Fields of Experts for Image-based Rendering

O. J. Woodford[1], I. D. Reid[1], P. H. S. Torr[2] and A. W. Fitzgibbon[3]
[1]Department of Engineering Science, University of Oxford
[2]Department of Computing, Oxford Brookes University
[3]Microsoft Research, Cambridge

Abstract

Image priors for novel view synthesis have traditionally been non-parametric models based on large libraries of image patch exemplars, producing high-quality results but making inference very slow. Recently a parametric framework, called Fields of Experts, has been proposed for image restoration that promises to speed up inference dramatically. In this paper we apply Fields of Experts for the first time to the problem of novel view synthesis, posed as a Markov random field labelling problem with very large cliques. Additionally, we introduce to computer vision for the first time a new optimization algorithm from statistical physics which reaches better minima than the ICM and simulated annealing algorithms to which such large-clique problems have previously been restricted.

1 Introduction

Novel view synthesis (NVS) is the task of generating a new view of a scene, given a set of input images of that scene. As Poggio *et al.* [13] point out, this is an ill-posed problem due to the many-to-one mapping of scenes to images, thus requiring strong priors to choose between the many potential solutions.

Current research focuses on patch-based priors. This allows for more realistic rendering of details poorly modelled by the piecewise-smooth priors conventionally used for stereo, such as hair and textured surfaces, instead constraining the synthesized image to have the same local statistics as the input image sequence. Fitzgibbon *et al.* [4] introduced a non-parametric prior for use in NVS, posing the problem in an energy minimization framework. However, inference using their prior, based on a large library of image patches, is slow. Although efforts have been made to overcome this [18], it remains the case that the minimization strategy is limited to iterated conditional modes.

Recently, Roth and Black [14] introduced a new image prior, known as Fields of Experts (FoE), based on a parametric model of patches, for the purpose of inpainting and denoising images. In this paper our first contribution is to employ FoE for the first time for the purposes of NVS, embedding it in the MRF framework of Fitzgibbon *et al.* The parametric prior promises faster inference, and the ability to obtain better minima, than exemplar-based priors.

It is a characteristic of patch-based priors that the corresponding MRFs contain large cliques. Techniques such as graph cuts [8] and loopy belief propagation [3] become exponentially more costly with increasing clique size, making them unsuitable for patch-based optimizations. Methods for large-clique size energy minimization currently famil-

iar to vision researchers—iterated conditional modes (ICM) [2] and simulated annealing (SA) [5]—both tend to find poor minima in our problem. Our second contribution is to introduce a method new to computer vision, called Chained Local Optimization (CLO) [11], and compare the results of all three methods on this problem.

1.1 Related work

Two areas of research relate to our problem: image priors and large-clique optimization methods.

The non-parametric prior introduced in [4] models the space of natural image patches using a distance transform over a library of example patches. This can loosely be thought of as a sum-of-experts or mixture model, where each exemplar models a small part of the high-dimensional space and the distributions over each exemplar are summed (or, in the case of the distance transform, the maximum value is chosen). However, a large library is required to correctly model the space of natural image patches, making inference using this prior very slow. As a result, Fitzgibbon *et al.* were only able to optimize their energy minimization framework using an approximation of ICM.

In diametric contrast, Hinton [6] developed a parametric prior made up of experts that each model a different uni-dimensional part of the high-dimensional patch space. These experts are then multiplied together in a framework called Products of Experts (PoE). The strength of this framework comes from its simplicity, which makes inference much faster. In the field of computer vision PoE models have been applied to hand-written digit classification [6], and to image denoising [17]. In the former case the experts could only be learned on and applied to small, identically sized images. In the latter case the experts were learned on 10×10 pixel images, then applied to all the 10×10 sub-images, or patches, in a larger image. However, since inference using this model does not account for the (very large) inter-dependence of overlapping patches, it is generally not suited to vision problems.

To this end, Roth and Black developed the "Fields of Experts" [14] framework for learning image priors. It explicitly models the inter-dependence of image patches by learning the experts over images rather than patches, and once learnt is applicable to any image size. It has been used with great success on the problems of denoising and inpainting (texture synthesis) [14], as well as determining optic flow [15].

We shall define large-clique optimization methods to be methods that increase in running time at worst polynomially with clique size. Minimizing the energy of an MRF given a large-clique energy term is an NP-hard problem. Current solutions tend to be iterative, with iteration time increasing linearly with clique and image size. The methods most familiar to the vision community are ICM [2], a deterministic algorithm guaranteed to find a local minimum (generally close to the initial MRF labelling), and simulated annealing (SA) [5], a much slower, stochastic approach that aims to find a stronger minimum.

