Learning Non-Local Range Markov Random Field for Image Restoration Supplementary Material

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

The supplementary materials are organized as follows. First, the computation procedures of gradients for Maximum A-Posteriori (MAP) inference and training of the Non-Local Range Markov Random Field (NLR-MRF) model are presented. Second, more results and comparisons are presented for image denoising.

1. Gradients for MAP Inference of NLR-MRF Model

In this section, we present how to compute the gradients when applying NLR-MRF model to infer the restored image using fixed number of gradient-descent procedures. For the convenience of computations, we use the matrix operation to substitute the convolution operation in the original formulation of NLR-MRF.

When applying the proposed MRF prior to image restoration, the Maximum A-Posteriori (MAP) estimation of the restored image can be derived by minimizing the following energies:

$$
\begin{aligned}
E(\mathbf{x};\Theta) &= \sum_p \sum_{i=1}^{N} \alpha_i \log(1+\frac{1}{2}(F_i\mathbf{x})_p^2), \\
E(\mathbf{x};\Theta) &= -\sum_p \sum_{i=1}^{N} \tau_i \log(\sum_{j=1}^{J} \alpha_{ij} N((F_i\mathbf{x})_p^2; 0, \sigma_i^2/s_j)),
\end{aligned}
$$

for student-t (ST) expert and Gaussian scale mixture (GSM) expert respectively, $N(\cdot)$ is the Gaussian function. We minimize these energies by gradient descent procedures, and the gradients can be computed as follows.

$$
\begin{aligned}
\frac{\partial E(\mathbf{x};\Theta)}{\partial \mathbf{x}} &= \sum_p \sum_{i=1}^{N} \alpha_i \frac{(F_i\mathbf{x})_p (F_i)_p}{(1+\frac{1}{2}(F_i\mathbf{x})_p^2)} \\
&= \sum_{i=1}^{N} \alpha_i \sum_p (F_i\mathbf{x})_p \frac{1}{1+\frac{1}{2}(F_i\mathbf{x}_p)^2} (F_i)_p \\
&= \sum_{i=1}^{N} \alpha_i \sum_p (F_i\mathbf{x})_p w_{ip} (F_i)_p \\
&= \sum_{i=1}^{N} \alpha_i F_i^T W_i F_i \mathbf{x}, \qquad (1)
\end{aligned}
$$

for student-t expert, $(F_i)_p$ denotes the p-th row of matrix F_i and

$$
W_i = \text{diag}(\{\frac{1}{1+\frac{1}{2}(F_i\mathbf{x})_p^2}\}_{p=1}^{M}). \qquad (2)
$$

For NLR-MRF model with Gaussian scale mixture expert,

$$\begin{aligned}\frac{\partial E(\mathbf{x};\Theta)}{\partial \mathbf{x}} &= -\sum_p\sum_{i=1}^N \tau_i \frac{\sum_{j=1}^J \alpha_{ij} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_j})\frac{-s_j}{\sigma_i^2}(F_i\mathbf{x})_p (F_i)_p}{\sum_{l=1}^J \alpha_{il} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_l})} \\ &= \sum_{i=1}^N \tau_i \sum_{j=1}^J \sum_p \frac{\alpha_{ij} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_j})\frac{s_j}{\sigma_i^2}}{\sum_{l=1}^J \alpha_{il} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_l})}(F_i\mathbf{x})_p (F_i)_p \\ &= \sum_{i=1}^N \tau_i \sum_{j=1}^J \sum_p w_{ijp}(F_i\mathbf{x})_p (F_i)_p \\ &= \sum_{i=1}^N \tau_i \sum_{j=1}^J F_i^T W_{ij} F_i \mathbf{x}, \end{aligned} \tag{3}$$

where $W_{ij} = \text{diag}(\{\frac{\alpha_{ij} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_j})\frac{s_j}{\sigma_i^2}}{\sum_{l=1}^J \alpha_{il} N((F_i\mathbf{x})_p^2; 0, \frac{\sigma_i^2}{s_l})}\}_{p=1}^M)$.

2. Gradients for Training NLR-MRF Model

In this section, we present the computations of gradients used in learning the parameters Θ in NLR-MRF model. Given the gradients of cost function with respect to the model parameters, the involved parameters can be learned using gradient-based optimization algorithm.

We first present the general framework for computing the gradients of cost function with respect to the parameters of NLR-MRF model with general expert function. Then specify these gradients for NLR-MRF model with two typical expert functions, i.e., student-t expert function and GSM expert function.

2.0.1 General Framework for Gradients Computation

The parameters in NLR-MRF model are discriminatively learned by optimizing the following problem:

$$\begin{aligned}\Theta^* &= \text{argmin}_\Theta L(\mathbf{x}^K(\Theta), \mathbf{t}) \\ &\quad \text{where } \mathbf{x}^K(\Theta) = \text{GradDesc}_K\{E(\mathbf{x}, \Theta)\}, \end{aligned} \tag{4}$$

where GradDesc_K means K steps of gradient descent procedures to minimize $E(\mathbf{x}, \Theta)$:

$$\mathbf{x}^k = \mathbf{x}^{k-1} - g(\mathbf{x}^{k-1}; \Theta), \tag{5}$$

where $g(\mathbf{x}^{k-1}; \Theta) = \frac{\partial E(\mathbf{x}^{k-1};\Theta)}{\partial \mathbf{x}^{k-1}}$, $k = 1, \cdots, K$ and $\mathbf{x}^0$ is the degraded image $\mathbf{y}$.

