial areas in images. This allows more efficient parametrization of image structures, and potentially better recognition performance.

A number of recursive models have been proposed. The HMAX model formulated the increase in shape complexity as the interaction of two types of operations: template matching and max pooling [27]. Units performing these operations are stacked in layers, where the template-matching units on the bottom of each layer receive inputs from small patches of the image, and the pooling units on top of each layer output more complex visual features. Implementations of this idea have shown promising classification results [22]. The units in the original HMAX model were pre-designed, not learned, while the recent improvements have included random template selection from a training corpus [27].

Learning mid-level visual features in a recursive hierarchical framework has motivated several recent works on convolutional networks [25, 2], deep Boltzmann Machines [12, 18], hyperfeatures [1], fragment-based hierarchies [30], stochastic grammars [34] and compositional object representations [9, 33]. The underlying ideas behind these approaches are similar, but they differ in the type of representation used.

Convolutional networks stack one or several feature extraction stages, each of which consists of a filter bank layer, non-linear transformation layers, and a pooling layer that combines filter responses over local neighborhoods using a pooling operation, thereby achieving invariance to small distortions [25]. In [1], the authors propose a representation with "hyperfeatures" which are recursively formed as quantized local histograms over the labels from the previous layer, which in turn are quantized local histograms over smaller spatial areas in images. Compositional hierarchies [9, 33, 24] and stochastic grammars [34] define objects in terms of spatially related parts which are then recursively defined in terms of simpler constituents from the layers below.

Most of these models process information in a feedforward manner, which is not robust to local ambiguities in the visual input. In order to disambiguate local information, contextual cues imposed by more global image representation may need to be used. A probabilistic Bayesian framework offers a good way of recruiting such top-down information [19] into the model.

In this paper, we propose a recursive Bayesian probabilistic model as a representation for the intermediately complex visual features. Our model is based on Latent Dirichlet Allocation [4], a latent factor mixture model extensively used in the field of text analysis. Here, we extend this representation to a recursively defined probabilistic model over image features and show its advantage over the flat approach as well as a feedforward implementation.

Due to overlapping terminology, we emphasize that our recursive LDA model is inherently different from Hierarchical-LDA [3] which forms a hierarchy of topics over the same vocabulary. In contrast, our hierarchy is over recursively formed inputs, with a fixed base vocabulary. Similar structure is seen in the Pachinko Allocation Model [20], which generalizes the LDA model from a single layer of latent topic variables to an arbitrary DAG of variables. As we explain later, this structure is similar to our model, but crucially, it is missing a spatial grid that allows us to model image patches of increasing spatial sup-

port. This allows us to keep the base vocabulary small, while for PAM it would grow with the size of the support in the image.

3. Recursive LDA Model

In contrast to previous approaches to latent factor modeling, we formulate a layered approach that derives progressively higher-level spatial distributions based on the underlying latent activations of the lower layer.

For clarity, we will first derive this model for two layers (L_0 and L_1) as shown in Figure 2a, but will show how it generalizes to an arbitrary number of layers. For illustration purposes, the reader may visualize a particular instance of our model that observes SIFT descriptors (such as the L_0 patch in Figure 2a) as discrete spatial distributions on the L_0-layer. In this particular case, the L_0 layer models the distribution of words from a vocabulary of size $V = 8$ gradient orientation bins over a spatial grid X_0 of size 4×4. As words in our vocabulary correspond to orientation bins, their frequency represents the histogram energy in the corresponding bin of the SIFT descriptor.

The mixture model of T_0 components is parameterized by multinomial parameters $\phi_0 \in \mathbb{R}^{T_0 \times X_0 \times V}$; in our particular example $\phi_0 \in \mathbb{R}^{T_0 \times (4\times 4) \times 8}$. The L_1 aggregates the mixing proportions obtained at layer L_0 over a spatial grid X_1 to an L_1 patch. In contrast to the L_0 layer, L_1 models a spatial distribution over L_0 components. The mixture model of T_1 components at layer L_1 is parameterized by multinomial parameters $\phi_1 \in \mathbb{R}^{T_1 \times X_1 \times T_0}$.

