Learning Non-Local Range Markov Random Field for Image Restoration

Jian Sun
School of Science, Xi'an Jiaotong University
jiansun@mail.xjtu.edu.cn

Marshall F. Tappen
EECS, University of Central Florida
mtappen@eecs.ucf.edu

Abstract

In this paper, we design a novel MRF framework which is called Non-Local Range Markov Random Field (NLR-MRF). The local spatial range of clique in traditional MRF is extended to the non-local range which is defined over the local patch and also its similar patches in a non-local window. Then the traditional local spatial filter is extended to the non-local range filter that convolves an image over the non-local ranges of pixels. In this framework, we propose a gradient-based discriminative learning method to learn the potential functions and non-local range filter bank. As the gradients of loss function with respect to model parameters are explicitly computed, efficient gradient-based optimization methods are utilized to train the proposed model. We implement this framework for image denoising and inpainting, the results show that the learned NLR-MRF model significantly outperforms the traditional MRF models and produces state-of-the-art results.

1. Introduction

The Markov Random Field (MRF) [11, 30, 10, 4] provides an effective framework for modeling the statistical prior of natural images [24, 14]. Image models based on MRFs have been widely investigated and applied to vision problems, e.g., image denoising [21, 1, 13, 32, 26, 22, 27, 17], inpainting [20], segmentation [2], stereo [25], etc.

In this paper, we focus on the continuously-valued high-order MRF model [26], which is often constructed to model the marginal distributions of image responses to a filter bank [23]. In designing the model, two issues are commonly encountered. The first issue is the construction of the random field, i.e., the spatial range of clique and the potential function. Pairwise MRFs [4] model the statistical dependency between the neighboring pixels. High-order MRF models [21, 20, 1] extend these models to larger neighborhoods to learn the MRF models with larger cliques, enabling them to better capture image structures. The potentials themselves are often a robust function [25], student-t distribution [21], or Gaussian scale mixture model [23, 28]. These functions are typically used to model the heavy-tailed distribution of image responses.

The other major issue lies in learning the parameters. Sampling-based methods [32, 9, 21, 23] and discriminative learning methods [27, 26, 22, 1] are the two most common approaches for fitting an MRF model to natural images. The sampling-based methods utilize efficient sampling methods to learn image prior by fitting the statistics of natural images. These methods are well founded on statistical theories and exhibit great performance in learning natural image prior in a general way. The discriminative learning methods learn the model parameters by constructing a loss function [12] between the inferred image by MRF model and the target image, and optimize the loss function by variational method [26], implicit differentiation [22], or coordinate descent algorithm [1]. This category of methods has exhibited excellent performance in image restoration applications.

In this paper, we propose a novel framework of MRF model, which is called non-local range MRF (NLR-MRF) model. The major contribution is that a non-local range Markov random field is constructed by extending the local range clique around pixel to be a non-local range clique composed of several similar patches in a non-local window around each pixel. Using this clique, traditional spatial convolution is extended to a 3-dimensional non-local range convolution over the spatially adaptive non-local ranges of pixels. In this framework, both of the local image structures and the dependencies among the similar patches are captured. As will be shown in Section 5, this model leads to significant performance increases in MRF-based image denoising and inpainting.

We propose a gradient-based discriminative learning framework for learning the non-local range MRF model. This training method is motivated by active random field [1], which was trained using a randomized descent procedure. In this work, we will show how to explicitly compute the gradients of loss function with respect to the parameters for NLR-MRF model. This enables us to use gradient-based optimization methods to flexibly and efficiently train the system. As a side-benefit, the gradient computation framework is general for both NLR-MRF model and the traditional Field of Experts (FoE) [21] model with

many differential potential functions, providing a general framework for discriminatively learning the FoE-like MRF models.

The following sections are organized as follows. The related works are discussed in Section 2. In Section 3, the non-local range MRF is introduced. In Section 4, the training algorithm of NLR-MRF model is presented by explicitly computing the gradients of loss function with respect to model parameters. In Section 5, experimental evaluations are presented. Finally, this work is concluded in Section 6.

2. Related Work

Product of Experts (PoE) [9] and Field of Experts (FoE) [21] are typical high-order MRF models in which the range of clique is defined as the local patch around each pixel. Steerable random field (SRF) [20] learns the image responses to local derivative filter that is aligned to the local directions of image structures. In sparse long-range random field [13], the sparse and long-range clique with size three is designed to capture the second-order image statistics. All of these MRF models capture the high-order dependency of image structure around each pixel. In this work, the characteristics of self-similarity in image patches is introduced to design the non-local range of clique in MRF which captures both the local image structures and also the dependencies among the non-locally similar patches.

Training the high-order MRF is a challenging work. Zhu and Mumford [32] propose to use Gibbs sampling to learn the statistics of image responses to a pre-defined filter bank. PoE [9] and FoE [9] jointly learn the filter bank and fit the distribution of image responses to the filter bank using contrastive divergence. In [23], auxiliary-variable Gibbs sampler and Bayesian minimum mean squared error estimate are proposed to exploit more general and efficient MRF model using the potentials of Gaussian scale mixture (GSM) model [14]. These sampling-based methods have sound statistical interpretation, however, they are generally computationally challenging in applications.

Discriminative learning methods learn the parameters of MRF by minimizing a loss function between the estimated images and the target images. Variational optimization method [26] discriminatively learns MRF model through deriving the quadratic up-bound for student-t expert function, however, it is not easy to find the up-bounds for general expert functions. In [22], the gradients of cost function w.r.t. model parameters are calculated by implicit differentiation. The above methods require that the MAP estimation is optimized to achieve the final results at convergence. Barbu [1] proposes an Active Random Field (ARF) that discriminatively trains the MRF model combined with a sub-optimal inference algorithm using fixed number of gradient descent iterations and produces excellent denoising results. The model parameters are learned by randomized descent procedure that randomly selects a variable and modify it by a small amount if the cost function decreases. Our training framework is motivated by ARF, however, we propose a general gradient-based training framework to learn the parameters by explicitly computing the gradients of loss function with respect to parameters for NLR-MRF model with any differentiable potential functions. As the gradients are explicitly computed, it is efficient and flexible to learn NLR-MRF model using gradient-based optimization algorithms.