The gradient of loss function with respect to any parameter $\theta \in \Theta$ in the NLR-MRF model can be computed as

$$\frac{\partial L(\mathbf{x}^K, \mathbf{t})}{\partial \theta} = \frac{\partial L}{\partial \mathbf{x}^K}\frac{\partial \mathbf{x}^K}{\partial \theta}. \tag{6}$$

If the cost function L is defined as the minus PSNR value between the restored image $\mathbf{x}^K$ and the target image $\mathbf{t}$:

$$L(\mathbf{x}^K, \mathbf{t}) = -20\log_{10}\frac{255}{\sqrt{\frac{1}{M}||\mathbf{x}^K - \mathbf{y} - \text{mean}(\mathbf{x}^K - \mathbf{y})||^2}}, \tag{7}$$

where M is the number of pixels in image. $\frac{\partial L}{\partial \mathbf{x}^K}$ can be easily derived as:

$$\frac{\partial L(\mathbf{x}^K, \mathbf{t})}{\partial \mathbf{x}^K} = -\frac{20}{\ln 10}\frac{(\mathbf{x}^K - \mathbf{y} - \text{mean}(\mathbf{x}^K - \mathbf{y}))(1 - \frac{1}{M})}{||\mathbf{x}^K - \mathbf{y} - \text{mean}(\mathbf{x}^K - \mathbf{y})||^2}. \tag{8}$$

$\frac{\partial \mathbf{x}^K}{\partial \theta}$ can be iteratively computed from the final iteration K to the first iteration in Equation (5) . In the final iteration K, according to the iteration formula in Equation (5), the gradient of $\mathbf{x}^K$ w.r.t. $\theta \in \Theta$ is computed as

$$\frac{\partial \mathbf{x}^K}{\partial \theta} = \frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^{K-1}} \frac{\partial \mathbf{x}^{K-1}}{\partial \theta} - \frac{\partial g(\mathbf{x}^{K-1}; \Theta)}{\partial \theta}, \tag{9}$$

in which $\mathbf{x}^{K-1}$ is also dependent on the parameter θ. $\frac{\partial \mathbf{x}^{K-1}}{\partial \theta}$ is similarly computed based on the previous iteration of gradient descent, and inserted into Equation (9). After iterating this procedure in K steps and utilizing the fact that $\frac{\partial \mathbf{x}^0}{\partial \theta} = 0$, the gradient $\frac{\partial \mathbf{x}^K}{\partial \theta}$ can be computed as

$$\begin{aligned}
\frac{\partial \mathbf{x}^K}{\partial \theta} &= \frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^{K-1}} \frac{\partial \mathbf{x}^{K-1}}{\partial \theta} - \frac{\partial g(\mathbf{x}^{K-1}; \Theta)}{\partial \theta} \\
&= \frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^{K-1}} \left(\frac{\partial \mathbf{x}^{K-1}}{\partial \mathbf{x}^{K-2}} \frac{\partial \mathbf{x}^{K-2}}{\partial \theta} - \frac{\partial g(\mathbf{x}^{K-2}; \Theta)}{\partial \theta} \right) - \frac{\partial g(\mathbf{x}^{K-1}; \Theta)}{\partial \theta} \\
&\cdots \\
&= -\sum_{k=1}^{K} \frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^k} \frac{\partial g(\mathbf{x}^{k-1}; \Theta)}{\partial \theta},
\end{aligned}$$

where $\frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^k} = \prod_{t=k}^{K-1} \frac{\partial \mathbf{x}^{t+1}}{\partial \mathbf{x}^t}, (k = 1, \cdots, K-1)$ and $\frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^K} = I$. In summary, the gradient of $L(\mathbf{x}^K, \mathbf{t})$ w.r.t. θ is

$$\begin{aligned}
\frac{\partial L(\mathbf{x}^K, \mathbf{t})}{\partial \theta} &= -\sum_{k=1}^{K} \frac{\partial L}{\partial \mathbf{x}^K} \frac{\partial \mathbf{x}^K}{\partial \mathbf{x}^k} \frac{\partial g(\mathbf{x}^{k-1}; \Theta)}{\partial \theta} \\
&= -\sum_{k=1}^{K} \frac{\partial L}{\partial \mathbf{x}^k} \frac{\partial g(\mathbf{x}^{k-1}; \Theta)}{\partial \theta}.
\end{aligned} \tag{10}$$