The spatial grid is considered to be deterministic at each layer and position variables for each word x are observed. However, the distribution of words / topics over the grid is not uniform and may vary across different components. We thus have to introduce a spatial (multinomial) distribution χ at each layer which is computed from the mixture distribution ϕ. This is needed to define a full generative model.

The model for a single layer with a grid of size 1×1 is equivalent to LDA, which is therefore a special case of our recursive approach.

3.1. Generative Process

Given symmetric Dirichlet priors α, β_0, β_1 and the number of mixture components T_0 and T_1 for layer L_1 and L_0 respectively, we define the following generative process, also illustrated in Figure 2b.

Mixture distributions are sampled globally according to:

- $\phi_1 \sim \text{Dir}(\beta_1)$ and $\phi_0 \sim \text{Dir}(\beta_0)$: sample L_1 and L_0 multinomial parameters
- $\chi_1 \leftarrow \phi_1$ and $\chi_0 \leftarrow \phi_0$: compute spatial distributions from mixture distributions

For each document, $d \in \{1, \dots, D\}$ top level mixing proportions $\theta^{(d)}$ are sampled according to:

- $\theta^{(d)} \sim \text{Dir}(\alpha)$: sample top level mixing proportions

For each document d, $N^{(d)}$ words w are sampled according to:

- $z_1 \sim \text{Mult}(\theta^{(d)})$: sample L_1 mixture distribution
- $x_1 \sim \text{Mult}(\chi_1^{(z_1,\cdot)})$: sample spatial position on L_1 given z_1
- $z_0 \sim \text{Mult}(\phi_1^{(z_1,x_1,\cdot)})$: sample L_0 mixture distribution given z_1 and x_1 from L_1
- $x_0 \sim \text{Mult}(\chi_0^{(z0,\cdot)})$: sample spatial position on L_0 given z_0
- $w \sim \text{Mult}(\phi_0^{(z_0,x_0,\cdot)})$: sample word given z_0 and x_0

According to the proposed generative process, the joint distribution of the model parameters given the hyperparameters can be factorized as:

$$p(w, z_{0,1}, \phi_{0,1}, x_{0,1}, \chi_{0,1}, \theta | \alpha, \beta_{0,1}) = P_{\phi_0} P_{\phi_1} \prod_{d=1}^{D} P_d \tag{1}$$

where

$$P_{\phi_i} = \prod_{t_1=1}^{T_1} p(\phi_i^{(t_i,\cdot,\cdot)} | \beta_i) \; p(\chi_i^{(t_i,\cdot)} | \phi_i^{(t_i,\cdot,\cdot)})$$

$$P_d = p(\theta^{(d)}|\alpha) \prod_{n=1}^{N^{(d)}} P_{z_1} P_{z_0} \; p(w^{(d,n)} | \phi_0^{(z_0,x_0,\cdot)})$$

$$P_{z_1} = p(z_1^{(d,n)} | \theta^{(d)}) \; p(x_1^{(d,n)} | \chi_1^{(z_1,\cdot)})$$

$$P_{z_0} = p(z_0^{(d,n)} | \phi_1^{(z_1,x_1,\cdot)}) \; p(x_0^{(d,n)} | \chi_0^{(z_0,\cdot)})$$

We use the superscript in parentheses to index each variable uniquely in the nested plates. Whenever a "$\cdot$" is specified, we refer to the whole range of the variable. As an example, $\phi_i^{(t_i,\cdot,\cdot)}$ refers to the multinomial parameters of topic t_i over the spatial grid and the topics of the lower layer L_0. The spatial distribution $\chi_0 \in \mathbb{R}^{T_0 \times X_0}$ and $\chi_1 \in \mathbb{R}^{T_1 \times X_1}$ are directly computed from ϕ_0 and ϕ_1 respectively by summing the multinomial coefficients over the vocabulary.