The characteristics of self-similarity in natural image is widely investigated in image restoration [5, 6, 15, 8], motion estimation [29], super-resolution [31], etc. In this work, this characteristics is utilized into the MRF-based image prior, which captures the advantages of BM3D [6] and non-local means [5] in a flexible MRF framework. It can also be applied to more low level vision applications besides denoising and inpainting, e.g., super-resolution, demonsacing, that will be investigated in our future work.

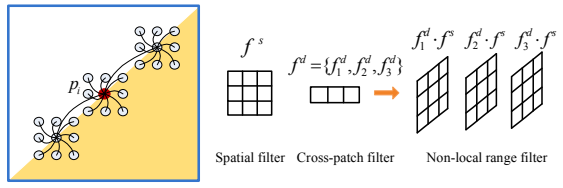
3. Non-Local Range Markov Random Field

Markov Random Field (MRF) is formally defined over a graph $G =< V, E >$. V is the set of nodes representing the random variables $\mathbf{x} = \{x_v\}_{v \in V}$, E are the edges connecting the nodes, and the clique $c \in C$ is defined by a neighborhood system, which indicates the factorization of the probability density of MRF. By Hammersley-Clifford theorem, the probability density of MRF is a Gibbs distribuion: $p(\mathbf{x}) = \frac{1}{Z(\Theta)} \exp(-\sum_{c \in C} V_c(\mathbf{x}; \Theta))$, where $V_c(\cdot)$ is the potential function over clique c, and Z is the partition function to normalize the density. In applications of low level vision, image lattice coupled with a neighborhood system is mapped to MRF, and the potential function is modeled as the robust function [25], student-t distribution [19, 21, 1] or Gaussian scale mixture model [28, 23].

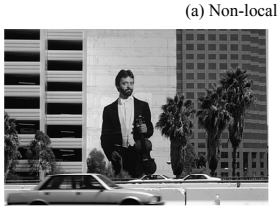
3.1. Basic Definition

The clique of MRF is generally defined by the local neighborhoods around pixels in the traditional MRF. In this work, we extend the clique of MRF to be defined over non-local range in natural image, which is called ***Non-Local Range Markov Random Field*** (NLR-MRF).

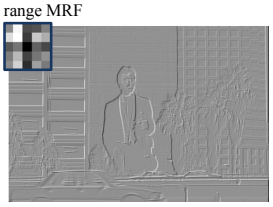
In non-local range MRF, the clique of each pixel is composed of the local patch around the pixel and also its top similar patches searched by block matching with mean squared error as patch similarity in the non-local neighborhood around the pixel, and we define the involved pixels in the clique as the ***non-local range*** of this pixel. For example, as shown in Figure 1 (a), the non-local range of pixel p_i (shown in the red dot) is composed of its connected pixels both in the local patch and the similar patches in a larger neighborhood. Apparently, the non-local ranges of pixels are adaptively located over the image region depending on the different locations of the similar patches.



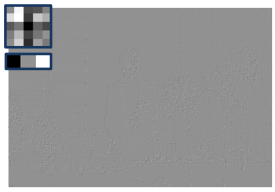
(a) Non-local range MRF



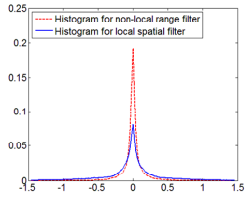
(b) Natural image



(c) Response map of local spatial filter



(d) Response map of non-local range filter



(e) Histogram of filter responses

Figure 1. Non-Local Range MRF (NLR-MRF) and the maps of image responses to local and non-local range filter. In NLR-MRF, The clique of each pixel is composed of the patch around the pixel and the top similar patches in its non-local window. The non-local range filter is the combination of local spatial filter and the cross-patch filter. Compared with the local spatial filter, the response map of non-local range filter is much more sparser, which is beneficial to better preserve the image details in image restoration.

Based on this definition, the filter that convolves an image is also extended to the ***non-local range filter***, which is defined over the non-local ranges of pixels. As the non-local range of each pixel is composed of a set of patches, the filter over the non-local range should be a 3-dimensional filter (for gray-scale image). Due to the high similarities among the patches in a non-local range, we define the non-local range filter, denoted as f, as the combination of two separate filters: a spatial filter f^s defined over each patch and a 1-dimensional cross-patch filter f^t defined across different patches in a non-local range. Assume that the local spatial filter is of size $w \times w$, and there are d similar patches in the non-local range, then f is a three dimensional filter:

$$f = \{f^1; f^2; \cdots; f^d\} = \{f_1^t f^s; f_2^t f^s; \cdots; f_d^t f^s\}. \quad (1)$$

It is composed of d spatial filters scaled by the elements in cross-patch filter $f^t = \{f_1^t, \cdots, f_d^t\}$. Then the ***non-local range convolution*** can be computed as

$$(f * I)(x, y) = \sum_{l=1}^{d} \sum_{p=0}^{w-1} \sum_{q=0}^{w-1} f^{d-l+1}(w-p, w-q) I(x_l + p, y_l + q), \quad (2)$$

where (x_l, y_l) is the left-top position of the l-th most similar patch in the non-local range of (x, y). Obviously, when $l = 1$, the most similar patch is just the local patch surrounding (x, y). As f is separable, this is equivalent to convolving the image by the local spatial filter f^s, and followed by the convolution with cross-patch filter f^t over the patch centers in the non-local range of each pixel.