2.0.2 Gradients for NLR-MRF Model with Student-T Expert

We now present how to compute $\frac{\partial L}{\partial \mathbf{x}^k}$ and $\frac{\partial g(\mathbf{x}^{k-1};\Theta)}{\partial \theta}$ in Equation (10) for the NLR-MRF model with student-t expert. Based on the Equations (1) and (5), $\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k}$ $(k = 0, \cdots, K-1)$ can be computed as

$$\begin{aligned}
\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k} &= I - \sum_{i=1}^{N} (\alpha_i F_i^T W_i^k F_i + \alpha_i F_i^T \frac{\partial W_i^k}{\partial \mathbf{x}^k} F_i \mathbf{x}^k) \\
&= I - \sum_{i=1}^{N} (\alpha_i F_i^T W_i^k F_i + \alpha_i F_i^T \text{diag}(F_i \mathbf{x}^k) \frac{\partial \overrightarrow{W_i^k}}{\partial \mathbf{x}^k}) \\
&= I - \sum_{i=1}^{N} (\alpha_i F_i^T W_i^k F_i + \alpha_i F_i^T \text{diag}(F_i \mathbf{x}^k) \text{diag}(\{ \frac{-(F_i \mathbf{x}^k)_p}{[1 + \frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2} \}_{p=1}^{M}) F_i) \\
&= I - \sum_{i=1}^{N} \alpha_i F_i^T (W_i^k - U_i^k) F_i,
\end{aligned} \tag{11}$$

where W_i^k is defined as in Equation (2) for $\mathbf{x}^k$, and $U_i^k = \text{diag}(\{ \frac{(F_i \mathbf{x}^k)_p^2}{[1+\frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2} \}_{p=1}^{M})$ is also a diagonal matrix, and $\overrightarrow{W_i^k}$ is the vector of the diagonal values of W_i^k. The second equality in Equation (11) holds for the fact that $W_i^k (F_i \mathbf{x}^k) = \text{diag}(F_i \mathbf{x}^k) \overrightarrow{W_i^k}$, and the third equality holds due to

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

Then we compute $\frac{\partial g(\mathbf{x}^k;\Theta)}{\partial \theta}$ for any parameter $\theta \in \Theta$. For NLR-MRF model with student-t expert, the involved parameters are $\Theta = \{\lambda_i, \gamma_i, \alpha_i\}_{i=1}^N$, where α_i is the parameter of student-t distribution, and λ_i, γ_i are the coefficients for spatial filter and cross-patch filter, i.e., $F_i^s = \sum_{m=1}^{N_s} \lambda_{i,m} B_m^s,\ F_i^t = \sum_{n=1}^{N_t} \gamma_{i,n} B_n^t$ and $F_i = F_i^t F_i^s$. It is easy to derive that

$$\frac{\partial g(\mathbf{x}^k, \Theta)}{\partial \alpha_i} = F_i^T W_i^k F_i \mathbf{x}^k.$$

The gradients of $g(\mathbf{x}^k, \Theta)$ w.r.t. filter coefficients are

$$\begin{aligned}
& \frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \lambda_{i,m}} \\
= & \frac{\partial (\sum_{i=1}^N \alpha_i F_i^T W_i^k F_i \mathbf{x}^k)}{\partial \lambda_{i,m}} \\
= & \sum_{i=1}^N \alpha_i [\frac{\partial F_i^T}{\partial \lambda_{i,m}} W_i^k F_i + F_i^T \frac{\partial W_i^k}{\partial \lambda_{i,m}} F_i + F_i^T W_i^k \frac{\partial F_i}{\partial \lambda_{i,m}}] \mathbf{x}^k \\
= & \sum_{i=1}^N \alpha_i [(F_i^t B_m^s)^T W_i^k F_i + F_i^T \frac{\partial W_i^k}{\partial \lambda_{i,m}} F_i + F_i^T W_i^k F_i^t B_m^s] \mathbf{x}^k \\
= & \sum_{i=1}^N \alpha_i [(F_i^t B_m^s)^T W_i^k F_i + F_i^T (W_i^k - U_i^k) F_i^t B_m^s] \mathbf{x}^k,
\end{aligned}$$

where the final equality is true because

$$\begin{aligned}
& \sum_{i=1}^N \alpha_i F_i^T \frac{\partial W_i^k}{\partial \lambda_{i,m}} F_i \mathbf{x}^k \\
= & -\sum_{i=1}^N \alpha_i F_i^T \text{diag}(\{\frac{(F_i \mathbf{x}^k)_p (F_i^t B_m^s \mathbf{x}^k)_p}{[1 + \frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2}\}_{p=1}^P) F_i \mathbf{x}^k \\
= & -\sum_{i=1}^N \alpha_i F_i^T \text{diag}(\{\frac{(F_i \mathbf{x}^k)_p}{[1 + \frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2}\}_p) \text{diag}(F_i^t B_m^s \mathbf{x}^k) F_i \mathbf{x}^k \\
= & -\sum_{i=1}^N \alpha_i F_i^T \text{diag}(\{\frac{(F_i \mathbf{x}^k)_p}{[1 + \frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2}\}_p) \text{diag}(F_i \mathbf{x}^k) F_i^t B_m^s \mathbf{x}^k \\
= & -\sum_{i=1}^N \alpha_i F_i^T \text{diag}(\{\frac{(F_i \mathbf{x}^k)_p (F_i \mathbf{x}^k)_p}{[1 + \frac{1}{2}(F_i \mathbf{x}^k)_p^2]^2}\}_p) F_i^t B_m^s \mathbf{x}^k \\
= & -\sum_{i=1}^N \alpha_i F_i^T U_i^k F_i^t B_m^s \mathbf{x}^k.
\end{aligned}$$

