Fields of Experts: A Framework for Learning Image Priors

Stefan Roth Michael J. Black
Department of Computer Science, Brown University, Providence, RI, USA
{roth,black}@cs.brown.edu

Abstract

We develop a framework for learning generic, expressive image priors that capture the statistics of natural scenes and can be used for a variety of machine vision tasks. The approach extends traditional Markov Random Field (MRF) models by learning potential functions over extended pixel neighborhoods. Field potentials are modeled using a Products-of-Experts framework that exploits non-linear functions of many linear filter responses. In contrast to previous MRF approaches all parameters, including the linear filters themselves, are learned from training data. We demonstrate the capabilities of this Field of Experts *model with two example applications, image denoising and image inpainting, which are implemented using a simple, approximate inference scheme. While the model is trained on a generic image database and is not tuned toward a specific application, we obtain results that compete with and even outperform specialized techniques.*

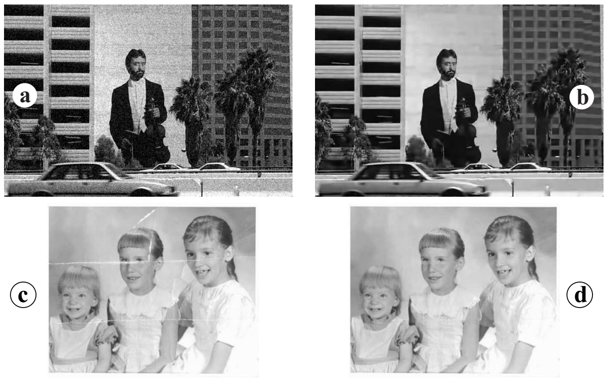

Figure 1. **Image reconstruction using a Field of Experts.** **(a)** Example image with additive Gaussian noise ($\sigma = 20$, PSNR $=$ 22.51dB). **(b)** Denoised image. (PSNR $=$ 28.79dB). **(c)** Photograph with scratches. **(d)** Image inpainting using the FoE model.

1. Introduction

The need for prior models of image structure occurs in many machine vision problems including stereo, optical flow, denoising, super-resolution, and image-based rendering to name a few. Whenever one has "noise" or uncertainty, prior models of images (or depth maps, flow fields, *etc.*) come into play. Here we develop a method for learning rich Markov random field (MRF) image priors by exploiting ideas from sparse image coding. The resulting *Field of Experts* (FoE) models the prior probability of an image in terms of a random field with overlapping cliques, whose potentials are represented as a Product of Experts [11].

We show how the model is trained on a standard database of natural images [16] and develop a diffusion-like scheme that exploits the prior for approximate Bayesian inference. To demonstrate the modeling power of the FoE model, we use it in two different applications: image denoising and image inpainting [3]. Despite the generic nature of the prior and the simplicity of the approximate inference, we obtain state of the art results that, until now, were not possible with MRF approaches. Figure 1 illustrates the application of the FoE model for image denoising and image inpainting. Below we provide a detailed quantitative analysis of the performance in these tasks with respect to the state of the art.

Modeling image priors is challenging due to the high-dimensionality of images, their non-Gaussian statistics, and the need to model correlations in image structure over extended image neighborhoods. It has been often observed that, for a wide variety of linear filters, the marginal filter responses are non-Gaussian, and that the responses of different filters are usually not independent [13, 20].

Sparse coding approaches attempt to address some of the issues in modeling complex image structure. In particular, they model structural properties of images in terms of a set of linear filter responses. Starting from a variety of simple assumptions, numerous authors have obtained sparse representations of local image structure in terms of the statistics of filters that are local in position, orientation, and scale [18, 24]. These methods, however, focus on image *patches* and provide no direct way of modeling the statistics of whole *images*.

Markov random fields on the other hand have been widely used in machine vision but exhibit serious limitations. In particular, MRF priors typically exploit hand-crafted clique potentials and small neighborhood systems [9], which limit the expressiveness of the models and only crudely capture the statistics of natural images. Typical models consider simple nearest neighbor relations and model first derivative filter responses. There is a sharp contrast between the rich, patch-based priors obtained by sparse coding methods and the extremely local (e. g. first order) priors employed by most MRF methods.

Zhu and Mumford took a step toward more practical MRFs with the introduction of the FRAME model [27], which allowed MRF priors to be learned from training data. This method, however, still relies on a hand-selected set of image filters from which an image prior is built. The approach is complicated by its use of discrete filter histograms and the reported image reconstruction results appear to fall well below the current state of the art. Another line of work modeled more complex spatial properties using multiple, non-local pairwise pixel interactions [10, 25]. These models have so far only been exploited for texture synthesis rather than for modeling generic image priors.