3.2. Learning and inference

For learning the model parameters we infer for each observed word occurrence $w^{(d,n)}$ the latent allocations $z_0^{(d,n)}$ and $z_1^{(d,n)}$, which indicate which mixture distributions were sampled at L_1 and L_0. Additionally, we observe the position variables $x_0^{(d,n)}$ and $x_1^{(d,n)}$, which trace each word occurrence through the X_0 and X_1 grids to $\theta^{(d)}$, as visualized in Figure 2a.

As we seek to perform Gibbs Sampling over the latent variables z for inference, we condition on the observed variables x and integrate out the multinomial parameters of the model. The equations are presented in Figure 3. In equation (3) we are able to eliminate all terms referring to χ, as

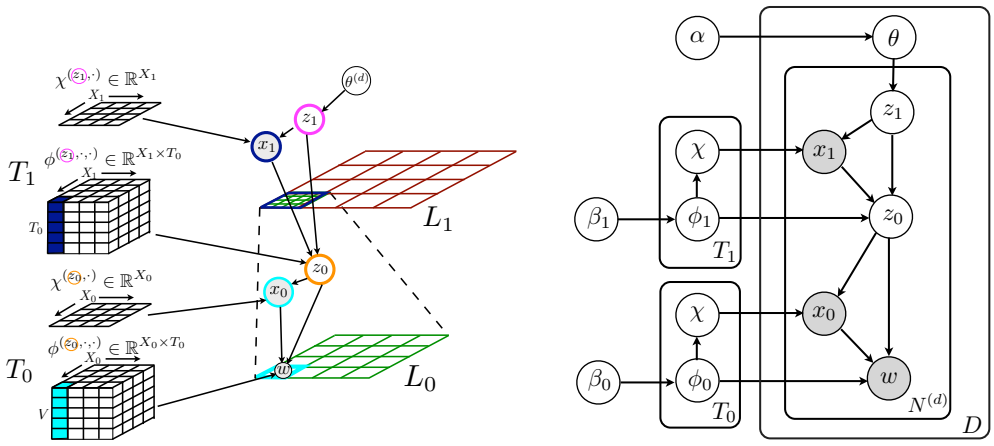


Figure 2: a) Concept for the Recursive LDA model. b) Graphical model describing two-layer RLDA.

$$p(w, z_{0,1}|x_{0,1}, \alpha, \beta_{0,1}) = \int_\theta \int_{\phi_1} \int_{\phi_0} \int_{\chi_1} \int_{\chi_0} p(w, z_1, z_0, \phi_1, \phi_0, \theta, \chi_1, \chi_0|x_1, x_0, \alpha, \beta_1, \beta_0) d\chi_0 d\chi_1 d\phi_0 d\phi_1 d\theta \tag{2}$$

$$= \int_\theta \int_{\phi_1} \int_{\phi_0} p(w, z_1, z_0, \phi_1, \phi_0, \theta|x_1, x_0, \alpha, \beta_1, \beta_0) d\phi_0 d\phi_1 d\theta \tag{3}$$

$$= \underbrace{\int_\theta p(\theta|\alpha) p(z_1|\theta) d\theta}_{\text{top layer}} \underbrace{\int_{\phi_1} p(\phi_1|\beta_1) p(z_0|\phi_1, z_1, x_1) d\phi_1}_{\text{intermediate layer}} \underbrace{\int_{\phi_0} p(w|\phi_0, z_0, x_0) p(\phi_0|\beta) d\phi_0}_{\text{evidence layer}} \tag{4}$$

$$p(w, z_{1,\ldots,L}|x_{1,\ldots,L}, \alpha, \beta_{1,\ldots,L}) = \underbrace{\int_\theta p(\theta|\alpha) p(z_L|\theta) d\theta}_{\text{top layer } L} \prod_{l=1}^{L-1} \underbrace{\int_{\phi_l} p(\phi_l|\beta_l) p(z_{l-1}|\phi_l, z_l, x_l) d\phi_l}_{\text{layer } l} \underbrace{\int_{\phi_0} p(w|\phi_0, z_0, x_0) p(\phi_0|\beta) d\phi_0}_{\text{evidence layer}} \tag{5}$$

