Fields of Experts

Stefan Roth* Michael J. Black†

Received: 22 January 2008 / Accepted: 17 November 2008

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 provides a practical method for learning high-order Markov random field (MRF) models with potential functions that extend over large pixel neighborhoods. These clique potentials are modeled using the Product-of-Experts framework that uses 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 specialized techniques.

Keywords: Markov random fields · low-level vision · image modeling · learning · image restoration

1 Introduction

The need for prior models of image or scene structure occurs in many machine vision and graphics problems including stereo, optical flow, denoising, super-resolution, image-based rendering, volumetric surface reconstruction, and texture synthesis to name a few. Whenever one has "noise" or uncertainty, prior models of images (or depth maps, flow fields, three-dimensional volumes, etc.) come into play. Here we develop a method for learning priors for low-level vision problems that can be used in many standard vision, graphics, and image processing algorithms. The key idea is to formulate these priors as a high-order Markov random field (MRF) defined over large neighborhood systems. This is facilitated by exploiting ideas from sparse image patch representations. The resulting *Field of Experts* (FoE) models the prior probability of an image, or other low-level representation, in terms of a random field with overlapping cliques, whose potentials are represented as a Product of Experts (Hinton, 1999). While this model applies to a wide range of low-level representations, this paper focuses on its applications to modeling images. In other work (Roth and Black, 2007b) we have already studied the application to modeling vector-valued optical flow fields; other potential applications will be discussed in more detail below.

To study the application of Fields of Experts to modeling natural images, we train the model on a standard database of natural images (Martin et al., 2001) and develop a diffusion-like scheme that exploits the prior for approximate Bayesian inference. To demonstrate the power of the FoE model, we use it in two different applications: image denoising and image inpainting (Bertalmío et al., 2000) (i. e., filling in missing pixels in an image). Despite the generic nature of the prior and the simplicity of the approximate inference, we obtain results near the state of the art that, until now, were not possible with MRF approaches. Fig. 1 illustrates the application of the FoE model to image denoising and image inpainting. We perform a detailed analysis of various aspects of the model and use image denoising as a running example for quantitative comparisons with the state of the art. We also provide quantitative results for the problem of image inpainting.

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 (Huang and Mumford, 1999; Srivastava et al., 2002; Portilla et al., 2003).

As discussed in more detail below, there have been a number of attempts to overcome these difficulties and to model the statistics of small image patches as well as of entire images. Image patches have been modeled using a variety of sparse coding approaches or other sparse representations (Olshausen and Field, 1997; Teh et al., 2003). Many of these models, however, do not easily generalize to models for entire images, which has limited their impact for machine vision applications. Markov

*Department of Computer Science, TU Darmstadt, Darmstadt, Germany, Email: sroth@cs.tu-darmstadt.de.

The work for this paper was performed while SR was at Brown University.

†Department of Computer Science, Brown University, Providence, RI, USA, Email: black@cs.brown.edu.

(a) (b) (c) (d)

Figure 1: Image restoration using a Field of Experts. (a) Image from the Corel database with additive Gaussian noise ($\sigma = 15$, PSNR = 24.63dB). (b) Image denoised using a Field of Experts (PSNR = 30.72dB). (c) Original photograph with scratches. (d) Image inpainting using the FoE model.

random fields on the other hand can be used to model the statistics of entire images (Geman and Geman, 1984; Besag, 1986). They have been widely used in machine vision, but often exhibit serious limitations. In particular, MRF priors typically exploit hand-crafted clique potentials and small neighborhood systems, which limit the expressiveness of the models and only crudely capture the statistics of natural images. A notable exception to this is the FRAME model by Zhu et al. (1998), which learns clique potentials for larger neighborhoods from training data by modeling the responses of a set of predefined linear filters.

The goal of the current paper is to develop a framework for learning expressive yet generic prior models for low-level vision problems. 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 the use of more elegant optimization methods. The core contribution 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 Product-of-Experts (PoE) framework (Hinton, 1999), which is a generic method for learning high dimensional probability distributions. Previous efforts to model images using Products of Experts (Teh et al., 2003) were patch-based and hence inappropriate for learning generic priors for images or other low-level representations 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 improved modeling power makes them practical for complex tasks[1].

[1] This paper is an extended version of (Roth and Black, 2005).

2 Background and Previous Work

Formal models of image or scene structure play an important role in many vision problems where ambiguity, noise, or missing sensor data make the recovery of world or image structure difficult or impossible. Models of *a priori* structure are used to resolve, or regularize, such problems by providing additional constraints that impose prior assumptions or knowledge. For low-level vision applications the need for modeling such prior knowledge has long been recognized (Geman and Geman, 1984; Poggio et al., 1985), for example due to their frequently ill-posed nature. Often these models entail assuming spatial smoothness or piecewise-smoothness of various image properties. While there are many ways of imposing prior knowledge, we focus here on probabilistic prior models, which have a long history and provide a rigorous framework within which to combine different sources of information. Other regularization methods, such as deterministic ones, including variational approaches (Poggio et al., 1985) will only be discussed briefly.

For problems in low-level vision, such probabilistic prior models of the spatial structure of images or scene properties are often formulated as Markov random fields (MRFs) (Wong, 1968; Kashyap and Chellappa, 1981; Geman and Geman, 1984; Besag, 1986; Marroquin et al., 1987; Szeliski, 1990) (see (Li, 2001) for a recent overview and introduction). Markov random fields have found many areas of application including image denoising (Sebastiani and Godtliebsen, 1997), stereo (Sun et al., 2003), optical flow estimation (Heitz and Bouthemy, 1993), texture classification (Varma and Zisserman, 2005), to name a few. MRFs are undirected graphical models, where the nodes of the graph represent random variables which, in low-level vision applications, typically correspond to image measurements such as pixel intensities, range values, surface normals, or optical flow vectors. Formally, we let the image measurements $\mathbf{x}$ be represented by nodes V in a graph $G = (V, E)$, where E are the edges con-

necting nodes. The edges between the nodes indicate the factorization structure of the probability density $p(\mathbf{x})$ described by the MRF. More precisely, the maximal cliques $\mathbf{x}_{(k)},\ k = 1, \dots, K$ of the graph directly correspond to factors of the probability density. The Hammersley-Clifford theorem (Moussouris, 1974) establishes that we can write the probability density of this graphical model as a Gibbs distribution

$$p(\mathbf{x}) = \frac{1}{Z} \exp\left(-\sum_k U_k(\mathbf{x}_{(k)}) \right), \tag{1}$$

where $\mathbf{x}$ is an image, $U_k(\mathbf{x}_{(k)})$ is the so-called potential function for clique $\mathbf{x}_{(k)}$, and Z is a normalizing term called the partition function. In many cases, it is reasonably assumed that the MRF is homogeneous; i. e., the potential function is the same for all cliques (or in other terms $U_k(\mathbf{x}_{(k)}) = U(\mathbf{x}_{(k)})$). This property gives rise to the translation-invariance of an MRF model for low-level vision applications[2]. Equivalently, we can also write the density under this model as

$$p(\mathbf{x}) = \frac{1}{Z} \prod_k f_k(\mathbf{x}_{(k)}), \tag{2}$$

which makes the factorization structure of the model even more explicit. Here, $f_k(\mathbf{x}_{(k)})$ are the factors defined on clique $\mathbf{x}_{(k)}$, which, in an abuse of terminology, we also sometimes call potentials.

Because of the regular structure of images, the edges of the graph are usually chosen according to some regular neighborhood structure. In almost all cases, this neighborhood structure is chosen *a priori* by hand, although the type of edge structure and the choice of potentials varies substantially. The vast majority of models use a pairwise graph structure; each node (i. e., pixel) is connected to its 4 direct neighbors to the left, right, top, and bottom (Geman and Geman, 1984; Besag, 1986; Sebastiani and Godtliebsen, 1997; Tappen et al., 2003; Neher and Srivastava, 2005). This induces a so-called pairwise MRF, because the maximal cliques are simply pairs of neighboring nodes (pixels), and hence each potential is a function of two pixel values:

$$p(\mathbf{x}) = \frac{1}{Z} \exp\left(-\sum_{(i,j)\in E} U(x_i, x_j) \right) \tag{3}$$

Moreover, the potential is typically defined in terms of some robust function of the difference between neighboring pixel values

$$U(x_i, x_j) = \rho(x_i - x_j), \tag{4}$$

where a typical ρ-function is shown in Figure 2. The truncated quadratic ρ-function in Figure 2 allows spatial discontinuities by not heavily penalizing large neighbor differences.

The difference between neighboring pixel values also has an intuitive interpretation, as it approximates a horizontal or vertical image derivative. The robust function can thus be understood as modeling the statistics of the first derivatives of the images. These statistics, as well as the study of the statistics of natural images in general have received a lot of attention in the literature (Ruderman, 1994; Olshausen and Field, 1996; Huang and Mumford, 1999; Srivastava et al., 2003). A review of this literature is well beyond the scope of this paper and the reader is thus referred to above papers for an overview.

Despite their long history, MRF methods have often produced disappointing results when applied to the recovery of complex scene structure. One of the reasons for this is that the typical pairwise model structure severely restricts the image structures that can be represented. In the majority of the cases, the potentials are furthermore hand-defined, and consequently are only *ad hoc* models of image or scene structure. The resulting probabilistic models typically do not well represent the statistical properties of natural images and scenes, which leads to poor application performance. For example, Figure 2 shows the result of using a pairwise MRF model with a truncated quadratic potential function to remove noise from an image. The estimated image is characteristic of many MRF results; the robust potential function produces sharp boundaries but the result is piecewise smooth and does not capture the more complex textural properties of natural scenes.

For some years it was unclear whether the limited application performance of pairwise MRFs was due to limitations of the model, or due to limitations of the optimization approaches used with non-convex models. Yanover et al. (2006) have recently obtained global solutions to low-level vision problems even with non-convex pairwise MRFs. Their results indicate that pairwise models are incapable of producing very high-quality solutions for stereo problems and suggest that richer models are needed for low-level modeling.

Gimel'farb (1996) proposes a model with multiple and more distant neighbors, which are able to model more complex spatial properties (see also Zalesny and van Gool, 2001). Of particular note, this method learns the neighborhood structure that best represents a set of training data; in the case of texture modeling, different textures result in quite different neighborhood systems. This work however has been limited to modeling specific classes of image texture and our experiments with modeling more diverse classes of generic image structure suggest these methods do not scale well beyond narrow, class-specific, image priors.

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

(a) (b) (c)

Figure 2: Typical pairwise MRF potential and results: (a) Example of a common robust potential function (negative log-probability). This truncated quadratic is often used to model piecewise smooth surfaces. (b) Image with Gaussian noise added. (c) Typical result of denoising using an ad-hoc pairwise MRF (obtained using the method of Felzenszwalb and Huttenlocher (2004)). Note the piecewise smooth nature of the restoration and how it lacks the textural detail of natural scenes.

Figure 3: Filters representing first and second order neighborhood systems (Geman and Reynolds, 1992). The left two filters correspond to first derivatives, the right three filters to second derivatives.

2.1 High-order Markov random fields

There have been a number of attempts to go beyond these very simple pairwise models, which only model the statistics of first derivatives in the image structure (Geman et al., 1992; Zhu and Mumford, 1997; Zhu et al., 1998; Tjelmeland and Besag, 1998; Paget and Longstaff, 1998). The basic insight behind such high-order models is that the generality of MRFs allows for richer models through the use of larger maximal cliques. One approach uses the second derivatives of image structure. Geman and Reynolds (1992), for example, formulate MRF potentials using polynomials determined by the order of the (image) surface being modeled ($k = 1, 2, 3$ for constant, planar, or quadric).

In the context of this work, we think of these polynomials as defining linear filters, $\mathbf{J}_i$, over local neighborhoods of pixels. For the quadric case, the corresponding 3×3 filters are shown in Figure 3. In this example, the maximal cliques are square patches of 3×3 pixels and their corresponding potential for clique $\mathbf{x}_{(k)}$ centered at pixel k is written as

$$U(\mathbf{x}_{(k)}) = \sum_{i=1}^{5} \rho(\mathbf{J}_i^{\mathrm{T}} \mathbf{x}_{(k)}), \tag{5}$$

where the $\mathbf{J}_i$ are the shown derivative filters. When ρ is a robust potential, this corresponds to the weak plate model (Blake and Zisserman, 1987).

The above models are capable of representing richer structural properties beyond the piecewise spatial smoothness of pairwise models, but have remained largely hand-defined. The designer decides what might be a good model for a particular problem and chooses a neighborhood system, the potential function, and its parameters.

2.2 Learning MRF models

Hand selection of parameters is not only somewhat arbitrary and can cause models to only poorly capture the statistics of the data, but is also particularly cumbersome for models with many parameters. There exist a number of methods for learning the parameters of the potentials from training data (see (Li, 2001) for an overview). In the context of images, Besag (1986) for example uses the pseudo-likelihood criterion to learn the parameters of a parametric potential function for a pairwise MRF from training data. Applying pseudo-likelihood in the high-order case is, however, hindered by the fact that computing the necessary conditionals is often difficult.