However, there is a great deal of research being carried out on this problem in other scientific fields, in particular physics. One area of interest is that of Iterated Local Search (ILS) methods, of which there is a survey [10], which aim to stochastically search through the local minima found by another method such as ICM. We introduce an algorithm in this class called Chained Local Optimization (CLO) [11], and compare its performance against those of ICM and SA on our MRF labelling problem.

1.2 Problem statement

Our aim is to produce an image, $\mathcal{V}$—the new view of a scene to be synthesized—comprising pixels V_j, which defines the colour (a vector in the chosen colour space) of pixel j, where $j \in S$ and S is the list of all pixels in the output image. As input we are given a list of n colour choices, $\mathbf{C}_j$, for each pixel, making up a colour cube, $\mathcal{C}$. We are also given a cube of data likelihood energies, $\mathcal{E}$, the values of which correspond to the colours in $\mathcal{C}$. These values represent the photoconsistency [16] of each output pixel with the input images of the scene.

Our problem is one of ascertaining the most likely labelling, $L_j = 1, .., n$, of each pixel, where $V_j = \mathbf{C}_j[L_j]$ (or, more generally, $\mathcal{V} = \mathcal{C}[\mathcal{L}]$). Following [4], we pose the problem in an energy minimization framework, with the energy given by

$$E(\mathcal{L}) = \mathcal{E}[\mathcal{L}] + \lambda E_{\text{texture}}(\mathcal{C}[\mathcal{L}]) \tag{1}$$

where E_{texture} is the energy due to the image prior, defined below. We calculate $\mathcal{C}$ and $\mathcal{E}$ from the sequence of input images using the method described in [18], and set λ experimentally (in this work, $\lambda = 100$ was found to give the best results).

2 Parametric image priors

An image prior is a function $E_{\text{texture}}(\mathcal{V})$ which evaluates the likelihood that $\mathcal{V}$ is from our target distribution. In order to learn the form of E_{texture} from training examples, it is necessary to restrict the form of E_{texture} to be a function over image *patches* $\mathbf{x}$, typically small windows centered at each pixel.

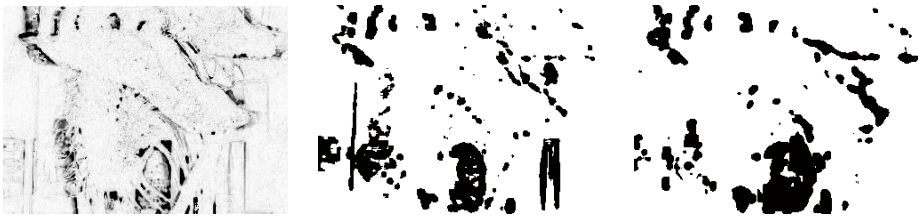
2.1 Products of Experts

High-dimensional data often satisfies many different low-dimensional constraints simultaneously. An efficient way to model such data is therefore to multiply together several probability distributions that are expert over each of these low-dimensional spaces—this is the premise of Hinton's Products of Experts [6].

In the PoE framework experts in the model each work on a linear one-dimensional subspace of the data, in our case images, by projecting the vectorized patch, $\mathbf{x}$, onto a linear component, $\mathbf{J}_i$. The distribution of each expert is modelled by the Student-t distribution, chosen because its heavy tails match the highly kurtotic marginal distributions that the responses of linear filters applied to natural images have been shown to exhibit [7]. The prior therefore takes the form

$$p(\mathbf{x}) = \frac{1}{Z(\Theta)} \prod_{i=1}^{N} \left(1 + \frac{1}{2}(\mathbf{J}_i^\top \mathbf{x})^2\right)^{-\alpha_i} \qquad \alpha_i > 0 \tag{2}$$

where N is the number of experts in the model, and $\{\mathbf{J}_1, .., \mathbf{J}_N, \alpha_1, .., \alpha_N\} = \Theta$ are the model parameters. $Z(\Theta)$, the partition function, is a normalization constant that ensures that $p(\mathbf{x})$ integrates to one. Evaluating the partition function for a distribution of this form is algebraically intractable, but there exist algorithms for both learning the model parameters (§2.3) and applying the model (§3) that do not require $Z(\Theta)$ to be known.



a b c

Figure 1: **FoE versus PoE as an image prior.** (a) The difference between a ground truth image and an image, $\mathcal{I}$, synthesized, using no prior, from other input images. (b), (c) The 10% of pixels with the highest E_{texture}, calculated using the learnt FoE parameters and the estimated PoE model parameters used to initialize the FoE learning respectively. An omniscient image prior would predict accurately all the errors in (a). In practice, FoE (b) predicts more of the errors than PoE (c), indicating its usefulness in the NVS task.

