The Annals of Applied Statistics
2017, Vol. 11, No. 3, 1275–1308
DOI: 10.1214/17-AOAS1025

MAXIMUM LIKELIHOOD FEATURES FOR GENERATIVE IMAGE MODELS[1]

BY LO-BIN CHANG*, ERAN BORENSTEIN†, WEI ZHANG‡ AND STUART GEMAN§

Ohio State University, Amazon†, Smartleaf‡ and Brown University§*

Most approaches to computer vision can be thought of as lying somewhere on a continuum between generative and discriminative. Although each approach has had its successes, recent advances have favored discriminative methods, most notably the convolutional neural network. Still, there is some doubt about whether this approach will scale to a human-level performance given the numbers of samples that are needed to train state-of-the-art systems. Here, we focus on the generative or Bayesian approach, which is more model based and, in theory, more efficient. Challenges include latent-variable modeling, computationally efficient inference, and data modeling. We restrict ourselves to the problem of data modeling, which is possibly the most daunting, and specifically to the generative modeling of image patches. We formulate a new approach, which can be broadly characterized as an application of "conditional modeling," designed to sidestep the high-dimensionality and complexity of image data. A series of experiments, learning appearance models for faces and parts of faces, illustrates the flexibility and effectiveness of the approach.

1. Introduction. Lately, discriminative approaches to computer vision, mostly employing convolutional neural networks, have dominated academic research and industrial applications. Whether biological-level performance can be achieved by these approaches remains a matter of debate. The key issue is context—the system of hierarchical relationships that humans rely on, effortlessly, to disambiguate occlusions and local uncertainties and to make sense of the identities, attributes, and poses of parts that make up a whole, such as the arrangements of edges in a contour, leaves in a tree, pedestrians walking or talking together, tiles making up a roof or bricks and windows making up a building. Existing systems cannot fully exploit these part/whole relationships. Whereas, in principle, context can be learned without explicit models, the numbers of samples needed to train state-of-the-art systems often exceed the numbers available to an individual in a lifetime, and are already challenging today's big-data repositories. The root of the problem

Received July 2016; revised January 2017.

[1]Supported in part by the Office of Naval Research under Contracts ONR N000141010933, N000141512267, and N000141512356, the Defense Advanced Research Projects Agency under Contract FA8650-11-1-7151, and the National Science Foundation under Grant 0964416.

Key words and phrases. Computer vision, image models, appearance models, generative models, conditional modeling, sufficiency, features.

is that the number of meaningful contextual arrangements grows very rapidly with the number of components, or parts, in a composition. Discriminative approaches may not scale to human performance.

Generative hierarchical models, with grammar-like structures, appear to have the right architectures to model context and address the unfavorable combinatorics. But these and other Bayesian methods rely on, and are particularly sensitive to, models of the data, that is, the conditional distributions on appearance given the presence of a contour, a texture, or, in general, a composition of parts making up a recognizable unit. Appearance modeling remains a major impediment to progress of the Bayesian (generative) approach.

Generative models of high-dimensional data, such as whole images or image patches, invariably encounter issues of data sparsity and computational complexity. Among the most common approaches to addressing these difficulties is through dimensionality reduction, which amounts to replacing the raw high-dimensional data with a small set of low-dimensional feature values. Obviously, the choice of features is critical. In this paper, we provide a mathematically coherent approach to learning features within a fully generative framework, meaning that the resulting probability distribution is on the pixel data rather than on the features *per se*. Since features are almost never sufficient to define the pixel data itself, our approach avoids the inherent loss of information incurred by modeling extracted features in place of raw pixel data.

The main idea, which amounts to an application of conditional modeling [cf. Reid (1995)], is to define a category-specific low-dimensional distribution on features, and then to assume that the conditional distribution on pixel intensities given the values of the features is universal, that is, independent of the category. The result is a category-specific distribution on pixel data that is a function of the features. One consequence is that the features themselves, which may have been unspecified, can be estimated using traditional statistical methods, such as maximum likelihood.

Formally, we consider image patches, such as a 30×40 rectangle of pixel intensities, denoted Y. (We will work with gray-level images, though no important changes are needed to adapt to color images.) The task is to develop a model for the distribution of Y given that it comes from a particular category of objects, say the right eye of a human face. In general, real images have complex dependency structures that are extremely difficult to model, even for a modest-sized image patch with only 1200 pixels. But suppose that there is a low-dimensional function (aka "statistic" or "feature") $s = s(y)$ whose value is particularly relevant to the determination of whether or not the observation $Y = y$ is an observation of a patch from the category of interest (e.g., right eyes). A time-honored example is $s(y) = \mathrm{corr}(T, y)$, where T is a template, perhaps a prototypical right eye, and *corr* is the normalized correlation ($s \in [-1, 1]$). Given the category of interest, there is some (typically unknown) distribution $p_S(s)$ on the random variable $S \triangleq s(Y)$,

and we can always use this to factor the distribution on Y:

$$p_Y(y) = p_S\big(s(y)\big)p_Y\big(y|S = s(y)\big). \tag{1.1}$$

By assumption, s is low dimensional, and, therefore, it is a relatively easy job to estimate its distribution, $p_S(s)$. We will generally assume a parametric form, $p_S(s) = p_S(s;\theta)$, although semiparametric or even nonparametric estimation would be reasonable options, depending on the dimensionality, which is just one for the example $s(y) = \mathrm{corr}(T, y)$. The remaining dimensions are in the conditional distribution, $p_Y(y|S = s)$, which must be estimated for each value of s. On the other hand, if we think of $s(Y)$ as carrying the bulk of the category-specific information about Y, then we can think of $p_Y(y|S = s)$ as capturing those aspects of the spatial arrangements of pixel intensities that are common across categories, having to do with neighborhood relationships, spacial scaling, the appearance of shadows and reflections, and so on. In other words, with a proper choice of s we can think of $p_Y(y|S = s)$ as being derived from some universal distribution, say $p_Y^o(y)$: given $p_Y^o(y)$, we replace $p_Y(y|S = s)$ by $p_Y^o(y|S = s)$ and (1.1) by

$$p_Y(y) = p_S\big(s(y)\big)p_Y^o\big(y|S = s(y)\big). \tag{1.2}$$

To look at this from another direction, we are assuming the existence of a kind of *background distribution*, $p_Y^o(y)$, some of whose aspects are nearly universal to image patches, and in particular independent of the category of the patch. Then, given a particular statistic $s(y)$ and distribution $p_S(s)$, associated with a particular category of image patch, we seek a distribution on Y under which $S = s(Y)$ has distribution $p_S(s)$ [i.e., $s(Y) \sim p_S(s)$] but which is otherwise similar to $p_Y^o(y)$. If we were to take "similar" to mean closest in the sense of Kullback–Leibler divergence, that is, minimizing

$$D\big(p_Y^o||p_Y\big) \triangleq \int p_Y^o(y)\log\frac{p_Y^o(y)}{p_Y(y)}\,dy$$

then we would recover (1.2), that is,

$$p_Y(y) = p_S\big(s(y)\big)p_Y^o\big(y|S = s(y)\big) = \operatorname*{argmin}_{\tilde{p}_Y : s(Y)\sim p_S} D\big(p_Y^o||\tilde{p}_Y\big).$$

In fact, we arrive at the same expression for p_Y when $D(p_Y^o||p_Y)$ is replaced by $D(p_Y||p_Y^o)$, as is shown, by straightforward calculation, in Section A.1 of the Appendix.

As an example, if we were to choose the uniform distribution to serve as a background model (so that under p_Y^o, Y has independent and uniformly distributed pixel intensities), then the category-specific model, (1.2), becomes the maximum entropy distribution subject to the category-specific constraint, $s(Y) \sim p_S$. In other words, maximum-entropy models [e.g., Zhu, Wu and Mumford (1998)] are a special case. Alternatively, we could define the background distribution, p^o, implicitly, to be the distribution on a set of "unstructured" patches from real images, say the set of all uniformly smooth patches from all natural images available on the

Internet. As it turns out (see Section 2), estimation of features and parameters in p_Y, as well as the classification of patches under the model p_Y, depend on the background only through p_S^o, which can be estimated, straightforwardly, by sampling background patches, provided that s is low dimensional.

Imagine now that we have a sample of patches from a category of interest, and we have identified a relevant low-dimensional statistic s, which may depend on a parameter vector, ϕ: $s(y) = s(y;\phi)$. In the case of the correlation statistic, for example, $s(y) = \mathrm{corr}(T, y)$ and $\phi = T$. We might, furthermore, model the category-specific distribution on $S = s(Y)$ in a parametric form, in which case we write $p_S(s) = p_S(s;\theta)$. The likelihood of θ and ϕ, given N patches $y_1, \ldots, y_N$ sampled from the category, is

$$\begin{aligned} L(y_1, \ldots, y_N; \theta, \phi) &= \prod_{k=1}^{N} p_Y(y_k) \\ &= \prod_{k=1}^{N} p_S\big(s(y_k;\phi);\theta\big) p_Y^o\big(y_k | S = s(y_k;\phi)\big) \end{aligned} \tag{1.3}$$

which we will seek to maximize over θ and ϕ, thereby learning both the category-specific features (through ϕ) and their category-specific distributions (through θ) in a fully generative model.

It is tempting to sidestep the high-dimensional conditional distribution, $p_Y^o(y_k | S = s(y_k;\phi))$, by replacing (1.3) with a likelihood that depends only on the feature values, $s(y_k)$,

$$L(y_1, \ldots, y_N; \theta, \phi) = \prod_{k=1}^{N} p_S\big(s(y_k;\phi);\theta\big) \tag{1.4}$$

as though we had observed the values of the features, and their associated statistics, rather than the pixel intensities themselves. Whereas (1.4) is consistent for θ, it is not consistent for ϕ, which is not surprising given that the modified likelihood ignores the dependency of the conditional distribution on the parameter ϕ. Put differently, s is sufficient for θ but not for ϕ.[2] See Section 4.2 for an experiment comparing the results of using (1.4) instead of (1.3). Another approach to avoiding the conditional distribution is to attempt to craft a model of the pixel data directly from the model of the statistic s: if we simply renormalize $p_S(s)$, then we have a proper parametric form for a data distribution

$$p_Y(y) = \frac{1}{Z(\theta,\phi)} p_S\big(s(y;\phi);\theta\big). \tag{1.5}$$

But now the distribution of the statistic, S, is no longer p_S. It is, instead, of the form $\frac{c(s)}{Z(\theta,\phi)} p_S(s)$, where $c(s)$ is a combinatorial factor (essentially the "entropic

[2]For fixed ϕ, θ is a maximum of (1.3) if and only if it is also a maximum of (1.4), but for fixed θ, the maxima can occur at entirely different values of ϕ.

term"), representing the number of assignments of pixel intensities, y, for which the statistic has the particular value s. If we were, for example, to design or learn T under the reasonable expectation that the probability of $s \triangleq \operatorname{corr}(T, y)$ is monotonic in s [e.g., $p_S(s) \propto e^{-\lambda(1-s)}$, $\lambda > 0$], then under p_Y, in (1.5), the distribution on $S = s(Y)$ will typically not be monotonic, since $s \in [-1, 1]$ and $c(s)$ is strongly peaked at $s = 0$. This is problematic since monotonicity motivated the choice of p_S in the first place.

Although the approach is quite general, most of our examples will involve the correlation statistic, $s = \operatorname{corr}(T, y)$, in which case the estimated parameter ϕ will be the template T. Modeling image patches through templates is a common practice in computer vision. Examples include Gaussian mixture models, in which an image patch is viewed as a sample from a mixture of Gaussian distributions; a different template, serving as the mean, is learned for each component of the mixture [e.g., Frey (2003), Frey and Jojic (1999), Kannan, Jojic and Frey (2002)]. Ullman and his collaborators [Borenstein and Ullman (2002), Sali and Ullman (1999), Ullman, Sali and Vidal-Niquet (2001)] selected templates from image patches that have the highest mutual information with the object category. They use these templates for object classification and segmentation. Others have defined candidate templates by running interest-point detectors on training sets, and then selecting from the surrounding image patches; see Agarwal, Awan and Roth (2004), Fergus, Perona and Zisserman (2003), Leibe and Schiele (2003), and Weber, Welling and Perona (2000). Heisele, Serre and Poggio (2007) and Heisele et al. (2001) designed a SVM algorithm to select patches from a collection of manually chosen seed points. The idea is to learn templates for facial parts that minimize the foreground-versus-background classification error. Si and Zhu (2012) used an information-based projection pursuit to select informative heterogeneous image patches as templates. The reference model, used for initialization, can be viewed as playing a role very similar to our background model. Allassonnière, Amit and Trouvé (2007) developed a Bayesian framework for *deformable* templates, which could be learned through a version of the EM algorithm. The approach was demonstrated by implementing a system for handwritten digit recognition. Sabuncu, Balci and Golland (2008) later adapted the method to the problem of registering and clustering whole brain MR images. The resulting templates defined clusters of individuals that were interpretable through their correlations with age and pathology.