For Markov random field modeling in general (i. e., not specifically for vision applications), maximum likelihood (ML) (Geyer, 1991) is probably the most widely used learning criterion. Nevertheless, due to its often extreme computational demands, it has long been avoided. Hinton (2002) recently proposed a learning rule for energy-based models, called contrastive divergence (CD), which resembles maximum likelihood, but allows for much more efficient computation. In this paper we apply contrastive divergence to the problem of learning Markov random field models of images; details will be discussed below. Other learning methods include iterative scaling (Darroch and Ratcliff, 1972; della Pietra et al., 1997), score matching (Hyvaärinen, 2005), discriminative training of energy-based models (LeCun and Huang, 2005), as well as a large set of variational (and related) approximations to maximum likelihood (Jordan et al., 1999; Yedidia et al., 2003; Welling and Sutton, 2005; Minka, 2005).

In this work, Markov random fields are used to model

prior distributions of images and potentially other scene properties, but in the literature, MRF models have also been used to directly model the posterior distribution for particular low-level vision applications. For these applications, it can be beneficial to train MRF models discriminatively (Ning et al., 2005; Kumar and Hebert, 2006). This is not pursued here.

In low-level vision applications, most of these learning methods have not found widespread use. Nevertheless, maximum likelihood has been successfully applied to the problem of modeling images (Zhu and Mumford, 1997; Descombes et al., 1999). One model that is of particular importance in the context of this paper is the FRAME model of Zhu et al. (1998). It took a step toward more practical MRF models, as it is of high-order and allows its parameters to be learned from training data, for example from a set of natural images (Zhu and Mumford, 1997). This method uses a "filter pursuit" strategy to select filters from a pre-defined set of standard image filters; the potential functions model the responses of these filters using a flexible, discrete, non-parametric representation. The discrete nature of this representation complicates its use, and, while the method exhibited good results for texture synthesis, the reported image restoration results appear to fall below the current state of the art.

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. (2000) propose an MRF model that uses example image patches and a measure of consistency between them to model scene structure. This idea has been exploited as a prior model for image based rendering (Fitzgibbon et al., 2003) and super-resolution (Pickup et al., 2004). The roots of these models are in example-based texture synthesis (Efros and Leung, 1999).

In contrast, our approach uses parametric (and differentiable) potential functions applied to filter responses. Unlike the FRAME model, we learn the filters themselves as well as the parameters of the potential functions. As we will show, the resulting filters appear quite different from standard filters and achieve better performance than do standard filters in a variety of tasks. A computational advantage of our parametric model is that it is differentiable, which facilitates various learning and inference methods.

2.3 Inference

To apply MRF models to actual problems in low-level vision, we compute a solution using tools from probabilistic inference. Inference in this context typically means either performing maximum a-posteriori (MAP) estimation, or computing expectations over the solution space. Common to all MRF models in low-level vision is the fact that inference is challenging, both algorithmically and computationally. The loopy structure of the underlying graph makes exact inference NP-hard in the general case, although special cases exist where polynomial time algorithms are known. Because of that, inference is usually performed in an approximate fashion, for which there are a wealth of different techniques. Classical techniques include Gibbs sampling (Geman and Geman, 1984), deterministic annealing (Hofmann et al., 1998), and iterated conditional modes (Besag, 1986). More recently, algorithms based on graph cuts (Kolmogorov and Zabih, 2004) have become very popular for MAP inference. Variational techniques and related ones, such as belief propagation (Yedidia et al., 2003), have also enjoyed enormous popularity, both for MAP inference and computing marginals. Nevertheless, even with such modern approximate techniques, inference can be quite slow, which has prompted the development of models that simplify inference (Felzenszwalb and Huttenlocher, 2004). While these may make inference easier, they typically give the answer to the wrong problem, as the model does not capture the relevant statistics well (cf. Fig. 2).

Inference in high-order MRF models is particularly demanding, because the larger size of the cliques complicates the (approximate) inference process. Because of that, we rely on very simple approximate inference schemes using the conjugate gradient method. Nevertheless, the applicability of more sophisticated inference techniques to models such as the one proposed here, promises to be a fruitful area for future work (cf. Potetz, 2007; Kohli et al., 2007).

2.4 Other regularization methods

It is worth noting that prior models of spatial structure are also often formulated as energy terms (e. g., log-probability) and used in non-probabilistic regularization methods (Poggio et al., 1985). While we pursue a probabilistic framework here, the methods are applicable to contexts where deterministic regularization methods are applied. This suggests that our FoE framework is applicable to a wide class of variational frameworks (see (Schnörr et al., 1996) for a review of such techniques).

Interestingly, many of these deterministic regularization approaches, for example variational (Schnörr et al., 1996) or nonlinear-diffusion related methods (Weickert, 1997), suffer from very similar limitations as typical MRF approaches. This is because they penalize large image derivatives similar to pairwise MRFs. Moreover, in order to show the existence of a unique global optimum, many models are restricted to be convex, and are furthermore mostly hand-defined. Non-convex regularizers often show superior performance in practice (Black et al., 1998), and the missing connection to the statistics of natural images or scenes can be viewed as problematic. There have been variational and diffusion-related approaches that try to overcome some of these limitations (Gilboa et al., 2004; Trobin et al., 2008).

2.5 Models of image patches

Even though typically motivated from an image-coding or neurophysiological point of view, there is a large amount of related work in the area of sparse coding and component analysis, which attempts to model complex image structure. Such models typically encode structural properties of images through a set of linear filter responses or components. For example, Principal Component Analysis (PCA) (Roweis and Ghahramani, 1999) of image patches yields visually intuitive components, some of which resemble derivative filters of various orders and orientations. The marginal statistics of such filters are highly non-Gaussian (Ruderman, 1994) and are furthermore not independent, making this model unsuitable for probabilistically modeling image patches.

Independent Component Analysis (ICA) (Bell and Sejnowski, 1995), for example, assumes non-Gaussian statistics and finds the linear components such that the statistical dependence between the components is minimized. As opposed to the principal components, ICA yields localized components, which resemble Gabor filters of various orientations, scales, and locations. Since the components (i. e., filters) $\mathbf{J}_i \in \mathbb{R}^n$ found by ICA are by assumption independent, one can define a probabilistic model of image patches $\mathbf{x} \in \mathbb{R}^n$ by multiplying the marginal distributions, $p_i(\mathbf{J}_i^{\mathrm{T}}\mathbf{x})$, of the filter responses:

$$p(\mathbf{x}) \propto \prod_{i=1}^{n} p_i(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}). \tag{6}$$

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 $\mathbf{J}_i$. However, in the 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. Somewhat similar to ICA are sparse-coding approaches (e. g., Olshausen and Field, 1996), which also represent image patches in terms of a linear combination of learned filters, but in a synthesis-based manner (see also Elad et al., 2006).

Most of these methods, however, focus on image *patches* and provide no direct way of modeling the statistics of whole *images*. Several authors have explored extending sparse coding models to full images. For example, Sallee and Olshausen (2003) propose a prior model for entire images, but inference with this model requires Gibbs sampling, which makes it somewhat problematic for many machine vision applications. Other work has integrated translation invariance constraints into the basis finding process (Hashimoto and Kurata, 2000; Wersing et al., 2003). The focus in that work, however, remains on modeling the image in terms of a sparse linear combination of basis filters with an emphasis on the implications for human vision. Modeling entire images has also been considered in the context of image denoising (Elad and Aharon, 2006). While these approaches are motivated in a way that is quite different from Markov random field approaches as emphasized here, they are similar in that they model the response to linear filters and even allow the filters themselves to be learned. Another difference is that the model of Elad and Aharon (2006) is not trained offline on a general database of natural images, but the parameters are instead inferred "online" in the context of the application at hand. While this may also have advantages, it for example makes the application to problems with missing data (e. g., inpainting) more difficult.

Popular approaches to modeling images also include wavelet-based methods (Portilla et al., 2003). Since neighboring wavelet coefficients are not independent, it is beneficial to model their dependencies. This has for example been done in patches using Products of Experts (Gehler and Welling, 2006) or over entire wavelet subbands using MRFs (Lyu and Simoncelli, 2007). While such a modeling of the dependencies between wavelet coefficients bears similarities to the FoE model, these wavelet approaches do not directly yield generic image priors due to the fact that they model the coefficients of an overcomplete wavelet transform. Their applicability has thus mostly been restricted to specific applications, such as denoising.

2.6 Products of Experts

Products of Experts (PoE) (Hinton, 1999) have also been used to model image patches (Welling et al., 2003; Teh et al., 2003) overcoming some of the limitations of the complete (square) ICA model in (6). Since the ideas behind this model are very important for understanding the model we propose here, we will discuss them in some detail. 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 or directions (corresponding to the basis vectors in sparse coding models). Projection onto these directions corresponds to filtering the image patch with the basis vector, $\mathbf{J}_i$. 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, Teh et al. (2003) propose the use of Student-t experts. The full Product of t-distribution (PoT) model can be written as

$$p(\mathbf{x}; \Theta) = \frac{1}{Z(\Theta)} \prod_{i=1}^{N} \phi(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \alpha_i), \tag{7}$$

where $\Theta = \{\theta_1, \ldots, \theta_N\}$ with $\theta_i = \{\alpha_i, \mathbf{J}_i\}$ are the parameters to be learned. The experts $\phi(\cdot\,; \cdot)$ have the form

$$\phi(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \alpha_i) = \left(1 + \frac{1}{2}(\mathbf{J}_i^{\mathrm{T}}\mathbf{x})^2\right)^{-\alpha_i}, \tag{8}$$

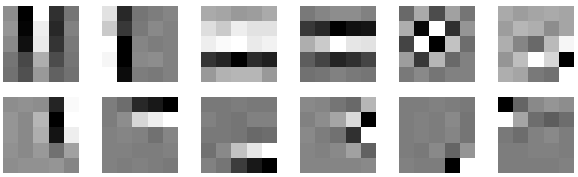


Figure 4: **Selection of the 5×5 filters** obtained by training the *Product-of-Experts* model on a generic image database.

and $Z(\Theta)$ is the normalizing, or partition, function. It is important to note that $\mathbf{x}$ here is now an image patch and not the full image. The α_i are assumed to be positive, which is needed to make the ϕ 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

$$p(\mathbf{x};\, \Theta) = \frac{1}{Z(\Theta)} \exp(-E_{\text{PoE}}(\mathbf{x}, \Theta)) \tag{9}$$

with

$$E_{\text{PoE}}(\mathbf{x}, \Theta) = -\sum_{i=1}^{N} \log \phi(\mathbf{J}_i^{\mathrm{T}}\mathbf{x};\, \alpha_i). \tag{10}$$

One important property of this model is that all parameters can be 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 (square or 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.

Fig. 4 shows a selection of the 24 filters obtained by training this PoE model on 5×5 image patches. Training was done on the image data as described in Section 3.4 using the learning algorithm described by Teh et al. (2003). The filters learned by this model are similar to those obtained using a non-parametric ICA technique or standard sparse coding approaches. Here the shape of the t-distribution has the effect of a sparseness prior. It is possible to train models that are several times over-complete (Olshausen and Field, 1997; Teh et al., 2003); the characteristics of the filters remain the same.

Despite the fact that the PoT models small image patches rather than defining a prior model over an entire image, Welling et al. (2003) suggest an algorithm that uses the filters to denoise images of arbitrary size. The resulting algorithm, however, does not easily generalize to other image restoration problems such as image inpainting. Our focus here is not on any specific application such as denoising, but rather on finding a good general purpose framework for priors in low-level vision. We argue that to that end it is important to model whole images and not just small patches.

3 Fields of Experts

3.1 Basic model

To overcome the limitations of pairwise MRFs and patch-based models we define a high-order Markov random field for entire images $\mathbf{x} \in \mathbb{R}^{L \times M}$ using a neighborhood system that connects all nodes in an $m \times m$ square region (cf. Geman et al., 1992; Tjelmeland and Besag, 1998; Zhu et al., 1998). This is done for all *overlapping* $m \times m$ regions of $\mathbf{x}$, which now denotes an entire image rather than a small image patch. Every such neighborhood centered on a node (pixel) $k = 1, \ldots, K$ defines a maximal clique $\mathbf{x}_{(k)}$ in the graph. Without loss of generality we usually assume that the maximal cliques in the MRF are square pixel patches of a fixed size. Other, non-square, neighborhoods can be used (cf. Geman and Reynolds, 1992), and will be discussed further in Section 5.3.

We propose to represent the MRF potentials as a Product of Experts (Hinton, 1999) with the same basic form as in Eq. (7). This means that the potentials are defined with a set of expert functions that model filter responses to a bank of linear filters. This global prior for low-level vision is a Markov random field of "experts", or more concisely a *Field of Experts* (FoE). More formally, Eq. (7) is used to define the potential function (written as factor):

$$f(\mathbf{x}_{(k)}) = f_{\text{PoE}}(\mathbf{x}_{(k)};\, \Theta) = \prod_{i=1}^{N} \phi(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}_{(k)};\, \boldsymbol{\alpha}_i). \tag{11}$$

Each $\mathbf{J}_i$ is a linear filter that defines the direction (in the vector space of the pixel values in $\mathbf{x}_{(k)}$) that the corresponding expert $\phi(\cdot;\, \cdot)$ is modeling, and $\boldsymbol{\alpha}_i$ is its corresponding (set of) expert parameter(s). $\Theta = \{\mathbf{J}_i, \boldsymbol{\alpha}_i \,|\, i = 1, \ldots, N\}$ is the set of all model parameters. The number of experts and associated filters, N, is not prescribed in a particular way; we can choose it based on criteria such as the quality of the model and computational expense (see also Section 5.3). Since each factor can be unnormalized, we neglect the normalization component of Eq. (7) for simplicity. Note that we assume in this paper that the image $\mathbf{x}$ is a continuous-valued random vector; discrete-valued spatial data can be dealt with in similar ways (Stewart et al., 2008).