To model more complex local statistics a number of authors have turned to empirical probabilistic models captured by a database of image patches. Freeman *et al.* [7] propose an MRF model that uses example image patches and a measure of consistency between them to model scene structure. This idea has recently been exploited as a prior model for image based rendering [6] and is related to example-based texture synthesis [5]. Other MRF models used the Parzen window approach [19] to define the field potentials. Jojic *et al.* [14] use a miniature version of an image or a set of images, called the epitome, to describe an image. While it may be possible to use this method as a generic image prior, this possibility has not yet been explored.

The goal of the current paper is to develop a framework for learning rich, generic prior models of natural images (or any class of images). In contrast to example-based approaches, we develop a *parametric representation* that uses examples for training, but does not rely on examples as part of the representation. Such a parametric model has advantages over example-based methods in that it generalizes better beyond the training data and allows for more elegant computational techniques. The key idea is to extend Markov random fields beyond FRAME by modeling the local field potentials with learned filters. To do so, we exploit ideas from the Products-of-Experts (PoE) framework [11]. Previous efforts to model images using Products of Experts [24] were patch-based and hence inappropriate for learning generic priors for images of arbitrary size. We extend these methods, yielding a *translation-invariant* prior. The Field-of-Experts framework provides a principled way to learn MRFs from examples and the greatly improved modeling power makes them practical for complex tasks.

2. Sparse Coding and Product of Experts

The statistics of small image patches have received extensive treatment in the literature. In particular, sparse coding methods [18] represent an image patch in terms of a linear combination of learned filters, or "bases", $\mathbf{J}_i \in \mathbf{R}^n$,

$$\min_{\mathbf{a},\mathbf{J}} E(\mathbf{a}, \mathbf{J}) = \sum_j \left\| \mathbf{x}^{(j)} - \sum_i a_{i,j} \mathbf{J}_i \right\|^2 + \lambda \sum_{i,j} S(a_{i,j})$$

where $\mathbf{x}^{(j)} \in \mathbf{R}^n$ are vectorized image patches and $S(a_{i,j})$ is a sparseness prior that penalizes non-zero coefficients, $a_{i,j}$. Variations of this formulation lead to principal components, independent components, or more specialized filters.

Independent component analysis (ICA) [2] can be used to define a probabilistic model for images patches. Since the components found by ICA are by assumption independent, one can simply multiply their marginal distributions to obtain a prior model. However, in case of image patches of n pixels it is generally impossible to find n fully independent linear components, which makes the ICA model only an approximation.

Welling *et al.* [24] went beyond this limitation with a model based on the *Products-of-Experts* framework [11]. The idea behind the PoE framework is to model high-dimensional probability distributions by taking the product of several expert distributions, where each expert works on a low-dimensional subspace that is relatively easy to model. Usually, experts are defined on linear one-dimensional subspaces (corresponding to the basis vectors in sparse coding models). Notice that projecting an image patch onto a linear component ($\mathbf{J}_i^{\mathrm{T}}\mathbf{x}$) is equivalent to filtering the patch with a *linear filter* described by $\mathbf{J}_i$. Based on the observation that responses of linear filters applied to natural images typically exhibit highly kurtotic marginal distributions that resemble a Student-t distribution, Welling *et al.* [24] propose the use of Student-t experts. The full Product of t-distribution (PoT) model can be written as

$$p(\mathbf{x}) = \frac{1}{Z(\Theta)} \prod_{i=1}^{N} \phi_i(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \alpha_i), \quad \Theta = \{\theta_1, \ldots, \theta_N\}, \quad (1)$$

where $\theta_i = \{\alpha_i, \mathbf{J}_i\}$ and the experts ϕ_i have the form

$$\phi_i(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \alpha_i) = \left(1 + \frac{1}{2}(\mathbf{J}_i^{\mathrm{T}}\mathbf{x})^2\right)^{-\alpha_i},$$

and $Z(\Theta)$ is the normalizing, or partition, function. The α_i are assumed to be positive, which is needed to make the ϕ_i proper distributions, but note that the experts themselves are not assumed to be normalized. It will later be convenient to rewrite the probability density in Gibbs form as $p(\mathbf{x}) = \frac{1}{Z(\Theta)} \exp(-E_{\mathrm{PoE}}(\mathbf{x}, \Theta))$ with

$$E_{\mathrm{PoE}}(\mathbf{x}, \Theta) = -\sum_{i=1}^{N} \log \phi_i(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \alpha_i). \quad (2)$$

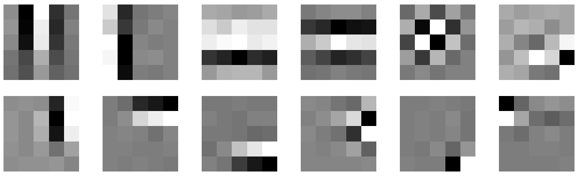

Figure 2. **Selection of the** 5×5 **filters** obtained by training the *Products-of-Experts* model on an generic image database.