Inference algorithms dispense with the partition function by writing the probability as a Gibbs distribution, $p(\mathbf{x}) = \exp\left(-E(\mathbf{x})\right)/Z(\Theta)$ and minimizing the energy:

$$E_{\text{PoE}}(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i \log\left(1 + \frac{1}{2}(\mathbf{J}_i^\top \mathbf{x})^2\right) \tag{3}$$

2.2 Fields of Experts

FoE [14] was developed to overcome the shortcomings of PoE for vision applications. Its aim is to learn a parametric prior over $m \times m$ image patches that accounts for the inter-dependence of neighbouring (overlapping) image patches. The energy is therefore explicitly dependent on all the vectorized $m \times m$ image patches, $\mathbf{x}_{1,..,K}$, in an image, $\mathcal{V}$, thus taking the form

$$E_{\text{FoE}}(\mathcal{V}) = \sum_{k=1}^{K}\sum_{i=1}^{N} \alpha_i \log\left(1 + \frac{1}{2}(\mathbf{J}_i^\top \mathbf{x}_k)^2\right) \tag{4}$$

This model also has the desirable properties that it is translation invariant and can be used on images of varying size.

2.3 Learning parameters

The parameters of the FoE model are learned from a set of training images, $\mathbb{I} = \{\mathcal{I}_1, .., \mathcal{I}_D\}$, by maximizing the likelihood of the set under the model, given by $\prod_{d=1}^{D} p(\mathcal{I}_d)$. Since one can't calculate $Z(\Theta)$ there is no closed form solution; instead, a gradient ascent is performed on the log-likelihood.

As [14] explains, this ordinarily requires a numerical integration over $p(\mathcal{I})$ per parameter update. Using Monte Carlo integration this requires many MCMC sampling cycles, with the result that the learning process is prohibitively long. However, Hinton [6] shows that the gradient of the log-likelihood can be approximated using as few as one MCMC sampling cycle per iteration if the MCMC sampler is initialized with the training set,

$\mathbb{I}$, in a method known as *contrastive divergence*. Following [14], the parameter update equation is therefore

$$\Theta^{(t+1)} = \Theta^{(t)} + \eta \left(\left\langle \frac{\partial E_{\text{FoE}}}{\partial \Theta^{(t)}} \right\rangle_{\mathbb{I}^1} - \left\langle \frac{\partial E_{\text{FoE}}}{\partial \Theta^{(t)}} \right\rangle_{\mathbb{I}^0} \right) \tag{5}$$

where η is a user-defined step size, $\langle f(x) \rangle_X$ denotes the average of $f(x)$ over the samples, X, and $\mathbb{I}^n$ represents $\mathbb{I}$ transformed by n cycles of MCMC.

The data vectors, $\mathcal{I}_1, .., \mathcal{I}_D$, in our learning problem have a high dimensionality. Simple MCMC sampling techniques, such as Metropolis sampling, exhibit random walk behaviour, thus are very poor at exploring the data space. Instead, we use the Hamiltonian Monte Carlo (HMC, also known as hybrid Monte Carlo) sampler [12], which explores the space more effectively by making use of the gradient of the log-likelihood w.r.t. the data.

2.3.1 Implementation details

We learn filters for 5×5 patches, a patch size shown to be adequate for the task of NVS [4, 18]. We train the model on 12000 15×15 cropped image regions, sampled uniformly from the set of input images of the scene for which an image is to be synthesized. In contrast to [14], we train on RGB image patches (with an intensity range of $[0, 1]$) and do not whiten the training data. In addition, we homogenize the patch vectors when evaluating E_{FoE} during both training and inference, so that the projection of patches becomes $\mathbf{J}_i^\top \mathring{\mathbf{x}}$ where $\mathring{\mathbf{x}} = [\mathbf{x}^\top, 1]^\top$. This introduces one more parameter to learn for each filter, but it ensures that $\mathcal{V} = \mathtt{0}$ is not necessarily given the greatest likelihood.