$\frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \gamma_{i,n}}$ can be computed similar to the above computations:

$$\frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \gamma_{i,n}} = \sum_{i=1}^N \alpha_i [(B_n^t F_i^s)^T W_i^k F_i + F_i^T (W_i^k - U_i^k) B_n^t F_i^s] \mathbf{x}^k.$$

Given the above computations, we can derive the gradients: $\frac{\partial L}{\partial \lambda_{i,n}}, \frac{\partial L}{\partial \gamma_{i,m}}, \frac{\partial L}{\partial \alpha_i}$ from Equation (10).

2.0.3 Gradients for NLR-MRF Model with GSM Expert

We now compute the gradients of cost function with respect to all the parameters in NLR-MRF model with Gaussian scale mixture expert. To make the computations more clear, we denote

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

Based on the Equations (3) and (5), $\frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k}$ can be computed as

$$\begin{aligned} \frac{\partial \mathbf{x}^{k+1}}{\partial \mathbf{x}^k} &= I - \sum_{i=1}^N \tau_i \sum_{j=1}^J (F_i^T W_{ij}^k F_i + F_i^T \frac{\partial W_{ij}^k}{\partial \mathbf{x}^k} F_i \mathbf{x}^k) \\ &= I - \sum_{i=1}^N \tau_i \sum_{j=1}^J (F_i^T W_{ij}^k F_i + F_i^T \text{diag}(F_i \mathbf{x}^k) \frac{\partial \overrightarrow{W_{ij}^k}}{\partial \mathbf{x}^k}) \\ &= I - \sum_{i=1}^N \tau_i \sum_{j=1}^J (F_i^T W_{ij}^k F_i + F_i^T \text{diag}(F_i \mathbf{x}^k) \text{diag}(\{-\frac{s_j}{\sigma_i^4} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p\}_{p=1}^M) F_i) \\ &= I - \sum_{i=1}^N \tau_i \sum_{j=1}^J F_i^T (W_{ij}^k - U_{ij}^k) F_i, \end{aligned} \tag{12}$$

where $\overrightarrow{W_{ij}^k}$ denotes the diagonal vector of W_{ij}^k, and the third equality holds because

$$\frac{\partial \overrightarrow{W_{ij}^k}}{\partial \mathbf{x}^k} = \text{diag}(\{-\frac{s_j}{\sigma_i^4} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p\}_{p=1}^M) F_i. \tag{13}$$

which can be computed by calculus.

Then we compute the gradient of $g(\mathbf{x}^k; \Theta)$ $(k = 1, \cdots, K)$ with respect to all the model parameters, i.e.,

$$\Theta = \{\tau_i, \{\alpha_{ij}\}, \sigma_i, \lambda_i, \gamma_i\}_{i=1,\cdots,N; j=1,\cdots J}.$$

First, the gradient of $g(\mathbf{x}^k; \Theta)$ w.r.t. filter coefficients are computed as

$$\begin{aligned} & \frac{\partial g(\mathbf{x}^k, \Theta)}{\partial \lambda_{i,m}} \\ = & \frac{\partial (\sum_{i=1}^N \tau_i \sum_{j=1}^J F_i^T W_{ij}^k F_i \mathbf{x}^k)}{\partial \lambda_{i,m}} \\ = & \sum_{i=1}^N \tau_i \sum_{j=1}^J (\frac{\partial F_i^T}{\partial \lambda_{i,m}} W_{ij}^k F_i \mathbf{x}^k + F_i^T \frac{\partial W_{ij}^k}{\partial \lambda_{i,m}} F_i \mathbf{x}^k + F_i^T W_{ij}^k \frac{\partial F_i}{\partial \lambda_{i,m}} \mathbf{x}^k) \\ = & \sum_{i=1}^N \tau_i \sum_{j=1}^J ((F_i^t B_m^s)^T W_{ij}^k F_i \mathbf{x}^k + F_i^T \frac{\partial W_{ij}^k}{\partial \lambda_{i,m}} F_i \mathbf{x}^k + F_i^T W_{ij}^k F_i^t B_m^s \mathbf{x}^k), \end{aligned} \tag{14}$$

where the second term in the above formula can be computed as:

$$
\begin{aligned}
F_i^T \frac{\partial W_{ij}^k}{\partial \lambda_{i,m}} F_i \mathbf{x}^k &= F_i^T \text{diag}(\{-\frac{s_j}{\sigma_i^4} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p (F_i^t B_m^s \mathbf{x}^k)_p\}_{p=1}^M) F_i \mathbf{x}^k \\
&= F_i^T \text{diag}(\{-\frac{s_j}{\sigma_i^4} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p (F_i \mathbf{x}^k)_p\}_{p=1}^M) F_i^t B_m^s \mathbf{x}^k \\
&= -F_i^T U_{ij}^k F_i^t B_m^s \mathbf{x}^k.
\end{aligned}
$$