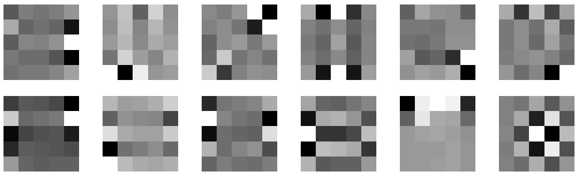

Figure 3. **Selection of the** 5×5 **filters** obtained by training the *Fields-of-Experts* model on a generic image database.

One important property of this model is that all parameters can be automatically learned from training data, i. e., both the α_i and the image filters $\mathbf{J}_i$. The advantage of the PoE model over the ICA model is that the number of experts N is not necessarily equal to the number of dimensions n (i. e. pixels). The PoE model permits fewer experts than dimensions (under-complete), equally many (complete), or more experts than dimensions (over-complete). The over-complete case is particularly interesting because it allows dependencies between filters to be modeled and consequently is more expressive than ICA.

The procedure for training the PoT model will be described in the following section in the context of our generalization to the FoE model. Figure 2 shows a selection of the 24 filters obtained by training this PoE model on 5×5 image patches. The training data contains about 60000 image patches randomly cropped from the Berkeley Segmentation Benchmark [16] and converted to grayscale. The filters learned by this model are the same kinds of Gabor-like filters obtained using a non-parametric ICA technique or standard sparse coding approaches. It is possible to train models that are several times over-complete [18, 24]; the characteristics of the filters remain the same.

A key characteristic of these methods is that they focus on the modeling of small image patches rather than defining a prior model over an entire image. Despite that, Welling *et al.* [24] suggest an algorithm for denoising images of arbitrary size. The resulting algorithm, however, does not easily generalize to other image reconstruction problems.

Some effort has gone into extending sparse coding models to full images [21]. Inference with this model requires Gibbs sampling, which makes it somewhat less attractive for many machine vision applications.

3. Fields of Experts

3.1. Basic model

While the model described in the preceding section provides an elegant and powerful way of learning prior distributions on small image patches, the results do not generalize immediately to give a prior model for the whole image. For several reasons simply making the patches bigger is not a viable solution: (1) the number of parameters to learn would be too large; (2) the model would only work for one specific image size and would not generalize to other image sizes; and (3) the model would not be translation invariant, which is a desirable property for generic image priors.

The key insight here is that we can overcome these problems by combining ideas from sparse coding with Markov random field models. To that end, let the pixels in an image be represented by nodes V in a graph $G = (V, E)$, where E are the edges connecting nodes. We define a neighborhood system that connects all nodes in an $m \times m$ rectangular region. Every such neighborhood centered on a node (pixel) $k = 1, \ldots, K$ defines a maximal clique $\mathbf{x}_{(k)}$ in the graph. The Hammersley-Clifford theorem establishes that we can write the probability density of this graphical model as a Gibbs distribution $p(\mathbf{x}) = \frac{1}{Z} \exp\left(-\sum_k V_k(\mathbf{x}_{(k)})\right)$, where $\mathbf{x}$ is an image and $V_k(\mathbf{x}_{(k)})$ is the potential function for clique $\mathbf{x}_{(k)}$. We make the additional assumption that the MRF is homogeneous; i. e., the potential function is the same for all cliques (or in other terms $V_k(\mathbf{x}_{(k)}) = V(\mathbf{x}_{(k)})$). This property gives rise to translation-invariance of an MRF model[1]. Without loss of generality we assume the maximal cliques in the MRF are square pixel patches of a fixed size; other, non-square, neighborhoods could be used [8].

Instead of defining the potential function V by hand, we learn it from training images. To enable that, we represent the MRF potentials as a Product of Experts with the same basic form as in (1). More formally, we use the energy term from (2) to define the potential function, i. e., $V(\mathbf{x}_{(k)}) = E_{\text{PoE}}(\mathbf{x}_{(k)}, \Theta)$. Overall, we thus write the probability density of a full image under the FoE model as $p(\mathbf{x}) = \frac{1}{Z(\Theta)} \exp(-E_{\text{FoE}}(\mathbf{x}, \Theta))$ with

$$E_{\text{FoE}}(\mathbf{x}, \Theta) = -\sum_k \sum_{i=1}^{N} \log \phi_i(\mathbf{J}_i^{\mathrm{T}} \mathbf{x}_{(k)}; \alpha_i), \qquad (3)$$

or equivalently

$$p(\mathbf{x}) = \frac{1}{Z(\Theta)} \prod_k \prod_{i=1}^{N} \phi_i(\mathbf{J}_i^{\mathrm{T}} \mathbf{x}_{(k)}; \alpha_i), \qquad (4)$$

where ϕ_i and θ_i are defined as before. The important difference with respect to the PoE model in (1) is that we here

[1] When we talk about translation-invariance, we disregard the fact that the finite size of the image will make this property hold only approximately.

take the product over all neighborhoods k.