Figure 3: RLDA conditional probabilities for a two-layer model, which is then generalized to an L-layer model.

all variables x are observed and the deterministic transformation between ϕ and χ has probability one. Our formulation now closely resembles the Latent Dirichlet Allocation but adds an additional layer in between that also performs spatial grouping via the variables x. This formulation easily generalizes to L layers, as shown in equation (5).

The derivation of the Gibbs Sampling equations is analogous to derivations for LDA (for an excellent reference, see [11]), with the addition of the observed position variables x. Due to space constraints, the derivations are presented in the supplementary materials.

4. Experiments

To evaluate the performance of our probabilistic recursive model, we test on the classification dataset Caltech 101 [7], which has been the most common evaluation of hierarchical local descriptors. We show performance for different numbers of components at each of the layers and explore *single-layer*, *feed-forward* (FLDA), and *fully generative* (RLDA) models, showing that 1) classification performance improves with an additional layer over the single-layer model, and 2) the fully generative RLDA model improves over the feed-forward-only FLDA analog.

Our work seeks to improve local features for low and mid-level vision independent of any specific object recognition methods, and we do not innovate in that regard. We note that we test our model using only a single feature, and compare it only to other single-descriptor approaches, focusing on hierarchical models.

4.1. Implementation

The feature representation of an image for our approach are SIFT descriptors of size 16×16 pixels, densely extracted from the image with a stride of 6 pixels. Individ-

Table 1: Comparison of RLDA with 128 components at layer L_1 and 1024 components at layer L_2 to other hierarchical approaches. Under layers, "bottom" refers to the L_1 features of the RLDA model, "top" to L_2 features, and "both" denotes the feature obtained by stacking both layers.

Approach			Caltech-101	
Model		Layer(s) used	**15**	**30**
Our Model	RLDA (1024t/128b)	bottom	$56.6 \pm 0.8\%$	$62.7 \pm 0.5\%$
	RLDA (1024t/128b)	top	$66.7 \pm 0.9\%$	$72.6 \pm 1.2\%$
	RLDA (1024t/128b)	both	$\mathbf{67.4 \pm 0.5}$	$\mathbf{73.7 \pm 0.8}\%$
Hierarchical Models	Sparse-HMAX [22]	top	51.0%	56.0%
	CNN [16]	bottom	–	$57.6 \pm 0.4\%$
	CNN [16]	top	–	$66.3 \pm 1.5\%$
	CNN + Transfer [2]	top	58.1%	67.2%
	CDBN [18]	bottom	$53.2 \pm 1.2\%$	$60.5 \pm 1.1\%$
	CDBN [18]	both	$57.7 \pm 1.5\%$	$65.4 \pm 0.4\%$
	Hierarchy-of-parts [8]	both	60.5%	66.5%
	Ommer and Buhmann [24]	top	–	$61.3 \pm 0.9\%$

Table 2: Results for different implementations of our model with 128 components at layer L_1 and 128 components at L_2. For LDA models, "bottom" refers to using SIFT patches as input, while "top" refers to using 4×4 SIFT superpatches.