It is well known that the responses of an image convolved by a high-frequency filter obey a heavy-tailed distribution, and only the high-frequency edges or textures have larger responses. Please refer to Figure 1 (c) for example, it shows the response map of a natural image convolved by a 5×5 filter. However, when convolved by a non-local range filter composed of the same spatial filter and a cross-patch filter $f^t = [-0.15, -0.35, 0.5]$, the response map shown in Figure 1 (d) becomes much sparser than that in Figure 1 (c), even along edges and in the textured region. That is because the high responses in Figure 1 (c) are further removed by the convolution with a high-frequency cross-patch filter. This phenomenon is beneficial to better preserve image structures or textures when applying NLR-MRF model to image restoration.

3.2. Formulation of NLR-MRF model

The non-local range MRF is based on the Field of Experts (FoE) framework with potential function modeling image responses to a filter bank. Assume that there are N filters in the filter bank, then the probability density of an image I is modeled as:

$$p(I; \Theta) = \frac{1}{Z(\Theta)} \prod_{c \in C} \prod_{i=1}^{N} \phi(f_i * I_{(c)}; \Theta), \quad (3)$$

where Θ are the model parameters, $f_i * I_{(c)}$ denotes the value of non-local range convolution at each clique $c \in C$. The clique c is defined as the non-local range of each pixel, and f_i denotes the i-th non-local range filter, which can be decomposed into pairs of filters $\{f_i^t, f_i^s\}$, and ϕ is the expert function modeling heavy-tailed marginal distribution of filter responses. For the convenience of computation, the non-local range convolution can be written as matrix multiplication: $F_i \mathbf{x} = F_i^t F_i^s \mathbf{x}$, where F_i^s, F_i^t are the convolution matrices corresponding to spatial filter f_i^s and cross-patch filter f_i^t respectively, $\mathbf{x}$ is the image vector.

We will investigate two typical types of expert functions, i.e., Student-T (ST) expert:

$$\phi((F_i \mathbf{x})_p; \Theta) = [1 + \frac{1}{2}(F_i^t F_i^s \mathbf{x})_p^2]^{-\alpha_i}, \quad (4)$$

and Gaussian Scale Mixture (GSM) expert:

$$\phi((F_i \mathbf{x})_p; \Theta) = \sum_{j=1}^{J} \alpha_{ij} N((F_i^t F_i^s \mathbf{x})_p^2; 0, \sigma_i^2 / s_j), \quad (5)$$

where $N(\cdot)$ is the Gaussian function, α_{ij} is the weight of j-th Gaussian component for the i-th filter. σ_i^2 is the base variance shared by all the Gaussian components in the i-th filter and s_j is the scale parameter to scale the variance of the j-th Gaussian component shared by all the filters, which is set to

be 11 scales, i.e., $\{s_j\}_{j=1}^{11} = \{\exp(-5, -4, -3, \cdots, +5)\}$ to support different shapes of Gaussian components.

When applying NLR-MRF to image restoration, the inferred image $\mathbf{x}$ (the vector representation of image) is derived by Maximum A Posteriori (MAP) estimation. It is commonly implemented by minimizing its minus logarithm that is an energy function:

$$\text{argmin}_{\mathbf{x}}\{E(\mathbf{x}|\mathbf{y}, \Theta) = E_{\text{data}}(\mathbf{y}|\mathbf{x}) + E_{\text{prior}}(\mathbf{x}; \Theta)\}, \quad (6)$$

where E_{data} is deduced by likelihood and $E_{\text{prior}} = -\sum_p \sum_i \log\phi((F_i\mathbf{x})_p; \Theta)$. The key component in gradient based optimization is the computation of the gradient of $E_{\text{prior}}(\mathbf{x}; \Theta)$. For student-t expert, the gradient can be computed as

$$\frac{\partial E_{\text{prior}}(\mathbf{x}; \Theta)}{\partial \mathbf{x}} = \sum_{i=1}^{N} \alpha_i F_i^T W_i F_i \mathbf{x}, \quad (7)$$

where $W_i = \text{diag}(\frac{1}{1+\frac{1}{2}(F_i\mathbf{x})^2})$, and $\text{diag}(\cdot)$ denotes a diagonal matrix with diagonal vector as in the bracket. For GSM expert, assume that there are J components in each Gaussian scale mixture model, then the gradient is

$$\frac{\partial E_{\text{prior}}(\mathbf{x}; \Theta)}{\partial \mathbf{x}} = \sum_{i=1}^{N} \tau_i \sum_{j=1}^{J} F_i^T W_{ij} F_i \mathbf{x}, \quad (8)$$

where $W_{ij} = \text{diag}(\frac{s_j}{\sigma_i^2} \frac{\alpha_{ij} N(F_i\mathbf{x}; 0, \sigma_i^2/s_j)}{\sum_{l=1}^{J} \alpha_{il} N(F_i\mathbf{x}; 0, \sigma_i^2/s_l)})$, τ_i is added to scale the contribution of the i-th filter to the expert function.

4. Discriminative Learning Framework

Now the challenging work is how to learn the involved parameters in NLR-MRF model. We design an efficient and general framework to train non-local range MRF with general expert functions. Motivated by the success of the ARF model [1], we discriminatively train the model parameters by optimizing the result created by executing a fixed number of gradient-descent optimization steps. Given the pair of degraded image $\mathbf{y}$ and the ground-truth high-quality image $\mathbf{t}$, the parameters are learned by minimizing the loss function between the inferred image $\mathbf{x}^K$ and target image $\mathbf{t}$:

$$\begin{aligned} \Theta^* &= \text{argmin}_{\Theta} L(\mathbf{x}^K(\Theta), \mathbf{t}) \\ &\text{where } \mathbf{x}^K(\Theta) = \text{GradDesc}_K\{E(\mathbf{x}|\mathbf{y}, \Theta)\}, \end{aligned} \quad (9)$$

where GradDesc_K means the K-steps of gradient descent procedures using NLR-MRF model to infer the restored image, and the restored image is initialized by $\mathbf{x}^0 = \mathbf{y}$. We denote the learned NLR-MRF model as **NLR-MRF model with K iterations** in the following sections. Unlike the randomized coordinate descent algorithm in [1], we explicitly compute the gradients of the loss function with respect to model parameters, which enable efficient and flexible gradient-based algorithms to learn NLR-MRF model with general types of expert functions.

