A Generative Perspective on MRFs in Low-Level Vision  
Supplemental Material  
Uwe Schmidt* Qi Gao* Stefan Roth  
Department of Computer Science, TU Darmstadt  

1. Derivations  
1.1. Sampling the Prior  

We first rewrite the model density from Eqs. (1) and (2) of the main paper as 

\[ p(x; \Theta) = \sum_z \frac{1}{Z(\Theta)} e^{-\epsilon \|x\|^2/2} \prod_{c \in C} \prod_{i=1}^N p(z_{ic}) \cdot N(J_i^T x_c; 0, \sigma_i^2/s_{z_{ic}}), \] 

(1)  

where we treat the scales \( z \in \{1, \ldots, J\}^{N \times |C|} \) for each expert and clique as random variables with \( p(z_{ic}) = \alpha_{z_{ic}} \) (i.e., the GSM mixture weights). Instead of marginalizing out the scales, we can also retain them explicitly and define the joint distribution (cf. [13])  

\[ p(x, z; \Theta) = \frac{1}{Z(\Theta)} e^{-\epsilon \|x\|^2/2} \prod_{c \in C} \prod_{i=1}^N p(z_{ic}) \cdot N(J_i^T x_c; 0, \sigma_i^2/s_{z_{ic}}). \]  

(2)  

The conditional distribution \( p(x|z; \Theta) \) can be derived as the multivariate Gaussian  

\[ p(x|z; \Theta) \propto e^{-\epsilon \|x\|^2/2} \prod_{c \in C} \prod_{i=1}^N \exp \left( -\frac{s_{z_{ic}}}{2\sigma_i^2} (J_i^T x_c)^2 \right) \]  

\[ \propto \exp \left( -\frac{1}{2} x^T \left( \epsilon I + \sum_{i=1}^N \sum_{c \in C} \frac{s_{z_{ic}}}{\sigma_i^2} w_i w_i^T \right) x \right) \]  

\[ \propto N \left( x; 0, \left( \epsilon I + \sum_{i=1}^N W_i Z_i W_i^T \right)^{-1} \right). \]  

(3)  

where the \( w_{ic} \) are defined such that \( w_i^T x \) is the result of applying filter \( J_i \) to clique \( c \) of the image \( x \). \( Z_i = \text{diag}\{s_{z_{ic}}/\sigma_i^2\} \) are diagonal matrices with entries for each clique, and \( W_i \) are filter matrices that correspond to a convolution of the image with filter \( J_i \), i.e. \( W_i^T x = [w_{i1}^T x, \ldots, w_{i|C|}^T x]^T = [J_i^T x_{(c_1)}, \ldots, J_i^T x_{(c_{|C|})}]^T \).

Following Levi and Weiss [4, 11], we further rewrite the covariance as the matrix product  

\[ \Sigma = \left( \epsilon I + \sum_{i=1}^N W_i Z_i W_i^T \right)^{-1} = \left[ \begin{array}{cc} Z_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & Z_N \end{array} \right] \left[ \begin{array}{c} W_1^T \\ \vdots \\ W_N^T \end{array} \right] \]  

\[ = (WZW^T)^{-1}. \]  

(4)  

and sample \( y \sim N(0, I) \) to obtain a sample \( x \) from \( p(x|z; \Theta) \) by solving the least-squares problem  

\[ WZW^T x = W \sqrt{Z} y. \]  

(5)
By using the well-known property

\[ y \sim \mathcal{N}(0, I) \Rightarrow Ay \sim \mathcal{N}(0, AIA^T), \]

it follows that

\[ x = (WZW^T)^{-1} W\sqrt{Z}y \sim \mathcal{N}\left( x; 0, \left( (WZW^T)^{-1} W\sqrt{Z} \right) I \left( (WZW^T)^{-1} W\sqrt{Z} \right)^T \right) \]

\[ \sim \mathcal{N}\left( x; 0, (WZW^T)^{-1} \right) \]  

is indeed a valid sample from the conditional distribution as derived in Eq. (3). Since the scales are conditionally independent given the image by construction, the conditional distribution \( p(z|\theta) \) is readily given as

\[ p(z_{ic}|x; \theta) \propto p(z_{ic}) \cdot \mathcal{N}(J^T x_{(c)}; 0, \sigma_i^2/s_{z_{ic}}). \]  

