Noise-Blind Image Deblurring
– Supplementary Material –

Meiguang Jin
University of Bern
Switzerland

Stefan Roth
TU Darmstadt
Germany

Paolo Favaro
University of Bern
Switzerland

A. Upper and Lower Bounds

Our approach proposes a Bayesian generalization to Maximum a-Posteriori (MAP). It is interesting to understand what relationship exists between MAP, the proposed expected loss, and the loss lower bound. It turns out that the logarithm of *Bayes' utility* (BU) is bounded from below by our proposed bound and from above by the log of the MAP problem. We have

$$\begin{aligned}
\log \max_{\hat{x}} p(y|\hat{x};\sigma_n)p(\hat{x}) &= \log \max_{\hat{x}} p(\hat{x},y;\sigma_n) && \text{(MAP)}\\
&= \max_x \log \max_{\hat{x}} p(\hat{x},y;\sigma_n) \int G(\bar{x},x)\, d\bar{x}\\
&\geq \max_x \log \int p(\bar{x},y;\sigma_n) G(\bar{x},x)\, d\bar{x} && \text{(BU)}\\
&\geq \max_x \int G(\bar{x},x) \log p(\bar{x},y;\sigma_n)\, d\bar{x}. && \text{(lower bound)}
\end{aligned}$$

The second equation is due to the definition of G (it integrates to 1 over $\bar{x}$). Notice that as $\sigma \to 0$, the lower bound tends towards the MAP and thus they both converge to the logarithm of BU.

B. A General Family of Image Priors

We here show the applicability of our framework to a general family of image priors whose negative log-likelihood can be written as a concave function ϕ of terms $\frac{|F_{ik}\bar{x}-\mu_j|^2}{2\sigma_j^2}$. As in the main paper, F_i is a linear filter or a Toeplitz matrix, *e.g.*, $F_i = \nabla$. Again as before, F_{ik} yields the kth entry of the output and is therefore a row vector; μ_j and σ_j are parameters. Consequently, we have

$$-\log p(\bar{x}) = \phi\left(\xi_{111},\dots,\xi_{IJK}\right) \qquad \text{with} \qquad \xi_{ijk} = \frac{|F_{ik}\bar{x}-\mu_j|^2}{2\sigma_j^2}. \tag{39}$$

Table 5 summarizes a few choices of ϕ for some popular image priors. Also notice that the type-1 Gumbel prior falls into this family as

$$\phi\left(\xi_0,\xi_{111},\dots,\xi_{IJK}\right) = \xi_0 - \sum_{ijk} w_{ij} e^{-\xi_{ijk}} \qquad \text{with} \qquad \xi_0 = \frac{|\bar{x}|^2}{2\sigma_0^2},\ \xi_{ijk} = \frac{|F_{ik}\bar{x}-\mu_j|^2}{2\sigma_j^2} \tag{40}$$

is a concave function jointly in all its arguments. These priors can have simple surrogate functions that yield simple Majorization Minimization (MM) [10] iterations. The concavity of ϕ gives the following inequality and surrogate function $\psi(\bar{x}|x^\tau)$

$$\log p(\bar{x}) \geq \log p(x^\tau) - \sum_{ijk} \frac{\partial \phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial \xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}} \left(\frac{|F_{ik}\bar{x}-\mu_j|^2}{2\sigma_j^2} - \frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}\right) \doteq -\psi(\bar{x}|x^\tau). \tag{41}$$

Image prior	$-\log p(\bar{x})$	$\phi(\lvert\nabla\bar{x}(\cdot)\rvert_2^2)$	$\phi'(\lvert\nabla\bar{x}(\cdot)\rvert_2^2)$
Gaussian	$\frac{1}{2\sigma_x^2}\lvert\nabla\bar{x}\rvert_{2,2}^2$	$\phi(w) = \frac{1}{2\sigma_x^2}\int w(z)\,dz$	$\phi'(w) = \frac{1}{2\sigma_x^2}$
Total Variation	$\frac{1}{2\sigma_x^2}\lvert\nabla\bar{x}\rvert_{2,1}$	$\phi(w) = \frac{1}{2\sigma_x^2}\int (w(z)+\epsilon)^{\frac{1}{2}}\,dz$	$\phi'(w) = \frac{1}{4\sigma_x^2}(w(z)+\epsilon)^{-\frac{1}{2}}$
Sparsity ($p<1$)	$\frac{1}{2\sigma_x^2}\lvert\nabla\bar{x}\rvert_{2,p}$	$\phi(w) = \frac{1}{2\sigma_x^2}(\int (w(z)+\epsilon)^{\frac{p}{2}}\,dz)^{\frac{1}{p}}$	$\phi'(w) = (\int (w(z)+\epsilon)^{\frac{p}{2}}\,dz)^{\frac{1}{p}-1}\frac{(w(z)+\epsilon)^{\frac{p}{2}-1}}{4\sigma_x^2}$

Table 5. Examples of image priors, their negative log probability density functions, the corresponding ϕ functions, and derivative ϕ'. The small coefficient $\epsilon > 0$ is to avoid division by zero.