4.1. Computation of Gradients

It is non-trivial to compute the gradients of cost function with respect to model parameters for NLR-MRF model. Due to the limited space, only the major formulations are presented and all the related deductions can be found in the ***supplemental material***.

The major technique in computation is to utilize matrix operation instead of the convolution operation in the original definition of NLR-MRF model. We represent the spatial and cross-patch filters as the linear combinations of two dictionaries of basis filters B^s and B^t computed by PCA over the natural image patch samples: $F_i^s = \sum_{m=1}^{N_s} \lambda_{i,m} B_m^s$, $F_i^t = \sum_{n=1}^{N_t} \gamma_{i,n} B_n^t$, in which B_m^s and B_n^t are the m-th and n-th convolution matrices of basis filters for spatial filter and cross-patch filter respectively, N_s, N_t are the number of basis filters. Then learning a filter is to learn its coefficients over the basis filters.

We also found very limited improvement using E_{data} as [1], therefore we safely discard E_{data} and only use E_{prior} for inference. Then the inference problem in Equation (9) can be explicitly written as $\mathbf{x}^{t+1} = \mathbf{x}^t - g(\mathbf{x}^t; \Theta)$, where $g(\mathbf{x}^t; \Theta) = \partial E_{\text{prior}}(\mathbf{x}^t; \Theta)/\partial \mathbf{x}^t$, $t = 0, 1, \cdots, K-1$. Given the training pairs of observed image and desired image $\{\mathbf{y}_l, \mathbf{t}_l\}_{l=1}^{D}$, the gradient of loss function L w.r.t. any parameter $\theta \in \Theta$ is

$$\frac{\partial L(\{\mathbf{x}_l^K, \mathbf{t}_l\})}{\partial \theta} = \sum_l \frac{\partial L(\mathbf{x}_l^K, \mathbf{t}_l)}{\partial \theta} = -\sum_l \sum_{k=1}^{K} \frac{\partial L}{\partial \mathbf{x}_l^k} \frac{\partial g(\mathbf{x}_l^{k-1})}{\partial \theta}, \quad (10)$$

where $\mathbf{x}_l^0$ is the observed image $\mathbf{y}_l$. To make the formulation more clear, we remove the index l of training pairs in the following computations. In the implementation, minus PSNR is used as the loss function, therefore

$$L(\mathbf{x}^K, \mathbf{t}) = -20 \log_{10} \frac{255}{\sqrt{\frac{1}{M}||\mathbf{x}^K - \mathbf{t} - \text{mean}(\mathbf{x}^K - \mathbf{t})||^2}}, \quad (11)$$

M is the number of pixels. By the chain rule of calculus, $\frac{\partial L(\mathbf{x}^K, \mathbf{t})}{\partial \mathbf{x}^t}$ is computed as

$$\frac{\partial L(\mathbf{x}^K, \mathbf{t})}{\partial \mathbf{x}^t} = \frac{\partial L}{\partial \mathbf{x}^K} \prod_{k=t}^{K-1} \frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k}. \quad (12)$$

Note: Equation (10) presents a general framework to compute gradients of loss function w.r.t. model parameters for NLR-MRF model with any differentiable expert functions, and the key is how to compute $\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k}$ and $\frac{\partial g(\mathbf{x}^k)}{\partial \theta}$ for different expert functions. This gradient computation method can also be applied to train the traditional FoE model with local spatial filter bank by setting the cross-patch filter to be with a single value of one.

Next we will compute the gradients for the NLR-MRF model with student-t expert and GSM expert, which are two commonly used expert functions.

4.1.1 Gradients for NLR-MRF with Student-t Expert

We now compute the gradients of loss function w.r.t. parameters $\Theta = \{\lambda_i, \gamma_i, \alpha_i\}_{i=1}^N$ for NLR-MRF model with student-t expert. First, if denote:

$$W_i^k = \text{diag}(\{\frac{1}{1+\frac{1}{2}(F_i\mathbf{x}^k)_p^2}\}_{p=1}^M),$$
$$U_i^k = \text{diag}(\{\frac{(F_i\mathbf{x}^k)_p^2}{[1+\frac{1}{2}(F_i\mathbf{x}^k)_p^2]^2}\}_{p=1}^M),$$

then $\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k} = I - \sum_{i=1}^N \alpha_i F_i^T(W_i^k - U_i^k)F_i$, where M is the number of pixels.

Second, the gradients of $g(\mathbf{x}^k;\Theta)$ w.r.t. parameters in Θ are listed as follows.

$$\frac{\partial g(\mathbf{x}^k)}{\partial \lambda_{i,m}} = \sum_{i=1}^N \alpha_i[(F_i^tB_m^s)^TW_i^kF_i + F_i^T(W_i^k - U_i^k)F_i^tB_m^s]\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \gamma_{i,n}} = \sum_{i=1}^N \alpha_i[(B_n^tF_i^s)^TW_i^kF_i + F_i^T(W_i^k - U_i^k)B_n^tF_i^s]\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \alpha_i} = F_i^TW_i^kF_i\mathbf{x}^k.$$

Given the above computations, it is easy to derive the gradients of $\frac{\partial L}{\partial \lambda_{i,m}}$, $\frac{\partial L}{\partial \gamma_{i,n}}$, $\frac{\partial L}{\partial \alpha_i}$ from Equation (10).