Therefore,

$$
\frac{\partial g(\mathbf{x}^k, \Theta)}{\partial \lambda_{i,m}} = \sum_{i=1}^N \sum_{j=1}^J \tau_i [(F_i^t B_m^s)^T W_{ij}^k F_i + F_i^T (W_{ij}^k - U_{ij}^k) F_i^t B_m^s] \mathbf{x}^k.
$$

We can similarly derive that

$$
\frac{\partial g(\mathbf{x}^k, \Theta)}{\partial \gamma_{i,n}} = \sum_{i=1}^N \sum_{j=1}^J \tau_i [(B_n^t F_i^s)^T W_{ij}^k F_i + F_i^T (W_{ij}^k - U_{ij}^k) B_n^t F_n^s] \mathbf{x}^k.
$$

Second, the gradient of cost function w.r.t. the coefficient of j-th Gaussian component in the GSM model for i-th filter's responses is computed as

$$
\frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \alpha_{ij}} = \tau_i F_i^T \frac{\partial W_{ij}^k}{\partial \alpha_{ij}} F_i \mathbf{x}^k + \tau_i \sum_{l=1, l \neq j}^J F_i^T \frac{\partial W_{il}^k}{\partial \alpha_{ij}} F_i \mathbf{x}^k,
$$

where

$$
\begin{aligned}
\frac{\partial W_{ij}^k}{\partial \alpha_{ij}} &= \text{diag}(\{\frac{s_j}{\sigma_i^2} \frac{\partial}{\partial \alpha_{ij}} (\frac{\alpha_{ij} N((F_i \mathbf{x}^k)_p^2, 0, \sigma_i^2/s_j)}{\sum_{l=1}^J \alpha_{il} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_l)})\}_{p=1}^M) \\
&= \text{diag}(\{\frac{s_j w_{ijp}^k}{\sigma_i^2 \alpha_{ij}} (1 - w_{ijp}^k)\}_{p=1}^M),
\end{aligned}
$$

and

$$
\begin{aligned}
\frac{\partial W_{il(l \neq j)}^k}{\partial \alpha_{ij}} &= \text{diag}(\{\frac{s_l}{\sigma_i^2} \frac{\partial}{\partial \alpha_{ij}} (\frac{\alpha_{il} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_l)}{\sum_{q=1}^J \alpha_{iq} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_q)})\}_{p=1}^M) \\
&= \text{diag}(\{-\frac{s_l w_{ilp}^k w_{ijp}^k}{\sigma_i^2 \alpha_{ij}}\}_{p=1}^M).
\end{aligned}
$$

Therefore, $\frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \alpha_{ij}}$ can be further computed as

$$
\begin{aligned}
\frac{\partial g(\mathbf{x}^k; \Theta)}{\partial \alpha_{ij}} &= \tau_i F_i^T \text{diag}(\{\frac{s_j w_{ijp}^k}{\sigma_i^2 \alpha_{ij}} (1 - w_{ijp}^k)\}_{p=1}^M) F_i \mathbf{x}^k + \tau_i \sum_{l=1, l \neq j}^J F_i^T \text{diag}(\{-\frac{s_l w_{ilp}^k w_{ijp}^k}{\sigma_i^2 \alpha_{ij}}\}_{p=1}^M) F_i \mathbf{x}^k \\
&= \tau_i F_i^T \text{diag}(\{\frac{w_{ijp}^k}{\sigma_i^2 \alpha_{ij}} (s_j - \sum_{l=1}^J w_{ilp}^k s_l)\}_{p=1}^M) F_i \mathbf{x}^k \\
&= \tau_i F_i^T \text{diag}(\{\frac{w_{ijp}^k}{\sigma_i^2 \alpha_{ij}} (\sum_{l=1}^J w_{ilp}^k (s_j - s_l))\}_{p=1}^M) F_i \mathbf{x}^k \\
&= \tau_i F_i^T \text{diag}(\{\frac{1}{\sigma_i^2 \alpha_{ij}} w_{ijp}^k u_{ijp}^k\}_{p=1}^M) F_i \mathbf{x}^k.
\end{aligned}
$$

Third, the gradient of cost function w.r.t. the Gaussian base variance σ_i is computed as

$$\begin{aligned}
&\frac{\partial g(\mathbf{x}^k;\Theta)}{\partial \sigma_i} \\
= &\sum_{j=1}^{J} \tau_i F_i^T \mathrm{diag}(\{\frac{\partial (w_{ijp}^k \frac{s_j}{\sigma_i^2})}{\partial \sigma_i}\}_{p=1}^M) F_i \mathbf{x}^k \\
= &\sum_{j=1}^{J} \tau_i F_i^T \mathrm{diag}(\{\frac{-2s_j}{\sigma_i^3} w_{ijp}^k + \frac{s_j}{\sigma_i^5} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p^2\}_{p=1}^M) F_i \mathbf{x}^k \\
= &\sum_{j=1}^{J} \tau_i F_i^T (\frac{-2}{\sigma_i} W_{ij}^k + \frac{1}{\sigma_i} U_{ij}^k) F_i \mathbf{x}^k.
\end{aligned}$$