The generative models developed by Amit and collaborators [e.g., Amit, Geman and Fan (2004), Amit and Trouvé (2006, 2007)] are also closely related. These models generate a binary edge map rather than pixel intensities *per se*, but as pointed out in Amit, Geman and Fan (2004), they can also be viewed as generating intensities by assuming a uniform distribution on the set of intensity images consistent with the generated values of the binary features. In this direction, the intensity differences that are thresholded by Amit et al. to form edges could be generalized to zero-mean templates ("differential operators"), which could then be learned from data under the assumption of a conditionally uniform distribution.

Many of the other aspects of our approach (as discussed in Sections 2–4) would then be in place, including the use of mixtures over poses (aka "spreading" in Amit et al.) and over templates [Amit and Trouvé (2006)].

We begin the next section, Section 2, by observing that the likelihood equations can be manipulated to depend on the low-dimensional distribution $p_S^o(s)$ rather than the high-dimensional distribution $p_Y^o(y|S = s(y))$ that appears in (1.3). We observe, furthermore, that with no important changes the approach generalizes to mixture models, which will be used in Section 4 to estimate mixtures over statistics, s, as well as mixtures over poses (translations, scales, and rotations). In Section 3, we focus on the particulars of the estimation problem for the special case in which the chosen statistics are normalized correlations, and we formulate three background models for later comparison. The results from a variety of experiments are discussed in Section 4, including maximum-likelihood features learned within mixture models for noses and eyes, coarse features learned for whole faces, learning mixture models for patches drawn from natural images, the use of PCA templates in a generative model, and an approach to drawing samples from these models. Section 5 concludes with a summary and some challenges.

2. Background factoring, likelihood ratios, and the inclusion of mixtures. The goal is to learn both ϕ and θ in the category-specific model $p_Y(y) = p_S(s(y;\phi);\theta)p_Y^o(y|S = s(y;\phi))$, given a sample of image patches, $y_1, \ldots, y_N$, from the category of interest. If we happen to know ϕ, then $s(y;\phi)$ is a sufficient statistic for θ, and, as already observed, the maximum-likelihood estimator is the maximizer of (1.4). On the other hand, if ϕ is unknown then the likelihood is given by equation (1.3), which includes the problematic conditional distribution $p_Y^o(y|S = s(y;\phi))$. This is no doubt a complicated distribution, in that p_Y^o defines the small fraction of possible patches that are likely to actually show up in real images—those with a measure of continuity across neighboring pixels, along with the occasional specular reflection, shadow boundary, and so on. Of course, we can always factor p_Y^o as

$$p_Y^o(y) = p_S^o\big(s(y;\phi)\big)p_Y^o\big(y|S = s(y;\phi)\big). \tag{2.1}$$

Appearances aside, p_Y^o itself depends on neither θ *nor* ϕ. We can then rewrite the likelihood, (1.3), as

$$\begin{aligned} L(y_1, \ldots, y_N;\theta,\phi) &= \prod_{k=1}^{N} p_S\big(s(y_k;\phi);\theta\big)p_Y^o\big(y_k|S = s(y_k;\phi)\big) \\ &= p_{Y_{1:N}}^o(y_{1:N}) \prod_{k=1}^{N} \frac{p_S(s(y_k;\phi);\theta)}{p_S^o(s(y_k;\phi))}, \end{aligned} \tag{2.2}$$

where $p_{Y_{1:N}}^o(y_{1:N})$ is shorthand for $\prod_{k=1}^N p_Y^o(y_k)$, and is independent of the parameters. The point being that the ratio, $\prod_{k=1}^N p_S(s(y_k;\phi);\theta)/p_S^o(s(y_k;\phi))$, is

much more manageable than (1.3), at least when s is low dimensional (e.g., the one-dimensional correlation with a template $\phi = T$). Depending on how the background is conceived (see the discussion in the following section, Section 3), we typically have an inexhaustible supply of background patches, which makes it a relatively easy matter to estimate $p_Y^o(s(y_k;\phi))$ for any given value of ϕ.

Unless it is severely under-sampled, a typical category, such as our prototypical example, right eyes, is much too rich to model through a single, one-dimensional statistic. A more sensible approach is to use a mixture of models, of the same type as (1.2), but mixed over multiple statistics: $s(y) \to \{s_m(y)\}_{m\in\{1,\ldots,M\}}$. Each mixing component requires its own features and parameters $[s_m(y) \to s_m(y;\phi_m)$, $p_{S_m}(s_m(y;\phi_m)) \to p_{S_m}(s_m(y;\phi_m);\theta_m)]$, and its own mixing probability $\varepsilon_m, m = 1,\ldots,M$, leading to a more comprehensive category-specific patch model:

$$p_Y(y) = \sum_{m=1}^{M} \varepsilon_m p_{S_m}\big(s_m(y;\phi_m);\theta_m\big) p_Y^o\big(y|S_m = s_m(y;\phi_m)\big).$$

This same framework accommodates pose, so that a given statistic (say template-based) appears repeatedly in the mixture, for example, as an ensemble of rotations, scales, and within-patch translations of a given template. Various kinds of mixtures will be explored, experimentally, in Section 4.

The factorization of the background model, (2.1), can be used promiscuously for each of the statistics $s_m, m = 1,\ldots,M$, leading to the mixture-based generalization of (2.2):

$$\begin{aligned} &L\big(y_1,\ldots,y_N;\{\phi_m,\theta_m,\varepsilon_m\}_{m=1}^M\big) \\ &\quad= \prod_{k=1}^{N}\sum_{m=1}^{M} \varepsilon_m p_{S_m}\big(s_m(y_k;\phi_m);\theta_m\big) p_Y^o\big(y_k|S_m = s_m(y_k;\phi_m)\big) \\ &\quad= p_{Y_{1:N}}^o(y_{1:N}) \prod_{k=1}^{N}\sum_{m=1}^{M} \varepsilon_m \frac{p_{S_m}(s_m(y_k;\phi_m);\theta_m)}{p_{S_m}^o(s_m(y_k;\phi_m))}. \end{aligned} \tag{2.3}$$

We note, finally, that comparing categories, as in a classification experiment, is simply a matter of applying the Neyman–Pearson lemma and exploiting these same factorizations to avoid high-dimensional conditional distributions. For example, a test for "category 1," with data model $p_Y^1(y)$, against "category 2," with data model $p_Y^2(y)$, is made by thresholding on the ratio:

$$\begin{aligned} \frac{p_Y^1(y)}{p_Y^2(y)} &= \frac{p_Y^1(y)/p_Y^o(y)}{p_Y^2(y)/p_Y^o(y)} \\ &= \frac{\sum_{m=1}^{M^1} \varepsilon_m^1 p_{S_m^1}^1(s_m^1(y;\phi_m^1);\theta_m^1)/p_{S_m^1}^o(s_m^1(y;\phi_m^1))}{\sum_{m=1}^{M^2} \varepsilon_m^2 p_{S_m^2}^2(s_m^2(y;\phi_m^2);\theta_m^2)/p_{S_m^2}^o(s_m^2(y;\phi_m^2))}, \end{aligned}$$

where y is the observed patch and, as needed, the superscript 1 or 2 has been used to differentiate category-specific variables. Evidently, absent additional assumptions, a proper decision between object categories will need to take into account the background distributions on the category-specific sufficient statistics.

3. Normalized correlation, background models, estimation. In anticipation of the experiments in Section 4, we will focus on the specific choice of the correlation statistic for the features, $s_m, m = 1, \ldots, M$, and examine three candidate background distributions. Given a chosen set of features and a background model, we discuss the computational problem of maximizing the likelihood, (2.3), and propose some iterative algorithms for later experimentation.

3.1. *Normalized correlation.* Mostly, we have experimented with the particular statistic $s(y) = \text{corr}(T, y)$: the template T is then the "parameter" ϕ that defines the "feature" s. Normalized correlation is commonly used as a feature, largely because it is invariant to linear transformations of the pixel's intensities, making it robust to illumination artifacts.[3] Let n be the number of pixels in the patch, y. Without loss of generality, we can restrict T to have mean zero $[\frac{1}{n}\sum_i T(i) = 0$, where i indexes the elements of the array] and variance one $[\frac{1}{n}\sum_i T(i)^2 = 1]$, in which case

$$s(y) = s(y; T) = \frac{\langle T, y\rangle}{\hat{\sigma}(y)} \in [-1, 1],$$

where $\langle \cdot, \cdot \rangle$ is inner product and $\hat{\sigma}(y)^2 = \frac{1}{n}\sum_i (y(i) - \bar{y})^2$ is the sample variance.

More generally, we are interested in a mixture of models for some target category, one model for each of M templates, in which case

$$p_Y(y) = \sum_{m=1}^{M} \varepsilon_m p_{S_m}\big(s_m(y)\big) p_Y^o\big(y | S_m = s_m(y)\big), \tag{3.1}$$

where $s_m(y) = \text{corr}(T_m, y)$. If we think of the templates as prototypical examples of objects in the category, then it is natural to try a model for $p_{S_m}(s)$ which is a monotone increasing function on $[-1, 1]$. We have used the exponential function:

$$p_{S_m}(s) = \alpha_{\lambda_m} e^{-\lambda_m(1-s)}, \tag{3.2}$$

[3]Of course, there are many other statistics that are robust to illumination and worth exploring, depending on the application, such as Spearman's ρ and Kendall's τ, which are in fact invariant to all rank-preserving transformations, or statistics that include "nuisance parameters," responsible for normalizing the location or scale of the intensity distribution in a local image patch. We make note of the fact that the methods we are advocating, and in particular the factorizations that they rely on, do not depend on the particulars of the chosen statistics.

where, for each $m = 1, \ldots, M$, $\lambda_m > 0$ and

$$\alpha_{\lambda_m} = \left(\int_{s=-1}^{1} e^{-\lambda_m (1-s)}\, ds \right)^{-1} = \frac{\lambda_m}{1 - e^{-2\lambda_m}}$$

is the normalizing constant. Or, to make the connection to the more general notation of Section 2, we take $\phi_m = T_m$ and $\theta_m = \lambda_m$. Then, according to (2.3), the maximum likelihood estimators for the parameters (i.e., for the M templates, $\{T_m\}_{m=1,\ldots,M}$, and the $2M$ scalars, $\{\lambda_m, \varepsilon_m\}_{m=1,\ldots,M}$), given a sample $y_1, \ldots, y_N$ of patches from the target category, can be found by maximizing

$$\prod_{k=1}^{N} \sum_{m=1}^{M} \varepsilon_m \frac{\frac{\lambda_m}{1-e^{-2\lambda_m}} e^{-\lambda_m (1 - s_m(y_k))}}{p^o_{S_m}(s_m(y_k))}, \tag{3.3}$$

where $\varepsilon_1, \ldots, \varepsilon_M$ are constrained to define a probability mass function, and $s_m(y_k) = \mathrm{corr}(T_m, y_k)$.

Before turning to background models, we wish to make a final observation about the appropriateness of the exponential model or, for that matter, anything monotone on $[-1, 1]$. If, after training, we look at the empirical distribution on $S_m(y)$, using, say, $y = y_1, \ldots, y_N$, we find that it is indeed monotone increasing from -1 until entering a small neighborhood near the value 1, at which point it rapidly *decreases* to zero. This is because of the "entropic" or "combinatorial" term; there are very few ways to make the correlation nearly 1. On the other hand, such image patches are extremely rare and we have noticed little or no effect on performance when using a simple and convenient (reverse) exponential instead of a more appropriate but complex parametric form.