This model overcomes all the problems we cited above: The number of parameters is only determined by the size of the maximal cliques in the MRF and the number of filters defining the potential. Furthermore, the model applies to images of arbitrary size and is translation invariant because of the homogeneity of the potential functions.

Note that computing the partition function $Z(\Theta)$ is intractable. Nevertheless, most inference algorithms, such as the ones proposed in Section 4, do not the require this normalization term to be known. What distinguishes this model from that of [24] is that it explicitly models the overlap of image patches. These overlapping patches are highly correlated and the learned filters, $\mathbf{J}_i$, as well as the parameters α_i must account for this correlation. We refer to the resulting translation-invariant Product-of-Experts model as a Field of Experts to emphasize how the probability density of an entire image involves the combination of overlapping local experts.

3.2. Contrastive divergence learning

The parameters α_i as well as the linear filters $\mathbf{J}_i$ can be learned from a set of D training images $X = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(D)}\}$ by maximizing its likelihood. Maximizing the likelihood for the PoE and the FoE model is equivalent to minimizing the Kullback-Leibler divergence between the model and the data distribution, and so guarantees the model distribution to be as close to the data distribution as possible. Since there is no closed form solution for the parameters, we perform a gradient ascent on the log-likelihood. This leads to the parameters being updated with

$$\delta\theta_i = \eta \left[\left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_p - \left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_X \right],$$

where η is a user-defined learning rate, $\langle \cdot \rangle_X$ denotes the average over the training data X, and $\langle \cdot \rangle_p$ the expectation value with respect to the model distribution $p(\mathbf{x})$. While the average over the training data is easy to compute, there is no general closed form solution for the expectation over the model distribution. However, it can be computed approximately using Monte Carlo integration by repeatedly drawing samples from $p(\mathbf{x})$ using MCMC sampling. In our implementation, we use a hybrid Monte Carlo (HMC) sampler [17], which is more efficient than many standard sampling techniques such as Metropolis sampling. The advantage of the HMC sampler stems from the fact that it uses the gradient of the log-density to explore the space more effectively.

Despite using efficient MCMC sampling strategies, training such a model in this way is still not very practical, because it may take a very long time until the Markov chain approximately converges. Instead of running the Markov chain until convergence we use the idea of *contrastive divergence* [12] to initialize the sampler at the data points and only run it for a small, fixed number of steps. If we denote the data distribution as p^0 and the distribution after j MCMC iterations as p^j, the contrastive divergence parameter update is written as

$$\delta\theta_i = \eta \left[\left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_{p^j} - \left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_{p^0} \right].$$

The intuition here is that running the MCMC sampler for just a few iterations starting from the data distribution will draw the samples closer to the target distribution, which is enough to estimate the parameter updates. Hinton [12] justifies this more formally and shows that contrastive divergence learning is typically a good approximation to a maximum likelihood estimation of the parameters.

3.3. Implementation details

In order to correctly capture the spatial dependencies of neighboring cliques (or image patches), the size of the images in the training data set should be substantially larger than the clique size. On the other hand, large images would make the required MCMC sampling inefficient. We train this model on 2000 randomly cropped image regions that have 3 times the width and height of the maximal cliques (i. e., in case of 5×5 cliques we train on 15×15 images). Our training data again is taken from fifty images from the Berkeley Segmentation Database (natural scenes, people, buildings, etc.) [16]. Welling *et al.* [24] noted that in their PoE model the filter learning usually benefits from whitening the data distribution, since this removes potential scaling issues due to the very non-spherical covariance of image patches. To avoid similar problems in our model, we apply a whitening transform to all the clique pixels before computing the update for the filters. The transform furthermore ignores any changes to the average gray level in the clique, which reduces the number of dimensions of the filters by 1. We enforce the positivity of the α_i by updating their logarithm. However, we found that the learning algorithm also works without this constraint. In our experiments we used contrastive divergence with a single step of HMC sampling. Each HMC step consisted of 30 leaps; the leap size was adjusted automatically, so that the acceptance rate was near 90%. We performed 3000 update steps with $\eta = 0.01$. We found the result to not be very sensitive to the exact value of the learning rate nor the number of contrastive divergence steps. Figure 3 shows a selection of the 24 filters learned by training the FoE model on 5×5 pixel cliques. These filters respond to various edge and texture features at multiple orientations and scales and, as demonstrated below, capture important structural properties of images. They appear to