1.2. Conditional Sampling

In order to avoid extreme values at the less constrained boundary pixels [5] during learning and model analysis, or to perform inpainting of missing pixels given the known ones, we rely on conditional sampling. In particular, we sample the pixels \( x_A \) given fixed \( x_B \) and scales \( z \) according to the conditional Gaussian distribution

\[ p(x_A|x_B, z; \theta), \]  

where \( A \) and \( B \) denote the index sets of the respective pixels. Without loss of generality, we assume that

\[ x = \begin{bmatrix} x_A \\ x_B \end{bmatrix}, \quad \Sigma = (WZW^T)^{-1} = \begin{bmatrix} A & C \\ C^T & B \end{bmatrix}^{-1}, \]  

where the square sub-matrix \( A \) has as many rows and columns as the vector \( x_A \) has elements, etc. The conditional distribution of interest can now be derived as

\[ p(x_A|x_B, z; \theta) \propto \exp \left( -\frac{1}{2} \begin{bmatrix} x_A \\ x_B \end{bmatrix}^T \begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \begin{bmatrix} x_A \\ x_B \end{bmatrix} \right) \]

\[ \propto \exp \left( -\frac{1}{2} (x_A + A^{-1} C x_B)^T A (x_A + A^{-1} C x_B) \right) \]

\[ \propto \mathcal{N}(x_A; -A^{-1} C x_B, A^{-1}). \]  

The matrices \( A \) and \( C \) are given by the appropriate sub-matrices of \( W \) and \( Z \), and allow for the same efficient sampling scheme. The mean \( \mu = -A^{-1} C x_B \) can also be computed by solving a least squares problem. Sampling the conditional distribution of scales \( p(z|x_A, x_B; \theta) = p(z|x; \theta) \) remains as before.

1.3. Sampling the Posterior for Image Denoising

Assuming additive i. i. d. Gaussian noise with known standard derivation \( \sigma \), the posterior given scales \( z \) can be written as

\[ p(x|y, z; \theta) \propto p(y|x) \cdot p(x|z; \theta) \]

\[ \propto \exp \left( -\frac{1}{2\sigma^2} \|y - x\|^2 \right) \cdot \exp \left( -\frac{1}{2} x^T \Sigma^{-1} x \right) \]

\[ \propto \exp \left( -\frac{1}{2} \left( -2x^T \frac{y}{\sigma^2} + x^T \left( \frac{1}{\sigma^2} + \Sigma^{-1} \right) x \right) \right) \]

\[ \propto \mathcal{N}(x; \Sigma y/\sigma^2, \Sigma), \]  

where \( \Sigma = (I/\sigma^2 + \Sigma^{-1})^{-1} \) and \( \Sigma \) as in Eq. (4). The conditional distribution of the scales \( p(z|x, y; \theta) = p(z|x; \theta) \) remains as before.
2. Image Restoration

To further illustrate the image restoration performance of our approach, we provide the following additional results:

• Table 1 repeats Tab. 1 of the main paper and additionally gives the numerical results of MAP estimation with graph cuts and $\alpha$-expansion [1]. Note that in most cases, $\alpha$-expansion performs slightly worse in terms of PSNR than conjugate gradients, even (and in fact particularly) for non-convex potentials. Also, using a Student-t potential [3] does not show favorable results.

• Table 2 shows the results of the same experiment as in Tab. 1, but reports the performance in terms of the perceptually more relevant structural similarity index (SSIM) [10]. Note that all of the conclusions reported in the main paper also hold for this perceptual quality metric.

• Table 3 repeats Tab. 2 of the main paper, and additionally reports standard deviations as well as SSIM performance. The SSIM supports the same conclusions about relative performance as the PSNR.

• Figs. 1–6 show denoising results for 6 of the 68 images, for which the average performance is reported in Tab. 2 of the main paper. Note that in contrast to the tested previous approaches, combining our learned models with MMSE leads to good performance on relatively smooth as well as on strongly textured images.