3.2. *Background models.* To complete the formulation, we need to specify a background model $p^o_Y(y)$, or, at the least, its marginalized distribution on each of the M random variables $S_m = \mathrm{corr}(T_m, Y)$, $m = 1, \ldots, M$. We have experimented with three background models. Many variations are plausible, and, not surprisingly, better models produce better results (see Section 4).

i. *Independent and identically distributed pixel intensities ("i.i.d. background model").* Of course, backgrounds are not i.i.d., but this is a convenient place to start. Let n be the size of the image region being modeled, as measured by the total number of pixels, for example, $n = 1200$ for the 30×40 sized patches used in the experiments with right eye appearance modeling reported in Section 4.1. Then, by an application of the (Lyapunov) central limit theorem (Section A.2 of the Appendix), $\mathrm{corr}(T, y)$ is approximately normal, with mean zero and variance $1/n$. Hence, we approximate $p^o_{S_m}(s_m(y_k))$ by

$$\sqrt{\frac{n}{2\pi}} e^{-\frac{n s_m(y_k)^2}{2}}. \tag{3.4}$$

ii. *Smooth patches drawn from natural images ("natural-image background model").* If we think of the goal of modeling patches as creating a library of appearance models for common parts and objects (edges, boundaries, eyes, mouths, faces, and so on), and "background" as referring to regions that are essentially unstructured and less informative, or at least less semantically meaningful, then we might define, more-or-less by fiat, background patches to be those that are absent from the sharp boundaries that characterize most familiar structures. With this idea in mind, consider the ensemble of all patches in natural images that have a bounded maximum gradient, in the following sense:

$$\max_{i\in\{1,\ldots,n\}} \frac{|\nabla_i(y)|}{\hat{\sigma}(y)} < \eta, \tag{3.5}$$

where $|\cdot|$ is length in $\mathbb{R}^2$, and the gradient at location i, ∇_i, is the discrete approximation that uses the difference in neighboring horizontal and vertical pixel intensities to approximate the horizontal and vertical partial derivatives, respectively. The denominator (the sample standard deviation) is included to provide a measure of lighting invariance, which is quite precise for so-called linear data, but rather crude for "log" data or other camera-specific manipulations of the intensities. In our experiments, we used the threshold $\eta = 0.3$.

The problem with (3.5), as it stands, is that natural images have a weak but detectable average gradient, that runs from top-to-bottom and dark-to-light. When using the correlation statistic, a weak gradient is as important as a strong gradient, because of the normalization. To eliminate this effect, we defined the ensemble of background patches to be the result of (3.5), but applied to natural images that had first been randomly and uniformly rotated.

To collect data, we selected high-quality and uncompressed images of natural surroundings from various web sites devoted to photography. As for the distribution on the correlation statistics, S_m, we found that these were well approximated by zero-mean Gaussian distributions, with a variance that is somewhat dependent on the particular template, T_m. Hence, we modeled the background distribution on S_m with $N(0, \sigma_m^2)$, and used the simple empirical estimate of σ_m^2 for any given template T_m.

iii. *Gaussian random fields ("GRF background model").* We also experimented with a Gaussian random field model, reasoning that GRFs might make for a good fit to the smooth patches defined in (ii), given the absence of sharp boundaries in the ensemble defined by (3.5). The GRF was generated by convolving an i.i.d. array of standard normal random variables with a circularly-symmetric Gaussian kernel, mean zero and standard deviation 5, to produce a random field model of pixel intensities. The kernel bandwidth, 5, was adjusted, "by eye," to best match the appearance of samples from the ensemble of smooth patches. Needless to say, these natural-image patches could be more carefully modeled (e.g., by estimating a covariance matrix), especially given the essentially unlimited supply of examples.

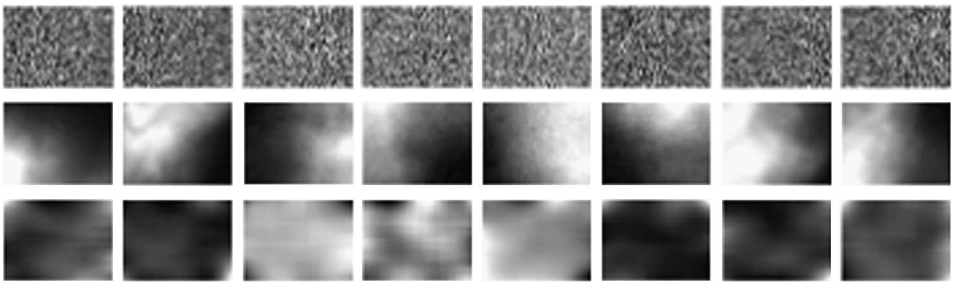


FIG. 1. Background models. *Eight random samples from each of three background models* (*see Section* 3.2). *Top row*: *i.i.d. uniform*. *Middle row*: *randomly selected "structureless" patches* [*see* (3.5)] *from randomly selected natural images*. *Bottom row*: *samples from a Gaussian random field formulated to look like the ensemble of structureless patches*.

In any case, the resulting empirical distribution of S_m is again very nearly zero-mean Gaussian, with variance that depends (weakly) on the template T_m, which is likely a reflection of stronger versions of the central limit theorem that allow for (restricted) dependency among the random variables. As in the previous model, we have available an inexhaustible supply of examples for estimating σ_m^2, for each m.

Figure 1 shows eight 30×40 image patches sampled from each of the three models.

Obviously, none of these background models are correct, in the sense of producing a reasonable model of what might be seen away from any given category or small set of categories of objects. Indeed, background is a relative thing—relative to a library of already-modeled objects. And unless we have built a rather complete library, it is not reasonable to think of backgrounds as exclusively smooth (much less i.i.d.). On the other hand, as an idealization, smooth background models are not unreasonable in that (1) they represent an appropriate *goal* for a system that is learning to recognize structure, and (2) when conditioned on a well-chosen sufficient statistic, they give excellent qualitative results as can be seen by sampling, Section 4.6, and performance results as can be seen in ROC experiments, Section 4.1.

3.3. *Estimation*. We follow the maximum-likelihood principle. Given a sample of image patches, $y_1, \ldots, y_N$, from a given target category (e.g., instances of right eyes), we wish to learn a category-specific appearance model, $p_Y(y)$. If we use the mixture model (3.1) for $p_Y(y)$, and the correlation statistics $S_m = \mathrm{corr}(T_m, Y)$, with exponential distributions (3.2), for p_{S_m}, $m = 1, \ldots, M$, then the likelihood for the mixing probabilities $\varepsilon_1, \ldots, \varepsilon_M$, the templates $T_1, \ldots, T_M$, and the exponents $\lambda_1, \ldots, \lambda_M$ is (3.3), but multiplied by a factor [namely, $p^o_{Y_{1:N}}(y_{1:N})$] that is independent of these parameters.

Consider first the simple i.i.d. background model [model (i)], in which case, up to the CLT, $p^o_{S_m}(s)$ is $N(0, 1/\sqrt{n})$ [referring to equation (3.4)]. With this approxi-

mation,

$$L\big(y_1,\ldots,y_N;\{\varepsilon_m,T_m,\lambda_m\}_{m=1}^M\big) = p^o_{Y_{1:N}}(y_{1:N})\prod_{k=1}^N\sum_{m=1}^M \varepsilon_m \frac{\frac{\lambda_m}{1-e^{-2\lambda_m}}e^{-\lambda_m(1-s_m(y_k))}}{\sqrt{\frac{n}{2\pi}}e^{-\frac{ns_m(y_k)^2}{2}}} \tag{3.6}$$

with $s_m(y_k)=\mathrm{corr}(T_m,y_k)$. Since there are no additional parameters in the denominator terms, $p^o_{S_m}(s_m(y_k))$, $m=1,\ldots,M$, estimation can proceed using a version of EM, Dempster, Laird and Rubin (1977). The only aspect of our implementation that is worth noting is that we use a gradient ascent algorithm for computing the templates, $T_1,\ldots,T_M$, in the "M" step, constrained by the assumed standardizations, $\sum_i T_m(i)=0$ and $\sum_i T_m(i)^2=1$, for each m. Templates were initialized using i.i.d. standard normal random variables, followed by a location and scale change to satisfy the constraints, and we started with equal mixing probabilities ($\varepsilon_m=1/M,\forall m$) and all of the λ's set to one.

Concerning models (ii) and (iii), we have already observed that in both cases the marginal distributions on the statistics $S_1,\ldots,S_M$ are well approximated as zero-mean normals, with standard deviations that vary, to a degree, as a function of m. Thus, the likelihood, for each of these models, has the form

$$L\big(y_1,\ldots,y_N;\{\varepsilon_m,T_m,\lambda_m\}_{m=1}^M\big) = p^o_{Y_{1:N}}(y_{1:N})\prod_{k=1}^N\sum_{m=1}^M \varepsilon_m \frac{\frac{\lambda_m}{1-e^{-2\lambda_m}}e^{-\lambda_m(1-s_m(y_k))}}{\frac{1}{\sqrt{2\pi\sigma_m^2}}e^{-\frac{s_m(y_k)^2}{2\sigma_m^2}}}. \tag{3.7}$$

The equation is somewhat deceptive, since it would appear that the likelihood depends on the additional parameters $\sigma_1,\ldots,\sigma_M$. But in fact each σ_m is actually a function of the corresponding template, T_m: $\sigma_m=\sigma_m(T_m)$. Specifically, σ_m is the standard deviation of the statistic $S_m=\mathrm{corr}(T_m,Y)$ under the particular background distribution $p^o_Y(y)$ on Y. The i.i.d. case was special, in that the standard deviation was known, before hand, to be well approximated by $1/\sqrt{n}$.

For any given template, T_m, the standard deviation of $\sigma_m(T_m)$ is easy to estimate from the wealth of examples that are easily produced for each of the two models. But the general, analytic, form of the relationship is complicated. This makes the inner-loop maximization over T_m more difficult. Many approaches could be taken. We chose a simple modification of the EM procedure, which involved alternating between running EM at fixed values of the standard deviations and updating the standard deviations at fixed values of the templates. We used the parameters delivered by the i.i.d. model for initialization. The details are presented below as pseudocode; see Algorithm 1.

We did not experiment extensively with other approaches, some of which would likely have been more effective. We will briefly discuss one alternative in the concluding section, Section 5.

Algorithm 1 Estimation of $\{\varepsilon_m, T_m, \lambda_m\}_{m=1}^M$ under background model (ii) or (iii)

% initialization
for $m = 1$ to M **do**
  $\varepsilon_m \leftarrow \frac{1}{n}$, $\lambda_m \leftarrow 1$, $T_m \leftarrow$ i.i.d random template (standardized)
end for

% EM with i.i.d. background model for initial estimates of ε_m, λ_m, T_m
repeat
  Expectation, equation (3.6)
  Maximization, equation (3.6)
until convergence of $\{\varepsilon_m, T_m, \lambda_m\}_{m=1}^M$

% alternate between sample estimates of a *common* background σ, and
% EM estimates of ε_m, λ_m, T_m
repeat
  collect background samples
  estimate $\sigma_1(T_1), \ldots, \sigma_M(T_M)$ *assuming common* σ: $\sigma = \sigma_1 = \cdots = \sigma_M$
  repeat
    Expectation, equation (3.7), common and fixed σ
    Maximization, equation (3.7), common and fixed σ
  until convergence of $\{\varepsilon_m, T_m, \lambda_m\}_{m=1}^M$
until convergence of $\{\varepsilon_m, T_m, \lambda_m, \sigma\}_{m=1}^M$

% estimate individual σ's
for $m = 1$ to M **do**
  estimate $\sigma_m = \sigma_m(T_m)$ from background samples
end for

% final estimates of category-specific parameters $\{\varepsilon_m, \lambda_m\}_{m=1}^M$
repeat
  Expectation, equation (3.7), $\sigma_1, \ldots, \sigma_M$ and $T_1, \ldots, T_M$ fixed
  Maximization, equation (3.7), $\sigma_1, \ldots, \sigma_M$ and $T_1, \ldots, T_M$ fixed
until convergence of $\{\varepsilon_m, \lambda_m\}_{m=1}^M$

4. Experiments. The feasibility and flexibility of the approach are demonstrated through a series of experiments with ensembles of image patches. We learn coarse and fine features, and we learn mixture models, mixing over both features and poses for a given category of parts or objects, including, simply, the category of random patches from a library of natural images. We demonstrate the potential advantage of using full-data likelihoods, as opposed to models that are only partially specified in that they include only feature probabilities rather than probabilities of