σ / PSNR	Lena	Barbara	Boats	House	Peppers
1 / 48.13	47.84	47.86	47.69	48.32	47.81
2 / 42.11	42.92	42.92	42.28	44.01	42.96
5 / 34.15	38.12	37.19	36.27	38.23	37.63
10 / 28.13	35.04	32.83	33.05	35.06	34.28
15 / 24.61	33.27	30.22	31.22	33.48	32.03
20 / 22.11	31.92	28.32	29.85	32.17	30.58
25 / 20.17	30.82	27.04	28.72	31.11	29.20
50 / 14.15	26.49	23.15	24.53	26.74	24.52
75 / 10.63	24.13	21.36	22.48	24.13	21.68
100 / 8.13	21.87	19.77	20.80	21.66	19.60

Table 1. **Peak signal-to-noise ratio (PSNR)** in dB for images (from [1]) denoised with FoE prior.

lack, however, the clearly interpretable structure of the filters learned using the standard PoE model (cf. Figure 2). This may result from the filters having to account for the correlated image structure in overlapping patches.

Training the FoE model is computationally intensive but occurs off-line. As we will see, there are relatively efficient algorithms for approximate inference that make the use of the FoE model practical.

4. Applications and Experiments

There are many computational methods for exploiting MRF models in image denoising and other applications. The methods include Gibbs sampling [9], deterministic annealing, mean-field methods, belief propagation, non-linear diffusion, as well as many related PDE methods [23]. While a Gibbs sampler has formal convergence properties, it is computationally intensive. Instead we derive a gradient ascent-based method for approximate inference that performs well in practice.

4.1. Image denoising

Currently, the most accurate denoising methods in the literature fall within the category of wavelet "coring" in which the image is 1) decomposed using a large set of wavelets at different orientations and scales; 2) the wavelet coefficients are modified based on their prior probability; and 3) the image is reconstructed by inverting the wavelet transform. For an excellent review and quantitative evaluation of the state of the art see [20]. The most accurate of these methods model the fact that the marginal statistics of the wavelet coefficients are non-Gaussian and that neighboring coefficients in space or scale are not independent. Portilla *et al.* [20] model these dependencies using a Gaussian scale mixture and derive a Bayesian decoding algorithm that appears to be the most accurate of this class of methods. They use a pre-determined set of filters and hand select a few neighboring coefficients (e. g. across adjacent scales) that intuition and empirical evidence suggest are statistically dependent.

In contrast to the above schemes we focus on a Bayesian formulation with a spatial prior term. Given an observed image $\mathbf{y}$, our goal is to find the true image $\mathbf{x}$ that maximizes the posterior probability $p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x}) \cdot p(\mathbf{x})$. As is common in the denoising literature, our experiments assume that the true image has been corrupted by additive, i. i. d. Gaussian noise with zero mean and known standard deviation σ. We thus write the likelihood as

$$p(\mathbf{y}|\mathbf{x}) \propto \prod_j \exp\left(-\frac{1}{2\sigma^2}(\mathbf{y}_j - \mathbf{x}_j)^2\right),$$

where j ranges over the pixels in the image. Our method generalizes to other kinds of noise distributions, as long as the noise distribution is known (and its logarithm is differentiable).

Maximizing the posterior probability of a graphical model such as the FoE is generally hard. In order to emphasize the practicality of the proposed model, we refrain from using expensive inference techniques. Instead we perform a gradient ascent on the logarithm of the posterior probability. The gradient of the log-likelihood is written as

$$\nabla_{\mathbf{x}} \log p(\mathbf{y}|\mathbf{x}) = \frac{1}{\sigma^2}(\mathbf{y} - \mathbf{x}).$$

Fortunately, the gradient of the log-prior is also simple to compute [26]:

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}) = \sum_{i=1}^{N} \mathbf{J}_i^- * \psi_i(\mathbf{J}_i * \mathbf{x}),$$

where $\mathbf{J}_i * \mathbf{x}$ denotes the convolution of image $\mathbf{x}$ with filter $\mathbf{J}_i$. We also define $\psi_i(y) = \partial/\partial y \ \log \phi_i(y; \alpha_i)$ and let $\mathbf{J}_i^-$ denote the filter obtained by mirroring $\mathbf{J}_i$ around its center pixel [26]. Note that $-\log \phi_i$ is a standard robust error function when ϕ_i has heavy tails, and that ψ_i is proportional to its influence function [4].

By introducing an iteration index t, an update rate η, and an optional weight λ, we can write the gradient ascent denoising algorithm as:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \eta \left[\sum_{i=1}^{N} \mathbf{J}_i^- * \psi_i(\mathbf{J}_i * \mathbf{x}^{(t)}) + \frac{\lambda}{\sigma^2}(\mathbf{y} - \mathbf{x}^{(t)})\right]$$

As observed by Zhu and Mumford [26], this is related to non-linear diffusion methods. If we had only two filters (x- and y-derivative filters) then this equation is similar to standard non-linear diffusion filtering with a data term. Even though denoising proceeds in very similar ways in both cases, our prior model uses many more filters than non-linear diffusion. The key advantage of the FoE model is that it tells us how to build richer prior models that combine more filters over larger neighborhoods in a principled way.