Following [14], we ensure that the α parameter of each expert remains positive by updating its logarithm, use 30 leaps per HMC step and adjust the leap size to maintain an acceptance rate of around 90%. We learn 30 linear experts to model our 75D patch space, thus our representation is under-complete. We initialize our filters by calculating the eigenvectors of 5×5 sub-patches from all the training patches, and setting the filters $\mathbf{J}_{1,..30}$ to the eigenvectors with the smallest eigenvalues, with the last value of each filter (due to the homogenization of the data) set to 0. This gives the lowest variance projection of 5×5 patches onto the experts' subspace, a good starting point for learning a PoE model, but which also worked well for FoE, as indicated by Figure 1.

The learning process is slow, but we have found that a large step size, η, can cause the HMC acceptance rate to become unstable, which in turn produces inaccurate gradient estimates and an unstable gradient ascent. Throughout the learning process we therefore set η to give a maximum change of 0.01 over all parameters at each update. Instead, to accelerate learning we start the process using only 1000 image patches selected randomly from $\mathbb{I}$, and slowly increase the number of patches used up to the full set as the training progresses. The number of parameter updates remains the same, but using a smaller patch set early on reduces the time per iteration. We have found that using this method produces a stable parameter set after around 2000 iterations.

3 Energy minimization

The second major focus of our work is on computing high-quality minima of the energy. To this end we introduce a new algorithm to computer vision, and compare it with the state of the art large-clique optimizers, ICM and SA.

3.1 Iterated conditional modes

At each iteration, ICM [2] minimizes the conditional energy of each pixel's label, which is to say its energy given the current labelling elsewhere. Let us define $S\backslash j$ to be the list of all output pixels except pixel j, such that, for example, $\mathcal{L}_{S\backslash j}$, is the labelling of the output image excepting pixel j. Thus, the label update equation for pixel j using ICM is

$$L_j^{(t+1)} = \underset{l \in \{1,..,n\}}{\operatorname{argmin}} \; E(\mathcal{L}_{S\backslash j}^{(t)}, L_j = l) \tag{6}$$

This update is guaranteed to not increase the total energy, as the pixel label can, at worst, remain the same. We can therefore be sure that, by updating the label of each pixel in the output image sequentially using Equation 6, then repeating the cycle until no pixel labels change between iterations, we will find a local energy minimum in a deterministic fashion.

3.2 Simulated annealing

SA [5] chooses a labelling for each pixel by sampling from its conditional probability distribution, which can be derived from the conditional energy. In addition, the conditional probability distribution is made dependent on a global control parameter, T, known as the "temperature", as follows

$$p(L_j = l|\mathcal{L}_{S\backslash j}) = \frac{\exp\left(-E(\mathcal{L}_{S\backslash j}, L_j = l)/T\right)}{\sum_{i=1}^{n} \exp\left(-E(\mathcal{L}_{S\backslash j}, L_j = i)/T\right)} \tag{7}$$

The algorithm works by sampling all pixel labels from the above distribution, either synchronously or sequentially, then lowering the temperature according to a cooling schedule and repeating the process. At $T = \infty$ the distribution is uniform, and the algorithm is essentially a greedy, random search through all labellings that will eventually find the global minimum. As $T \to 0$ the mass of the distribution becomes increasingly concentrated on the minimum energy label. It can therefore be seen that ICM is just a special case of SA where the MRF is essentially frozen, yielding a local minimum close to the initial labelling.

The aim of this sampling with slow cooling (annealing) is to strike a balance between these two extremes of computational expense and local optimization. The stochasticity of the sampling process allows the labelling, $\mathcal{L}$, to hop out of local minima, while the annealing causes the energy gradient to remain effective in gradually reducing the labelling energy as it approaches the global minimum.

3.3 Chained local optimization

As shown in §3.1, ICM can be used to find a local minimum labelling, $\mathcal{L}^*$, deterministically, given an initial labelling, $\mathcal{L}$. Since many labellings reduce to the same local minimum, the space of local minimum labellings, $\mathbb{L}^*$, is much smaller than the space of all labellings, $\mathbb{L}$. The stochastic search of SA is therefore very inefficient, as it traverses the space $\mathbb{L}$. Rather, one should stochastically traverse the smaller space $\mathbb{L}^*$. We do this using CLO [11], which works by taking a locally minimized (in our case via ICM) labelling, $\mathcal{L}_1^*$, stochastically perturbing or "kicking" it and performing another local minimization to

give $\mathcal{L}_2^*$. An optimal labelling is then selected from $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$, and the process repeated using the optimal labelling as a starting point.