Overall, the Field-of-Experts model is thus defined as

$$p_{\text{FoE}}(\mathbf{x};\, \Theta) = \frac{1}{Z(\Theta)} \prod_{k=1}^{K} \prod_{i=1}^{N} \phi(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}_{(k)};\, \boldsymbol{\alpha}_i). \tag{12}$$

All components retain their definitions from above. It is very important to note here that this definition does

not imply that we take a *trained* PoE model with fixed parameters Θ and use it directly to model the potential function. This would be incorrect, because the PoE model described in Section 2.6 was trained on independent patches. In case of the FoE, the pixel regions $\mathbf{x}_{(k)}$ that correspond to the maximal cliques are overlapping and thus not independent. Instead, we use the *untrained* PoE model to define the potentials, and learn the parameters Θ in the context of the full MRF model. What distinguishes this model from that of Teh et al. (2003) is that it explicitly models the overlap of image patches and the resulting statistical dependence; the filters $\mathbf{J}_i$, as well as the expert parameters $\boldsymbol{\alpha}_i$ must account for this dependence (to the extent they can). It is also important to note that the FoE parameters $\Theta = \{\mathbf{J}_i, \boldsymbol{\alpha}_i \,|\, i = 1, \dots, N\}$ are shared between all maximal cliques and their associated factors. This keeps the number of parameters moderate, because it only depends on the size of the maximal cliques and the number of experts, but not on the size of the image itself. Beyond that, the model applies to images of an arbitrary size and is translation invariant because of the homogeneity of the potential functions. This means that the FoE model can be thought of as a translation-invariant PoE model.

Comparing the FoE to the FRAME model of Zhu et al. (1998), we should note that while the models look similar (both are high-order MRF models with "experts" modeling linear filter responses), there are important differences. While the FRAME model allows learning some of the parameters of the potential functions from data, the candidate set of filters used to define the potentials is chosen by hand. In the model developed here, we learn the filters alongside the other parameters; to enable that, our expert functions are parametric and thus less flexible.

Similar to the PoE (at least in its overcomplete form) (Teh et al., 2003) and to most Markov random field models (Li, 2001), computing the partition function $Z(\Theta)$ of the FoE is generally intractable. One important fact to note is that the partition function depends on the parameters, Θ, of our model. Nevertheless, most inference algorithms, such as the ones discussed in Section 3.5, do not require this normalization term to be known. During learning, on the other hand, we do need to take the normalization term into account, as we will see shortly.

We should also note that the FoE model has certain similarities to convolutional neural networks (Ning et al., 2005). Both types of models apply banks of linear filters to whole images in a convolutional fashion and model the filter responses using a non-linear function. A crucial difference is that convolutional networks are typically trained discriminatively in the context of a specific application, whereas the probabilistic nature of the FoE allows us to learn a generic prior that can be directly used in different applications.

We will frequently work with the log of the FoE model, and it is thus convenient to rewrite the model as

$$\begin{aligned} p_{\text{FoE}}(\mathbf{x};\, \Theta) &= \frac{1}{Z(\Theta)} \exp\left\{-E_{\text{FoE}}(\mathbf{x};\, \Theta)\right\} \\ &= \frac{1}{Z(\Theta)} \exp\left\{\sum_{k=1}^{K} \sum_{i=1}^{N} \psi(\mathbf{J}_i^{\mathrm{T}} \mathbf{x}_{(k)};\, \boldsymbol{\alpha}_i)\right\}, \end{aligned} \tag{13}$$

where log-experts are defined as $\psi(\cdot;\, \boldsymbol{\alpha}_i) = \log \phi(\cdot;\, \boldsymbol{\alpha}_i)$.

3.2 The experts

To make this general framework more specific, we have to choose appropriate expert functions $\phi(y;\, \boldsymbol{\alpha})$; y here stands for the response to one of the linear filters. Similar to the PoE model, we have substantial freedom in doing so. The important criteria for choosing experts from a mathematical point of view are that the expert and its log are continuous and differentiable with respect to y and $\boldsymbol{\alpha}$; we will rely on these criteria during learning and inference. From a modeling perspective, we want to choose experts that in the context of the full model give rise to statistical properties that resemble the data we want to model. As mentioned above, natural images and other scene properties have heavy-tailed marginal distributions, which motivates the use of heavy-tailed, highly kurtotic experts.

There are two experts that we consider here: (1) The very heavy-tailed Student t-distribution as it has been used in the PoE framework for modeling image patches (Teh et al., 2003) (cf. Eq. (8)). (2) A less heavy-tailed expert that is loosely based on the L1 norm, which has been successfully applied to a number of problems in image restoration (e. g., Donoho et al., 2006). Since the L1 norm is not differentiable, we employ the "smooth" penalty function proposed by Charbonnier et al. (1997), which leads to the following expert:

$$\phi_{\text{C}}(y;\, \alpha, \beta) = e^{-\alpha\sqrt{\beta + y^2}}. \tag{14}$$

We fix the offset β to 1, but because y can be arbitrarily scaled through the filter norms, this incurs no loss of generality. One aspect to note is that the Charbonnier expert is convex (more precisely its energy is convex). Consequently, FoE models with Charbonnier experts have a convex energy.

Later on, we will require the logarithm of the expert functions, as well as the partial derivatives of the log w. r. t. α and y. Since none of these are hard to derive, we omit the details for brevity.

3.3 Contrastive divergence learning

The parameters, $\theta_i \in \Theta$, which include the expert parameters $\boldsymbol{\alpha}_i$ and the elements of the filters $\mathbf{J}_i$, can

be learned from a set of D training images $X = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(D)}\}$, like those in Figure 5, by maximizing its likelihood. Maximizing the likelihood of a training set of images 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 that the model distribution is as close to the data distribution as possible under the model. Since there is no closed form solution for the ML parameters, we perform a gradient ascent on the log-likelihood. Taking the partial derivative of the log-likelihood with respect to a parameter θ_i leads to the parameter update

$$\delta\theta_i = \eta \left[\left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_p - \left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_X \right], \qquad (15)$$

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 $p(\mathbf{x}; \Theta)$. 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 by repeatedly drawing samples from $p(\mathbf{x}; \Theta)$ using Markov chain Monte Carlo (MCMC) sampling. In our implementation, we use a hybrid Monte Carlo (HMC) sampler (Neal, 1993), 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* (Hinton, 2002) 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

$$\delta\theta_i = \eta \left[\left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_{p^j} - \left\langle \frac{\partial E_{\text{FoE}}}{\partial \theta_i} \right\rangle_{p^0} \right]. \qquad (16)$$

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 (2002) justifies this more formally and shows that contrastive divergence learning is typically a good approximation to a maximum likelihood estimation of the parameters.

Figure 5: Subset of the images used for training. The training database has images of animals, landscapes, people, architecture, etc.

3.4 Implementation details

In order to correctly capture the spatial dependencies of neighboring cliques (or equivalently the overlapping 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. As a trade-off, we train on image regions that have 3 to 5 times the width and height of the maximal cliques; e. g., in case of 5×5 cliques we train on 15×15 images. The training data contains 20000 image regions randomly cropped from the images of the Berkeley Segmentation Benchmark (Martin et al., 2001). The color images were converted to the YCbCr color space, from which we obtained gray scale versions by ignoring the chromatic channels Cr and Cb. While we do not explore modeling color images here, the FoE has been applied to color images as well. McAuley et al. (2006) describe an extension to RGB images, in which the cliques and filters are simply extended to the third dimension (corresponding to the color channels). Even though the authors did not use the full learning approach proposed here, they still obtained encouraging color denoising results.

Instead of using the entire dataset at each iteration of the contrastive divergence procedure, we split the data into "mini batches" of 200 images each, and used only the data from one batch at each iteration. This so-called stochastic gradient ascent procedure (Bottou, 2004) sped up learning considerably. In most of our experiments we used 5000 stochastic gradient iterations, each performing a single contrastive divergence step (i. e., $j = 1$) with a learning rate of $\eta = 0.01$. The contrastive divergence step relied on hybrid Monte Carlo sampling using 30 leaps; the leap size was adjusted automatically, so that the acceptance rate was near 90%. In our experiments we found the results were not very sensitive to the exact values of these parameters.

Due to the necessary Monte Carlo sampling, the pa-

rameter updates are stochastic and thus exhibit a certain degree of variation due to sampling. To stabilize the learning procedure, we introduce a momentum term as suggested by Teh et al. (2003); each parameter update is a weighted sum of the previous update (weight 0.9) and the intended update (weight 0.1) as determined by the current samples. The stochastic character of the updates also makes it difficult to establish automated convergence criteria. We thus manually monitor convergence. The learning algorithm can furthermore be stabilized by ensuring that the expert parameters α_i are positive. Positive expert parameters are required to make each expert have a proper probability density, but we should note that due to the "overcompleteness" of the FoE model, not all the α_i have to be positive for the FoE to represent a proper probability density. In most of our experiments, we ensure positivity of the expert parameters by updating their logarithm.

As we will discuss in some more detail alongside the experiments in Section 5.3, we investigated representing the filter vectors in 3 different bases. In other terms, instead of learning the filters $\mathbf{J}$ directly, we represented the filters as $\mathbf{J} = \mathbf{A}^{\mathrm{T}}\tilde{\mathbf{J}}$, where $\mathbf{A}$ is the basis in which the filters are defined, and learn the basis representation $\tilde{\mathbf{J}}$ using contrastive divergence. It is important to note that this does not change the learning objective in any way, but due to the fact that we use a local stochastic learning rule, it may still lead to different learned parameters. For most of our experiments we use an inverse whitening transformation as the basis (see Section 5.3.1). Furthermore, we also make the model invariant to global changes in gray level by removing the basis vector that represents uniform patches.

Fig. 6 shows the filters learned by training a FoE model with 5×5 pixel cliques, 24 filters, and Student-t experts. 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 lack, however, the clearly interpretable structure of the filters learned using the standard PoE model (cf. Fig. 4). We conjecture that this results from the filters having to account for the statistical dependency of the image structure in overlapping patches, and show in Section 5.3 that these somewhat unusual filters are important for application performance.

Despite the stochastic gradient procedure and the use of efficient sampling techniques, learning is still computationally intensive. Training a 3×3 model with 8 filters on 15×15 patches takes 8 CPU hours on a single PC (Intel Pentium D, 3.2 GHz). Training a 5×5 model with 24 filters requires roughly 24 CPU hours. We should note though that training occurs offline ahead of application time, and is done only once per kind of data to be modeled.

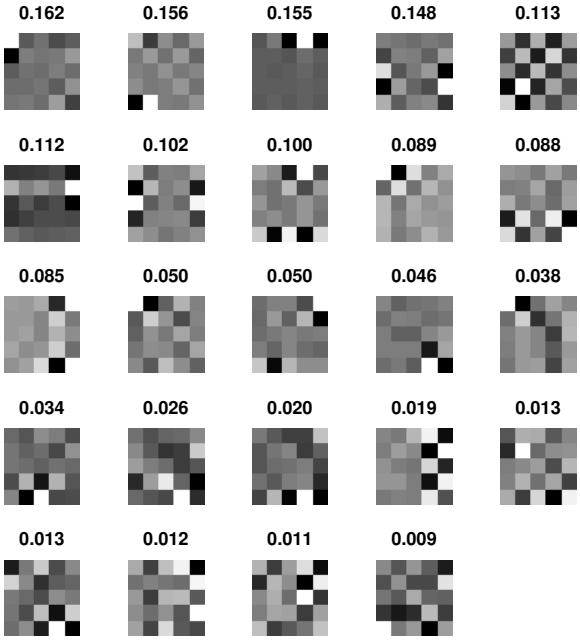


Figure 6: 5×5 filters obtained by training the *Field-of-Experts* model with Student-t experts on a generic image database. Each filter is shown with the corresponding α_i, which can be viewed as weights multiplying the log experts in the energy formulation of the model.

3.5 Inference

There are a number of methods that can be used for probabilistic inference with the Field-of-Experts model. In most of the applications of the model, we are interested in finding the solution that has the largest posterior probability (MAP estimation). As already discussed in Section 2.3 in the context of general MRF models, inference with FoEs will almost necessarily have to be approximate, not only because of the loopy graph structure, but also because of the high dimensionality of images, and the large state space (large number of gray levels, even when discretized). In principle, sampling techniques, such as Gibbs sampling, can be employed in conjunction with the FoE model. Due to the cost of sampling the posterior model, they are computationally very intensive.