Approach				Caltech-101	
	Model	Basis size	Layer(s) used	**15**	**30**
128-dim models	LDA	128	"bottom"	$52.3 \pm 0.5\%$	$58.7 \pm 1.1\%$
	RLDA	128t/128b	bottom	$55.2 \pm 0.3\%$	$62.6 \pm 0.9\%$
	LDA	128	"top"	$53.7 \pm 0.4\%$	$60.5 \pm 1.0\%$
	FLDA	128t/128b	top	$55.4 \pm 0.5\%$	$61.3 \pm 1.3\%$
	RLDA	128t/128b	top	$59.3 \pm 0.3\%$	$66.0 \pm 1.2\%$
	FLDA	128t/128b	both	$57.8 \pm 0.8\%$	$64.2 \pm 1.0\%$
	RLDA	128t/128b	both	$61.9 \pm 0.3\%$	$68.3 \pm 0.7\%$

ual descriptors were processed by our probabilistic models, and results of inference were used in a classification framework described in section 4.2. Because LDA requires discrete count data and SIFT dimensions are continuous-valued, normalization of the maximum SIFT value to 100 tokens was performed; this level of quantization was shown to maintain sufficient information about the descriptor.

We trained and compared the following three types of models:

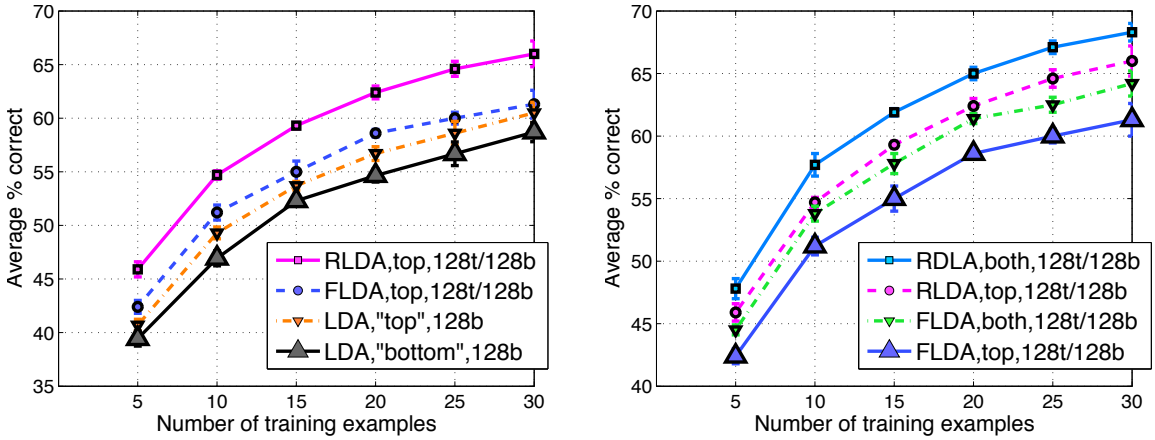
1. **LDA**: LDA models with various numbers of components (128, 1024, and 2048) trained on 20K randomly extracted SIFT patches. We also trained LDA models on "superpatches" consisting of 4×4 SIFT patches, to give the same spatial support as our two-layer models.

2. **FLDA**: The feed-forward model first trains an LDA model on SIFT patches, as above. Topic activations are output and assembled as 4×4 superpatches. Another LDA model is learned on this input. We tested 128 components at the bottom layer, and 128 and 1024 components at the top layer.

3. **RLDA**: The full model was trained on the same size patches as FLDA described above: SIFT descriptors in a 4×4 spatial arrangement, with model parameters set accordingly. We tested 128 components at the bottom, and 128 and 1024 components at the top layer.

4.2. Evaluation

The setup of our classification experiments follows the Spatial Pyramid Match, a commonly followed approach in Caltech-101 evaluations [17]. A spatial pyramid with 3 layers of 4×4, 2×2, and 1×1 grids was constructed on top of our features. Guided by the best practices outlined in a recent comparison of different pooling and factorization functions [5], we used max pooling for the spatial pyramid aggregation. For classification, we used a linear SVM, following the state-of-the-art results of Yang et al. [32]. Caltech-101 is a dataset comprised of 101 object categories,



(a) Caltech-101: 128 topic models, LDA-SIFT vs FLDA vs RLDA

(b) Caltech-101: 128 topic models, L_2 vs L_1+L_2 models

Figure 4: Comparison of classification rates on Caltech-101 of the one-layer model (LDA-SIFT) with the feed-forward (FLDA) and full generative (RLDA) two-layer models for different number of training examples, all trained with 128 at L_1 and 128 at L_2 layers. We also compare to the stacked L_1 and L_2 layers for both FLDA and RLDA, which performed the best.

with differing numbers of images per category [7]. Following standard procedure, we use 30 images per category for training and the rest for testing. Mean accuracy and standard deviation are reported for 8 runs over different splits of the data, normalized per class.