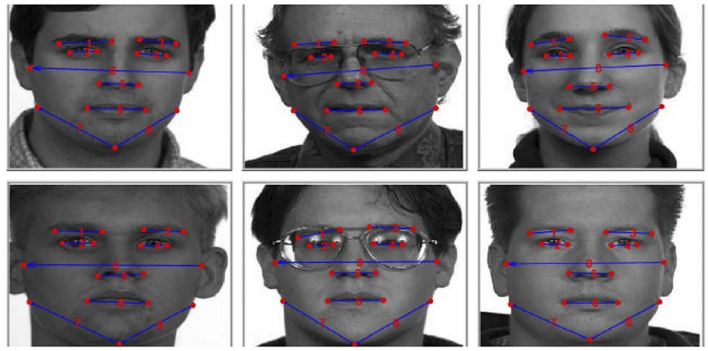


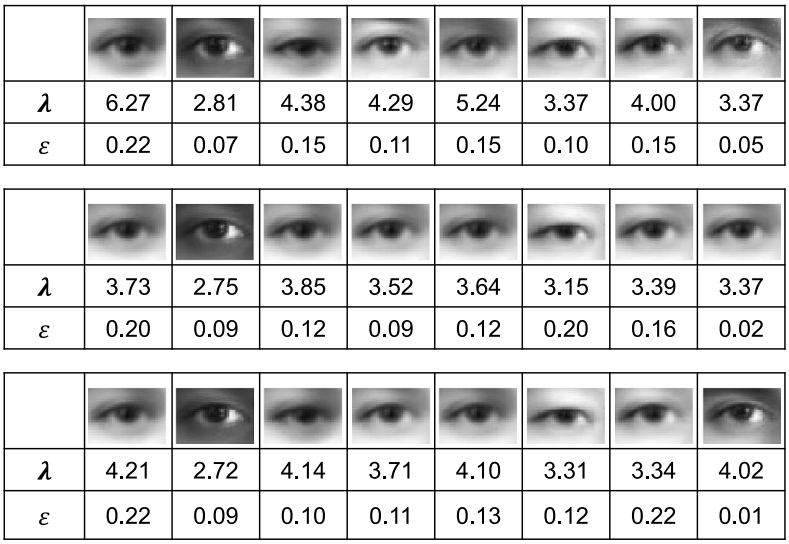
FIG. 2. Feret database. *Six examples drawn from the Feret Database. Each of the* 499 *faces in the database is manually labelled with* 19 *landmarks, locating the eyes, eye brows, nose, mouth, chin, and cheeks.*

the pixel intensities themselves. We illustrate the fully generative nature of the approach by devising and experimenting with an approximate sampling scheme. We examine the importance of the background model through an informal comparison of the appearances of category-based samples, and a formal comparison of ROC performances in an object detection problem. We also compare performances to a generative version of principal component analysis (PCA) and to the time-honored Gaussian mixture model.

4.1. *Eye models and eye detection.* The training data was extracted from 499 images taken from the Feret Database (http://www.itl.nist.gov/iad/humanid/feret/feret_master.html). Each image in the database consists of a face and 19 manually labeled landmarks (Figure 2). The landmarks were used to define $N = 499$ 30×40 training patches, $y_1, \ldots, y_N$, each with the right eye centered in the patch, and each with the same orientation and scale—a selection of 70 training patches is shown in Figure 3.

Appearance models were learned using each of the three background models (Section 3.2) and the respective category-specific likelihoods: (3.6) for model (i)

FIG. 3. Right-eye training set. *Sampling from the* 499 *right-eye image patches extracted from the Feret Database and resized to* 30×40 *pixels.*



λ	6.27	2.81	4.38	4.29	5.24	3.37	4.00	3.37
ε	0.22	0.07	0.15	0.11	0.15	0.10	0.15	0.05

λ	3.73	2.75	3.85	3.52	3.64	3.15	3.39	3.37
ε	0.20	0.09	0.12	0.09	0.12	0.20	0.16	0.02

λ	4.21	2.72	4.14	3.71	4.10	3.31	3.34	4.02
ε	0.22	0.09	0.10	0.11	0.13	0.12	0.22	0.01

FIG. 4. Estimated templates. *Each panel shows the eight templates* (*prototypical right-eye patches*) *inferred from the training set under one of the three background models* (*Section* 3.2), *using an approximate maximum-likelihood algorithm* (*Section* 3.3). *The corresponding parameters* $\lambda_1, \ldots, \lambda_8$ [*see equation* (3.2)], *for the correlation distributions, and* $\varepsilon_1, \ldots, \varepsilon_8$, *for the mixing weights, appear underneath the templates. The background models for the top, middle, and bottom panels are the i.i.d., natural-image, and GRF models, respectively* (*Section* 3.2).

and (3.7) for models (ii) and (iii). In all cases, the number of mixing components was set, arbitrarily, at eight ($M = 8$). Figure 4 shows the resulting templates, $T_1, \ldots, T_8$, and mixing probabilities, $\varepsilon_1, \ldots, \varepsilon_8$, as well as the exponents $\lambda_1, \ldots, \lambda_8$ that define the feature distributions. A cursory examination of the results indicates only a weak dependence on the background models, especially for the templates. In part, this reflects the decision to initialize parameter values, under models (ii) and (iii), with those already determined under model (i); see the algorithm, and the earlier discussion. But it also raises the question of whether generative models based on more realistic backgrounds will perform better at object detection.

To explore this question, we compared three likelihood-ratio classifiers, one for each of the three background models:

$$H_t(y) = \begin{cases} 1 & \text{if } \dfrac{p_Y(y)}{p_Y^o(y)} > t, \\ -1 & \text{if } \dfrac{p_Y(y)}{p_Y^o(y)} \le t. \end{cases}$$

In each case, p_Y is the eight-component mixture model, as estimated above, and p_Y^o is the corresponding background model. We used the 499 right-eye training images as positive samples, and collected 4740 random 30×40-pixel image patches,

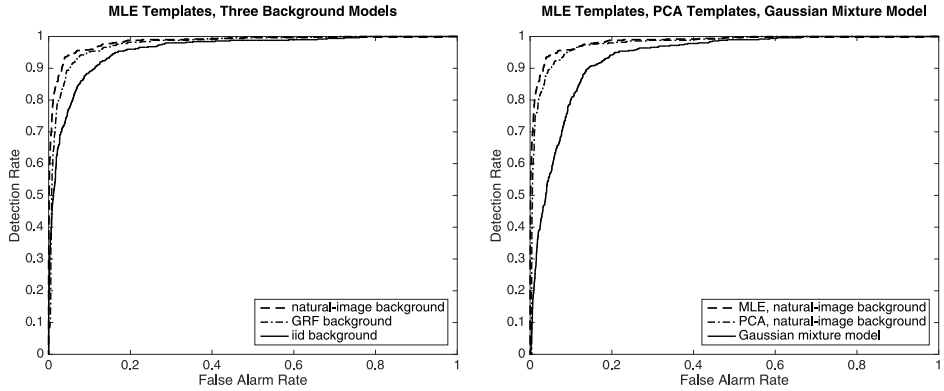


FIG. 5. Classification experiments. *Left-hand panel*: *Features learned under each of the three background models* (*Section* 3.2) *were used in a classification experiment*, *distinguishing right–eye patches from randomly selected background patches*. *The three ROC curves show the averaged performances from* 10*-fold cross-validations*, *under the natural-image* (*dashed line*), *GRF* (*dashed/dotted line*), *and i.i.d.* (*solid line*) *models*. *Right-hand panel*: *Same classification experiment*, *comparing* (i) *the eight-fold mixture model from Section* 4.1, *trained under the natural-image background model* (*dashed line—same curve as in left-hand panel*); (ii) *the eight-component generative PCA model developed in Section* 4.7 (*dashed/dotted line*); *and* (iii) *a standard*, *eight-fold Gaussian mixture model* (*solid line*).

from natural images downloaded from the Internet, as negative samples. The set of the positive and negative samples was partitioned into 10 equal-size subsets for 10-fold cross-validation: for each of the three background models, each of the 10 subsets was used for testing a classifier estimated from the data in the remaining 9 subsets. For each background model and each testing set, a ROC curve was swept out by varying t from zero to infinity. Then, for each background model, the 10 ROC curves were averaged to produce the three results shown in the left-hand panel of Figure 5 (where the curves are labeled "i.i.d. background," "GRF background," and "natural-image background").

Despite the similar appearances of samples from the smooth natural background (ii) and GRF (iii) models, the ensemble of smooth-background patches performs somewhat better than the GRF model. And both models significantly improve on the i.i.d. model. Keeping in mind that each background model defines a fully generative category-specific appearance model, another means of comparison is through an examination of samples from these generative models. See Section 4.6 where we compare (approximate) samples from appearance models for mouths, based on each of the three background models.

The model estimated from smooth image patches was also compared to a standard, eight-fold Gaussian mixture model and to a generative version of PCA, based on an eight-dimensional sufficient statistic (developed in Section 4.7). Concerning the Gaussian mixture, we estimated a single "on-target" covariance function from the training set, and a second, single, covariance function for the "null" model from

background patches. Performance was based on likelihood ratios. The results for both the PCA and Gaussian mixture models are summarized in the ROC curves of the right-hand panel of Figure 5. All of the models represented in the two panels were trained on the same data. The Gaussian mixture model performs poorly as compared to any one of the models built out of sufficient statistics, none of which are Gaussian.

Going forward, we will be using the natural-image model for the background, unless otherwise noted.

4.2. *Data likelihood versus feature likelihood.* Suppose, instead, we use feature likelihoods in place of data likelihoods. Are the results substantially different? We compared an eight-template mixture model ($M = 8$) learned from *feature* likelihoods:

$$\prod_{k=1}^{N} \sum_{m=1}^{M} \varepsilon_m \frac{\lambda_m}{1 - e^{-2\lambda_m}} e^{-\lambda_m (1 - s_m(y_k))} \tag{4.1}$$

to the eight-template mixture model learned previously using *data* likelihoods. Feature likelihoods, (4.1), perform poorly. Indeed, up to minor variations, equation (4.1) produces a single template; there is no meaningful mixture. See Figure 6 for a comparison of the eight templates learned under the two approaches. Evidently, the complete data likelihood, which includes the *combinatorial* (or *entropic*) terms $p_Y^o(y|S_m = s_m(y)), m = 1, \ldots, M$, is substantially more sensitive to the correlations $s_m(y) = \text{corr}(T, y), m = 1, \ldots, M$, resulting in an appearance model which better fits the variability of eyes in the training data.

4.3. *Mixing over pose.* We cannot assume, in general, that a set of training samples for the appearance of an object or part will include a precise pose. Although the hand-labeled landmarks in the Feret Database are sufficient to compute the poses of faces and certain parts, which we used to advantage in learning a mixture of right-eye templates, most training sets are labeled with less detailed

FIG. 6. Data likelihood versus feature likelihood. *Top row*: *Right-eye patches learned from the full data likelihood. Bottom row*: *Patches learned using the feature likelihood. The eight templates in the bottom row are nearly identical to each other*; *the resulting mixture model fails to capture the variety of appearances seen in the training set* (*cf. Figure* 3).

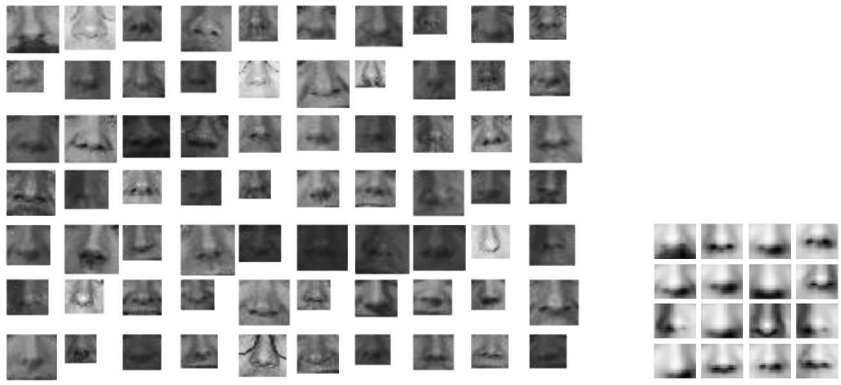


FIG. 7. Mixing over pose. *Left-hand panel*: 120 *training patches drawn from the* 499 *faces in the Feret Database*, *with variations in rotation and scale*. *Right-hand panel*: 16 *learned templates*, *each of size* 15×18, *learned from the fully generalized model*, *including mixtures over multiple templates*, *spatial shifts*, *scales*, *and rotations*.

information. Furthermore, landmarks, when provided, are usually subjective and, therefore, variable.