Recently, work on approximate inference in graphical models has focused on belief propagation (BP) (Yedidia et al., 2003), a message passing algorithm that typically leads to very good approximations (if it converges). Efficient variants of belief propagation have been recently applied to image restoration problems, which were previously infeasible due to the large number of states (gray values) in images (Felzenszwalb and Huttenlocher, 2004). For these efficient methods simple pairwise Markov random fields are used to model images. In the case of the FoE model on the other hand, applying belief propagation is unfortunately not

straightforward, because the larger cliques cause an explosion in the state space size: A 5×5 FoE model, for example, has maximal cliques with 25 variables (as compared to just 2 in the pairwise case). Every factor node in the factor graph representation of the MRF, on which the BP message passing scheme is based (cf. Yedidia et al., 2003), thus subsumes 25 pixels. Assuming 256 gray levels at each pixel, each factor node has 256^{25} states, which makes it intractable to store the beliefs and messages. Some progress has been made in applying belief propagation to high-order MRF models including FoEs (Lan et al., 2006; Potetz, 2007). So far, this has been limited in practice to cliques of 2×2 pixels.

In our experiments in Sections 4 and 5, we instead use very simple gradient-based optimization techniques for approximate MAP inference. They find a local optimum, but require much less computational power and memory than BP. To perform the optimization, we require the gradient of the log-density with respect to the image itself. Following Zhu and Mumford (1997), we can express the gradient using simple convolution operations (see (Roth, 2007) for details):

$$\nabla_{\mathbf{x}} \log p_{\text{FoE}}(\mathbf{x}; \Theta) = \sum_{i=1}^{N} \mathbf{J}_{-}^{(i)} * \boldsymbol{\psi}'(\mathbf{J}^{(i)} * \mathbf{x}; \boldsymbol{\alpha}_i). \quad (17)$$

Here $\mathbf{J}^{(i)}$ is a convolution filter corresponding to $\mathbf{J}_i$, $\mathbf{J}_{-}^{(i)}$ is a convolution filter that has been obtained by mirroring $\mathbf{J}^{(i)}$ around its center, and $\boldsymbol{\psi}'$ is the derivative of the log-expert. In contrast to the FRAME model (Zhu et al., 1998), this derivative can be computed without making approximations to the model due to the parametric nature of the experts. Because the overall expression is based on convolutions, it is very simple and efficient to implement. Moreover, these gradient-based techniques bear interesting connections to nonlinear diffusion and many related PDE methods (Weickert, 1997).

4 Example Applications

To illustrate the capabilities of the Field-of-Experts model as a prior model of images, we demonstrate its use in experiments on image denoising and image inpainting.

4.1 Image denoising

Image denoising is a widely studied problem. Some of the most accurate techniques are based on overcomplete wavelet decompositions and model the joint statistics of several neighboring wavelet coefficients (Portilla et al., 2003; Gehler and Welling, 2006; Lyu and Simoncelli, 2007). Another widely used category of techniques is based on partial differential equations or variational approaches and includes nonlinear diffusion (Weickert, 1997). Recently, a number of authors have proposed denoising algorithms based on non-local averaging (Buades et al., 2004; Kervrann and Boulanger, 2006). A more thorough review of denoising techniques is beyond the scope of this paper, but is given in the mentioned references.

In contrast to a number of the above schemes, we focus on a Bayesian formulation with a probabilistic prior model of the spatial properties of images. As in a general Bayesian image restoration framework, our goal is to find the true image $\mathbf{x}$ given an observed image $\mathbf{y}$ by maximizing the posterior probability $p(\mathbf{x}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{x}) \cdot p(\mathbf{x})$. To simplify the evaluation, we assume homogeneous, pixelwise independent Gaussian noise, as is common in the denoising literature (for an example of FoE denoising with real noise and more realistic noise models see (Moldovan et al., 2006)). Accordingly, we write the likelihood as

$$p(\mathbf{y}|\mathbf{x}) \propto \prod_{k=1}^{L \cdot M} \exp\left(-\frac{1}{2\sigma^2}(y_k - x_k)^2\right), \quad (18)$$

where k ranges over the pixels in the image. Furthermore, we use the FoE model as the prior, i. e., $p(\mathbf{x}) = p_{\text{FoE}}(\mathbf{x})$.

To emphasize the practicality of the proposed model, we performed a simple gradient-based local optimization of the logarithm of the posterior probability. Together with an optional weight ω for the log-prior, we can write the gradient of the log-posterior as follows:

$$\nabla_{\mathbf{x}} \log p(\mathbf{x}|\mathbf{y}) = \omega \cdot \left[\sum_{i=1}^{N} \mathbf{J}_{-}^{(i)} * \boldsymbol{\psi}'(\mathbf{J}^{(i)} * \mathbf{x}; \boldsymbol{\alpha}_i)\right] + \frac{1}{\sigma^2}(\mathbf{y} - \mathbf{x}). \quad (19)$$

If we performed a standard gradient ascent based on this expression, we could directly relate the algorithm to nonlinear diffusion methods (Zhu and Mumford, 1997). In particular, if we had only two filters (x- and y-derivative filters) then the gradient ascent procedure would be very similar to standard nonlinear diffusion filtering with a data term. Instead of standard gradient ascent, we use a conjugate gradient method for optimization in our experiments based on the implementation of Rasmussen (2006). The optional weight ω can be used to adjust the strength of the prior compared to the likelihood. If both prior and likelihood were very accurately modeled and if we could find the global optimum of the denoising objective, such a weight would not be necessary. In practice, we learn the value of this parameter on a validation set, which can substantially improve performance. To make the interpretation easier, we parametrized this weight as $\omega(\lambda) = \frac{\lambda}{1-\lambda}$, where $\lambda \in (0, 1)$.

Even though denoising proceeds in very similar ways to nonlinear diffusion, our prior model uses many more filters. The key advantage of the FoE model over standard diffusion techniques is that it tells us how to build

richer prior models that combine more filters over larger neighborhoods in a principled way.

4.2 Image inpainting

In image inpainting (Bertalmío et al., 2000), the goal is to remove certain parts of an image, for example scratches on a photograph or unwanted occluding objects, without disturbing the overall visual appearance. Typically, the user supplies a mask, $\mathcal{M}$, of pixels that are to be filled in by the algorithm.

To define an appropriate likelihood, we assume that the masked pixels can take on any gray value with equal probability, and simply make the likelihood uniform there. Pixels that are not masked should not be modified at all; we can model this using a Dirac delta centered on the pixel value to be preserved. We thus write the likelihood for image inpainting as

$$p(\mathbf{y}|\mathbf{x}) = \prod_{k=1}^{L \cdot M} p(y_k|x_k) \propto \prod_{k=1}^{L \cdot M} \begin{cases} 1, & k \in \mathcal{M} \\ \delta(y_k - x_k), & k \notin \mathcal{M} \end{cases} . \tag{20}$$

To perform inpainting, we use a simple gradient ascent procedure, in which we leave the unmasked pixels untouched, while modifying the masked pixels only based on the FoE prior. We can do this by defining a mask matrix $\mathbf{M}$ that sets the gradient to zero for all pixels outside of the masked region $\mathcal{M}$:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \eta \mathbf{M} \left[\sum_{i=1}^{N} \mathbf{J}_i^- * \boldsymbol{\psi}'(\mathbf{J}_i * \mathbf{x}^{(t)}; \boldsymbol{\alpha}_i) \right] . \tag{21}$$

Here, η is the step size of the gradient ascent procedure. In contrast to other algorithms, we make no explicit use of the local image gradient direction; local structure information only comes from the responses to the learned filter bank. The filter bank as well as the $\boldsymbol{\alpha}_i$ are the same as in the denoising experiments.

Levin et al. (2003) 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 automatically determined features.

One important limitation of our approach is that it cannot fill in texture, but only "shading". Other techniques have been developed that can also fill in textured areas by synthesizing or copying appropriate textures (e. g., Criminisi et al., 2004).

5 Experimental Evaluation

Using the FoE model trained as in Section 3 (5×5 cliques with 24 filters and Student-t experts) we performed a number of denoising experiments. The experiments conducted here assume a known noise distribution, which allows us to focus on the effects of the prior alone. 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. The evaluation of the denoising performance relies on two measurements: (1) The peak signal-to-noise ratio (PSNR) defined as

$$\text{PSNR} = 20 \log_{10} \frac{255}{\sigma_e}, \tag{22}$$

where σ_e is the standard deviation of the pixelwise image error. PSNR is given in decibels (dB); a reduction of the noise by a factor of 2 leads to a PSNR increase of about 6dB. The PSNR is a very widely used evaluation criterion for denoising, but has the limitation that it does not fully reflect the perceptual quality of an image to the human observer. (2) Since the goal of most image restoration problems is to optimize perceived image quality, we also employ the structural similarity index (SSIM) (Wang et al., 2004). SSIM provides a perceptually more plausible image error measure, which has been verified in psychophysical experiments. SSIM values range between 0 and 1, where 1 is a perfect restoration.

We performed denoising using at most 5000 iterations of conjugate gradient. In essentially all of the cases, the ascent terminated in fewer than 5000 iterations because a local optimum had been reached. Experimentally, we found that the best results were obtained with an additional weight as introduced above, which furthermore depended on the amount of noise added. We determined the appropriate λ trade-off parameter for denoising using an automatic training procedure that was carried out for each noise standard deviation that we used. We manually picked a representative set of 10 images from the *training* database, cropped them randomly to 200×200 pixels, and added synthetic Gaussian noise of the appropriate standard deviation. Each artificially corrupted image was then denoised using the conjugate gradient method, and the optimal λ parameter with respect to the PSNR was determined in a two stage process: First, we denoised the training set using all $\lambda \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. We then fit a cubic spline through the PSNR values for these λ values and found the value $\hat{\lambda}$ that maximized the PSNR. In the second stage, the search was refined to $\lambda \in \hat{\lambda} + \{-0.06, -0.04, -0.02, 0, 0.02, 0.04, 0.06\}$. The PSNR values for all λ values were again fit with a cubic spline, and the value λ^* that maximized the PSNR across all 10 training images was chosen.

Results were obtained for two sets of *test* images. The first set consisted of images commonly used in denoising experiments (Lena, Boats, etc.; obtained from (Portilla, 2006a)). Table 1 provides PSNR and SSIM values for this set and various levels of additive Gaussian noise (cf. Portilla et al., 2003). Portilla et al. (2003) report some

Table 1: Peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) for images (from Portilla, 2006a) denoised with a FoE prior.

	PSNR in dB						SSIM (Wang et al., 2004)					
σ	Noisy	Lena	Barbara	Boats	House	Peppers	Noisy	Lena	Barbara	Boats	House	Peppers
1	48.13	47.84	47.86	47.69	48.32	47.81	0.993	0.991	0.994	0.994	0.993	0.993
2	42.11	42.92	42.92	42.28	44.01	42.96	0.974	0.974	0.983	0.978	0.982	0.981
5	34.15	38.12	37.19	36.27	38.23	37.63	0.867	0.936	0.957	0.915	0.930	0.950
10	28.13	35.04	32.83	33.05	35.06	34.28	0.661	0.898	0.918	0.860	0.880	0.923
15	24.61	33.27	30.22	31.22	33.48	32.03	0.509	0.876	0.884	0.825	0.866	0.901
20	22.11	31.92	28.32	29.85	32.17	30.58	0.405	0.854	0.841	0.788	0.850	0.879
25	20.17	30.82	27.04	28.72	31.11	29.20	0.332	0.834	0.805	0.754	0.836	0.853
50	14.15	26.49	23.15	24.53	26.74	24.52	0.159	0.741	0.622	0.614	0.763	0.735
75	10.63	24.13	21.36	22.48	24.13	21.68	0.096	0.678	0.536	0.537	0.692	0.648
100	8.13	21.87	19.77	20.80	21.66	19.60	0.066	0.615	0.471	0.473	0.622	0.568

of the most accurate results on these test images and their method is tuned to perform well on this dataset. We obtained signal-to-noise ratios that were close to their results (mostly within 0.5dB), and in some cases even surpassed their results (by about 0.3dB). Note that their wavelet model was actually trained on the noisy version of the image to be denoised. To the best of our knowledge, no other generic Markov random field approach has so far been able to closely compete with such wavelet-based methods on this dataset. Also note that the prior was not trained on, or tuned to these examples.