We first tested the one-layer LDA model over SIFT features. For 128 components we obtain classification accuracy of 58.7%. Performance increases as we increase the number of components: for 1024 topics the performance is 68.8%, and for 2048 it is 70.4% (all numbers here and on refer to the case of 30 training examples).

The emphasis of our experiments was to evaluate the contribution of additional layers to classification performance. We tested the FLDA and RLDA models (described in 4.1) in the same regime as described above.

As an initial experiment, we constrained the number of topics to 128 and trained three models: single-layer, FLDA, and RLDA. The two-layer models were trained with 128 topics on the bottom layer, corresponding to SIFT descriptors, and 128 on the top.

First, only the top-layer topics were used for classification. The dimensionality of vectors fed into the classifier was therefore the same for the two-layer and the single-layer models. We present the results in Table 2.

Both of the two-layer models improved on the one-layer model, obtaining classification rates of 61.3% for FLDA and 66.0% for RLDA, compared to the single-layer results of 58.7%, when trained on SIFT patches, and 60.6%, when trained on 4×4 SIFT "superpatches" (which are reported in Table 2 under LDA "top"). Detailed results across different numbers of training examples are shown in Figure 4a. The figure shows that the RLDA model always outperforms the FLDA model, which in turn always outperforms the one-layer model.

We have also tested the classification performance of stacking both L_1 and L_2 features into a feature vector. This feature contains more information, as both spatially larger and smaller features are taken into account. This is a standard practice for hierarchical models [18, 14]. The results, presented in Table 2 (under "both") and in Figure 4b, show that using information from both layers improves performance further, by about 3% for FLDA and 2% for RLDA.

We also compared the performance of just the first layer L_1 obtained with the RLDA model to the single-layer model (which also forms the first layer of the FLDA model). Interestingly, the decomposition learned in L_1 of the RLDA model outperforms the single layer model (62.6% vs 58.7%). This demonstrates that learning the layers jointly is beneficial to the performance of both bottom and top layers, separately examined.

Following this evaluation, we also learned two-layer models with 1024 topics in the top layer and 128 in the bottom layer. The results hold, but the difference between the models becomes smaller—we hypothesize that single-feature performance on the dataset begins saturating at these levels. RLDA top layer performs at 72.6% vs. 72.5% for FLDA and 68.8% for the single-layer model. The bottom layer of RLDA achieves 62.7% classification accuracy vs 62.6% for FLDA. Using both layers, RLDA gets 73.7% while FLDA gets 72.9%.

The comparison with related hierarchical models is given in Table 1, which shows that while the baseline single-layer LDA-SIFT model performs worse than most of the approaches, the proposed RLDA model outperforms the ex-

isting work by more than 5%.

Our best performance of 73.7% compares well to the current best performing approaches: sparse coding with 73.2% [32] and 75.7% [5] (using increased spatial support and much denser sampling), and LCC [31] with 73.5%. State-of-the art single-feature performance has been shown by keeping all training data in a non-parametric classification scheme, obtaining 78.5% [15].

We note that in the existing literature on hierarchical models, most authors report best performance with just one learned layer, while adding another layer seems to decrease the recognition performance [27, 25, 8]. In contrast, in our model, the performance increases by adding another layer to the representation. The most important result, however, is that the full Bayesian model as proposed in the paper outperforms the feed-forward approach. This supports the idea that inference carried out in the Bayesian approach results in more stable estimations of the feature activations, and thus better and more robust recognition performance. This highlights the importance of feedback in hierarchical recognition.