• Fig. 7 provides a different view of the summary results in Tab. 2 of the paper. Instead of the average performance, we show a per-image comparison between the denoising results of the discriminative approach of [8] (using MAP) and the results of our generatively-trained $3 \times 3$ FoE (using MMSE). Note that the PSNR and particularly the SSIM show a substantial performance advantage for our approach.

• Fig. 8 shows an uncropped version of the inpainting result in Fig. 7 of the paper. Additionally, one other inpainting result is provided as further visual illustration.

3. Sampling the Prior and Posterior

The following additional results illustrate properties of the auxiliary-variable Gibbs sampler.

• Fig. 9 shows five subsequent samples (after reaching the equilibrium distribution) from all models listed in Table 1. Note how samples from common pairwise models appear too “grainy”, while those from previous FoE models are too smooth and without discontinuities.

• Fig. 10 shows two larger samples from our learned models. Note that our pairwise model leads to locally uniform samples with occasional discontinuities that appear spatially isolated (“speckles”). Our learned high-order model, on the other hand, leads to smoothly varying samples with occasional spatially correlated discontinuities, which appear more realistic.

• Fig. 11 illustrates the convergence of the sampling procedure for the prior and the posterior (in case of denoising).

• Fig. 12 illustrates the advantages of running multiple parallel samplers.

References

### Table 1. Average PSNR (dB) of denoising results for 10 test images [3].

<table>
<thead>
<tr>
<th>Model</th>
<th>MAP (λ=1)</th>
<th>MAP (opt. λ)</th>
<th>MMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>conj. gradient</td>
<td>α-expansion</td>
<td>conj. gradient</td>
</tr>
<tr>
<td></td>
<td>σ=10</td>
<td>σ=20</td>
<td>σ=10</td>
</tr>
<tr>
<td>pairwise (marginal fitting)</td>
<td>28.35</td>
<td>23.96</td>
<td>27.32</td>
</tr>
<tr>
<td>pairwise (generalized Laplacian [9])</td>
<td>27.35</td>
<td>24.27</td>
<td>29.36</td>
</tr>
<tr>
<td>pairwise (Student-t [3])</td>
<td><strong>30.27</strong></td>
<td><strong>26.48</strong></td>
<td><strong>30.14</strong></td>
</tr>
<tr>
<td>pairwise (ours)</td>
<td>27.92</td>
<td>23.81</td>
<td>28.19</td>
</tr>
<tr>
<td>15 × 15 FoE from [12]</td>
<td>30.33</td>
<td>25.15</td>
<td>– –</td>
</tr>
<tr>
<td>3 × 3 FoE (ours)</td>
<td>30.33</td>
<td>25.15</td>
<td>– –</td>
</tr>
</tbody>
</table>

### Table 2. Average SSIM [10] of denoising results for 10 test images [3].

<table>
<thead>
<tr>
<th>Model</th>
<th>MAP (λ=1)</th>
<th>MAP (opt. λ)</th>
<th>MMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>conj. gradient</td>
<td>α-expansion</td>
<td>conj. gradient</td>
</tr>
<tr>
<td></td>
<td>σ=10</td>
<td>σ=20</td>
<td>σ=10</td>
</tr>
<tr>
<td>pairwise (marginal fitting)</td>
<td>0.787</td>
<td>0.599</td>
<td>0.745</td>
</tr>
<tr>
<td>pairwise (generalized Laplacian [9])</td>
<td>0.745</td>
<td>0.559</td>
<td>0.832</td>
</tr>
<tr>
<td>pairwise (Laplacian)</td>
<td>0.784</td>
<td>0.602</td>
<td>0.855</td>
</tr>
<tr>
<td>pairwise (Student-t [3])</td>
<td><strong>0.885</strong></td>
<td><strong>0.720</strong></td>
<td><strong>0.854</strong></td>
</tr>
<tr>
<td>pairwise (ours)</td>
<td>0.763</td>
<td>0.595</td>
<td>0.838</td>
</tr>
<tr>
<td>5 × 5 FoE from [7]</td>
<td>0.515</td>
<td>0.445</td>
<td>– –</td>
</tr>
<tr>
<td>15 × 15 FoE from [12]</td>
<td>– –</td>
<td>– –</td>
<td>0.838</td>
</tr>
<tr>
<td>3 × 3 FoE (ours)</td>
<td>– –</td>
<td>– –</td>
<td><strong>0.838</strong></td>
</tr>
</tbody>
</table>