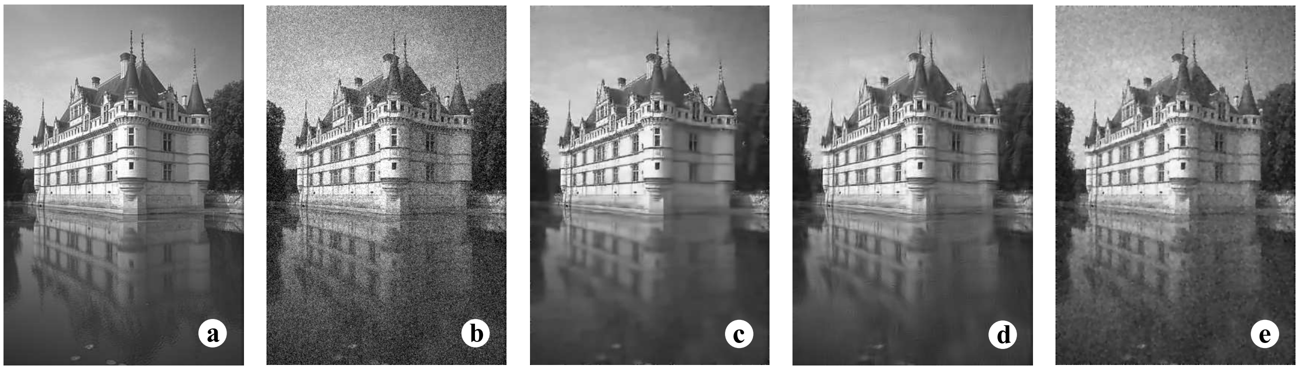

Figure 4. **Denoising results.** **(a)** Original noiseless image. **(b)** Image with additive Gaussian noise ($\sigma = 25$); PSNR $= 20.29$dB. **(c)** Denoised image using a Field of Experts; PSNR $= 28.72$dB. **(d)** Denoised image using the approach from [20]; PSNR $= 28.90$dB. **(e)** Denoised image using standard non-linear diffusion; PSNR $= 27.18$dB.

Denoising experiments

Using the FoE model trained as in the previous section on the Berkeley database we perform a number of denoising experiments. The experiments conducted here assume a known noise distribution. The extension of our exposition to "blind" denoising, for example using robust data terms or automatic stopping criteria, will remain the subject of future work. We used an FoE prior with 24 filters of 5×5 pixels. We chose the update rate η to be between 0.02 and 1 depending only on the amount of noise added, and performed 2500 iterations. While potentially speeding up convergence, large update rates may result in numerical instabilities, which experimentally disappear for $\eta \leq 0.02$. We found, however, that running with large step sizes and subsequently "cleaning up" the image with 250 iterations with $\eta = 0.02$ shows no worse results than performing the denoising only with $\eta = 0.02$. Experimentally, we found that the best results are obtained with an additional weight λ for the likelihood term, which furthermore depends on the amount of noise added. We automatically learn the optimal λ value for each noise level using the same training data set that was used to train the FoE model. This is done by choosing the best value from a small candidate set of λ's.

Results are obtained for two sets of images. The first set consists of images commonly used in denoising experiments [20]. Table 1 provides the peak signal-to-noise ratio (PSNR $= 20 \log_{10}(255/\sigma_e)$) for this set with various levels of additive Gaussian noise and denoised with the FoE model (cf. [20]). Portilla *et al.* [20] report the most accurate results on these test images and their method is tuned to perform well on this dataset. We obtain signal-to-noise ratios that are close to their results (mostly within 0.5dB), and in some cases even surpass their results (by about 0.3dB). To the best of our knowledge, no other MRF approach has so far been able to closely compete with such wavelet-based methods on this dataset. Also note that the prior is not trained on, or tuned to these examples. Our expectation is that the use of more and/or larger filters, and of better MAP estimation techniques will improve these results further.

To test more varied and realistic images we denoised a second test set consisting of 68 images from the test section of the Berkeley data set. For various noise levels we denoised the images using the FoE model, the method from [20] (using the software and default settings provided at [1]), simple Wiener filtering (using MATLAB's `wiener2`), and a standard non-linear diffusion scheme [23] with a data term. This last method employed a robust Huber function and can be viewed as an MRF model using only local first derivative filters. For this standard non-linear diffusion scheme, a λ weight for the prior term was trained as in the FoE case and the stopping time was selected to produce the optimal denoising result (in terms of PSNR). Figure 4 shows the performance of each of these methods (except for the Wiener filter) for one of the test images. Visually and quantitatively, the FoE model outperforms both Wiener filtering and non-linear diffusion and nearly matches the performance of the specialized Wavelet technique.