The kick plays a pivotal role in the efficiency of CLO. A good kick heuristic should perturb the labelling in a direction that will lead to a different, preferably lower, local minimum. For this purpose we developed the "blowtorch" approach.

Blowtorching Blowtorching perturbs pixel labels in $\mathcal{L}_1^*$ by sampling them from the same distribution as used by SA, defined in Equation 7. However, a high temperature, T, is used, for one iteration only, and in order to reduce computation, only pixel labels deemed to be a poor choice are perturbed. Candidate pixels for this process are chosen using the FoE energy of Equation 4, which provides an energy for each patch in an image. Each CLO iteration we choose the 10% of pixels whose average clique energies are greatest, demonstrated in Figure 1, as these are the least natural pixels (according to our prior) in the image.

The perturbed labelling is then "quenched" using ICM, producing labelling $\mathcal{L}_2^*$. As a consequence of the localized nature of blowtorching, differences between $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$ tend to be fragmented, surrounded by areas of identical labelling. It is possible to select the optimal labelling by splicing together fragments from $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$, using a method described in [10] which we call "optimal splicing".

Optimal splicing Optimal splicing takes $\mathcal{L}_1^*$ and swaps in all the fragments of $\mathcal{L}_2^*$ that lower the energy of $\mathcal{L}_1^*$ (or vice-versa). A fragment is defined to be the largest group of pixels, S', grown from a seed pixel, for which $L_{1j}^* \neq L_{2j}^*, j \in S'$ and all the pixels share a clique with at least one other pixel in S', such that each of the fragments can be swapped between $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$ independently. We generate the fragments using morphological operations on the binary image, $\mathcal{L}_1^* \neq \mathcal{L}_2^*$, and calculate the energy of each fragment in $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$ thus

$$E(L_{S'}) = \sum_{j \in S'} \mathbf{E}_j[L_j] + \sum_{k \in K'} \sum_{i=1}^{N} \alpha_i \log\left(1 + \frac{1}{2}(\mathbf{J}_i^\top \mathbf{x}_k)^2\right) \tag{8}$$

where K' is the list of patches that contain at least one pixel from S'. We then select the lowest energy fragments from $\mathcal{L}_1^*$ and $\mathcal{L}_2^*$ for the output labelling[1].

4 Experiments and Results

We tested the FoE prior and optimization methods by synthesizing images of a dinosaur skeleton from a 63 frame pal video filmed in a museum. The output image used for our comparisons is a novel viewpoint within the bounds of the original sequence. Each of the optimization methods were run 6 times to get a representative measure of performance.

All three minimization algorithms were placed in a common framework. Labellings were initialized to that which minimized $\mathcal{E}[\mathcal{L}]$. Each iteration pixels were updated sequentially in a stipple pattern, so that the same pixel in every non-overlapping patch would be

[1] As the optimal splice is simply a binary-graph partitioning problem it can also be solved optimally using graph cuts [9]. However, our large clique size generates a large number of edges in the graph, making graph cuts a less efficient solution.

Optimization method	Mean energy $\pm$ s.d. (in arbitrary units)	Minimum energy	Time (s)
None	1.418$\pm$ 0	1.418	0
ICM	1.208$\pm$ 0	1.208	3000
ICM with non-parametric prior*	N/A	N/A	87500
SA	1.300$\pm$ 0.001	1.299	60000
CLO	1.196$\pm$ 0.001	1.195	60000

Table 1: **Optimization performance**. A comparison of the optimization methods tested in terms of energies of labellings found and computation time. 'None' refers to the maximum likelihood labelling used to initialize all the optimization methods. *The computation time for this method is estimated for comparison.

updated first, followed by another pixel in each patch. This helped to avoid asymmetry in the image caused by an ordered update.

In ICM, only pixels sharing a patch with a pixel that changed in the previous iteration were updated. The algorithm ended when no pixels were updated in the previous iteration.

In SA every pixel was updated each iteration. The temperature, T, started at about 10% of the mean conditional energy of a pixel, and decreased linearly to zero over 250 iterations. ICM was then applied to ensure a local minimum was found.

In CLO 100 iterations were carried out, with the SA steps using a temperature of about 10% of the mean conditional energy on the selected pixels.