### Table 3. Denoising results for 68 test images [7, 8] (σ = 25).

<table>
<thead>
<tr>
<th>Model</th>
<th>Learning</th>
<th>Inference</th>
<th>PSNR in dB</th>
<th>SSIM [10]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>average</td>
<td>std. dev.</td>
</tr>
<tr>
<td>5 × 5 FoE from [7]</td>
<td>CD (generative)</td>
<td>MAP w/λ</td>
<td>27.44</td>
<td>2.36</td>
</tr>
<tr>
<td>5 × 5 FoE from [8]</td>
<td>discriminative</td>
<td>MAP</td>
<td>27.86</td>
<td>2.09</td>
</tr>
<tr>
<td>pairwise (ours)</td>
<td>CD (generative)</td>
<td>MMSE</td>
<td>27.54</td>
<td>2.05</td>
</tr>
<tr>
<td>3 × 3 FoE (ours)</td>
<td>CD (generative)</td>
<td>MMSE</td>
<td>27.95</td>
<td>2.30</td>
</tr>
<tr>
<td>Non-local means [2]</td>
<td>– (MMSE)</td>
<td>–</td>
<td>27.50</td>
<td>2.12</td>
</tr>
<tr>
<td>BLS-GSM [6]</td>
<td>– MMSE</td>
<td>–</td>
<td><strong>28.02</strong></td>
<td><strong>2.24</strong></td>
</tr>
</tbody>
</table>


Figure 1. Denoising results for test image “Castle”: (e, f) MMSE, (c, d, g) MAP w/λ, (h) MAP.
Figure 2. Denoising results for test image “Birds”: (e, f) MMSE, (c, d, g) MAP w/\(\lambda\), (h) MAP.
Figure 3. Denoising results for test image “LA”: (e, f) MMSE, (c, d, g) MAP w/\(\lambda\), (h) MAP.
Figure 4. Denoising results for test image “Goat”: (e, f) MMSE, (c, d, g) MAP w/λ, (h) MAP. 
Figure 5. Denoising results for test image “Wolf”: (e, f) MMSE, (c, d, g) MAP w/λ, (h) MAP.
Figure 6. Denoising results for test image “Airplane”: (e, f) MMSE, (c, d, g) MAP w/λ, (h) MAP.
Figure 7. Comparing the denoising performance ($\sigma = 25$) in terms of (a) PSNR and (b) SSIM for 68 test images between our $3 \times 3$ FoE (using MMSE) and the $5 \times 5$ FoE from [8] (using MAP). A red circle above the black line means performance is better with our approach.

Figure 8. MMSE-based image inpainting with our learned models.
Figure 9. Five subsequent samples (l. to r.) from various MRF models after reaching the equilibrium distribution. The boundary pixels are removed for better visualization.
Figure 10. 256 × 256 pixel sample from our learned models after reaching the equilibrium distribution. The boundary pixels are removed for better visualization.

Figure 11. Monitoring the convergence of sampling. (a) Sampling a 50 × 50 image from the learned pairwise MRF prior conditioned on a 1-pixel boundary. Three chains and over-dispersed starting points (red, dashed – interior of the boundary image; blue, solid – median-filtered version; black, dash-dotted – noisy version). Approximate convergence is reached after 25 iterations ($\hat{R} < 1.1$). (b) Sampling the posterior ($\sigma = 20$, image size 160 × 240) with four chains and over-dispersed starting points (red, dashed – noisy image; blue, dash-dotted – Gauss filtered version; green, solid – median filtered version; black, dotted – Wiener filtered version). Approximate convergence is reached after 24 iterations.

Figure 12. Efficiency of sampling-based MMSE denoising with different number of samplers. Learned pairwise MRF, $\sigma = 20$, image size 160 × 240. (a) In case of parallel computing (one sampler per computing core), faster convergence of the denoised image can be achieved. (b) Even when using sequential computing, multiple samplers can improve performance, as the samples are less correlated.