4.1.2 Gradients for NLR-MRF with GSM Expert

For the NLR-MRF model with GSM expert, we will learn the parameters $\Theta = \{\tau_i, \{\alpha_{ij}\}, \sigma_i, \lambda_i, \gamma_i\}_{i=1,\cdots,N;j=1,\cdots J}$. Similar to the computations in Section 4.1.1, if denote:

$$w_{ijp}^k = \frac{\alpha_{ij}N\left((F_i\mathbf{x}^k)_p^2; 0, \frac{\sigma_i^2}{s_j}\right)}{\sum_{l=1}^J \alpha_{il}N\left((F_i\mathbf{x}^k)_p^2; 0, \frac{\sigma_i^2}{s_l}\right)},$$
$$u_{ijp}^k = \frac{\sum_{l=1}^J \alpha_{il}N\left((F_i\mathbf{x}^k)_p^2; 0, \frac{\sigma_i^2}{s_l}\right)(s_j - s_l)}{\sum_{l=1}^J \alpha_{il}N\left((F_i\mathbf{x}^k)_p^2; 0, \frac{\sigma_i^2}{s_l}\right)},$$
$$W_{ij}^k = \text{diag}(\{\frac{s_j}{\sigma_i^2}w_{ijp}^k\}_{p=1}^M),$$
$$U_{ij}^k = \text{diag}(\{\frac{s_j}{\sigma_i^4}w_{ijp}^ku_{ijp}^k(F_i\mathbf{x}^k)_p^2\}_{p=1}^M),$$

then $\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k} = I - \sum_{i=1}^N \tau_i \sum_{j=1}^J F_i^T(W_{ij}^k - U_{ij}^k)F_i$. Second, the gradients of $g(\mathbf{x}^k;\Theta)$ w.r.t. parameters are listed as follows.

$$\frac{\partial g(\mathbf{x}^k)}{\partial \lambda_{i,m}} = \sum_{i=1}^N\sum_{j=1}^J \tau_i[(F_i^tB_m^s)^TW_{ij}^kF_i + F_i^T(W_{ij}^k - U_{ij}^k)F_i^tB_m^s]\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \gamma_{i,n}} = \sum_{i=1}^N\sum_{j=1}^J \tau_i[(B_n^tF_i^s)^TW_{ij}^kF_i + F_i^T(W_{ij}^k - U_{ij}^k)B_n^tF_n^s]\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \alpha_{ij}} = \tau_iF_i^T\text{diag}(\{\frac{1}{\sigma_i^2\alpha_{ij}}w_{ijp}^ku_{ijp}^k\}_{p=1}^M)F_i\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \sigma_i} = \sum_{j=1}^J \tau_iF_i^T(\frac{-2}{\sigma_i}W_{ij}^k + \frac{1}{\sigma_i}U_{ij}^k)F_i\mathbf{x}^k,$$
$$\frac{\partial g(\mathbf{x}^k)}{\partial \tau_i} = \sum_{j=1}^J F_i^TW_{ij}^kF_i\mathbf{x}^k.$$

Based on the above formulations, the gradients of loss function w.r.t. the model parameters can be derived by inserting the above equations into Equation (10).

4.1.3 Implementation Details

Algorithm 1 : Compute the gradients of loss function with respect to NLR-MRF model parameters

Input: Training pair $\{\mathbf{y}, \mathbf{t}\}$, current parameters Θ and the number of gradient descent steps K during inference.

Output: $\frac{\partial L}{\partial \theta}$, for any $\theta \in \Theta$.

1: **Forward inference:** Iteratively perform K steps of gradient descent procedures using the current parameters Θ, and save the inferred images $\{\mathbf{x}^t\}_{t=1,\cdots,K}$ in each step.
2: Compute $\frac{\partial L}{\partial \mathbf{x}^K}$.
3: **for** $t = K$ **to** 1 **do**
4: Compute $\frac{\partial g(\mathbf{x}^{t-1})}{\partial \theta}(\theta \in \Theta)$ and $\frac{\partial \mathbf{x}^t}{\partial \mathbf{x}^{t-1}}$.
5: $\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial \theta} - \frac{\partial L}{\partial \mathbf{x}^t}\frac{\partial g(\mathbf{x}^{t-1})}{\partial \theta}$.
6: $\frac{\partial L}{\partial \mathbf{x}^{t-1}} = \frac{\partial L}{\partial \mathbf{x}^t}\frac{\partial \mathbf{x}^t}{\partial \mathbf{x}^{t-1}}$.
7: **end for**
8: **return** $\frac{\partial L}{\partial \theta}, \theta \in \Theta$.

All of the above gradients can be efficiently computed by the convolution or per-pixel operations. First, the matrix multiplication $F\mathbf{x}$ is just the convolution of image I (the matrix form of $\mathbf{x}$) by the non-local range filter f corresponding to F. As the non-local ranges of pixels are not fixed and symmetric in the non-local range convolution, $F^T\mathbf{x}$ can not be implemented by the convolution with the mirror of filter f as in the traditional convolution. Assume that the non-local filter is with size $w \times w \times d$, then the filtered image I' corresponding to $F^T\mathbf{x}$ is computed by cumulating the weighted pixel values for all the pixels in the non-local range of each pixel (x, y): $I'(x_l + p, y_l + q) = I'(x_l+p, y_l+q) + f^{d-l+1}(w-p, w-q)I(x_l+p, y_l+q)$, for all $l \in \{1, \cdots, d\}$, $p, q \in \{0, \cdots, w-1\}$, and (x_l, y_l) is the left top position of the l-th patch in the non-local range of (x, y). Second, the diagonal matrix multiplying a vector $\mathbf{x}$ can be implemented by the per-element multiplication between its diagonal vector and $\mathbf{x}$. The iterative algorithm to compute the gradients is presented in Algorithm 1.

4.2. Training Methods

Given the gradients of loss function with respect to parameters in NLR-MRF model, we use gradient-based algorithms to discriminatively learn the parameters.