The second equality holds because

$$\begin{aligned}
&\frac{\partial (w_{ijp}^k \frac{s_j}{\sigma_i^2})}{\partial \sigma_i} \\
= &\frac{\partial}{\partial \sigma_i}(\frac{s_j}{\sigma_i^2} \frac{\alpha_{ij} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_j)}{\sum_{l=1}^J \alpha_{il} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_l)}) \\
= &\frac{s_j}{\sigma_i^2}(\frac{-2}{\sigma_i} w_{ijp}^k + \frac{\partial}{\partial \sigma_i}(\frac{\alpha_{ij} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_j)}{\sum_{l=1}^J \alpha_{il} N((F_i \mathbf{x}^k)_p^2; 0, \sigma_i^2/s_l)})) \\
= &\frac{s_j}{\sigma_i^2}(\frac{-2}{\sigma_i} w_{ijp}^k + \sigma_i^{-3} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p^2) \\
= &\frac{-2s_j}{\sigma_i^3} w_{ijp}^k + \frac{s_j}{\sigma_i^5} w_{ijp}^k u_{ijp}^k (F_i \mathbf{x}^k)_p^2.
\end{aligned}$$

Finally, it is easy to derive the gradient of cost function w.r.t. the parameter τ_i:

$$\frac{\partial g(\mathbf{x}^k;\Theta)}{\partial \tau_i} = \sum_{j=1}^{J} F_i^T W_{ij}^k F_i \mathbf{x}^k.$$

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

3. More Results

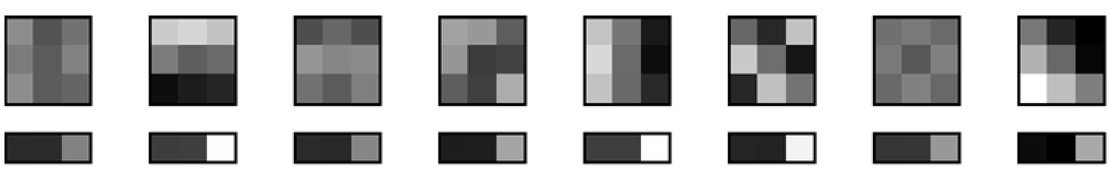


Figure 1. $3 \times 3 \times 3$ non-local range filter bank learned for NLR-MRF model with GSM expert and 4 iterations (the standard deviation of noise is 25) . Spatial filter and cross-patch filter are presented at top and bottom of each sub-figure.

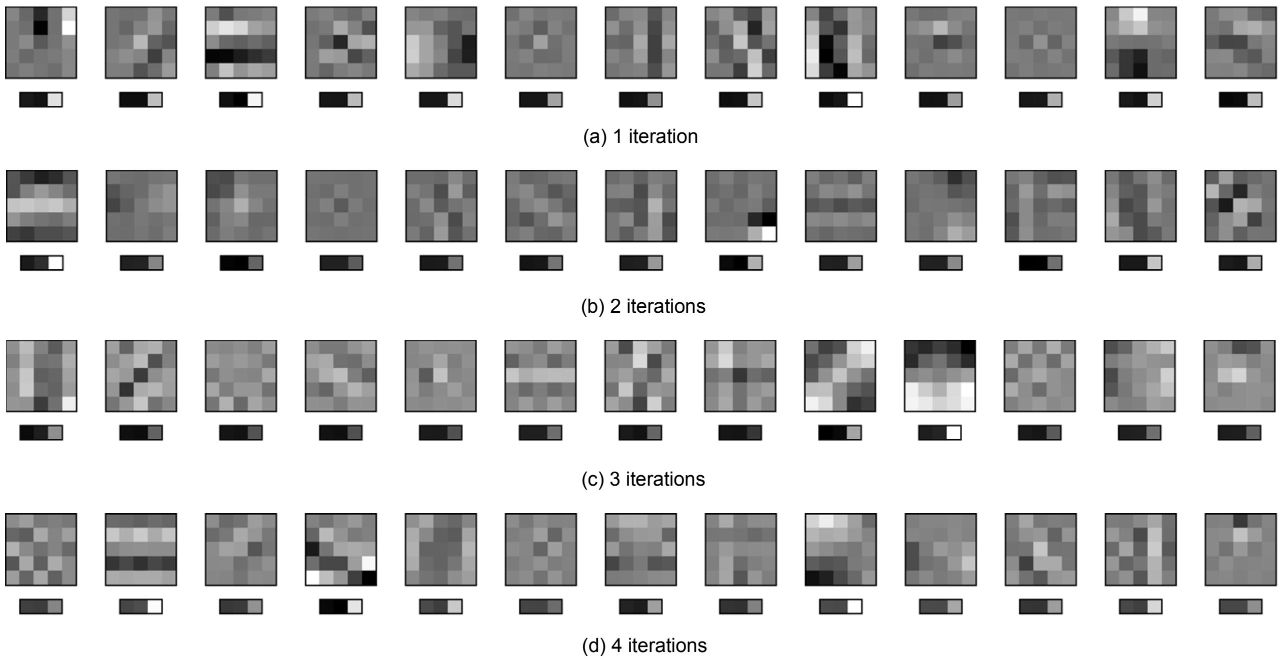


Figure 2. $5 \times 5 \times 3$ non-local range filter bank learned for NLR-MRF model with GSM expert (the standard deviation of noise is 15). Spatial filter and cross-patch filter are presented at top and bottom of each sub-figure.