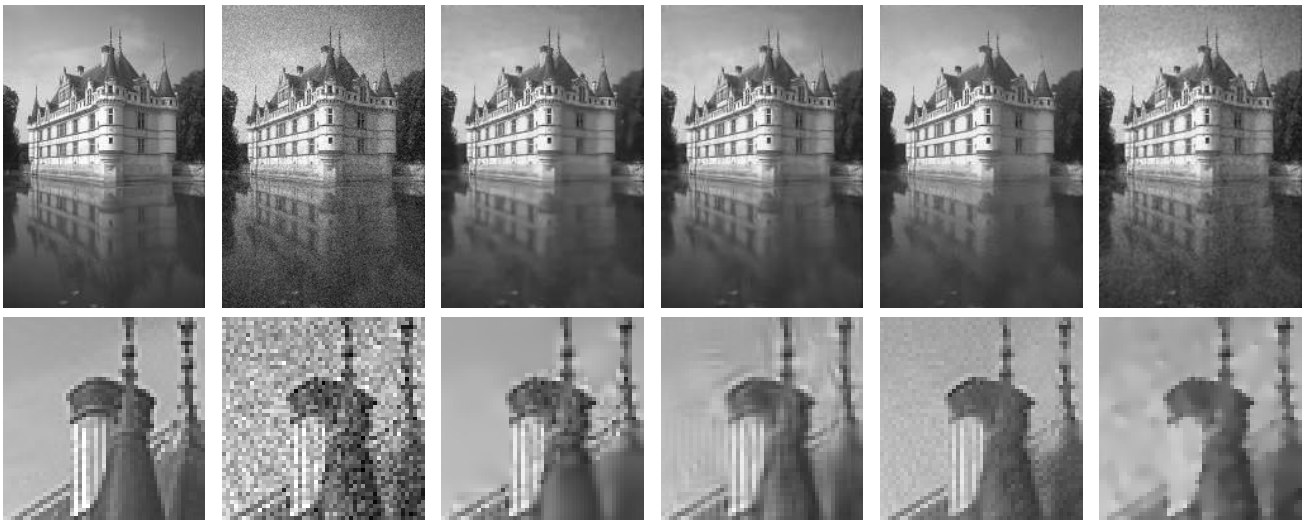
5.1 Denoising experiments

To test more varied and realistic images we denoised a second test set consisting of 68 images from the separate test section of the Berkeley segmentation dataset (Martin et al., 2001). Figure 9(a) shows example images from this test set. For various noise levels we denoised the images using the FoE model, the method of Portilla et al. (2003) (using the software and default settings provided by Portilla (2006b)), simple Wiener filtering (using MATLAB's `wiener2` with a 5×5 window), and a standard nonlinear diffusion scheme (Weickert, 1997) with a data term. For this last method, the diffusivity was modeled using a robust Huber function. This algorithm can be viewed as gradient ascent inference for an MRF model using only first derivative filters. For this standard nonlinear 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) giving the best case result. Note that in case of the FoE denoising was *not* stopped at the point of optimal PSNR, but rather automatically at convergence. Figure 7 shows the performance of these methods for one of the test images. Visually and quantitatively, the FoE model outperformed both Wiener filtering and nonlinear diffusion and nearly matched the performance of the specialized wavelet denoising technique. FoE denoising results for other images from this set are shown in Figure 8.

Figure 9 shows a performance comparison of the various denoising techniques over all 68 images from the test set at various noise levels. The FoE model consistently outperformed both Wiener filtering and standard nonlinear diffusion in terms of PSNR, while closely matching the performance of the current state of the art in image denoising (Portilla et al., 2003). A signed rank test showed that the performance differences between the FoE and the other methods were mostly statistically significant at a 95% confidence level (indicated by an asterisk on the respective bar). In terms of SSIM, the relative performance was very similar to that measured using the PSNR, with two notable exceptions: (1) When looking at the SSIM, the FoE performed slightly worse than nonlinear diffusion for two of the four noise levels, but the performance difference was not statistically significant in these cases. In the two cases where the FoE outperformed standard diffusion, the difference was significant, on the other hand. We should also keep in mind that nonlinear diffusion was helped substantially by the fact that it was stopped at the optimal PSNR, which is not possible in real applications. (2) On one of the noise levels, the FoE performed on par with the method of Portilla et al. (2003) (i. e., there was no significant performance difference). Overall, this means that in the majority of the cases the FoE performed significantly better than Wiener filtering and nonlinear diffusion, but also that the Wavelet method was still significantly better than the FoE (at a 95% confidence level). In making this comparison, it is important to keep in mind that Fields of Experts are generic image models with wide range of applications well beyond just image denoising.

5.2 Inpainting experiments

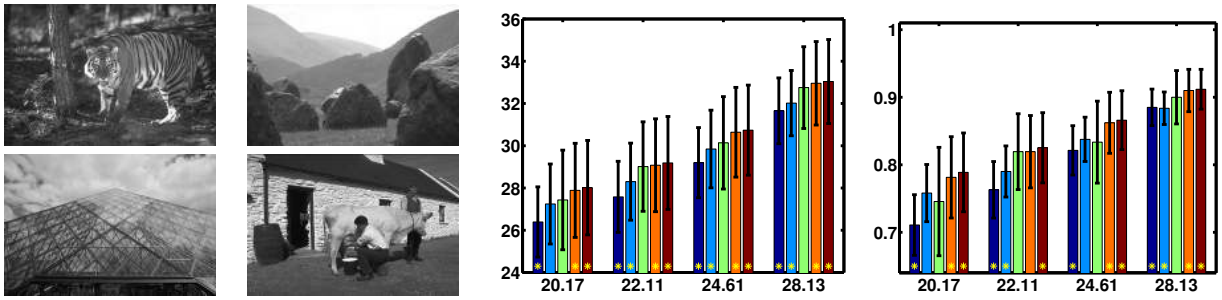
Figure 10 shows the result of applying our inpainting scheme in a text removal application in which the mask corresponds to all the pixels that were occluded by the text. The color image was converted to the YCbCr color model, and the algorithm was independently applied to all 3 channels. Since the prior was trained only on gray



(a) (b) (c) (d) (e) (f)

Figure 7: Denoising with a Field of Experts: Full image (top) and detail (bottom). (a) Original noiseless image. (b) Image with additive Gaussian noise ($\sigma = 25$); PSNR = 20.29dB. (c) Denoised image using a Field of Experts; PSNR = 28.72dB. (d) Denoised image using the approach of Portilla et al. (2003); PSNR = 28.90dB. (e) Denoised image using non-local means (Buades et al., 2004); PSNR = 28.21dB. (f) Denoised image using standard non-linear diffusion; PSNR = 27.18dB.

Figure 8: Other Field of Experts denoising results. Noisy input (left) and denoised image (right).



(a) Subset of the images used to evaluate the FoE model.

(b) PSNR in dB

(c) SSIM (Wang et al., 2004)

Figure 9: Denoising results on Berkeley database. Example images and denoising results for the following models (from left to right): Wiener filter, standard nonlinear diffusion, FoE model, and the two variants from (Portilla, 2006b). The horizontal axes denote the amount of noise added to the images (PSNR in dB). The error bars correspond to one standard deviation. The yellow asterisks denote cases where the performance differs significantly from that of the FoE model.

(a) (b) (c)

Figure 10: **Inpainting with a Field of Experts.** (a) Original image with overlaid text. (b) Inpainting result from diffusion algorithm using the FoE prior. (c) Close-up comparison between a (left), b (middle), and the results of Bertalmío et al. (2000) (right).

scale images, this is obviously suboptimal, but nevertheless gives good results. In order to speed up convergence we ran 5000 iterations of Eq. (21) with $\eta = 10$. Since such a large step size may lead to some numerical instabilities, we followed this with 250 more iterations with $\eta = 0.01$.

The inpainted result is very similar to the original and qualitatively superior to that by Bertalmío et al. (2000). Quantitatively, our method improved the PSNR by about 1.5dB (29.06dB compared to 27.56dB); the SSIM showed a sizable improvement as well (0.9371 compared to 0.9167; where higher is better). Note that to facilitate quantitative comparison with the results of Bertalmío et al. (2000), we measured these results using a GIF version of the input image that was used there[3]. To get a better idea of the performance of the FoE on high-quality input, we also measured results on a JPEG version of the same image. The PSNR was 32.22dB in that case and the SSIM was 0.9736. The advantage of the FoE prior can be seen in the continuity of edges which is better preserved compared with (Bertalmío et al., 2000). Figure 10(c) also shows a few detail regions comparing our method (center) with (Bertalmío et al., 2000) (right). We can see, for example, that the axle and the wheels of the carriage have been restored very well. Similar qualitative differences can be seen in many parts of the restored image.

Figure 11 shows various image inpainting results for test images that were corrupted using synthetic masks. An application of this inpainting algorithm to a problem of scratch removal in a photograph is shown in Figure 1. Furthermore, Gisy (2005) conducted a detailed study of Fields of Experts in conjunction with image inpainting. The reader is referred to his work for more detailed inpainting experiments.

5.3 Quantitative evaluation of FoE parameters

To evaluate the influence of the various parameters and design decisions on the quality of the learned FoE models, we performed a series of experiments. As an example, we varied the size or the number of the filters.

[3] Personal communication with Marcelo Bertalmío.

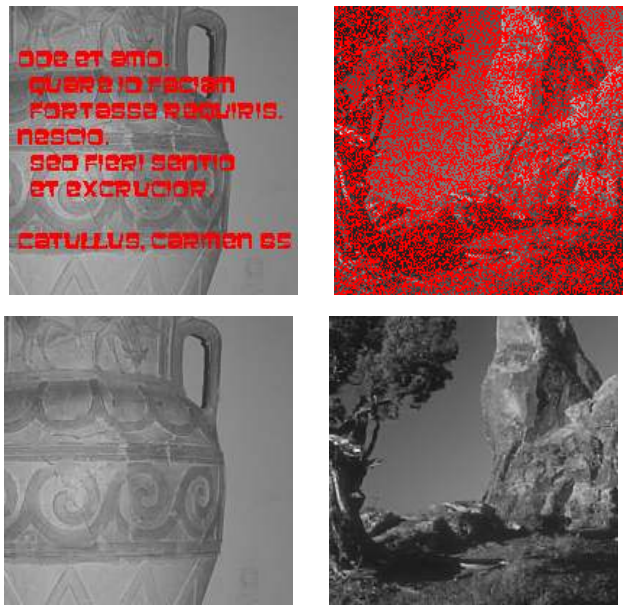


Figure 11: **Other image inpainting results.** The top row show the masked images; the red areas are filled in by the algorithm. The bottom row show the corresponding restored images that were obtained using a 5×5 FoE model with 24 filters.

Unfortunately, we cannot directly compare the goodness of various models by considering their likelihood, because it is intractable to compute the partition function for FoE models (note, however, that it may be possible to use likelihood bounds such as the ones derived by Weiss and Freeman (2007)). Instead, we evaluated FoE models in the context of image restoration applications, mostly using image denoising with the same basic setup as in Section 5.1. Note also that the performance numbers presented here are not fully indicative of the quality of the FoE model itself, but instead describe the performance of the model in the context of a particular application *and* a particular approximate inference scheme.

The general setup of the experiments was the following: The models were trained on 20000 image patches of 15×15 pixels as described in Section 3.4, except where indicated otherwise. All models suppressed the mean intensity either through choosing an appropriate basis for the filters, or by subtracting the mean from the data and the filters. Except for explicit special cases, the models were initialized with $\alpha_i = 0.01$, and a random set of filters drawn i. i. d. from a unit covariance Gaussian (possibly in a transformed space, as indicated alongside the experiments). If not indicated otherwise, we ran contrastive divergence with one step, where each step was performed using hybrid Monte-Carlo sampling with 30 leaps tuned so that the acceptance rate was around 90%. We always performed 5000 iterations of contrastive divergence with a learning rate of 0.01. As a baseline, we used models with 3×3 cliques and 8 filters, since those were faster to train and also led to faster inference due to faster convolution operations. In some of the cases, we also considered 5×5 cliques with 24 filters.

Once the models were trained, we determined the appropriate λ trade-off parameter for denoising that weighs the FoE prior against the Gaussian image likelihood. We used the same procedure as described in Section 5.1.

Using the estimated weight λ^*, every model was evaluated on 68 images from the test portion of the Berkeley segmentation database (Martin et al., 2001) (this is the same set as was used in Section 5.1). We added i. i. d. Gaussian noise with $\sigma = 20$ to every image, and subsequently denoised the images with the conjugate gradient method described above.

To analyze the FoE model, we evaluated the effects of the following aspects on performance in the respective section:

- 5.3.1: Choice of the filter basis $\mathbf{A}$.
- 5.3.2: Size and shape of the filters.
- 5.3.3: Choice of the number of filters.
- 5.3.4: Using fixed, random filters as opposed to learning them.
- 5.3.5: Using fixed filters from patch-based models, instead of learning them.
- 5.3.6: Choice of the expert function.

We measured the performance using the described denoising task and give both PSNR and SSIM results averaged over all 68 test images. To reduce the influence of the image border on the measurements, we ignore 10 pixels around the border when computing the PSNR and SSIM.

5.3.1 Learning the filters in transformed spaces

In this first set of experiments, we evaluated how the choice of the basis $\mathbf{A}$, in which the filters $\mathbf{J}_i$ are defined, affects the performance. As we discussed in Section 3.4, defining the filters in other bases does not change the actual learning objective. But since contrastive divergence learning entails a local gradient ascent procedure, it is susceptible to local optima, and the choice of the filter basis may thus prove important for convergence to a good local optimum. We used three different bases here: (1) A basis that defines the filters in their original space; that is $\mathbf{A}_O = \mathbf{I}$. This means that every filter coefficient in this space directly corresponds to a clique pixel. (2) A basis based on whitening the filter space defined as $\mathbf{A}_W = \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}^{\mathrm{T}}$, where $\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}}$ is an eigendecomposition of the covariance matrix $\mathbf{\Sigma}$ of natural image patches that have the same size as the filters. If we

chose the matrix of all transformed filters $\tilde{\mathbf{J}}$ as the identity matrix, then the filters $\mathbf{J} = \mathbf{A}_W^{\mathrm{T}}\tilde{\mathbf{J}}$ in the original space are just the principal components scaled according to their standard deviation. This means that the low-frequency principal components have a larger norm than the high-frequency ones, which makes it easier to find low-frequency filters. (3) A basis based on an "inverse" whitening, defined as $\mathbf{A}_I = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^{\mathrm{T}}$. If we chose the transformed filters $\tilde{\mathbf{J}}$ as the identity matrix in this case, then the filters $\mathbf{J} = \mathbf{A}_I^{\mathrm{T}}\tilde{\mathbf{J}}$ in the original space are also the principal components, but are now scaled according to their inverse standard deviation. In this case the high-frequency principal components have a larger norm than the low-frequency components, which makes it easier to find high-frequency filters.