We compare the performance of the FoE image prior for NVS with the earlier, non-parametric image prior from [4], synthesizing the same image using our own implementation of their method. Their optimization method, as described, does not fully implement ICM, as the conditional energy of a pixel is calculated using only the patch that pixel is centred on rather than all the patches the pixel is a member of. For comparison with full ICM using FoE we extrapolate the optimization time in Table 1 to account for this. As the table shows, inference using the parametric FoE prior is much faster. A comparison of the synthesized images in Figure 2 shows that FoE tends to soften edges more than the non-parametric prior, thus removing more of the high-frequency artifacts, and showing a general qualitative improvement overall.

We compare the optimization methods on computation time, minimum energy and image quality. Table 1 shows that the stochastic SA and CLO algorithms are substantially slower than the deterministic ICM algorithm, while CLO finds the lowest energy. SA actually found a higher energy minimum than ICM. However, it should be noted that the performance of the stochastic methods are dependent on parameters such as temperature, cooling schedule and number of iterations. The parameters used struck a balance between running time and minimum energy. Qualitatively, ICM was found to leave some artifacts which were local minima in the energy space. CLO was able to remove most of these, producing an improved image. While SA was able to reduce some artifacts it jumped into worse local minima in other regions, generating more artifacts overall.

5 Conclusion

We have introduced FoE as a prior for NVS, allowing much faster rendering of images (using ICM), with reduced artifacts. This speedup means that we can now employ algo-

(a) No prior

(d) FoE prior, optimized with CLO

(e) Non-parametric prior

(A), (A') No prior

(B), (B') FoE prior, optimized with ICM

(C), (C') FoE prior, optimized with SA

(D), (D') FoE prior, optimized with CLO

(E), (E') Non-parametric prior

Figure 2: **Novel view synthesis.** (a), (A), (A') Maximum data likelihood image, rendered without priors (with zooms). (B), (B') Zooms of the image rendered using the FoE prior and optimized with ICM. (C), (C') Zooms of the image rendered using the FoE prior and optimized with SA. (d), (D), (D') The lowest energy image rendered using the FoE prior and optimized with CLO (with zooms). (e), (E), (E') Image rendered using the method from [4] (with zooms).

rithms more sophisticated than ICM. In particular, we have introduced the CLO algorithm to computer vision for the first time, and shown that it can achieve better minima than simulated annealing or ICM. In addition, we have suggested a method to speed up FoE model learning.

It remains the case that the effectiveness of CLO depends on the "kick" given by the random perturbations. Our current implementation is still unable to fix problems caused by systematic errors in the maximum likelihood solution, for example those caused by large occlusions. Our current research is to improve the perturbation strategy in order to generate more structured moves, which may be possible using a variation on the Swendsen-Wang MCMC sampling strategy of Barbu *et al.* [1].

Acknowledgements Many thanks to EPSRC, the Imaging Faraday Partnership and Sharp Laboratories of Europe for their support through an industrial CASE award, and to Andrew Zisserman and Vladimir Kolomogorov for their kind assistance and advice.

References

[1] A. Barbu and S. Zhu. Graph partition by Swendsen-Wang cuts. In *Proc. ICCV*, volume 1, pages 320—327, 2003.

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

[3] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. In *Proc. CVPR*, pages 261–268, 2004.

[4] A. Fitzgibbon, Y. Wexler, and A. Zisserman. Image-based rendering using image-based priors. In *Proc. ICCV*, volume 2, pages 1176–1183, Oct 2003.

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

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

[7] J. Huang and D. Mumford. Statistics of natural images and models. In *Proc. CVPR*, pages 1541–1547, 1999.

[8] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Proc. ECCV*, volume 3, page 82, 2002.

[9] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE PAMI*, 26(2):147–159, 2004.

[10] H. R. Lourenço, O. C. Martin, and T. Stützle. Iterated local search. In *Handbook of Metaheuristics*, pages 321–353, 2002.

[11] O. C. Martin and S. W. Otto. Partitioning of unstructured meshes for load balancing. *Concurrency: Practice and Experience*, 7(4):303–314, 1995.

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

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

[14] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *Proc. CVPR*, volume 2, pages 860–867, 2005.

[15] S. Roth and M. J. Black. On the spatial statistics of optical flow. In *Proc. ICCV*, pages 42–49, 2005.

[16] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Proc. CVPR*, pages 1067–1073, 1997.

[17] M. Welling, G. Hinton, and S. Osindero. Learning sparse topographic representations with products of student-t distributions. In *NIPS*, pages 1359–1366, 2003.

[18] O. Woodford and A. W. Fitzgibbon. Fast image-based rendering using hierarchical image-based priors. In *Proc. BMVC.*, volume 1, pages 260–269, 2005.