The training method for image inpainting will be discussed in the next section. For image denoising, the training set of noisy/noise-free image pairs is constructed over the 40 images as [21, 1]. To incrementally learn the NLR-MRF model, we first use stochastic gradient descent algorithm to fast train the MRF model with traditional spatial filter bank initialized randomly, and the gradients are computed by setting the cross-patch filter as a single element filter with value of one. Second, we initialize the NLR-MRF

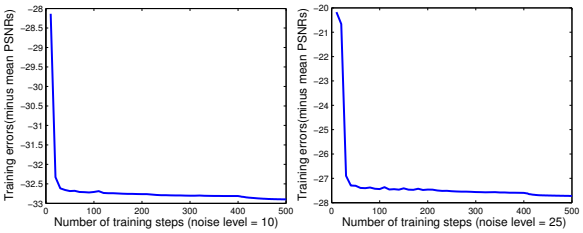


Figure 2. Training curves for the NLR-MRF model with 1 iteration and thirteen non-local filters of size $5 \times 5 \times 3$. The left and right sub-figures plot the training errors (i.e., the minus mean PSNRs over training set) with respect to the increasing training steps for noise levels of 10 and 25 respectively.

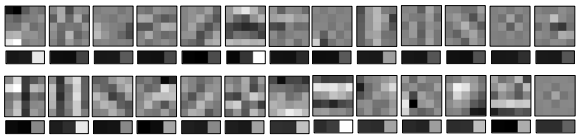


Figure 3. Learned non-local range filter banks ($5 \times 5 \times 3$). The first and second rows show the non-local range filter banks of NLR-MRF model with 3 iterations for noise levels of 10 and 25 respectively. Pairs of spatial filter and cross-patch filter are presented at top and bottom of each sub-figure.

by the above learned MRF model, and the non-local filters are initialized by the pairs of learned spatial filter and the cross-patch filter with the only non-zero element 1 in its final element. Then stochastic gradient descent algorithm is used to learn the NLR-MRF model by fast decreasing the training loss. Stochastic gradient descent provides a good estimation for model parameters with high speed. Finally, conjugate gradient descent is used to further improve the training accuracy by optimizing the model parameters. Experimentally, we use 200, 200 and 100 steps in these three training procedures to learn the optimal model parameters. About twenty hours were required to train the NLR-MRF model with 1 iteration and thirteen $5 \times 5 \times 3$ filters.

Figure 2 presents the training curves for NLR-MRF model with 1 iteration and thirteen non-local range filters of size $5 \times 5 \times 3$ (Please refer to the figure for details). In Figure 3, we present the learned non-local filter banks of NLR-MRF model with 3 iterations for noise levels of 10 and 25 respectively. Interestingly, we observe Gabor-like patterns in different orientations and frequencies in some of the learned spatial filters.

5. Results

We apply the NLR-MRF model to the applications of image denoising and inpainting. The performance for image denoising is evaluated over the 68 Berkeley test images as [21, 1] and the standard denoising test images. We present the results of MRF-based models: FoE [21], ARF [1], MRF with Bayesian minimum mean squared error estimate (MRF-MMSE) [23], and the other state-of-the art denoising methods: wavelet-based method (BLS-GSM) [18], non-local means method (NL) [5], sparse representation method (KSVD) [7] and BM3D algorithm [6]. All of the compared results come from the original papers or produced by the codes from the author's web page.

Table 1. Denoising results in average PSNRs over 68 Berkeley images. Thirteen filters of size 5×5 and $5 \times 5 \times 3$ are learned for ARF and our model. Ours-k means NLR-MRF with k iterations.

Noise levels	10	15	20	25	50	Average
FoE [21]	32.68	30.50	28.78	27.60	23.25	28.56
NL [5]	32.36	30.08	28.55	27.26	23.72	28.39
KSVD [7]	33.11	30.85	29.37	28.29	25.17	29.36
BLS-GSM [18]	33.03	30.77	29.30	28.23	25.30	29.33
BM3D [6]	33.32	31.08	29.62	28.57	25.44	29.61
ARF-1 [1]	32.65	30.52	28.89	27.75	24.57	28.88
ARF-2 [1]	32.65	30.64	29.19	28.07	24.87	29.08
ARF-3 [1]	32.75	30.70	29.26	28.14	25.10	29.19
ARF-4 [1]	32.74	30.70	29.28	28.21	25.13	29.21
Ours-1 (ST)	32.80	30.47	28.95	27.81	24.62	28.93
Ours-2 (ST)	33.03	30.78	29.21	28.13	25.16	29.26
Ours-3 (ST)	33.13	30.91	29.39	28.26	25.33	29.40
Ours-4 (ST)	33.18	30.97	29.46	28.32	25.38	29.46
Ours-1 (GSM)	32.99	30.66	29.09	27.95	24.72	29.08
Ours-2 (GSM)	33.13	30.87	29.38	28.30	25.16	29.37
Ours-3 (GSM)	33.17	30.92	29.46	28.39	25.29	29.45
Ours-4 (GSM)	33.20	30.97	29.50	28.48	25.38	29.51

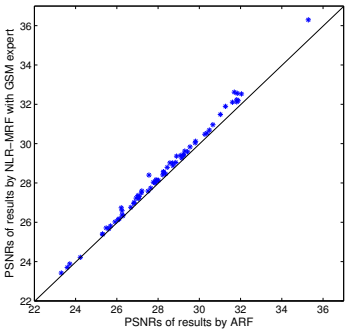


Figure 4. Scatter plots of the PSNRs over 68 Berkeley images produced by ARF and NLR-MRF with GSM expert (4 iterations). The standard deviation of noises is 25. The average PSNRs produced by ARF and NLR-MRF models are 28.21 and 28.48 respectively.