Then, this yields the inequality

$$\int G(\bar{x},x)\log p(\bar{x})\,d\bar{x} \geq -\int G(\bar{x},x)\psi(\bar{x}|x^\tau)\,d\bar{x} \tag{42a}$$

$$= -\int G(\bar{x},x)\sum_{ijk}\frac{\partial\phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial\xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}}\frac{|F_{ik}\bar{x}-\mu_j|^2}{2\sigma_j^2}\,d\bar{x} + \text{const} \tag{42b}$$

$$= -\sum_{ijk}\frac{\partial\phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial\xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}}\frac{|F_{ik}x-\mu_j|^2}{2\sigma_j^2} + \text{const}, \tag{42c}$$

where all the constant terms do not depend on x.

C. Noise-Adaptive Deblurring

The noise-adaptive algorithm in the general image prior family is analogous to that in the main paper for the Gumbel prior. We need to put all the terms together and solve the maximization of the lower bound

$$\arg\max_{x,\sigma_n}\int G(\bar{x},x)\log p(\bar{x},y;\sigma_n)\,d\bar{x}. \tag{43}$$

Thus, we obtain the following iterative algorithm

$$(x^{\tau+1},\hat{\sigma}_n) = \arg\min_{x,\sigma_n}\frac{|y-k*x|^2+M\sigma^2|k|^2}{2\sigma_n^2} + N\log\sigma_n + \sum_{ijk}\frac{\partial\phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial\xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}}\frac{|F_{ik}x-\mu_j|^2}{2\sigma_j^2}. \tag{44}$$

We can now solve explicitly for $\hat{\sigma}_n$ and, as in the main paper, obtain

$$\hat{\sigma}_n^2 = \tfrac{1}{N}\left[|y-k*x|^2+M\sigma^2|k|^2\right]. \tag{45}$$

This closed form can be incorporated in an iterative algorithm and yields

$$x^{\tau+1} = \arg\min_x \tfrac{N}{2}\log\left[|y-k*x|^2+M\sigma^2|k|^2\right] + \sum_{ijk}\frac{\partial\phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial\xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}}\frac{|F_{ik}x-\mu_j|^2}{2\sigma_j^2}. \tag{46}$$

In the noise-blind deblurring formulation we can explicitly obtain the gradient descent iteration

$$x^{\tau+1} = x^\tau - \alpha\left[\lambda^\tau K^\top(Kx^\tau-y) + \sum_{ijk}F_{ik}^\top\frac{\partial\phi\left(\xi_{111},\dots,\xi_{ijk},\dots\right)}{\partial\xi_{ijk}}\Big|_{\xi_{ijk}=\frac{|F_{ik}x^\tau-\mu_j|^2}{2\sigma_j^2}}\frac{F_{ik}x^\tau-\mu_j}{\sigma_j^2}\right]. \tag{47}$$

for some small step $\alpha > 0$, where x^τ is the solution at gradient descent iteration τ and, as in the main paper, the noise adaptivity is given as $\lambda^\tau = \frac{N}{|y-Kx^\tau|^2+M\sigma^2|k|^2}$.

TV-L_2 Noise-Adaptive Deblurring. Following from Eq. (47) the case of TV-L_2 is then readily obtained as the gradient descent iteration

$$x^{\tau+1} = x^{\tau} - \alpha \left[\lambda^{\tau} K^{\top}(Kx^{\tau} - y) + \nabla^{\top} \left(\frac{\nabla x^{\tau}}{4\sigma_x^2 |\nabla x^{\tau}|_{2,1}} \right) \right] \tag{48}$$

for some small step $\alpha > 0$, where ∇ denotes the finite difference operator along the two coordinate axes, and x^{τ} and λ^{τ} are defined as before.