When training with these three different bases we found filters with high-frequency, and seemingly non-regular structures in all three cases. Figure 6 shows the filters obtained when training with the "inverse" whitening basis $\mathbf{A}_I$, which compared to the filters obtained with the other bases (see (Roth, 2007) for details) are the most localized. While the filters exhibit some qualitative differences depending on the choice of basis, there are even stronger quantitative differences. As Table 2 shows, the denoising performance deteriorated when using the whitened basis as opposed to training in the original space. Using "inverse" whitening, on the other hand, led to quantitatively superior results. These findings were consistent for models with 3×3 cliques and 5×5 cliques. As we have seen, whitening the space in which training is performed can impact the nature of the recovered filters and one must be careful in evaluating the resulting filters with respect to any such processing (cf. Hinton and Teh, 2001). Furthermore, we found that updating the logarithm of the expert parameters α_i to enforce their positivity led to better results in almost all of the cases compared to updating the α_i directly, sometimes to significantly better results.

These quantitative findings strongly suggest that high-frequency filters are important to achieving good performance with FoE models in an image denoising application. As we will see below, experiments with various random filters led to results that are fully consistent with this observation. Since the "inverse" whitening basis encourages high-frequency filters and consistently led to the best quantitative results, we used it as baseline for most of our experiments, unless indicated otherwise. Furthermore, since updating the log of the expert parameters led to better results, we also adopted this as part of the baseline for the remaining experiments.

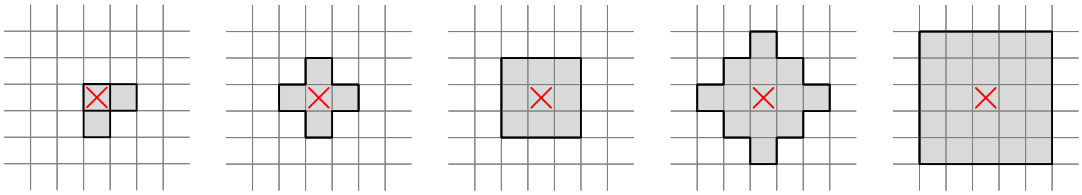
5.3.2 Varying the clique size and shape

In the second set of experiments, we evaluated how the size and the shape of the maximal cliques influenced the performance of the Field-of-Experts model. Figure 12 shows some of the clique shapes and sizes that were evaluated. The simplest conceivable model based on the FoE framework is the regular pairwise Markov random field shown in Figure 12(a), where each node is connected to its top, bottom, left, and right neighbors. Here, there are two types of maximal cliques: pairs of nodes connected by either horizontal or vertical edges. These can be modeled in the FoE framework by restricting 2×2 filters to pairs of horizontal or vertical pixels as depicted in the figure. In Figure 12(b), we see a more complicated non-square clique structure, where the 4-neighborhood around a central pixel (marked red) is fully connected. This clique shape was achieved by forcing the filter coefficients of 3×3 filters to be zero outside of the diamond shape. In Figure 12(c), we see a simple, square 3×3 clique, where a pixel and its 8 neighbors are all fully connected. We can also also have larger diamond-shaped cliques as shown in Figure 12(d), where the filter coefficients of 5×5 filters were forced to be zero outside of the diamond. Finally, in Figure 12(e) we can see square 5×5 cliques that were obtained once again by fully connecting all nodes inside the square. Beyond what is shown here, we also evaluated the performance of models with 7×7 filters, both in the diamond-shaped and in the square case. In each case we used the general experimental setup as outlined above, in particular we used "inverse" whitening for defining the filter basis.

In each of the cases, we evaluated the performance using two experiments: First, we trained and tested models with a fixed number of 8 filters. Then we trained and tested models with $p - 1$ filters, where p is the number of nodes in the maximal cliques. Since, as for all the experiments in this section, we ignored the mean gray value component of each clique-sized patch, this means that there were as many experts per clique as there were

Table 2: Denoising performance of the FoE model when trained with different filter bases. The filters are trained either in original, whitened, or "inverse" whitened coordinates (see text). In the indicated cases the log of the expert parameters α_i was updated, otherwise the α_i were updated directly.

Model	3×3, 8 filters						5×5, 24 filters					
Filter basis	whitened, $\mathbf{A}_W$		original, $\mathbf{A}_O$		"inverse" whitened, $\mathbf{A}_I$		whitened, $\mathbf{A}_W$		original, $\mathbf{A}_O$		"inverse" whitened, $\mathbf{A}_I$	
Update α	direct	log	direct	log	direct	log	direct	log	direct	log	direct	log
PSNR in dB	27.24	27.34	28.09	28.06	28.40	28.79	27.12	25.76	27.90	28.31	28.37	29.07
SSIM	0.757	0.759	0.784	0.778	0.794	0.813	0.773	0.665	0.782	0.792	0.800	0.819

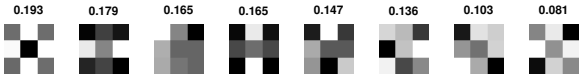


(a) (b) (c) (d) (e)

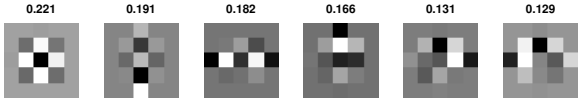
Figure 12: FoE clique structure for various clique shapes and sizes (illustrated by solid black outlines). (a) 2×1 cliques of pairwise MRF. (b) Diamond-shaped 3×3 clique or fully connected 4-neighborhood. (c) Square 3×3 clique or fully connected 8-neighborhood. (d) Diamond-shaped 5×5 clique. (e) Square 5×5 clique.

Table 3: Denoising performance of FoE models with various clique sizes and shapes.

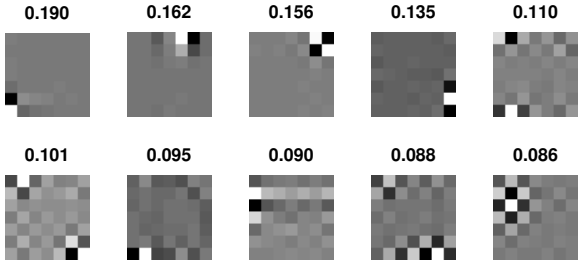
Size	2×1	3×3				5×5				7×7			
Shape	pairwise	diamond		square		diamond		square		diamond		square	
# of filters	1	4	8	4	8	8	12	8	24	8	24	8	48
PSNR in dB	26.58	27.81	27.90	28.63	28.79	28.81	28.88	28.80	29.07	28.78	28.98	28.74	29.04
SSIM	0.718	0.766	0.769	0.805	0.813	0.815	0.816	0.811	0.819	0.811	0.817	0.812	0.818



(a) Square 3×3 cliques with 8 filters.



(b) Diamond-shaped 5×5 cliques with 12 filters (first 6 shown).



(c) Square 7×7 cliques with 48 filters (first 10 shown).

Figure 13: Learned models with various clique sizes and shapes. The number above each filter denotes the corresponding expert parameter α_i.

degrees of freedom. Figure 13 shows some of the learned models (see also Figure 6 for a 5×5 model with square cliques). Table 3 gives the performance measurements from these experiments.

We can see that the pairwise model performed substantially worse than the high-order models with square cliques. This again showed that FoEs with large cliques are able to capture structure in natural images that cannot be captured using pairwise MRF models alone. In the 3×3 case, square cliques substantially outperformed diamond-shaped ones. For 5×5 and 7×7 FoEs, diamond-shaped cliques performed on par with square ones with few filters, but performed worse than square ones with many filters. Models with square 3×3 cliques and 8 filters already performed quite well, but a 5×5 model with 24 filters nevertheless outperformed the simpler model by a considerable margin. A 7×7 FoE with 48 filters was able to match the performance of the 5×5 model with 24 filters, but did not exceed it. Interestingly, the performance of models with larger cliques (5×5 and 7×7) was best when many filters were used; with only 8 filters the 3×3 FoE performed on par. We conclude that while cliques larger than 3×3 improved performance and captured more structure of natural images, models with square 3×3 cliques already captured a large amount of the variation in natural images, at least of the variation that can be captured with linear projections and Student t-experts. It is conceivable that this could be improved upon with other experts, or with experts that model nonlinear features of the clique pixels, but explorations of this are left for future work.

5.3.3 Varying the number of filters

The next set of experiments determined the impact of the number of filters on the denoising performance. To that end we trained and tested 3×3 models with 1 through 18 filters otherwise using the baseline setup. Figure 14 graphs the denoising performance as a function of the number of experts, measured both in terms of PSNR and SSIM and, as usual, averaged over all 68 images from the test set. We can see that about four to five filters were necessary to make the model perform well, and that the performance improvements became quite minor beyond eight filters. Nonetheless, there were small improvements in test set performance even for large numbers of filters, which is an important indication that we were not overfitting the training data.

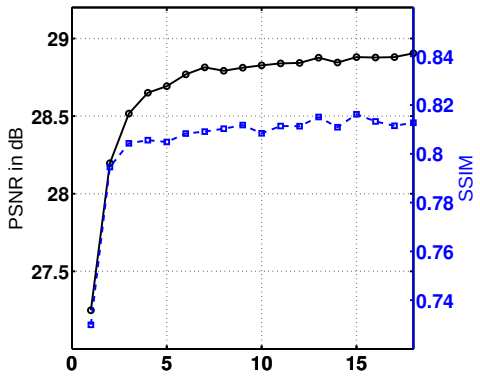


Figure 14: Denoising performance of 3×3 models with a varying number of filters shown in terms of PSNR (solid black, circular markers) and SSIM (dashed blue, square markers).

5.3.4 Using fixed, random filters

One might observe that the learned FoE filters look somewhat "random". In order to determine the effect of learning the filters, we performed two kinds of experiments. Here, we compared with fixed, random filters and only learned the parameters α_i and the norm of the filters (while keeping their direction fixed). In the following section, we will compare with using fixed filters that have been determined ahead of time using another method. For the random filter experiments we worked with three basic setups corresponding to different filter bases: (1) We first drew filters randomly in the original space (corresponding to $\mathbf{A}_O$ from above) by drawing all coefficients $J_{i,j}$ of the matrix $\mathbf{J}$ of all filters i. i. d. from a unit normal (i. e., $J_{i,j} \sim \mathcal{N}(0,1)$). (2) In the second case we drew filters randomly in whitened space so that $\mathbf{J} = \mathbf{A}_W^{\mathrm{T}} \tilde{\mathbf{J}}$, where $\tilde{J}_{i,j} \sim \mathcal{N}(0,1)$. Typical random filters obtained in this way look quite smooth, because the low-frequency principal components are scaled up by their (large) standard deviation. (3) In the final case we drew filters randomly using "inverse" whitening. Here, $\mathbf{J} = \mathbf{A}_I^{\mathrm{T}} \tilde{\mathbf{J}}$, where the coefficients of the transformed filter matrix $\tilde{\mathbf{J}}$ were again drawn from a unit normal. In this case, the typical filter samples were dominated by high-frequency structures, because the high-frequency principal components are scaled up in this case due to their small standard deviation. In all three cases we tested both 3×3 models with 8 filters and 5×5 models with 24 filters. As a reminder, training here means learning the parameters α_i and the norms of the filters. The norms of the filters were trained by only considering the component of the filter gradient that coincides with the fixed filter direction. This projects the filter gradient onto the manifold of matrices with fixed direction column vectors (i. e., filters), but variable filter length. Note that since we did not learn the filter direction in this experiment, we did not use any form of filter basis *during* training. Beyond what is reported here, we also investigated normalizing the random filter vectors prior to learning. Since we did not observe important performance changes, we do not report those results here (see (Roth, 2007) for results). Figure 15 shows portions of the 5×5 models for each of the 3 different filter bases.

The quantitative results shown in Table 4 once again underline the importance of high-frequency filters for achieving good denoising performance with Fields of Experts. Filters drawn in the whitened space performed poorly, particularly in case of the 5×5 model. Filters drawn in the original space performed better, but still not nearly as well as filters drawn in the "inverse" whitened space, which emphasizes high-frequencies with a certain structure. It is also interesting to note that this space was the only one where the 5×5 model outperformed the corresponding 3×3 model with random filters. When we compare the results to those in Table 3, we see that even with "inverse" whitening the performance with random filters was about 1dB short of that with learned filters. While the model generally performed well with random filters (at least with "inverse" whitening), learning the filters substantially improved performance.