Formally, we can just as well consider a mixture to be defined by prototypical templates *and* their poses, rather than just by the templates themselves. To illustrate the idea, we created a set of image patches, of variable sizes, containing noses extracted from the Feret Database set, but randomly rotated and scaled. Figure 7, left-hand panel, shows 120 examples of the resulting 499 patches, constituting a training set with unlabeled poses. Rotations and scales were chosen independently, and from the uniform distributions on $[-10°, 10°]$ and $[0.3, 0.5]$, respectively. The resulting images were cropped to contain the full width of the nose and, approximately, the bridge of the nose and a portion of the upper lip. The training patches ranged in size from 16×18 to 30×33.

The goal was to learn a set of M templates, each of size 15×18, through a mixture model that mixes over each of the templates at each of a discrete number of poses. Given a scale z, a rotation r, and a (two-dimensional) translation l, let $\Psi_{z,r,l}$ be the transformation that maps a template T into the image patch that results from first scaling, then rotating, and then translating T accordingly. In general, the transformed template, $\Psi_{z,r,l}(T)$, is defined on a parallelogram-shaped array of pixels. Let $\mathbb{Z}$ be the set of allowed scales and $\mathbb{R}$ the set of allowed rotations, and, for each $(z, r) \in \mathbb{Z} \times \mathbb{R}$, let $\mathbb{L}_{z,r}$ be the set of allowed translations. $\mathbb{R}$, $\mathbb{Z}$, and $\mathbb{L}_{z,r}$ are chosen sufficiently large so that the 15×18 sized templates can be positioned, through $\Psi_{z,r,l}$, to match images in the target range of scales, rotations, and translations, for example, the ensemble of noses derived from the Feret Database and sampled in the left-hand panel of Figure 7.

The appearance model needs to accommodate multiple-sized image patches, which can be accomplished with no important changes in the approach. Given

an image patch y, of variable dimensions, and given a transformation $\Psi_{z,r,l}$ with $z \in \mathbb{Z}$, $r \in \mathbb{R}$, and $l \in \mathbb{L}_{z,r}$, let $\mathcal{A}_{z,r,l}$ be the subset of pixels in y which are also contained in $\Psi_{z,r,l}(T)$ [i.e., the intersection of the pixels in y with those in $\Psi_{z,r,l}(T)$]. Observe that $\mathcal{A}_{z,r,l}$ is independent of T: prior to the transformation of pose, the templates are all the same size (15×18 in the current experiment). The component of the mixture model for y associated with the template T_m, $m = 1, \ldots, M$, is itself a mixture over poses, (z, r, l). For fixed m, z, r, and l, the feature is again a correlation, but this time confined to the pixels in $\mathcal{A}_{z,r,l}$:

$$s_{m,z,r,l} \triangleq \operatorname{corr}\bigl(\Psi_{z,r,l}(T_m)_{\mathcal{A}_{z,r,l}}, y_{\mathcal{A}_{z,r,l}}\bigr),$$

where, given any image patch $\tilde{y}$, and any subset of pixels in $\tilde{y}$, say $\mathcal{B}$, we write $\tilde{y}_{\mathcal{B}}$ for the corresponding set of pixel intensities.

Since the correlation is normalized, we assume that the distribution on $S_{m,z,r,l} = s_{m,z,r,l}(Y)$ depends only on m, and reuse the model (3.2):

$$p_{S_{m,z,r,l}}(s) = \frac{\lambda_m}{1 - e^{-2\lambda_m}} e^{-\lambda_m(1-s)}.$$

Finally, we assume that, given m, the choices of scale and rotation are independent, and that given z and r, the translation, l, is uniform on $\mathbb{L}_{z,r}$. Letting ε_m be the mixing probability on templates, and δ_z^m and η_r^m be the conditional probabilities on scale and rotation, given m, we arrive at the data likelihood, for mixed-sized patches $y_k, k = 1 \ldots, N$,

$$L\bigl(y_1, \ldots, y_N; \{\varepsilon_m, T_m, \lambda_m, \{\delta_z^m, \eta_r^m\}_{z\in\mathbb{Z}, r\in\mathbb{R}}\}_{m=1}^M\bigr)$$
$$= p^o_{Y_{1:N}}(y_{1:N}) \prod_{k=1}^N \sum_{m=1}^M \sum_{z\in\mathbb{Z}, r\in\mathbb{R}} \sum_{l\in\mathbb{L}_{z,r}} \varepsilon_m \delta_z^m \eta_r^m \frac{1}{|\mathbb{L}_{z,r}|} \frac{\frac{\lambda_m}{1-e^{-2\lambda_m}} e^{-\lambda_m(1-s_{m,z,r,l}(y_k))}}{p^o_{S_{m,z,r,l}}(s_{m,z,r,l}(y_k))}.$$

Here, $p^o_{S_{m,z,r,l}}$ refers to the smooth, natural-image background model [i.e., (ii) of Section 3.2], except that the variance necessarily scales with the number of pixels being correlated, that is, the size of $\mathcal{A}_{z,r,l}$. The natural scaling divides the variance by $n_{z,r,l} \triangleq |\mathcal{A}_{z,r,l}|$ [e.g., consider the i.i.d. case, (i) of Section 3.2], in which case

$$p^o_{S_{m,z,r,l}}\bigl(s_{m,z,r,l}(y_k)\bigr) = \sqrt{\frac{n_{z,r,l}}{2\pi\sigma_m^2}} e^{-\frac{n_{z,r,l} s_{m,z,r,l}(y_k)^2}{2\sigma_m^2}}.$$

In our experiments we used the scales $\mathbb{Z} = \{0.83, 1, 1.17\}$, the rotations $\mathbb{R} = \{-6.7°, 0.0°, 6.7°\}$ and, as remarked earlier, a set of translations, $\mathbb{L}_{z,r}$, chosen to be large enough to ensure the existence of poses that would register a transformed template image onto a sample from the target population of scaled and rotated noses. As in Section 4.1, training was by the modified expectation/maximization procedure presented, in the form of pseudocode, in the algorithm of Section 3.3. The result was the sixteen varied templates shown in the right-hand panel of Figure 7, each of which appears quite natural.

4.4. *Coarse representations*. Many considerations go into selecting good features for category detection. Eyes, noses, and mouths are obviously informative for the detection of faces, and nearly essential for the identification of individuals or ethnic characterization. Larger features, encompassing entire faces, are generally more specific than individual parts, but they are also more complex and typically too brittle for the identification of individuals. Ullman, Vidal-Naquet and Sali (2002) have argued that there is an intermediate range of complexity that characterizes the most informative features for a given classification task. There is also a computational tradeoff: most practical vision algorithms proceed from coarse-to-fine. A first pass over a large region narrows the search for a target object using features of low computational complexity, followed by an increasingly more focused, selective, and computationally intensive exploration [cf. Blanchard and Geman (2005)].

One way to produce low-complexity features for a given category, such as faces, is to build templates from down-sampled images. The resulting templates are of intermediate complexity, as argued for by Ullman et al., and of low computational cost, given their reduced sizes as measured by numbers of pixels. At the same time, it is desirable to build an appearance model of the original data, meaning a model of pixel intensities at full resolution. Among other reasons, this allows for the direct comparisons of likelihoods across scales. Both goals can be accomplished through a simple change in the definition of the correlation feature.

To illustrate, we experimented with appearance models for whole-face images using down-sampled data. Specifically, we down-sampled each image of a face in the Feret Database to a 10×10 image patch. Let $D(y)$ be the down-sampled image, and, given a 10×10 template, T_m, define the correlation feature $s_m(y) = \text{corr}(T_m, D(y))$. No further changes are required: we use the mixture model (3.1), with (truncated) exponential distribution (3.2) on s_m, and arrive at the likelihood (3.7), which is approximately maximized by the algorithm of Section 3.3. Figure 8, left-hand panel, shows 24 of the 499 downsampled faces. We trained $M = 8$ coarse templates, resulting in the 8 prototypical low-resolution faces seen in the right-hand panel.

We emphasize that the likelihood is still on the full-resolution pixel data, that is, we are still working from a generative model on pixel intensities [$s_m = s_m(y)$ and equation (3.1) is unchanged]. Indeed, if we let D_α represent down conversion by α, in rows and columns, and if we assume that $D_\alpha \circ D_\beta = D_{\alpha\beta}$, then the templates can be used to define a mixture model on any resolution higher than 10×10.

What would samples from the full-resolution model look like? Shortly, we will introduce an approximate sampling method which can be quite effective for evaluating the implications of the different modeling approaches (Section 4.6), especially when it comes to choosing an appropriate model for background. But sampling from a full-resolution distribution using a low-resolution sufficient statistic is quite challenging, and beyond the reach of our approximation. On the other hand, we could implement a brute-force approach, which is already instructive merely

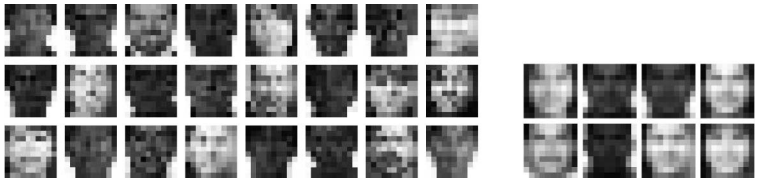


FIG. 8. Coarse representation. *Left-hand panel*: 24 *of the* 499 10×10 *down-converted training images*. *Right-hand panel*: 8 10×10 *templates learned from the coarsened* (*down-sampled*) *examples*, *under a generative mixture model of the full-resolution images*.

as a thought experiment. Refer to equation (3.1). Here, $M = 8$ and, along with the eight templates shown in Figure 8, the parameters specifying the mixing weights $(\varepsilon_1, \ldots, \varepsilon_8)$ and probability distributions of the sufficient statistics $(p_{S_1}, \ldots, p_{S_8})$ are also estimated. By "brute-force sampling," we mean this: (i) choose a mixing component (say "m") from the distribution specified by the mixing weights; (ii) assign a value, $S_m = s$, to the corresponding sufficient statistic using the corresponding distribution, $p_{S_m}(s)$; and (iii) (the brute-force part) search a large library of images for *full resolution* patches, y, that satisfy

$$s_m(y) \doteq \frac{\langle T_m, y \rangle}{\hat{\sigma}(y)} \approx s$$

in other words, sample from $p_Y^0(y|S_M = s)$. Typically (except for very unusual values of s), the chosen patch will be either a face, or something that looks very much like a face. The low-resolution "blocking" artifacts apparent in the learned templates (Figure 8) will not be visible.

4.5. *A mixture model for natural image patches*. The statistics of small randomly-selected image patches have been studied extensively. Most models belong to one of two categories: linear combinations of a set of patches that serve as a basis, possibly not orthonormal and oftentimes overcomplete, and random field models, which may or may not have a local neighborhood system. The approach to modeling studied in this paper is through mixtures, which is different from the usual random field models, which are rarely mixtures, and from the basis-type models in that only one component is active for any one sample, as opposed to a linear combination of components.

Examples of random field models include models based on learned filters, by Zhu and colleagues [e.g., Zhu and Mumford (1997), Zhu, Wu and Mumford (1998)], and Hinton's product-of-experts model [Hinton (1999)], which is also the starting point for the Markov random field models of Roth and Black ["Fields of Experts," Roth and Black (2009)]. The more frequent approach is through basis elements, which might simply be the large eigenvalue components (patches)

from a principal component analysis, as in the construction of structured background models for face detection in the work of Rajagopalan, Chellappa and Koterba (2005), or the use of sparse and overcomplete bases, as in the models by Olshausen and Field (1997), Aharon, Elad and Bruckstein (2006), Lee et al. (2007), and Mairal et al. (2009), wherein the emphasis is often on efficiently learning the bases, in addition to data fidelity. Finally, we mention the work of Welling, Hinton and Osindero (2003), which combines both random fields (product-of experts) and sparse coding, using an overcomplete basis.