EPLL Noise-Adaptive Deblurring. In the case of EPLL [37], we modify the original Eq. (4) in [37] by introducing our noise-adaptive term $\lambda^{\tau} = \frac{N}{|y - Kx^{\tau}|^2 + M\sigma^2 |k|^2}$ and obtain

$$x^{\tau+1} = \left(\lambda^{\tau} A^{\top} A + \beta \sum_j P_j^{\top} P_j \right)^{-1} \left(\lambda^{\tau} A^{\top} y + \beta \sum_j P_j^{\top} z_j^{\tau} \right). \tag{49}$$

D. Back-Propagation in the GradNet Architecture

In the following section we derive the gradients of GradNet with respect to its parameters. We first introduce the notation for the basic components of one stage in GradNet and then compute the gradients of this stage with respect to the parameters. The derivatives for the other stages will be similar.

Initialization. To obtain x_q^0 at the stage $\tau = 0$ we first pad the noisy blurry input using the constant boundary assumption (*i.e.*, zero derivative at the boundary). Then we apply 3 iterations of MATLAB's `edgetaper` function to the padded noisy blurry input. We find experimentally that this makes the reconstructed image converge faster at the boundaries.

Greedy Training. In the following derivation, we will omit the sample index q for simplicity. In the main paper, we defined $x^{\tau+1}$ as

$$\begin{aligned} x^{\tau+1} &= x^{\tau} - \left(\lambda^{\tau} H^{\top} H + \tfrac{I}{\sigma_0^2} + \gamma^{\tau} \sum_{ik} B_{ik}^{\tau\top} B_{ik}^{\tau} \right)^{-1} \left(\lambda^{\tau} K^{\top}(Kx^{\tau} - y) + \tfrac{x^{\tau}}{\sigma_0^2} - \sum_{ik} {F_{ik}^{\tau}}^{\top} \sum_j \hat{w}_{ij}^{\tau} \exp\left[-\tfrac{|F_{ik}^{\tau} x^{\tau} - \mu_j|^2}{2(\sigma^2 + \sigma_j^2)} \right] \right) \\ &= x^{\tau} - (\Lambda^{\tau})^{-1} \eta^{\tau} \end{aligned} \tag{50}$$

where we have

$$\lambda^{\tau} = \frac{N}{|y - Kx^{\tau}|^2 + M\sigma^2 |k|^2}, \tag{51}$$

$$\Lambda^{\tau} = \lambda^{\tau} H^{\top} H + \tfrac{I}{\sigma_0^2} + \gamma^{\tau} \sum_{ik} B_{ik}^{\tau\top} B_{ik}^{\tau}, \tag{52}$$

and

$$\eta^{\tau} = \lambda^{\tau} K^{\top}(Kx^{\tau} - y) + \frac{x^{\tau}}{\sigma_0^2} - \sum_{ik} {F_{ik}^{\tau}}^{\top} \sum_j \hat{w}_{ij}^{\tau} \exp\left[-\frac{|F_{ik}^{\tau} x^{\tau} - \mu_j|^2}{2(\sigma^2 + \sigma_j^2)} \right]. \tag{53}$$

In the stage τ, the gradient equals

$$\frac{\partial x^{\tau+1}}{\partial \Theta^{\tau}} = -(\Lambda^{\tau})^{-1} \left[-\frac{\partial \Lambda^{\tau}}{\partial \Theta^{\tau}} (\Lambda^{\tau})^{-1} \eta^{\tau} + \frac{\partial \eta^{\tau}}{\partial \Theta^{\tau}} \right]. \tag{54}$$

Hence, given the loss function for stage τ

$$L(\Theta^{\tau}) = \frac{1}{2} \left| C^{\tau+1}(x^{\tau+1} - x^{\mathrm{GT}}) \right|_2^2, \tag{55}$$

we can get the gradient of the loss function with respect to Θ^τ as

$$\frac{\partial L(\Theta^\tau)}{\partial \Theta^\tau} = \frac{\partial L(\Theta^\tau)}{\partial x^{\tau+1}} \frac{\partial x^{\tau+1}}{\partial \Theta^\tau} = P^{\tau+1} \left[\frac{\partial \Lambda^\tau}{\partial \Theta^\tau} (\Lambda^\tau)^{-1} \eta^\tau - \frac{\partial \eta^\tau}{\partial \Theta^\tau} \right] = -P^{\tau+1} \left[\frac{\partial \eta^\tau}{\partial \Theta^\tau} + \frac{\partial \Lambda^\tau}{\partial \Theta^\tau} (x^{\tau+1} - x^\tau) \right], \tag{56}$$