Table 1 compares the average PSNRs of denoising results produced by different algorithms over 68 Berkeley images with noise levels of 10, 15, 20, 25, 50. We present the results of NLR-MRF models with iterations 1, 2, 3, 4 and $5 \times 5 \times 3$ filters for both student-t expert and GSM expert, the filter bank with thirteen filters are learned as in [1], and more filters marginally increase the performance. First, our method significantly improves the performance of MRF-based models in denoising. The average PSNR of NLR-MRF model with 4 iterations and GSM expert is 29.51 which is better than 29.21 produced by the state-of-the-art MRF method ARF [1]. Please refer to Figure 4 for the scatter plots of per-image PSNRs. Second, the NLR-MRF model with GSM expert works better than student-t expert due to its flexibility in fitting the image statistics with more parameters. Third, compared with the other state-of-art denoising methods, our results using NLR-MRF with 4

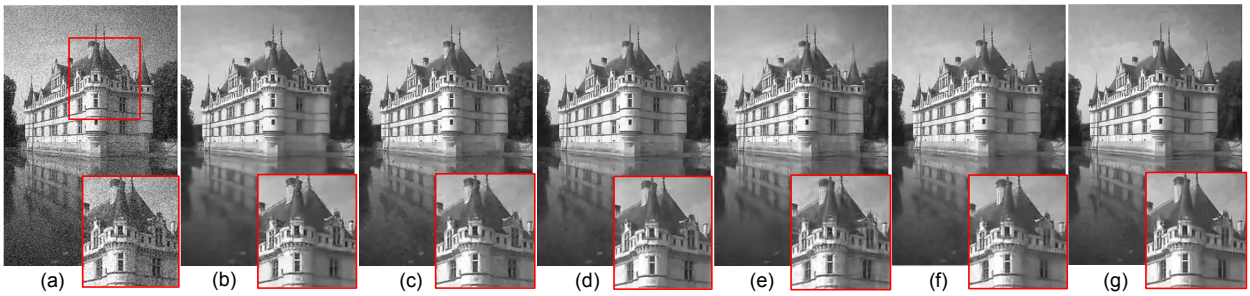


Figure 5. Denoising results comparison (Please refer to the electronic version and zoom-in for better comparison). (a) Noisy image (standard deviation of noises is 25). (b) Result of FoE (PSNR = 28.67). (c) Result of ARF (PSNR = 28.94). (d) Result of NLR-MRF (PSNR = 29.39). (e) Result of BLS-GSM (PSNR = 29.03). (f) Result of KSVD (PSNR = 29.05). (g) Result of BM3D (PSNR = 29.60).

Figure 6. Denoising results comparison (Please refer to the electronic version and zoom-in for better comparison). The standard deviation of added noises is 25. (a)-(e) show the results of Field of Experts model [21], MRF with Bayesian minimum mean squared estimation in [23] (eight 3×3 filters), active random field with 4 iterations [1] and NLR-MRF model with 4 iterations and GSM expert.

iterations and GSM expert are better than that of non-local means method, KSVD method, BLS-GSM method, and almost equal to the results of BM3D. We also trained NLR-MRF models with iterations 5, 6, 7, 8 and noise level of 25, and the test average PSNRs are 28.49, 28.51, 28.51, 28.52, and do not increase with more iterations.

Table 2. Image denoising results on standard test images (the added noise level is 25, and the results are measured by PSNR).

Noise levels	Barbara	Boats	House	Lena	Peppers	Average
NL [5]	28.71	28.10	30.32	29.95	28.45	29.11
BLS-GSM [18]	29.13	29.37	31.40	31.69	29.21	30.16
KSVD [7]	29.60	29.28	32.15	31.32	29.73	30.42
BM3D [6]	30.65	29.86	32.87	32.04	30.25	31.13
FoE [21]	27.04	28.72	31.11	30.82	29.20	29.38
ARF-4 [1]	27.59	29.14	31.18	30.87	29.54	29.66
Ours-4	28.94	29.55	32.15	31.51	30.03	30.43

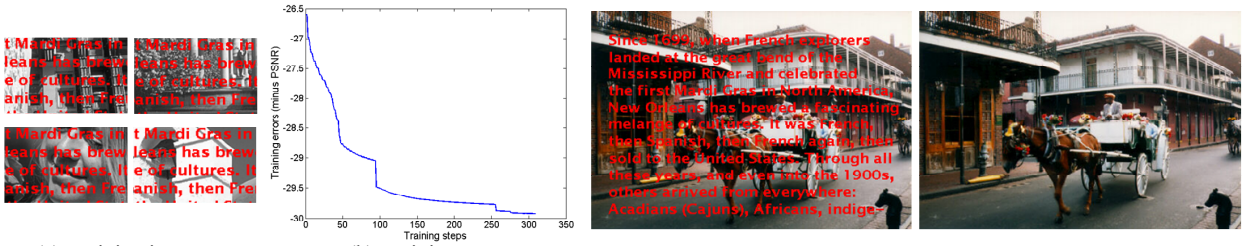
Table 3. Comparison with MRF-MMSE over 68 Berkeley images.

Methods	Expert function	Filter Bank	Mean PSNRs
MRF-MMSE [23]	GSM	3×3, 8 filters	27.95
NLR-MRF-4	GSM	$3 \times 3 \times 3$, 8 filters	28.16

Table 2 presents the results over 5 standard test images, and our method with 4 iterations, GSM expert and $5 \times 5 \times 3$ filter bank produces the second highest mean PSNRs among the state-of-the-art denoising methods, and performs much better than the other MRF-based denoising methods. In Table 3, we compare our model to MRF-MMSE with eight 3×3 filters [23]. The NLR-MRF model with eight $3 \times 3 \times 3$ non-local range filters achieves significantly higher performance which indicates the effectiveness of learning the cross-patch filters. Figure 5 and 6 present the denoising results. It is shown that NLR-MRF model well preserves image details and produces better results than the previous MRF-based models, and performs better or comparable to the other state-of-the art denoising methods.