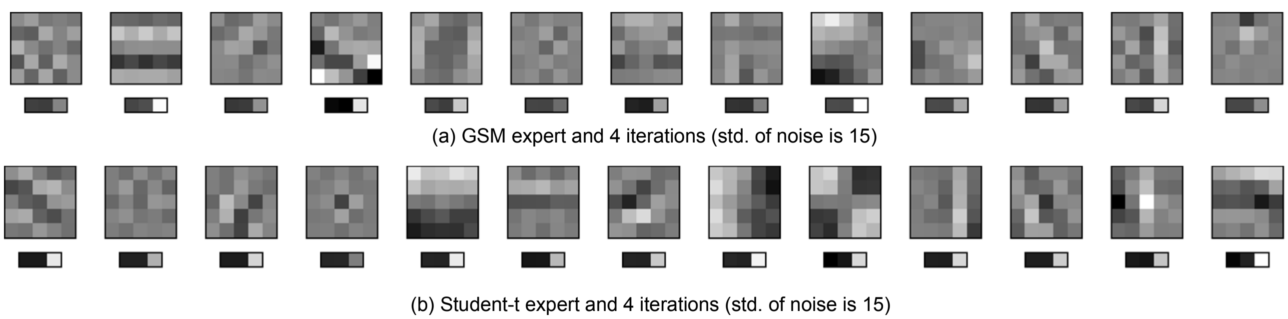


Figure 3. $5 \times 5 \times 3$ non-local range filter bank learned for NLR-MRF model with GSM/student-t expert and 4 iterations (the standard deviation of noise is 15). Spatial filter and cross-patch filter are presented at top and bottom of each sub-figure.

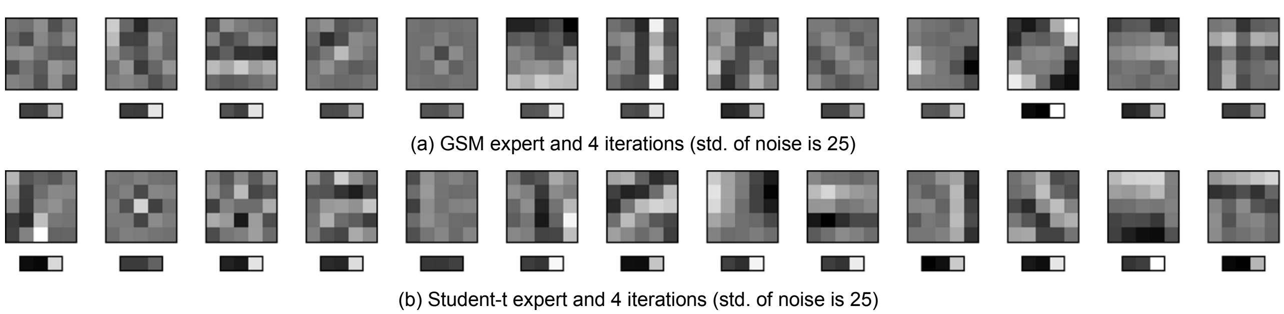
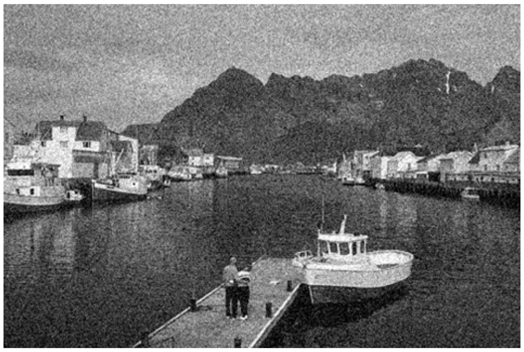


Figure 4. $5 \times 5 \times 3$ non-local range filter bank learned for NLR-MRF model with GSM/student-t expert and 4 iterations (the standard deviation of noise is 25). Spatial filter and cross-patch filter are presented at top and bottom of each sub-figure.



Noisy image

Original image

FoE with 5 × 5 filter bank
(PSNR = 28.22)

MRF-MMSE with 3 × 3 filter bank
(PSNR = 28.51)

ARF with 5 × 5 filter bank
(PSNR = 28.67)

NLR-MRF with 3 × 3 ×3 filter bank and GSM expert
(PSNR = 28.67)

NLR-MRF with 5 × 5 ×3 filter bank and student-t expert
(PSNR = 28.81)

NLR-MRF with 5 × 5 ×3 filter bank and GSM expert
(PSNR = 28.98)

Figure 5. FoE: field of experts model [2]; ARF: active random field [1]; MRF-MMSE: the MRF-based method in [3]. The standard deviation of noise is 25. In ARF and NLR-MRF methods, four iterations of gradient descent procedures are used to infer the noise-free images.