5.3.5 Using fixed filters from patch-based models

The next set of experiments was similar to the previous one in that we used a fixed set of filters, and only trained the parameters α_i as well as the norm of the filters. Instead of using randomly drawn filters, we here used filters obtained by various patch-based learning methods: (1) First, we used filters obtained by principal component analysis of image patches with the same size as the cliques. (2) We then used filters obtained by independent component analysis of image patches. In particular, we used the software of Gävert et al. (2005) using the standard settings and extracted 8 independent components for 3×3 patches and 24 for 5×5 patches. (3)

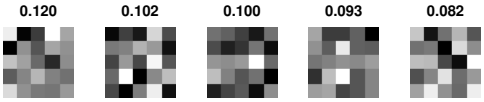
Table 4: Denoising performance of the FoE model with random filters. The filters are drawn either in original, whitened, or "inverse" whitened coordinates (see text).

Model	3×3, 8 filters			5×5, 24 filters		
Space	whitened	original	"inverse" whitened	whitened	original	"inverse" whitened
PSNR in dB	26.61	27.28	27.80	25.70	26.57	27.99
SSIM	0.746	0.761	0.779	0.694	0.748	0.783

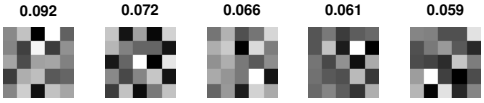
Table 5: Denoising performance of the FoE model with filters determined by PCA, ICA, and PoT.

Model	3×3, 8 filters			5×5, 24 filters		
Filters	PCA	ICA	PoT	PCA	ICA	PoT
PSNR in dB	27.86	28.02	28.12	28.08	28.37	28.51
SSIM	0.782	0.781	0.783	0.782	0.790	0.791

(a) Model with random filters in whitened space.

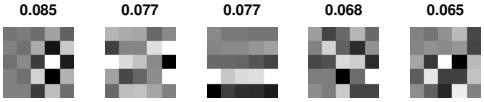


(b) Model with random filters in original space.

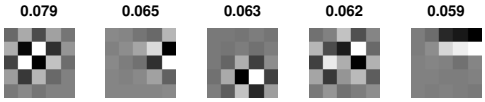


(c) Model with random filters in "inverse whitened" space.

(d) Model with PCA filters.



(e) Model with ICA filters.



(f) Model with PoT filters.

Figure 15: FoE models based on various kinds of fixed filters. Each model has 5×5 cliques and 24 filters. The number above each filter denotes the corresponding expert parameter α_i. The filters are sorted according to these expert parameters in descending order, and only every fifth filter is shown.

Finally, we used filters obtained by training a Product-of-Experts model with Student t-experts as described by Teh et al. (2003). Figure 15 shows parts of the learned 5×5 models; 24 filters were used in all three cases.

Table 5 shows the results from using these fixed filters. We found that PCA filters worked relatively poorly, which was not a surprise given that random filters drawn in the same space also did not perform very well. Nonetheless, the performance was better than with random filters drawn in the same space, which may be attributable to the fact that principal components contain some high-frequency filters. Filters from independent component analysis worked slightly better, particularly in the 5×5 case. Finally, filters obtained from a PoT model worked best in this comparison, but still performed between 0.4 and 0.6dB worse than fully learned filters. This further suggests the importance of learning the filters in conjunction with a high-order MRF model such as the FoE. It is furthermore very revealing to closely examine the learned models shown in Figure 15. As we can see, high-frequency filters had the highest weight (expert parameter) in all three cases, while smooth looking filters consistently received smaller weights. In particular, it is interesting how in the case of the PCA filters the sorting based on decreasing weight is almost exactly the reverse of a singular value-based sorting. This says that minor components were assigned more weight, and were thus more important than major components, which is consistent with the theoretical findings by Weiss and Freeman (2007). Both findings suggest that smooth derivative (e. g., Gabor) filters are not the best choice in conjunction with such a framework and that the kinds of filters found by the full learning algorithm are important for getting good performance.

5.3.6 Charbonnier expert

One interesting question is whether the quality of the model and the properties of the filters are affected by the choice of the Student-t expert. While a complete treatment of this question is warranted, here we analyzed the Charbonnier expert as introduced in Section 3.1. This is essentially a differentiable version of an exponential distribution (or an L1-norm, when viewed as energy). One advantage of the Charbonnier expert is that its energy is convex, which makes optimization in the context of denoising much easier. The Gaussian likelihood model we are using here is convex as well (more precisely the associated energy), which makes the posterior energy convex. This means that we can actually find global optima during denoising using the conjugate gradient algorithm. We trained both 3×3 models with 8 filters as well as 5×5 with 24 filters. Each kind of model was trained with the filters defined in the original space as well as the filters defined in the "inverse" whitened space. Except for the different expert function, training was largely identical to the Student-t case. Figure 16 shows the filters obtained by training a 5×5 model in the original filter space (note that inverse whitening led

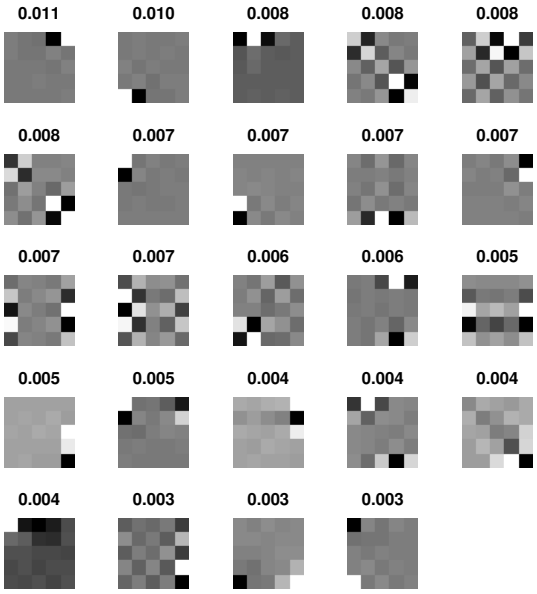


Figure 16: Filters learned for a 5×5 FoE model with Charbonnier experts. The number above each filter denotes the corresponding expert parameter α_i.

Table 6: Denoising performance of the FoE model with Charbonnier experts.

Model	3×3, 8 filters		5×5, 24 filters	
"Inverse" whitening	N	Y	N	Y
PSNR in dB	28.69	28.52	28.74	28.47
SSIM	0.806	0.792	0.808	0.790

to inferior performance in conjunction with Charbonnier experts). Similar to the Student-t case, we found high-frequency filters.

Table 6 shows the denoising results obtained with these Charbonnier experts. We can see that the model performed best when trained in the original space, and that the performance with 3×3 and 5×5 cliques was virtually identical. In either case, the performance was worse than with Student t-experts, in the case of 5×5 filters somewhat substantially (0.35 dB). This result reinforces the observation that non-convex regularization is important for achieving good performance in low-level vision applications (e. g., Black et al., 1998). While this obviously makes optimization more difficult, we observed better performance using non-convex models even though they rely on simple local optimization methods. It is interesting to note, however, that denoising using the Charbonnier expert was considerably faster, as many fewer conjugate gradient iterations were needed in practice (about 300 as opposed to 3000 or more iterations). In applications where computational performance is crucial, it may thus be interesting to use FoE models with convex experts (when they are viewed as energies).

5.3.7 Other experiments

We have conducted a range of other experiments to investigate the properties of Fields of Experts. For example, we tested how the denoising performance depends on the initialization of the model and found only relatively minor dependencies. While the learned filters generally differ between different initializations, they all share the same kind of high-frequency structures. Furthermore, we also analyzed whether it might be advantageous to model the image intensity in a different domain. As is the default with most digital images, the images from the database used here are gamma-compressed, but we also evaluated modeling linear or logarithmic images. However, neither affected performance favorably. More detailed results, including a range of other experiments, can be found in (Roth, 2007).

6 Discussion

Despite significantly increasing the modeling power compared to pairwise MRFs, Fields of Experts naturally have several limitations that should be addressed in the future. One such limitation is that the presented framework models images only at their original spatial scale (resolution), and is not able to model the scale invariance property of natural images. In particular, its 5×5 filters are too small to capture statistics at very coarse spatial scales, but computational considerations prevent us from making them much bigger. Our results with 7×7 filters furthermore indicate that simply making the filters larger will not necessarily help. The FRAME model (Zhu and Mumford, 1997), on the other hand, uses derivative filters of various spatial scales to capture marginal statistics across scales. Wavelet-based approaches (e. g., Portilla et al., 2003) also make use of multiple spatial scales and moreover model scale dependencies, which may explain their better denoising performance. While with the FoE we have observed improved denoising performance compared to standard MRF approaches, there is still more to be done with regards to modeling natural image statistics. As noted by Roth (2007), images sampled from the learned model do not look "natural", and moreover the marginal distributions of the learned model (obtained from sampled images) are not a good match to natural image marginals. We hypothesize that multi-scale (or longer-range) representations and better learning algorithms will be necessary to capture more properties of natural scenes. Considering the denoising and inpainting results from Section 5, we find that the FoE has a tendency to make relatively smooth regions even smoother; on the other hand, noise in highly textured areas is not fully removed. This should be further investigated. Another limitation is that we have only considered relatively sim-

ple parametric expert functions so far. More flexible Gaussian scale mixtures have also been used as experts (Weiss and Freeman, 2007; Roth and Black, 2007a), but non-parametric experts such as in the FRAME model might be considered as well.

Another issue that has not been addressed yet is that the clique sizes and shapes have been chosen a priori here, but may instead be selected automatically. In our experiments, we have furthermore assumed that the model is homogeneous. Certain applications may benefit from spatially varying statistics, which motivates further research on inhomogeneous models; this will likely require significantly more training data. In other work (Roth and Black, 2007a), we have shown that Markov random fields can significantly benefit from steering the filters to the local predominant image orientation. So far, this has only been done for MRFs based on simple derivative filters; learning steerable filters in an FoE-like framework seems like a promising avenue for future work. The presented FoE framework has solely focused on modeling the prior distribution of natural images or other low-level vision representations, and has been combined with simple hand-designed likelihood models. For certain applications it would be beneficial to learn these likelihood models as well, or to learn a model of the application-specific posterior directly.

An important issue that arises with the use of high-order MRF models is the increased difficulty of inference with such models. While, for the applications presented here, gradient-based optimization methods proved viable, they are less appropriate in other domains, such as stereo, where inference techniques such as graph cuts or belief propagation have been shown to be superior, at least in the context of pairwise MRF models of disparity. The increased clique size makes it difficult to apply these methods here, and so far only small cliques (Potetz, 2007) or restricted kinds of models (Kohli et al., 2007) can be handled. Further investigation of advanced inference techniques for high-order MRF models such as FoEs is needed.

7 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 expressive image priors we formulated a high-order Markov random field model based on extended cliques that capture local image statistics beyond simple pairwise neighborhoods. We modeled the potentials for these extended cliques based on the Product-of-Experts paradigm. The resulting Field of Experts 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, yet achieves performance close to the best special-purpose wavelet-based denoising algorithms. The advantage over most wavelet-based methods lies in the generality of the prior and its applicability across different vision problems.

There are many avenues for future work beyond further extending the capabilities of the model as discussed above. By making MRF models more powerful, many problems can be revisited with an expectation of improved results. We have already shown that optical flow estimation can benefit from FoE models (Roth and Black, 2007b), and expect that applications such as dense stereo estimation, object boundary detection and others will benefit as well. The methods may also be extended to non-image-based graphs such as surface meshes or MRF models of object parts.

Acknowledgements

We thank Stuart Andrews, Pietro Berkes, Alessandro Duci, Yoram Gat, Stuart Geman, Horst Haussecker, Thomas Hofmann, John Hughes, Dan Huttenlocher, Xiangyang Lan, Siwei Lyu, Oscar Nestares, Hanno Scharr, Eero Simoncelli, Yair Weiss, Max Welling, and Frank Wood for helpful discussions; Guillermo Sapiro and Marcelo Bertalmío for making their inpainting examples available for comparison; and Javier Portilla for making his denoising software available. This work was supported by Intel Research, NSF ITR grant 0113679, NSF IIS-0534858, NSF #0535075 and by 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

A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.*, 7(6):1129–1159, Nov. 1995.

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

J. Besag. On the statistical analysis of dirty pictures. *J. Roy. Stat. Soc. B*, 48(3):259–302, 1986.

M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger. Robust anisotropic diffusion. *IEEE Trans. Image Process.*, 7(3):421–432, Mar. 1998.

A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, 1987.

L. Bottou. Stochastic learning. In O. Bousquet and U. von Luxburg, eds., *Advanced Lectures on Machine Learning*,

number 3176 in Lecture Notes in Artificial Intelligence, pp. 146–168. Springer, Berlin, 2004.

A. Buades, B. Coll, and J.-M. Morel. A review of image denoising algorithms, with a new one. *SIAM Multiscale Modeling and Simulation*, 4(2):490–530, 2004.

P. Charbonnier, L. Blanc-Feéraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. Image Process.*, 6(2):298–311, Feb. 1997.

A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.*, 13(9):1200–1212, Sept. 2004.

J. N. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *The Annals of Mathematical Statistics*, 43(5):1470–1480, Oct. 1972.