For small image patches, another approach to modeling is through mixtures. We experimented with the correlation statistic, leaning $M = 16$ 10×10 templates from a collection of 12,753 15×15 image patches, randomly sampled from 59 natural images; see Figure 9 for some examples of the training patches. As in Section 4.3, we mixed over poses as well as templates, but here we allow only translations. There are 25 ways to situate a 10×10 template within a 15×15 image patch. In order to avoid learning separate templates for each possible shift of a particular structure, we defined the mixture over poses to correspond to these 25 localizations of the template within the image patch. By mixing over poses, instances of a particular structure that are situated at different locations in different 15×15 image patches are aggregated, and end up sharing a common template [as was done, similarly, in the approach taken by Papandreou, Chen and Yuille (2014)]. Following the notation developed in Section 4.3, let $\mathcal{A}_l$, be the locations of the 100 pixels within the 15×15 image patch, y, selected by translation $l = 1, \ldots, 25$, and let $y_{\mathcal{A}_l}$ be the corresponding pixel intensities. Then, for a given 10×10 template, T_m, and a given translation l, $s_{m,l}(y) = \text{corr}(T_m, y_{\mathcal{A}_l})$. As for the mixing probabilities, the natural assumption is that the translation, l, is uniform, independent of m.

Recognizing that most small image patches have little or not structure, we introduced an additional statistic, s_0, and an additional mixing component, $m = 0$, to model structureless patches. A simple and effective statistic for this purpose is the sample variance of $y_{\mathcal{A}_l}$,

$$s_{0,l}(y_{\mathcal{A}_l}) = \frac{1}{100} \sum_{i \in \mathcal{A}_l} (y_i - \bar{y})^2.$$

As for the distribution, $p_{S_{0,l}}(s)$, reasoning that a structureless background patch is more likely to have low variance than high variance, we chose

$$p_{S_{0,l}}(s) = \alpha_{\lambda_0} e^{-\lambda_0 s}, \tag{4.2}$$

where α_{λ_0} is the normalization, which depends on the maximum possible value for s, which in turn depends on the representation of pixel intensities. As it turns out, $p_{S_{0,l}}(s)$ is quite peaked at zero, meaning that λ_0 is sufficiently large that we can ignore the upper limit and simply model $p_{S_{0,l}}(s)$ by the exponential distribution $\lambda_0 e^{-\lambda_0 s}$, $s \geq 0$.

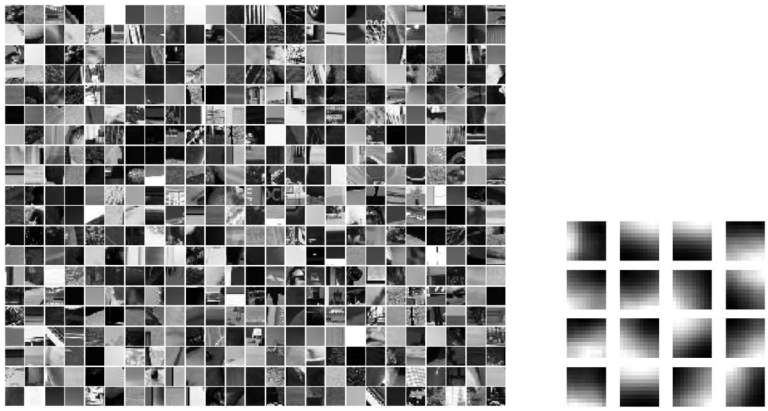


FIG. 9. Mixture model for natural image patches. *Left-hand panel*: 600 *examples from a training set of* 12,753 *randomly selected* 15×15 *natural-image patches*. *Right-hand panel*: *the* 16 10×10 *templates learned under a* 17*-fold mixture model. The additional mixing component, which carries* 78% *of the mass, uses the sample variance as a sufficient statistic and a distribution that encourages small values*; *see Section* 4.5.

In summary, the model for a 15×15 image patch y is

$$p_Y(y) = p_Y^o(y) \sum_{m=0}^{M} \sum_{l=1}^{25} \frac{1}{25} \varepsilon_m \frac{p_{S_{m,l}}(s_{m,l}(y_{\mathcal{A}_l}))}{p_Y^o(s_{m,l}(y_{\mathcal{A}_l}))},$$

where $p_{S_{m,l}}$ is the probability developed in Section 3.1 when $m > 0$, and the probability in (4.2) when $m = 0$. As usual, we used background model (ii), of Section 3.2, and the algorithm developed in Section 3.3 for approximate maximum likelihood learning of the parameters ε_0, λ_0, and $\{\varepsilon_m, T_m, \lambda_m\}$, $m = 1, \ldots, M$. The resulting 16 templates are displayed in the right-hand panel of Figure 9. As expected, the "null" component, $m = 0$, carried most of the mass: $\varepsilon_0 \approx 0.78$. Also as expected, the learned templates represented a collection of familiar structures, including boundaries, most typically vertical or horizontal, but also at other orientations, followed, in likelihood, by corners and lines, and then other less easily interpreted structures.

4.6. *Sampling*. When possible, examining samples from a generative model of images, or image patches, is an excellent way to evaluate the quality of the model. Using mouths instead of right eyes, we repeated the appearance modeling and inference methods of Section 4.1, obtaining a mixture of eight template-based distributions characterized by eight mixing probabilities, eight templates, and eight exponential parameters ($\{\varepsilon_m, T_m, \lambda_m\}$, $m = 1, \ldots, 8$). We devised an exact method for sampling under an i.i.d. Gaussian background model [a special case of model (i) of Section 3.2], and a closely related approximate method for models (ii) and (iii) (natural-image and Gaussian random field, respectively). By these

FIG. 10. Sampling the generative model. *An eight-fold mixture model for the appearance of mouths was trained on the Feret Database, under each of three background models* (*Section* 3.2). *Left-hand panel*: *exact samples under the i.i.d. model. Middle panel*: *approximate samples under the GRF model. Right-hand panel*: *approximate samples under the natural-image model.*

methods, eighteen samples for each of the three appearance models, corresponding to the three background models, were generated, and are shown in Figure 10. Most would agree that the subjective impression correlates with the classification performance, as measured by the experiments done in Section 4.1, where the best ROC performance was achieved by the model based on natural-image patches, closely followed by the GRF model, and with both outperforming the i.i.d. model.

When considering the appearance of a patch it makes sense to consider two patches, y and y', as equivalent if they are equivalent up to scale and location of their intensities, meaning that $y' = \alpha y + \beta$. After all, both parameters are absorbed in the choices made for display on a piece of paper or on an electronic screen. With this in mind, we search for standardized samples, that is, samples for which $\sum_{i\in\mathcal{B}} y_i = 0$ and $\sum_{i\in\mathcal{B}} y_i^2 = 1$, where $\mathcal{B}$ is an index set for the array of pixels in y and $n = |\mathcal{B}|$. Recall, from Section 3.1, that the templates, T_m, are also standardized.

The basic idea for sampling is projection: choose $y \sim p_Y^o$, $m \sim \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_M\}$, and $s_m \sim p_{S_m}$, and project y onto the surface defined by $\langle T_m, z\rangle = s_m$. Here, projection is easy because we chose sufficient statistics that define planar (affine) surfaces. (Sampling could be a great deal more difficult, if not entirely intractable, for various other choices for s_m.) In detail, with reference to Figure 11, let $E \subseteq \mathbb{R}^n$ be the $n-1$ dimensional subspace defined by $\sum_{i\in\mathcal{B}} y_i = 0$, and let $S^{n-1}(r)$ be the sphere in E, centered at the origin and having radius r. [So $S^{n-1}(r)$ is an $n-2$ dimensional manifold.] An exact sample, under model (i) with i.i.d. Gaussian background, is generated as follows:

1. Choose $m \in \{1, 2, \ldots, M\}$ according to the mixing probabilities $\{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_M\}$ and then choose $s_m \sim p_{S_m}$, as defined in equation (3.2). Let O_{s_m} be the $n-2$ dimensional affine subspace of vectors $z \in E$ for which $\langle T_m, z\rangle = s_m$, $(O_{s_m} = \{z \in E : \langle T_m, z\rangle = s_m\})$, keeping in mind that $T_m \in E$.
2. Choose y from the background model and project onto E, which is the same as subtracting $\bar{y}$: $y \to y - \bar{y}$.
3. Project $y - \bar{y}$ onto O_{s_m}, and denote the result by $\hat{y}$.

4. Observe that $\tilde{S}^{n-2} \triangleq S^{n-1}(1) \cap O_{s_m}$ is a sphere in $n-2$ dimensions centered at the projection of T_m onto O_{s_m}. The straight line passing through $\hat{y}$ and the center of $\tilde{S}^{n-2}$ intersects $\tilde{S}^{n-2}$ at two points. Move $\hat{y}$ to the closer of the two ("radial renormalization"), and denote the result by $\tilde{y}$.

Then $\tilde{y}$ is a standardized sample from the mixture distribution $p_Y(y)$, defined in (3.1). The reason is simple: By direct sample, the statistic s_m has the right distribution. What is more, under the i.i.d. Gaussian background model, all points on $\tilde{S}^{n-2}$ are equally likely, a property that is shared by $\tilde{y}$ due to the circular symmetry of the random choice $y - \bar{y}$. Hence, up to equivalence, $\tilde{y}$ is a sample from $p_Y^o(y|S_m = s_m)$.

What about approximate samples? The key to exact sampling under the i.i.d. Gaussian model is the independence of $\mathcal{P}_{T_m^\perp}(Y - \bar{Y})$ from $\mathcal{P}_{T_m}(Y - \bar{Y})$, under p_Y^o, where for any subspace U we write $\mathcal{P}_U$ for projection onto $U \cap E$. So, for example, it is sufficient that $\mathcal{P}_{T^\perp}(Y - \bar{Y})$ is independent of $\mathcal{P}_T(Y - \bar{Y})$ for arbitrary $T \in E$, which holds trivially in the case of i.i.d. Gaussian. For *approximate* sampling, one of these conditions needs to be approximately true. This will be difficult to verify for the GRF and implicitly-defined natural-image models, although some analytic evidence in favor of the approximation can be found in the analysis of projection pursuit by Diaconis and Freedman (1984), and the later refinements and extensions by other authors, for example, by Dümbgen and Del Conte-Zerial (2013). These results imply that the low-dimensional subspaces of $\mathcal{P}_{T^\perp}(Y - \bar{Y})$ will typically be nearly independent of $\mathcal{P}_T(Y - \bar{Y})$, a step in the right direction. In any case, for the examples in Figure 10, we simply used the exact procedure outlined above, whether or not p_Y^o was i.i.d. Gaussian.

4.7. *PCA and high-dimensional sufficient statistics.* There is nothing about our approach that requires one-dimensional features, as in the correlations studied in the previous examples. From an analytic (as opposed to computational) point of view, the development is unchanged when $s \to \vec{s}$.

A simple way to explore the idea of multidimensional features is to choose them, *a priori*, for a given category of image patches. The difference, then, is that these features do not have to be estimated as part of the generative model, as was done in the algorithm introduced in Section 3.3. As a specific example, consider adopting the first eight principal components to define, through correlations, an eight-dimensional feature. The principal components are estimated from the sample covariance matrix in the usual way, and what remains is to model and estimate the *joint* distribution on the eight correlations, one for each of the eight principal components.

Formally, let $T_1, \ldots, T_8$ be the eight eigenvectors with largest eigenvalues of the sample covariance matrix, based on N training samples, $y_1, \ldots, y_N$, from a particular category of interest. See Figure 12 for the eight principal components derived from the 499 right-eye image patches used to train the mixture model in

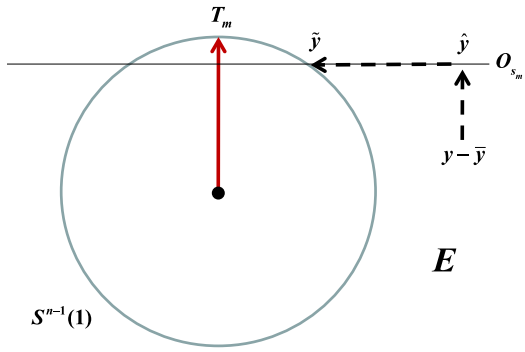


FIG. 11. Projection sampling. *The goal is to sample from the mixture* $p_Y(y) = \sum_{m=1}^{M} \varepsilon_m \times p_{S_m}(s_m(y)) p_Y^o(y|S_m = s_m(y))$, *modulo standardization of* y, *for a given background distribution* p_Y^o. *The picture is as seen from within the subspace* $E \triangleq \{z \in \mathbb{R}^n : \sum_i z_i = 0\}$. *The sphere of radius* r *within* E, *centered at the origin, is* $S^{n-1}(r)$, *and* $O_{s_m} = \{z \in E : \langle T_m, z\rangle = s_m\}$. *Choose* $m \sim \{\varepsilon_1, \ldots, \varepsilon_M\}$, $s_m \sim p_{S_m}$, *and* $y \sim p_Y^o$. *Project* y *onto* E *to get* $y - \bar{y}$, *and then onto* O_{s_m} *to get* $\hat{y}$. *Finally, move* $\hat{y}$ *to the nearest point in* $O_{s_m} \cap S^{n-1}(1)$, *to get the sample,* $\tilde{y}$. *Sampling is exact when* p_Y^o *is i.i.d. Gaussian and approximate for models* (ii) *and* (iii) *of Section* 3.2.