Figure 5 shows a performance comparison of the mentioned denoising techniques over all 68 images from the test set at various noise levels. In addition to PSNR we also computed a more perceptually-based similarity measure (SSIM) [22]. The FoE model consistently outperforms both Wiener filtering and standard non-linear diffusion, while closely matching the performance of the current state of the art in image denoising [20]. A signed rank test shows that the performance differences between the FoE and the other methods are statistically significant at a 95% confidence level (except for the SSIM of non-linear diffusion at the highest noise level).

4.2. Image inpainting

In image inpainting [3], the goal is to remove certain parts of an image, for example scratches on a photograph

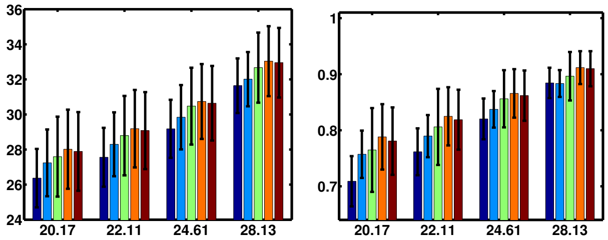

Figure 5. **Denoising results on Berkeley database.** Horizontal axis: PSNR (dB) of the noisy images. Error bars correspond to one standard deviation. **(left)** PSNR in dB for the following models (from left to right): Wiener filter, standard non-linear diffusion, FoE model, and the two variants of [20]. **(right)** Similarity index from [22] for these techniques.

or unwanted occluding objects, without disturbing the overall visual appearance. Typically, the user supplies a mask of pixels that are to be inpainted. Past approaches, such as [3], use a form of diffusion to fill in the masked pixels. This suggests that the diffusion technique we proposed for denoising may also be suitable for this task. In contrast to denoising, we only modify the subset of the pixels specified by the mask. At these pixels there is no observation and hence no likelihood term is used. Our simple inpainting algorithm propagates information using only the FoE prior:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \eta \mathbf{M} \left[\sum_{i=1}^{N} \mathbf{J}_i^- * \psi_i(\mathbf{J}_i * \mathbf{x}^{(t)}) \right]. \quad (5)$$

In this update scheme, the mask $\mathbf{M}$ sets the gradient to zero for all pixels outside of the masked region. In contrast to other algorithms, we make no explicit use of the local gradient direction; local structure information only comes from the responses of the learned filter bank. The filter bank as well as the α_i are the same as in the denoising experiments.

Levin *et al.* [15] have a similar motivation in that they exploit learned models of image statistics for inpainting. Their approach however relies on a small number of hand-selected features, which are used to train the model on the image to be inpainted. We instead use a generic prior and combine information from many more automatically determined features.

Figure 6 shows the result of applying this inpainting scheme in a text removal application in which the mask corresponds to all the pixels that were occluded by the text. The color image was converted to the YCbCr color model, and the algorithm was independently applied to all 3 channels. Since the prior was trained only on gray scale images, this is obviously suboptimal, but nevertheless gives good results. In order to speed up convergence we ran 500 iterations of (5) with $\eta = 10$. Since such a large step size may lead to some numerical instabilities, we "clean up" the image by applying 250 more iterations with $\eta = 0.01$.

The inpainted result (Figure 6 **(b)**) is very similar to the original and qualitatively superior to those in [3]. Quantitatively, our method improves the PSNR by about 1.5dB (29.06dB compared to 27.56dB); the image similarity metric from [22] shows a significant improvement as well (0.9371 compared to 0.9167; where higher is better). The advantage of the rich prior can be seen in the continuity of edges which is better preserved compared with [3]. Figure 6 **(c)** shows a few detail regions comparing our method (center) with [3] (right). Similar qualitative differences can be seen in many parts of the reconstructed image.

5. Summary and Conclusions

While Markov random fields are popular in machine vision for their formal properties, their ability to model complex natural scenes has been limited. To make it practical to model rich image priors we have extended approaches for the sparse coding of image patches to model the potentials of a homogeneous Markov random field capturing local image statistics. The resulting Fields-of-Experts model is based on a rich set of learned filters, and is trained on a generic image database using contrastive divergence. In contrast to previous approaches that use a pre-determined set of filters, all parameters of the model, including the filters, are learned from data. The resulting probabilistic model can be used in any Bayesian inference method requiring a spatial image prior. We have demonstrated the usefulness of the FoE model with applications to denoising and inpainting. The denoising algorithm is straightforward (approximately 20 lines of MATLAB code), yet achieves performance close to the best special-purpose wavelet-based denoising algorithms. The advantage over the wavelet-based methods lies in the generality of the prior and its applicability across different vision problems. We believe the results here represent an important step forward for the utility of MRF models and will be widely applicable.