S. della Pietra, V. della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(4):380–393, Apr. 1997.

X. Descombes, R. D. Morris, J. Zerubia, and M. Berthod. Estimation of Markov random field prior parameters using Markov chain Monte Carlo maximum likelihood. *IEEE Trans. Image Process.*, 8(7):954–963, July 1999.

D. L. Donoho, M. Elad, and V. N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Trans. Info. Theory*, 52(1):6–18, Jan. 2006.

A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *IEEE Int. Conf. on Comp. Vis. (ICCV)*, vol. 2, pp. 1033–1038, Sept. 1999.

M. Elad and M. Aharon. Image denoising via learned dictionaries and sparse representations. In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, vol. 1, pp. 895–900, June 2006.

M. Elad, P. Milanfar, and R. Rubinstein. Analysis versus synthesis in signal priors. In *Proc. of EUSIPCO*, Florence, Italy, Sept. 2006.

P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, vol. 1, pp. 261–268, June 2004.

A. Fitzgibbon, Y. Wexler, and A. Zisserman. Image-based rendering using image-based priors. In *IEEE Int. Conf. on Comp. Vis. (ICCV)*, vol. 2, pp. 1176–1183, Oct. 2003.

W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *Int. J. Comput. Vision*, 40(1):24–47, Oct. 2000.

H. Gävert, J. Hurri, J. Särelä, and A. Hyvärinen. FastICA software for MATLAB. http://www.cis.hut.fi/projects/ica/fastica/, Oct. 2005. Software version 2.5.

P. Gehler and M. Welling. Products of "edge-perts". In *Adv. in Neur. Inf. Proc. Sys. (NIPS)*, vol. 18, pp. 419–426, 2006.

D. Geman and G. Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):367–383, Mar. 1992.

S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6:721–741, Nov. 1984.

S. Geman, D. E. McClure, and D. Geman. A nonlinear filter for film restoration and other problems in image processing. *CVGIP: Graphical Models and Image Processing*, 54 (2):281–289, July 1992.

C. J. Geyer. Markov chain Monte Carlo maximum likelihood. In *Proceedings of the 23rd Symposium on the Interface*, Computing Science and Statistics, pp. 156–163, Seattle, Washington, Apr. 1991.

G. Gilboa, N. Sochen, and Y. Y. Zeevi. Image enhancement and denoising by complex diffusion processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(8):1020–1036, Aug. 2004.

G. L. Gimel'farb. Texture modeling by multiple pairwise pixel interactions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(11):1110–1114, Nov. 1996.

T. Gisy. Image inpainting based on natural image statistics. Diplom thesis, Eidgenössische Technische Hochschule, Zürich, Switzerland, Sept. 2005.

W. Hashimoto and K. Kurata. Properties of basis functions generated by shift invariant sparse representations of natural images. *Biological Cybernetics*, 83(2):111–118, July 2000.

F. Heitz and P. Bouthemy. Multimodal estimation of discontinuous optical flow using Markov random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(12):1217–1232, Dec. 1993.

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

G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Comput.*, 14(8):1771–1800, Aug. 2002.

G. E. Hinton and Y.-W. Teh. Discovering multiple constraints that are frequently approximately satisfied. In *Conf. on Uncert. in Art. Intel. (UAI)*, pp. 227–234, Aug. 2001.

T. Hofmann, J. Puzicha, and J. M. Buhmann. Unsupervised texture segmentation in a deterministic annealing framework. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(8): 803–818, Aug. 1998.

J. Huang and D. Mumford. Statistics of natural images and models. In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, vol. 1, pp. 1541–1547, June 1999.

A. Hyvaärinen. Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6:695–708, Apr. 2005.

M. I. Jordan, Z. Ghahramani, T. S. Jaakola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov. 1999.

R. L. Kashyap and R. Chellappa. Filtering of noisy images using Markov random field models. In *Proceedings of the Nineteenth Allerton Conference on Communication Control and Computing*, pp. 850–859, Urbana, Illinois, Oct. 1981.

C. Kervrann and J. Boulanger. Unsupervised patch-based image regularization and representation. In A. Leonardis, H. Bischof, and A. Prinz, eds., *Eur. Conf. on Comp. Vis. (ECCV)*, vol. 3954 of *Lect. Notes in Comp. Sci.*, pp. 555–567. Springer, 2006.

P. Kohli, M. P. Kumar, and P. H. S. Torr. $\mathcal{P}^3$ & beyond: Solving energies with higher order cliques. In *IEEE Conf.*

on Comp. Vis. and Pat. Recog. (CVPR), June 2007.

V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(2):147–159, Feb. 2004.

S. Kumar and M. Hebert. Discriminative random fields. *Int. J. Comput. Vision*, 68(2):179–201, June 2006.

X. Lan, S. Roth, D. P. Huttenlocher, and M. J. Black. Efficient belief propagation with learned higher-order Markov random fields. In A. Leonardis, H. Bischof, and A. Prinz, eds., *Eur. Conf. on Comp. Vis. (ECCV)*, vol. 3952 of *Lect. Notes in Comp. Sci.*, pp. 269–282. Springer, 2006.

Y. LeCun and F. J. Huang. Loss functions for discriminative training of energy-based models. In R. G. Cowell and Z. Ghahramani, eds., *Int. Works. on Art. Int. and Stat. (AISTATS)*, pp. 206–213, Jan. 2005.

A. Levin, A. Zomet, and Y. Weiss. Learning how to inpaint from global image statistics. In *IEEE Int. Conf. on Comp. Vis. (ICCV)*, vol. 1, pp. 305–312, Oct. 2003.

S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer, 2nd edition, 2001.

S. Lyu and E. P. Simoncelli. Statistical modeling of images with fields of Gaussian scale mixtures. In *Adv. in Neur. Inf. Proc. Sys. (NIPS)*, vol. 19, pp. 945–952, 2007.

J. Marroquin, S. Mitter, and T. Poggio. Probabilistic solutions of ill-posed problems in computational vision. *J. Am. Stat. Assoc.*, 82(397):76–89, Mar. 1987.

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. In *IEEE Int. Conf. on Comp. Vis. (ICCV)*, vol. 2, pp. 416–423, July 2001.

J. J. McAuley, T. Caetano, A. J. Smola, and M. O. Franz. Learning high-order MRF priors of color images. In *Int. Conf. on Mach. Learn. (ICML)*, pp. 617–624, June 2006.

T. Minka. Divergence measures and message passing. Technical Report MSR-TR-2005-173, Microsoft Research, Cambridge, UK, 2005.

T. M. Moldovan, S. Roth, and M. J. Black. Denoising archival films using a learned Bayesian model. In *IEEE Int. Conf. on Image Proc. (ICIP)*, pp. 2641–2644, Oct. 2006.

J. Moussouris. Gibbs and Markov random systems with constraints. *J. Stat. Phys.*, 10(1):11–33, Jan. 1974.

R. M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto, Ontario, Canada, Sept. 1993.

R. Neher and A. Srivastava. A Bayesian MRF framework for labeling using hyperspectral imaging. *IEEE Transactions on Geoscience and Remote Sensing*, 43(6):1363–1374, June 2005.

F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano. Toward automatic phenotyping of developing embryos from videos. *IEEE Trans. Image Process.*, 14(9):1360–1371, Sept. 2005.

B. A. Olshausen and D. J. Field. Natural image statistics and efficient coding. *Network: Comp. Neural*, 7(2):333–339, May 1996.

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

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

L. C. Pickup, S. J. Roberts, and A. Zisserman. A sampled texture prior for image super-resolution. In *Adv. in Neur. Inf. Proc. Sys. (NIPS)*, vol. 16, 2004.

T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:314–319, Sept. 1985.

J. Portilla. Benchmark images. http://www.io.csic.es/PagsPers/JPortilla/denoise/test_images/index.htm, 2006a.

J. Portilla. Image denoising software. http://www.io.csic.es/PagsPers/JPortilla/denoise/software/index.htm, 2006b. Software version 1.0.3.

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

B. Potetz. Efficient belief propagation for vision using linear constraint nodes. In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, June 2007.

C. E. Rasmussen. `minimize.m` - Conjugate gradient minimization. http://www.kyb.tuebingen.mpg.de/bs/people/carl/code/minimize/, Sept. 2006.

S. Roth. *High-Order Markov Random Fields for Low-Level Vision*. Ph.D. dissertation, Brown University, Department of Computer Science, Providence, Rhode Island, May 2007.

S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, vol. 2, pp. 860–867, June 2005.

S. Roth and M. J. Black. Steerable random fields. In *IEEE Int. Conf. on Comp. Vis. (ICCV)*, Oct. 2007a.

S. Roth and M. J. Black. On the spatial statistics of optical flow. *Int. J. Comput. Vision*, 74(1):33–50, Aug. 2007b.

S. Roweis and Z. Ghahramani. A unifying review of linear Gaussian models. *Neural Comput.*, 11(2):305–345, Feb. 1999.

D. L. Ruderman. The statistics of natural images. *Network: Comp. Neural*, 5(4):517–548, Nov. 1994.

P. Sallee and B. A. Olshausen. Learning sparse multiscale image representations. In *Adv. in Neur. Inf. Proc. Sys. (NIPS)*, vol. 15, pp. 1327–1334, 2003.

C. Schnörr, R. Sprengel, and B. Neumann. A variational approach to the design of early vision algorithms. *Computing Supplement*, 11:149–165, 1996.

G. Sebastiani and F. Godtliebsen. On the use of Gibbs priors for Bayesian image restoration. *Signal Processing*, 56(1): 111–118, Jan. 1997.

A. Srivastava, X. Liu, and U. Grenander. Universal analytical forms for modeling image probabilities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1200–1214, Sept. 2002.

A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu. On advances in statistical modeling of natural images. *J. Math. Imaging Vision*, 18(1):17–33, Jan. 2003.

L. Stewart, X. He, and R. S. Zemel. Learning flexible features

for conditional random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(8):1415–1426, Aug. 2008.

J. Sun, N.-N. Zhen, and H.-Y. Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7):787–800, July 2003.

R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *Int. J. Comput. Vision*, 5(3):271–301, Dec. 1990.

M. F. Tappen, B. C. Russell, and W. T. Freeman. Exploiting the sparse derivative prior for super-resolution and image demosaicing. In *Proceedings of the 3rd International Workshop on Statistical and Computational Theories of Vision*, Nice, France, Oct. 2003.

Y. W. Teh, M. Welling, S. Osindero, and G. E. Hinton. Energy-based models for sparse overcomplete representations. *J. Mach. Learn. Res.*, 4:1235–1260, Dec. 2003.

H. Tjelmeland and J. Besag. Markov random fields with higher-order interactions. *Scandinavian Journal of Statistics*, 25(3):415–433, Sept. 1998.

W. Trobin, T. Pock, D. Cremers, and H. Bischof. An unbiased second-order prior for high-accuracy motion estimation,. In *Pat. Recog., Proc. DAGM-Symp.*, vol. 5096 of *Lect. Notes in Comp. Sci.*, pp. 396–405. Springer, 2008.

M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *Int. J. Comput. Vision*, 62(1–2):61–81, Apr. 2005.

Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, Apr. 2004.

J. Weickert. A review of nonlinear diffusion filtering. In *Proceedings of Scale-Space Theory in Computer Vision*, vol. 1252 of *Lect. Notes in Comp. Sci.*, pp. 3–28, Berlin, Germany, 1997. Springer.

Y. Weiss and W. T. Freeman. What makes a good model of natural images? In *IEEE Conf. on Comp. Vis. and Pat. Recog. (CVPR)*, June 2007.

M. Welling and C. Sutton. Learning in Markov random fields with contrastive free energies. In R. G. Cowell and Z. Ghahramani, eds., *Int. Works. on Art. Int. and Stat. (AISTATS)*, pp. 389–396, Jan. 2005.

M. Welling, G. E. Hinton, and S. Osindero. Learning sparse topographic representations with products of Student-t distributions. In *Adv. in Neur. Inf. Proc. Sys. (NIPS)*, vol. 15, pp. 1359–1366, 2003.

H. Wersing, J. Eggert, and E. Körner. Sparse coding with invariance constraints. In *Int. Conf. on Art. Neur. Netw. (ICANN)*, pp. 385–392, June 2003.

E. Wong. Two-dimensional random fields and representation of images. *SIAM J. Appl. Math.*, 16(4):756–770, 1968.

C. Yanover, T. Meltzer, and Y. Weiss. Linear programming relaxations and belief propagation – An empirical study. *J. Mach. Learn. Res.*, 7:1887–1907, Sept. 2006.

J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. In G. Lakemeyer and B. Nebel, eds., *Exploring Artificial Intelligence in the New Millennium*, chapter 8, pp. 239–236. Morgan Kaufmann Pub., 2003.

A. Zalesny and L. van Gool. A compact model for viewpoint dependent texture synthesis. In *Proceedings of SMILE 2000 Workshop*, vol. 2018 of *Lect. Notes in Comp. Sci.*, pp. 124–143, 2001.

S. C. Zhu and D. Mumford. Prior learning and Gibbs reaction-diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(11):1236–1250, Nov. 1997.

S. C. Zhu, Y. Wu, and D. Mumford. Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. *Int. J. Comput. Vision*, 27(2): 107–126, Mar. 1998.