4.3. Role of Feedback in the Model

Additional evidence for the crucial role of feedback in the model comes from visualization of the average image patches corresponding to top- and bottom-layer components learned by the two models. Figure 5 shows that the full generative RLDA model uses lower-layer components in a notably different way than the feed-forward model, and learns different, more complex spatial structures at the top layer.

In the feed-forward model, the bottom-layer topics are in essence orientation filters. The second layer does not impose any additional structure on them, and therefore the top-layer topics appear to be the same simple orientations, localized within a bigger spatial support. In the fully generative RLDA model, the top-layer components seem to represent more interesting and potentially discriminative spatial structures.

We also found that the RLDA bottom-layer activations exhibit stronger correlations between topic activations in neighboring patches, which suggests that the model allows bottom layer inference to represent continuous structures across subpatches.

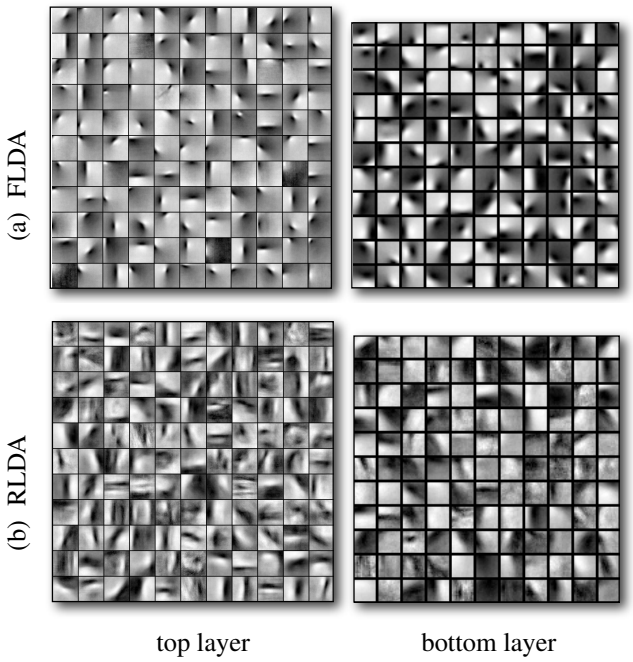


Figure 5: Comparison of the (a) feed-forward and (b) full generative two-layer models in terms of components learned, visualized as average image patches. While FLDA learns only localized edge orientations, RLDA learns more complex spatial structures.

5. Conclusions

We presented a probabilistic model for visual features of increasing complexity and spatial support. The layers of the hierarchy are trained jointly. We demonstrate performance that is among the recent best in single-descriptor approaches, and that outperforms existing hierarchical approaches. Most importantly, we show that adding another layer to our model significantly improves performance (something that is rarely true for layered models in vision), and that the full generative process performs better than the feed-forward approach. This emphasizes the necessity of feedback in hierarchical visual recognition models.

Probabilistic models are robust, make modular combinations easy, and form the basis of possible non-parametric extensions. The theorized goal of hierarchical approaches to vision is reaching object-level representations. With additional layers and variables, our model can be developed from merely a source of mid-level features to a full-scale object recognition method. With a non-parametric extension, the number of components would be inferred from the training data, which would appear especially important at the part- and object-level representation. These topics are subject of future work.

Acknowledgements: Various co-authors of this work were supported in part by a Feodor Lynen Fellowship granted by the Alexander von Humboldt Foundation; by awards from the US DOD and DARPA, including contract W911NF-10-2-0059; by NSF awards IIS-0905647 and IIS-0819984; by EU FP7-215843 project POETICON; and by Toyota and Google.