We also apply the NLR-MRF model to image inpainting in Figure 7. We incrementally train the NLR-MRF model by the following steps: first train NLR-MRF model with 50 iterations by minimizing the cost function between the degraded images and the target images, then iteratively train one more NLR-MRF model with 5 iterations at each step using the cost function between the restored images by the already trained models and the target images, until the training accuracy does not increase. In the example of Figure 7, NLR-MRF with iterations of 50, 5, 5, 5 and GSM expert (thirteen $5 \times 5 \times 3$ filters) are trained from the training set. By applying these four models successively on each channel (in RGB space) of test image in JPEG version [3, 21], the PSNR of inpainting result is 32.59. It is higher than the result by FoE model (32.22 dB in YCbCr space and 32.39 dB in RGB space), the result by running the code of MRF-MMSE method [23] (31.69 dB), and the result using the color version KSVD model [16] (32.45 dB).

The two computational steps for image restoration using NLR-MRF model are computing the non-local range by block matching and performing fixed number of gradient descent steps for inference. Our implementation combing C++ and matlab takes 37 seconds for block matching in



(a) Training images (b) Training curve (c) Test image (d) Inpainting result (PSNR = 32.59)

Figure 7. Learning NLR-MRF model ($5 \times 5 \times 3$ filters) for inpainting. Sixty-five steps of gradient descent in inference are learned to infer the results. The PSNR of the restored image is 32.59. It is higher than 32.22 (in YCbCr space) and 32.39 (in RGB space) by FoE model [21], 31.69 by running the codes of MRF-MMSE [23], and 32.45 by the color version KSVD algorithm [16].

31×31 search windows and 14 seconds for four iterations of gradient descent in inference, when thirteen $5 \times 5 \times 3$ non-local filter bank and GSM expert are utilized and the image size is 481×231 on a platform of Intel 3.2G CPU.

6. Conclusion

In this work, we proposed a novel framework of non-local range MRF model and a gradient-based discriminative learning method to train the model. By extending the traditional MRF with local clique to be a non-local range random field, both the local structures and dependencies among similar patches in natural images are learned by non-local range filters, which enable significant improvements in image restoration over the traditional MRF models.

The non-local range MRF model captures the advantages of BM3D and non-local means in a flexible MRF framework. It can be used as an image prior for the other low level vision problems besides denoising and inpainting, e.g., demosaicing, super-resolution, which deserve further investigations. We are also interested in extending the NLR-MRF model to color image and video restoration, then the non-local range filter will be defined over the multiple channels of color image or multiple frames in video.

Acknowledgement

Jian Sun was supported by NSF (No. 61003144) and 973 program (No. 2007CB311002) of China. Marshall F. Tappen was supported by NSF grants IIS-0905387, IIS-0916868, and a grant under the NGA NURI program.

References

[1] A. Barbu. Training an active random field for real-time image denoising. *IEEE Trans. on Image Processing*, 18(11):2451–2462, 2009.

[2] A. Barbu and S. C. Zhu. Generalizing swendsen-wang to sampling arbitrary posterior probabilities. *IEEE Trans. on PAMI*, 27(8):1239–1253, 2005.

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

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on PAMI*, 23(11):1222–1239, 2001.

[5] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. In *CVPR*, 2005.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. on Image Processing*, 16(8):2080–2095, 2007.

[7] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. on Image Processing*, 15(12):3736–3745, 2006.

[8] A. Elmoataz, O. Lezoray, and S. Bougleux. Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing. *IEEE Trans. on Image Processing*, 17(7):1047–1060, 2008.

[9] G. E. Hinton. Products of experts. In *Int. Conf. on Artificial Neural Networks (ICANN)*, volume 1, pages 1–6, 1999.

[10] P. Kohli and M. P. Kumar. Energy minimization for linear envelop mrfs. In *CVPR*, 2010.

[11] X. Lan, S. Roth, D. P. Huttenlocher, and M. J. Black. Efficient belief propagation with learned higher-order markov random fields. In *ECCV*, 2006.

[12] Y. LeCun and F. Huang. Loss functions for discriminative training of energy-based models. In *AIStats*, 2005.

[13] Y. P. Li and D. P. Huttenlocher. Sparse long-range random field and its application to image denoising. In *ECCV*, 2008.

[14] S. Lyu and E. P. Simoncelli. Modeling multiscale subbands of photographic images with fields of gaussian scale mixtures. *IEEE Trans. on PAMI*, 31(4):693–706, 2009.

[15] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *CVPR*, 2009.

[16] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17(1):53–69, 2008.

[17] J. J. McAuley, T. S. Caetano, A. J. Smola, and M. O. Franz. Learning high-order mrf priors of color images. In *ICML*, 2006.

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

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

[20] S. Roth and M. J. Black. Steerable random field. In *ICCV*, 2007.

[21] S. Roth and M. J. Black. Fields of experts. *International Journal of Computer Vision*, 82(2):205–229, 2009.

[22] K. G. G. Samuel and M. F. Tappen. Learning optimized map estimates in continuously-valued mrf models. In *CVPR*, 2009.

[23] U. Schmidt, Q. Gao, and S. Roth. A generative perspective on mrfs in low-level vision. In *CVPR*, 2010.

[24] 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, 2003.

[25] J. Sun, H. Y. Shum, and N. N. Zheng. Stereo matching using belief propagation. *IEEE Trans. on PAMI*, 25(7):787–800, 2003.

[26] M. F. Tappen. Utilizing variational optimization to learn markov random fields. In *CVPR*, 2007.

[27] M. F. Tappen, C. Liu, E. H. Adelson, and W. T. Freeman. Learning gaussian conditional random fields for low-level vision. In *CVPR*, 2007.

[28] Y. Weiss and W. T. Freeman. What makes a good model of natural images? In *CVPR*, 2007.

[29] M. Werlberger, T. Pock, and H. Bischof. Motion estimation with non-local total variation regularization. In *CVPR*, 2010.

[30] O. J. Woodford, C. Rother, and V. Kolmogorov. A global perspective on map inference for low-level vision. In *ICCV*, 2009.

[31] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang. Non-local kernel regression for image and video restoration. In *ECCV*, 2010.

[32] S. C. Zhu and D. Mumford. Prior learning and gibbs reaction-diffusion. *IEEE Trans. PAMI*, 19(11):1236–1250, 1997.