Noisy image

Original image

FoE with 5 × 5 filter bank
(PSNR = 27.37)

MRF-MMSE with 3 × 3 filter bank
(PSNR = 27.23)

ARF with 5 × 5 filter bank
(PSNR = 27.49)

NLR-MRF with 3 × 3 ×3 filter bank and GSM expert
(PSNR = 27.82)

NLR-MRF with 5 × 5 ×3 filter bank and student-t expert
(PSNR = 28.10)

NLR-MRF with 5 × 5 ×3 filter bank and GSM expert
(PSNR = 28.27)

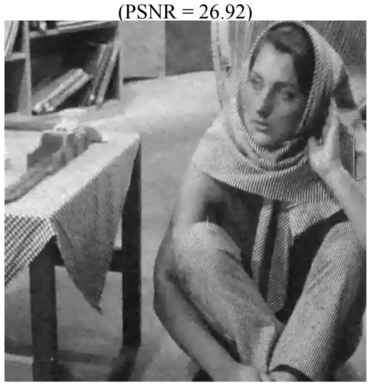
Figure 6. FoE: field of experts model [2]; ARF: active random field [1]; MRF-MMSE: the MRF-based method in [3]. The standard deviation of noise is 25. In ARF and NLR-MRF methods, four iterations of gradient descent procedures are used to infer the noise-free images.

Noisy image

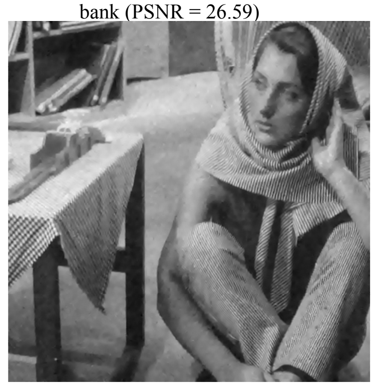
Original image

FoE with 5 × 5 filter bank (PSNR = 26.92)

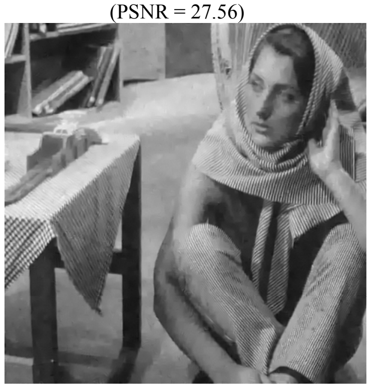
MRF-MMSE with 3 × 3 filter bank (PSNR = 26.59)



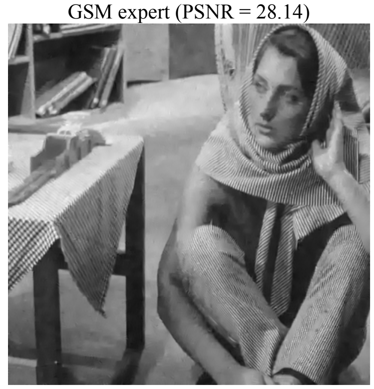
ARF with 5 × 5 filter bank (PSNR = 27.56)



NLR-MRF with 3 × 3 ×3 filter bank and GSM expert (PSNR = 28.14)



NLR-MRF with 5 × 5 ×3 filter bank and student-t expert (PSNR = 28.47)



NLR-MRF with 5 × 5 ×3 filter bank and GSM expert (PSNR = 28.92)

Figure 7. FoE: field of experts model [2]; ARF: active random field [1]; MRF-MMSE: the MRF-based method in [3]. The standard deviation of noise is 25. In ARF and NLR-MRF methods, four iterations of gradient descent procedures are used to infer the noise-free images.

Noisy image

Original image

FoE with 5 × 5 filter bank
(PSNR = 35.69)

MRF-MMSE with 3 × 3 filter bank
(PSNR = 35.55)

ARF with 5 × 5 filter bank
(PSNR = 35.17)

NLR-MRF with 3 × 3 ×3 filter bank and GSM expert
(PSNR = 35.01)

NLR-MRF with 5 × 5 ×3 filter bank and student-t expert
(PSNR = 35.95)

NLR-MRF with 5 × 5 ×3 filter bank and GSM expert
(PSNR = 36.31)

Figure 8. FoE: field of experts model [2]; ARF: active random field [1]; MRF-MMSE: the MRF-based method in [3]. The standard deviation of noise is 25. In ARF and NLR-MRF methods, four iterations of gradient descent procedures are used to infer the noise-free images.

Noisy image

Original image

FoE with 5 × 5 filter bank
(PSNR = 27.28)

MRF-MMSE with 3 × 3 filter bank
(PSNR = 27.92)

ARF with 5 × 5 filter bank
(PSNR = 27.86)

NLR-MRF with 3 × 3 ×3 filter bank and GSM expert
(PSNR = 27.78)

NLR-MRF with 5 × 5 ×3 filter bank and student-t expert
(PSNR = 27.96)

NLR-MRF with 5 × 5 ×3 filter bank and GSM expert
(PSNR = 28.02)

Figure 9. FoE: field of experts model [2]; ARF: active random field [1]; MRF-MMSE: the MRF-based method in [3]. The standard deviation of noise is 25. In ARF and NLR-MRF methods, four iterations of gradient descent procedures are used to infer the noise-free images.

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

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