There are many avenues for future work. By making MRF models much richer, many problems can be revisited with an expectation of improved results. Our current efforts are focused on learning prior models of optical flow, scene depth, color images, and object boundaries. The results here are applicable to image super-resolution, image sharpening, and graphics applications such as image based rendering [6] and others.

There are many avenues along which the FoE model itself can be studied in more detail, such as how the size of the cliques as well as the number of filters influence the quality of the prior. Furthermore, it would be interesting to explore an FoE model using fixed filters (e.g. standard derivative filters or even random filters) in which only the expert parameters α_i are learned from data. The Student-t expert distribution might also be replaced by another, more suitable form. Finally, the convergence and related prop-

Figure 6. **Inpainting with a Field of Experts.** **(a)** Original image with overlaid text. **(b)** Inpainting result from diffusion using the FoE prior. **(c)** Close-up comparison between **a** (left), **b** (middle), and the results from [3] (right).

erties of the diffusion-like algorithm that we propose for inference should be further studied.

Acknowledgments We thank S. Andrews, A. Duci, Y. Gat, S. Geman, H. Haussecker, T. Hoffman, O. Nestares, H. Scharr, E. Simoncelli, M. Welling, and F. Wood for helpful discussions; G. Sapiro and M. Bertalmío for making their inpainting examples available for comparison; and J. Portilla for making his denoising sofware available. This work was supported by Intel Research, NSF ITR grant 0113679 and NIH-NINDS R01 NS 50967-01 as part of the NSF/NIH Collaborative Research in Computational Neuroscience Program. Portions of this work were performed by the authors at Intel Research.

References

[1] http://decsai.ugr.es/~javier/denoise/index.html (software version 1.0.3).

[2] A. Bell and T. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Comp.*, 7(6):1129–1159, 1995.

[3] M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. *ACM SIGGRAPH*, pp. 417–424, 2000.

[4] M. Black, G. Sapiro, D. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Trans. Image Proc.*, 7(3):421–432, 1998.

[5] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. *ICCV*, v. 2, pp. 1033–1038, 1999.

[6] A. Fitzgibbon, Y. Wexler, and A. Zisserman. Image-based rendering using image-based priors. *ICCV*, v. 2, pp. 1176–1183, 2003.

[7] W. Freeman, E. Pasztor, and O. Carmichael. Learning low-level vision. *IJCV*, 40(1):24–47, 2000.

[8] D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *PAMI*, 14(3):367–383, 1992.

[9] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *PAMI*, 6(6):721–741, 1984.

[10] G. Gimel'farb. Texture modeling by multiple pairwise pixel interactions. *PAMI*, 18(11):1110–1114, 1996.

[11] G. Hinton. Product of experts. *ICANN*, v. 1, pp. 1–6, 1999.

[12] G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Comp.*, 14(7):1771–1800, 2002.

[13] J. Huang and D. Mumford. Statistics of natural images and models. *CVPR*, v. 1, pp. 1541–1547, 1999.

[14] N. Jojic, B. Frey, and A. Kannan. Epitomic analysis of appearance and shape. *ICCV*, v. 1, pp. 34–41, 2003.

[15] A. Levin, A. Zomet, and Y. Weiss. Learning how to inpaint from global image statistics. *ICCV*, v. 1, pp. 305–312, 2003.

[16] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *ICCV*, v. 2, pp. 416–423, 2001.

[17] R. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto, 1993.

[18] B. Olshausen and D. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997.

[19] R. Paget and I. Longstaff. Texture synthesis via a noncausal nonparametric multiscale Markov random field. *IEEE Trans. Image Proc.*, 7(6):925–931, 1998.

[20] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Proc.*, 12(11):1338–1351, 2003.

[21] P. Sallee and B. Olshausen. Learning sparse multiscale image representations. *NIPS 15*, pp. 1327–1334, 2003.

[22] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Proc.*, 13(4):600–612, 2004.

[23] J. Weickert. A review of nonlinear diffusion filtering. *Scale-Space Theory in Computer Vision*, pp. 3–28, 1997.

[24] M. Welling, G. Hinton, and S. Osindero. Learning sparse topographic representations with products of Student-t distributions. *NIPS 15*, pp. 1359–1366, 2003.

[25] A. Zalesny and L. van Gool. A compact model for viewpoint dependent texture synthesis. *SMILE 2000*, LNCS 2018, pp. 124–143, 2001.

[26] S. Zhu and D. Mumford. Prior learning and Gibbs reaction-diffusion. *PAMI*, 19(11):1236–1250, 1997.

[27] S. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. *IJCV*, 27(2):107–126, 1998.