References

[1] A. Agarwal and B. Triggs. Multilevel Image Coding with Hyperfeatures. *International Journal of Computer Vision*, 2008. 402

[2] A. Ahmed, K. Yu, W. Xu, Y. Gong, and E. P. Xing. Training Hierarchical Feed-Forward Visual Recognition Models Using Transfer Learning from Pseudo-Tasks. *ECCV*, 2008. 402, 405

[3] D. M. Blei, T. L. Griffiths, M. I. Jordan, and J. B. Tenenbaum. Hierarchical Topic Models and the Nested Chinese Restaurant Process. *Advances in Neural Information Processing Systems 16*, 2010. 402

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 2009. 402

[5] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning Mid-Level Features For Recognition. *CVPR 2010*, 2010. 401, 405, 407

[6] C. E. Connor, S. L. Brincat, and A. Pasupathy. Transformation of shape information in the ventral pathway. *Current Opinion in Neurobiology*, 17(2):140–147, 2007. 402

[7] L. Fei-Fei, R. Fergus, and P. Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. *CVPR 2004, Workshop on Generative-Model Based Vision*, 2004. 404, 406

[8] S. Fidler, M. Boben, and A. Leonardis. Similarity-based cross-layered hierarchical representation for object categorization. In *CVPR*, 2008. 405, 407

[9] S. Fidler and A. Leonardis. Towards Scalable Representations of Object Categories: Learning a Hierarchy of Parts. *CVPR 2007*, 2007. 401, 402

[10] M. Fritz, M. J. Black, G. R. Bradski, S. Karayev, and T. Darrell. An Additive Latent Feature Model for Transparent Object Recognition. *NIPS*, 2009. 401, 402

[11] G. Heinrich. Parameter estimation for text analysis. *Technical Report*, 2008. 404

[12] G. E. Hinton. Learning multiple layers of representation. *Trends in Cogn. Sciences*, 11(10):428–434, 2007. 402

[13] G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006. 401

[14] K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the Best Multi-Stage Architecture for Object Recognition? *CVPR*, 2009. 406

[15] C. Kanan and G. Cottrell. Robust classification of objects, faces, and flowers using natural image statistics. In *CVPR*, 2010. 407

[16] K. Kavukcuoglu, P. Sermanet, Y.-L. Boureau, K. Gregor, M. l. Mathieu, and Y. LeCun. Learning Convolutional Feature Hierarchies for Visual Recognition. *NIPS*, 2010. 405

[17] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. *CVPR 2006*, 2006. 401, 405

[18] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009. 401, 402, 405, 406

[19] T. S. Lee and D. Mumford. Hierarchical Bayesian inference in the visual cortex. *Journal of the Optical Society of America Association*, 2003. 401, 402

[20] W. Li and A. McCallum. Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations. *ICML 2006*, 2006. 402

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004. 401

[22] J. Mutch and D. G. Lowe. Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision*, 2008. 401, 402, 405

[23] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 2003. 401

[24] B. Ommer and J. M. Buhmann. Learning the compositional nature of visual objects. In *CVPR*, 2007. 402, 405

[25] M. A. Ranzato, F.-J. Huang, Y.-L. Boureau, and Y. LeCun. Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition. *CVPR*, 2008. 401, 402, 407

[26] E. T. Rolls and G. Deco. *Computational Neuroscience of Vision*. Oxford Univ. Press, 2002. 401, 402

[27] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Object recognition with cortex-like mechanisms. *PAMI*, 29(3):411–426, 2007. 401, 402, 407

[28] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their locations in images. In *ICCV*, 2005. 402

[29] J. M. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003. 401

[30] S. Ullman. Object recognition and segmentation by a fragment-based hierarchy. *TRENDS in Cognitive Sciences*, 2006. 402

[31] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained Linear Coding for Image Classification. *CVPR 2010*, 2010. 407

[32] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 401, 405, 407

[33] L. L. Zhu, C. Lin, H. Huang, Y. Chen, and A. Yuille. Unsupervised Structure Learning: Hierarchical Recursive Composition, Suspicious Coincidence and Competitive Exclusion. *ECCV 2008*, 2008. 401, 402

[34] S. Zhu and D. Mumford. A stochastic grammar of images. *Foundations and Trends in Computer Graphics and Vision*, 2(4):259–362, 2006. 402