Section 4.1. Now define eight corresponding statistics, through correlation, with the eigenvectors playing the role of templates: $s_m(y) = \text{corr}(T_m, y)$, $m = 1, \ldots, 8$. It would not make sense to think of the eigenvectors as defining mixing components, since they derive from elements of a basis rather than a set of prototypical appearances. But it does make sense to think of them as defining a single, eight-dimensional feature, $\vec{s} = (s_1, \ldots, s_8)$, sufficient for a single ($M = 1$) mixing component. [A related model, also using PCA templates, was proposed by Feldman and Younes (2006), using quantized inner products, $\langle T_m, y\rangle$, $m = 1, \ldots, M$, to define a feature vector $\vec{s}$ with ternary components. Feldman and Younes go on to learn a *joint distribution* on a sparse array of these feature vectors.] The only important difference from earlier experiments is that we need to devise an eight-dimensional joint distribution, $p_{\vec{S}}(\vec{s})$, rather than $M > 1$ one-dimensional distributions. A simple, more-or-less canned, approach is to use a Gaussian copula: estimate the inverse cumulative distribution of each of the statistics, transform the statistics to standard normals, and estimate the resulting means and covariances.

We evaluated the results by both sampling and via the classification experiment illustrated in the right-hand panel of Figure 5. Classification performance with the natural-image background model was almost as good as that of the mixture of

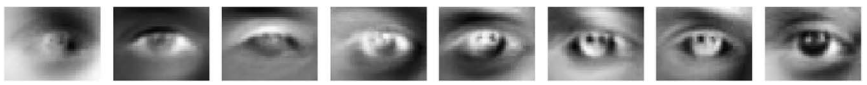


FIG. 12. PCA templates. *The eight largest-eigenvalue principal components estimated from* 499 30×40 *standardized right-eye image patches.*

FIG. 13. Samples from PCA-based mixture model. *The* 8 *principle components shown in Figure* 13 *were used to build an appearance model for right eyes*, *under the natural-image background model*, *that is*, *model* (ii) *of Section* 3.2. *Figure shows* 18 *approximate samples*, *generated by the projection method*, *Section* 4.6.

eight univariate sufficient statistics developed in Section 4.1, and much stronger than the Gaussian mixture model. As for samples, these were generated using the approximation developed in Section 4.6, the only difference being that O_{s_m} is now an $n-9$ (instead of $n-2$) dimensional affine subspace: $O_{s_m} = \{z \in E : \langle T_m, z\rangle = s_m, m = 1, \ldots, 8\}$. Eighteen examples are shown in Figure 13.

4.8. *Training*: *Variations on the theme.* We have argued for a more-or-less classical statistical approach to image modeling, based upon some of the time-honored tools of the trade: maximum-likelihood estimation, sufficiency, and likelihood ratios. Some approximations were necessary, especially in the training phase in which the sufficient statistics, themselves, were parametrized (by templates) and learned. One simplification, suggested by two of the referees, would be to learn templates from a traditional model, such as the Gaussian mixture model, and then simply use these as though they were the result of the more elaborate (yet still only approximate) MLE approach of Section 3.3. For example, using the eight means from an eight-fold Gaussian mixture model, constrained to have a common (but learned) covariance matrix, produces templates that are not so dissimilar from those that we estimated by approximate MLE. In terms of ROC performance, the background model remains important; best performance is consistently achieved under the "natural" model, though with somewhat higher error rates when compared to MLE templates.

Perhaps the best of both worlds would come from initializing MLE with the results of a Gaussian mixture model. Indeed, whether by this or other variations, we would be disappointed if there were not better ways to exploit the framework.

5. Discussion. The essence of the approach is to model the distribution on the appearance of a category of parts or objects through a factorization: a low-dimensional distribution determines the values of category-specific features, and a high-dimensional "background distribution" determines pixel intensities by conditioning on the values of the features. The distribution on features can generally

be learned with modest sample sizes, due to its low-dimensionality, but the features themselves are difficult, if not impossible, to learn without a model that fully specifies the data, that is, the pixel intensities within patches of images that contain the parts or objects of interest. The bulk of the dimensions lie in the conditional distributions on intensities given the values of the features. For these, we "borrow" from a background distribution, which is intended to model structureless image regions, as might be found in natural image patches that lack significant boundaries. There is no need to explicitly model the background distribution; only its low-dimensional marginal distributions on the features enter into the likelihood equations, or, for that matter, the likelihood ratios used to choose between competing categories.

The approach has several advantages. Since the likelihood is on the data, rather than on the features, the features themselves can be learned. In most of our experiments, we constructed features from correlations with templates, and then learned the templates from the likelihood equations, along with parameters characterizing the feature distributions, and, in the case of mixtures, the mixing probabilities. By considering mixtures over features *and* poses, we can learn templates from examples with unspecified and varied translations, scales, and rotations. For each template, the process of maximizing likelihood is essentially one of automatically aggregating data over a specified region, the pose space. Pixel-level generative models also have the advantage that they can be inspected visually by examining samples from the distribution, and we have provided a highly efficient algorithm for producing these, at least approximately. Most importantly, and our primary motivation, is that category-specific appearance models are an important component of a fully generative Bayesian model, which we advocate because it facilitates a comparison between competing interpretations of an image. Different categories are characterized by different features. A proper competition between opposing interpretations requires a comparison of the likelihoods of common data—the pixels—as opposed to category-specific features. In particular, if we use the same background distribution for all categories, then the factorization "trick" ensures that the likelihood ratio between two categories is easily computed.

Have we gone too far? A fully generative model such as ours is not likely to be pixel-level faithful to appearances in real scenes. Or, more to the point, is there really a universal conditional null distribution that supports meaningful comparisons of likelihood ratios across the spectrum of parts and objects? Maybe not, but it is not unreasonable to formalize the learning problem as one of successive approximations, involving an increasingly rich latent structure along with a growing vocabulary of ever-more-accurate appearance models. From this point of view, the background (or "combinatorial") factor in our model becomes increasingly irrelevant as the dimensionality and numbers of sufficient statistics increases. As for generative versus discriminative models, the jury is out, but in the meantime we subscribe to Grenander's maxim: "pattern synthesis equals pattern analysis."

Many directions for generalization come to mind. The last example (Section 4.7), which uses a multidimensional feature for a single-component model, suggests building mixtures of multi-dimensional features. Although our example used principal components as templates, there is no need to assume *a priori* features; these too could have been learned, for example, with a different set for each mixing component, or a shared set in which the components are distinguished by component-specific joint distributions. As for the background distribution on the feature vectors, the copula method used in Section 4.7 could be copied by simply training, instead, on samples from p_Y^o, which are always plentiful.

And many challenges remain unaddressed, for example, the choice for the number of mixing components, or more generally, the familiar tradeoff between model complexity and sample size. Bayesian approaches, or what amounts to the same thing, penalized likelihoods, come to mind, but we have not tried these. Another challenge is the lack of an exact "M" step for the templates (or features, in general), in the EM iteration. The problem is that the variance of the features under the background model is a complex function of the templates, $\sigma_m = \sigma_m(T_m)$, for which there is generally no analytic form, especially in the case of an implicitly defined background distribution, as in model (ii) of Section 3.2. The algorithm introduced in Section 3.3 sidesteps the issue by doing a single update of the σ's in the penultimate step. A better choice, based on the ease with which $\sigma(T_m)$ can be estimated for any *given* T_m, might be to approximate the full gradient, including terms involving $\sigma(T_m)$, by using a discrete version of gradient ascent. But this would introduce its own challenges, since the likelihood is not a well-defined function of the parameters without additional constraints, as is often true of mixture models. Density functions can be made infinite for some of the mixing components, for example, by driving σ_m to zero or λ_m to infinity. Unfortunately, even for common mixture models, including the prototypical mixture of Gaussians, heuristic approaches are still the state of the art.

Finally, we want to highlight the imposing challenge of building a coherent Bayesian model of full images, as opposed to just image patches. Grammars and other compositional structures have an apparent role to play in capturing *a priori* constraints on the relationships among parts that make up objects, and objects that make up scenes [Allassonnière, Amit and Trouvé (2007), Amit and Trouvé (2007), Felzenszwalb (2013), Felzenszwalb et al. (2010), Fergus, Perona and Zisserman (2003), Jin and Geman (2006), Ommer and Buhmann (2006), Yuille (2011), Zhu, Chen and Yuille (2009), Zhu and Mumford (2006), to highlight just a few examples]. But these models still need to be connected to the image, presumably through a conditional distribution on images, conditioning on the (latent) scene representation. As we have argued, pixel-level data modeling, as opposed to feature modeling alone, might ultimately give the best performance. But a full-blown generative model of pixel intensities for entire scenes will almost certainly have to accommodate *multiple, simultaneous representations* in many areas of the image. Humans annotate an object or part with a multitude of attributes, all of potential

relevance in and of themselves, and all of potential importance to the disambiguation of other parts of the image. A face can appear old and lively, sun-worn and lean, with intense eyes and narrow nose, all at the same time. It is artificial, and likely unnecessary, to segment the eyes or a nose from the surrounding parts of the face, or the head from the neck, or the neck from the torso. And it is impossible to imagine a spacial segmentation of attributes like old, lively, sun-worn, or lean. Each has something to say about features of the face. It follows that individual pixel intensities will participate in many of these features and, therefore, contribute to many sufficient statistics. The challenge is that the sufficiency approach leads to likelihood ratios in which the denominator is the *joint* distribution on the sufficient statistics under the null (background) model, and this distribution will evidently depend on the instance-by-instance poses of the represented parts, for example, the details of the placements of the corresponding templates, in the case of correlation statistics. What sorts of general and computationally efficient approximations are available for these joint distributions?

APPENDIX

A.1. Minimizing K-L divergence. The goal is to show that $\tilde{p}_Y(y) = p_S(s(y))p_Y^o(y|S = s(y))$ minimizes $D(p_Y^o||\tilde{p}_Y)$, under the constraint that $s(Y) \sim p_S$ when $Y \sim \tilde{p}_Y$.

We can always factor $p_Y^o(y)$ as $p_S^o(s(y))p_Y^o(y|S = s(y))$. Similarly, if, under $\tilde{p}_Y$, $s(Y) \sim p_S$, then $\tilde{p}_Y(y) = p_S(s(y))\tilde{p}_Y(y|S = s(y))$ and

$$
\begin{aligned}
D(p_Y^o||\tilde{p}_Y) &= \int_y p_S^o(s(y))p_Y^o(y|S = s(y)) \log \frac{p_S^o(s(y))p_Y^o(y|S = s(y))}{p_S(s(y))\tilde{p}_Y(y|S = s(y))}\, dy \\
&= \int_t \int_{y:s(y)=t} p_S^o(t)p_Y^o(y|S = t)\left[\log \frac{p_S^o(t)}{p_S(t)} + \log \frac{p_Y^o(y|S = t)}{\tilde{p}_Y(y|S = t)}\right] dy\, dt \\
&= D(p_S^o||p_S) + \int_t D(p_Y^o(\cdot|S = t)||\tilde{p}_Y(\cdot|S = t))\, dt \\
&\geq D(p_S^o||p_S)
\end{aligned}
$$

with equality, in the last step, if $\tilde{p}_Y(y|S = t) = p_Y^o(y|S = t)$ for every t and y.

The argument that $\tilde{p}_Y(y) = p_S(s(y))p_Y^o(y|S = s(y))$ also minimizes $D(\tilde{p}_Y||p_Y^o)$ is almost identical.

A.2. Correlation when the background is white noise. Among the three background models introduced in Section 3.2, the i.i.d. model is the least realistic but the most computationally convenient. We used it to initialize parameter values when learning templates, template distributions, and mixing probabilities. Its usefulness rests, in part, on the fact that the correlation, $\text{corr}(T, y)$, is then asymptotically (in the large-patch limit) normal, as asserted in the following lemma.

LEMMA. *Let* $Y^{(n)} = (y_1^{(n)}, y_2^{(n)}, \ldots, y_n^{(n)})$ *and* $\bar{y}^{(n)} = \frac{1}{n}\sum_{i=1}^n y_i^{(n)}$, *where* $y_1^{(n)}, y_2^{(n)}, \ldots, y_n^{(n)}$ *are i.i.d. random variables from a pixel-intensity distribution* G *with finite third moment* $E[(y_i^{(n)})^3]$. G *does not depend on* n. *Fix a sequence of (standardized) templates* $T^{(n)} = (t_1^{(n)}, t_2^{(n)}, \ldots, t_n^{(n)})$, *where* $\sum_{i=1}^n t_i^{(n)} = 0$ *and* $\sum_{i=1}^n (t_i^{(n)})^2 = n$, *and*

$$|t_i^{(n)}| \le M,$$

for some constant M *independent of* n. *Then*

$$\sqrt{n}\,\mathrm{corr}\big(Y^{(n)}, T^{(n)}\big) = \frac{\sum_{i=1}^n t_i^{(n)}(y_i^{(n)} - \bar{y}^{(n)})}{\sqrt{\sum_{i=1}^n (y_i^{(n)} - \bar{y}^{(n)})^2}}$$

converges in distribution to the standard normal, $N(0,1)$, *as* $n \to \infty$.

PROOF. Based on a version of Lyapunov's central limit theorem [cf. Chung (2001)]. Let $x_i = t_i^{(n)}(y_i^{(n)} - \bar{y}^{(n)})$ and $s_n^2 = \sum_{i=1}^n E[x_i^2]$. Then $E[x_i] = 0$ and

$$s_n^2 = \sum_{i=1}^n E\big[\big(t_i^{(n)}\big(y_i^{(n)} - \bar{y}^{(n)}\big)\big)^2\big] = nE\big[\big(y_1^{(n)} - \bar{y}^{(n)}\big)^2\big].$$

Hence, $s_n^3 = O(n^{3/2})$. Now since $E[(y_i^{(n)})^3] < \infty$

$$r_n \triangleq \sum_{i=1}^n E\big[\big|t_i^{(n)}\big(y_i^{(n)} - \bar{y}^{(n)}\big)\big|^3\big] \le MnE\big|y_1^{(n)} - \bar{y}^{(n)}\big|^3 = O(n)$$

and

$$\lim_{n\to\infty} \frac{r_n}{s_n^3} = O\left(\frac{1}{\sqrt{n}}\right) \longrightarrow 0.$$

This is the Lyapunov condition and, consequently,

$$\frac{\sum_{i=1}^n t_i^{(n)}(y_i^{(n)} - \bar{y}^{(n)})}{\sqrt{nE(y_i^{(n)} - \bar{y}^{(n)})^2}} = \frac{\sum_{i=1}^n x_i}{s_n} \longrightarrow N(0,1).$$

Furthermore, by the law of large numbers,

$$\frac{1}{n}\sum_{i=1}^n \big(y_i^{(n)} - \bar{y}^{(n)}\big)^2 \longrightarrow E\big[\big(y_1^{(n)} - \bar{y}^{(n)}\big)^2\big]$$

in probability.

Finally, by an application of Slutsky's theorem,

$$\sqrt{n}\,\mathrm{corr}\big(Y^{(n)}, T^{(n)}\big) = \frac{\sum_{i=1}^n t_i^{(n)}(y_i^{(n)} - \bar{y}^{(n)})}{\sqrt{nE(y_i^{(n)} - \bar{y}^{(n)})^2}} \cdot \frac{\sqrt{E(y_i^{(n)} - \bar{y}^{(n)})^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^n (y_i^{(n)} - \bar{y}^{(n)})^2}}$$
$$\longrightarrow N(0,1)$$

and the proof is complete. □

REMARKS. 1. The only condition on G is that the third moment exists, so the theorem applies to the commonly used Gaussian and uniform distributions on intensities. Referring to Section 3.2, the upshot is that the statistic, $s_m(y_k)$, is approximately $N(0, \frac{1}{n})$, which is equation (3.4).

2. The dimension, n, does not have to be all that large. It is easy to experiment with small patches, say $n = 10 \times 10 = 100$, $Y^{100} \sim \mathrm{U}\{0, 1, \ldots, 255\}^{100}$, and $T^{100} \in \{0, 1, \ldots, 255\}^{100}$, at which point the approximation is already excellent.

REFERENCES

AGARWAL, S., AWAN, A. and ROTH, D. (2004). Learning to detect objects in images via a sparse part-based representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **26** 1475–1490.

AHARON, M., ELAD, M. and BRUCKSTEIN, A. M. (2006). The KSVD: An algorithm for designing of overcomplete dictionaries for sparse representations. *IEEE Trans. Signal Process.* **54** 4311–4322.

ALLASSONNIÈRE, S., AMIT, Y. and TROUVÉ, A. (2007). Towards a coherent statistical framework for dense deformable template estimation. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **69** 3–29. MR2301497

AMIT, Y., GEMAN, D. and FAN, X. (2004). A coarse-to-fine strategy for multiclass shape detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **26** 1606–1621.

AMIT, Y. and TROUVÉ, A. (2006). *Generative Models for Labeling Multi-Object Configurations in Images* 362–381. Springer Berlin, Heidelberg.

AMIT, Y. and TROUVÉ, A. (2007). POP: Patchwork of parts models for object recognition. *Int. J. Comput. Vis.* **75** 267–282.

BLANCHARD, G. and GEMAN, D. (2005). Hierarchical testing designs for pattern recognition. *Ann. Statist.* **33** 1155–1202. MR2195632

BORENSTEIN, E. and ULLMAN, S. (2002). Class-specific, top-down segmentation. In *ECCV. LNCS* **2353** 109–122.

CHUNG, K. L. (2001). *A Course in Probability Theory*, 3rd ed. Academic Press, Inc., San Diego, CA. MR1796326

DEMPSTER, A. P., LAIRD, N. M. and RUBIN, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B* **39** 1–38. With discussion. MR0501537

DIACONIS, P. and FREEDMAN, D. (1984). Asymptotics of graphical projection pursuit. *Ann. Statist.* **12** 793–815. MR0751274

DÜMBGEN, L. and DEL CONTE-ZERIAL, P. (2013). On low-dimensional projections of high-dimensional distributions. In *From Probability to Statistics and Back: High-Dimensional Models and Processes. Inst. Math. Stat. (IMS) Collect.* **9** 91–104. IMS, Beachwood, OH. MR3186751

FELDMAN, T. and YOUNES, L. (2006). Homeostatic image perception: An artificial system. *Comput. Vis. Image Underst.* **102** 70–80.

FELZENSZWALB, P. (2013). A stochastic grammar for natural shapes. In *Shape Perception in Human and Computer Vision* (S. J. Dickinson and Z. Pizlo, eds.) 299–310. Springer, London.

FELZENSZWALB, P. F., GIRSHICK, R. B., MCALLESTER, D. and RAMANAN, D. (2010). Object detection with discriminatively trained part based models. *IEEE Trans. Pattern Anal. Mach. Intell.* **32** 1627–1645.

FERGUS, R., PERONA, P. and ZISSERMAN, A. (2003). Object class recognition by unsupervised scale-invariant learning. *CVPR* **2** 264–271.

FREY, B. J. (2003). Transformation-invariant clustering using the EM algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.* **25** 1–17.

FREY, B. J. and JOJIC, N. (1999). Transformed component analysis: Joint estimation of spatial transformations and image components. In *International Conference on Computer Vision* **2** 1190.

HEISELE, B., SERRE, T. and POGGIO, T. (2007). A component-based framework for face detection and identification. *Int. J. Comput. Vis.* **74** 167–181.

HEISELE, B., SERRE, T., PONTIL, M., VETTER, T. and POGGIO, T. (2001). Categorization by learning and combining object parts. In *NIPS*.

HINTON, G. E. (1999). Products of experts. In *Int. Conf. on Art. Neur. Netw.* (*ICANN*) **1** 1–6.

JIN, Y. and GEMAN, S. (2006). Context and hierarchy in a probabilistic image model. In *CVPR* 2145–2152.

KANNAN, A., JOJIC, N. and FREY, B. (2002). Fast transformation invariant factor analysis. In *Advances in Neural Information Processing Systems* **15**.

LEE, H., BATTLE, A., RAINA, R. and NG, A. Y. (2007). Efficient sparse coding algorithms. *Adv. Neural Inf. Process. Syst.* **19** 801–808.

LEIBE, B. and SCHIELE, B. (2003). Interleaved object categorization and segmentation. In *Proceedings of British Machine Vision Conference* (*BMVC*).

MAIRAL, J., BACH, F., PONCE, J. and SAPIRO, G. (2009). Online dictionary learning for sparse coding. In *Proceedings of the* 26*th International Conference on Machine Learning*.

OLSHAUSEN, B. A. and FIELD, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vis. Res.* **37** 3311–3325.

OMMER, B. and BUHMANN, J. M. (2006). Learning compositional categorization models. In *ECCV*.

PAPANDREOU, G., CHEN, L.-C. and YUILLE, A. (2014). Modeling image patches with a generic dictionary of mini-epitomes. In *Proc. IEEE Int. Conf. on Comp. Vision and Pat. Rec.* (*CVPR*).

RAJAGOPALAN, A. N., CHELLAPPA, R. and KOTERBA, N. T. (2005). Background learning for robust face recognition with PCA in the presence of clutter. *IEEE Trans. Image Process.* **14** 832–843.

REID, N. (1995). The roles of conditioning in inference. *Statist. Sci.* **10** 138–157, 173–189, 193–196. With comments by V. P. Godambe, Bruce G. Lindsay and Bing Li, Peter McCullagh, George Casella, Thomas J. DiCiccio and Martin T. Wells, A. P. Dawid and C. Goutis and Thomas Severini, with a rejoinder by the author. MR1368097

ROTH, S. and BLACK, M. J. (2009). Fields of experts. *Int. J. Comput. Vis.* **82** 205–229.

SABUNCU, M. R., BALCI, S. K. and GOLLAND, P. (2008). Discovering modes of an image population through mixture modeling. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention* (*MICCAI*). *LNCS* **5242** 381–389.

SALI, E. and ULLMAN, S. (1999). Combining class-specific fragments for object classification. In *Proc.* 10*th British Machine Vision Conference* **1** 203–213.

SI, Z. and ZHU, S.-C. (2012). Learning hybrid image templates (HIT) by information projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **34** 1354–1367.

ULLMAN, S., SALI, E. and VIDAL-NIQUET, M. (2001). A fragment-based approach to object representation and classification. In *International Workshop on Visual Form* 85–100.

ULLMAN, S., VIDAL-NAQUET, M. and SALI, E. (2002). Visual features of intermediate complexity and their use in classification. *Nat. Neurosci.* **5** 682–687.

WEBER, M., WELLING, M. and PERONA, P. (2000). Unsupervised learning of models for recognition. In *Proc. Sixth European Conf. Computer Vision* 18–22.

WELLING, M., HINTON, G. E. and OSINDERO, S. (2003). Learning sparse topographic representations with products of student-t distributions. In *Adv. in Neur. Inf. Proc. Sys.* (*NIPS*) **15** 1359–1366.

YUILLE, A. (2011). Towards a theory of compositional learning and encoding of objects. In *Computational Methods for the Innovative Design of Electrical Devices'*11 1448–1455.

ZHU, L., CHEN, Y. and YUILLE, A. (2009). Unsupervised learning of probabilistic grammar-Markov models for object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* **31** 114–128.

ZHU, S.-C. and MUMFORD, D. (1997). Prior learning and Gibbs reaction-diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **19** 1236–1250.

ZHU, S.-C. and MUMFORD, D. (2006). A stochastic grammar of images. In *Foundations and Trends in Computer Graphics and Vision* 259–362.

ZHU, S.-C., WU, Y. and MUMFORD, D. (1998). Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. *Int. J. Comput. Vis.* **27**.

L.-B. CHANG
DEPARTMENT OF STATISTICS
OHIO STATE UNIVERSITY
COLUMBUS, OHIO 43210
USA
E-MAIL: lobinchang@stat.osu.edu

E. BORENSTEIN
AMAZON
SUNNYVALE, CALIFORNIA 94089
USA

W. ZHANG
SMARTLEAF
CAMBRIDGE, MASSACHUSETTS 02139
USA

S. GEMAN
DIVISION OF APPLIED MATHEMATICS
BROWN UNIVERSITY
PROVIDENCE, RHODE ISLAND 02912
USA