where we define $P^{\tau+1} = (x^{\tau+1} - x^{\text{GT}})^\top (C^{\tau+1})^\top C^{\tau+1} (\Lambda^\tau)^{-1}$. Now we can obtain the derivative with respect to each parameter to be learned, *i.e.* $\sigma, \gamma^\tau, \hat{w}_{ij}^\tau$ and f_i^τ. To calculate $\frac{\partial L(\Theta^\tau)}{\partial \sigma}$, we can first take the derivative $\frac{\partial L(\Theta^\tau)}{\partial \lambda^\tau}$, after which $\frac{\partial L(\Theta^\tau)}{\partial \sigma}$ can be easily calculated by chain rule

$$\frac{\partial L(\Theta^\tau)}{\partial \lambda^\tau} = -P^{\tau+1} \left[K^\top (K x^\tau - y) + H^\top H (x^{\tau+1} - x^\tau) \right]. \tag{57}$$

$$\frac{\partial \lambda^\tau}{\partial \sigma} = \frac{2\sigma N M |k|^2}{(|y - K x^\tau|^2 + M\sigma^2 |k|^2)^2}. \tag{58}$$

Hence, we get

$$\frac{\partial L(\Theta^\tau)}{\partial \sigma} = \frac{\partial L(\Theta^\tau)}{\partial \lambda^\tau} \frac{\partial \lambda^\tau}{\partial \sigma} - P^{\tau+1} \sum_{ik} {F_{ik}^\tau}^\top \sum_j \hat{w}_{ij}^\tau \exp\left[-\frac{|F_{ik}^\tau x^\tau - \mu_j|^2}{2(\sigma^2 + \sigma_j^2)} \right] \frac{|F_{ik}^\tau x^\tau - \mu_j|^2 \sigma}{(\sigma^2 + \sigma_j^2)^2}, \tag{59}$$

$$\frac{\partial L(\Theta^\tau)}{\partial \gamma^\tau} = -P^{\tau+1} \sum_{ik} B_{ik}^{\tau\top} B_{ik}^\tau (x^{\tau+1} - x^\tau), \tag{60}$$

$$\frac{\partial L(\Theta^\tau)}{\partial \hat{w}_{ij}^\tau} = P^{\tau+1} \sum_k (F_{ik}^\tau)^\top \exp\left[-\frac{|F_{ik}^\tau x^\tau - \mu_j|^2}{2(\sigma^2 + \sigma_j^2)} \right]. \tag{61}$$

We omit the derivation of $\frac{\partial L(\Theta^\tau)}{\partial f_i^\tau}$ here. Since our work and shrinkage fields [25] make use of filters in the same way, the derivation is analogous. For details we refer to the supplementary material of [25].

Joint Training. After training each stage separately, we perform a joint training similarly to shrinkage fields [25] and the diffusion network [4], to which we refer to for more details.

E. Additional Experimental Results

In Table 7, 8 and 9 we show additional results with both the PSNR and SSIM metrics and intermediate results of our GradNet (after 1st stage and 4th stage). We also show two more visual results in Figs. 6 and 7 at the 2% and 1% noise levels corresponding to $\sigma = 5.10$ and 2.55. We can see that GradNet removes more blur in the flower region in Fig. 6. In the 3rd row of Fig. 7 we also show the globally contrast-adjusted difference between each reconstruction and the ground truth. We can see that GradNet compares favorably to EPLL on the dataset of Sun *et al*. Our noise-adaptive formulation is also able to deal with colored (spatially correlated) noise. We generate different amounts of white Gaussian noise 2%, 3%, and 4% (*i.e.*, $\sigma = 5.10,\ 7.65,\ 10.20$), convolve these noise images with a 3×3 uniform filter to make noise spatially correlated and then finally add them to the blurry image. Experiments are performed with 32 test images from [16]. Results in Table 6 show that our noise-adaptive formulation is robust to colored noise.

	PSNR			SSIM		
Method	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$
FD[13] (non-blind)	28.075	27.881	26.815	0.837	0.828	**0.795**
EPLL [37] + NE	27.352	24.099	21.683	0.709	0.556	0.444
EPLL [37] + NA	**31.356**	28.775	26.254	0.900	0.807	0.687
TV-L_2 + NA	30.818	29.086	**27.489**	0.879	0.810	0.738
BD [26]	29.020	26.495	24.675	0.789	0.685	0.602
GradNet 7S	31.104	**29.121**	26.882	**0.903**	**0.840**	0.741

Table 6. Average PSNR(dB) and SSIM on 32 images from [16] for three different colored noise levels.

Method	PSNR				SSIM			
	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$
FD [13] (non-blind)	30.029	28.396	27.315	26.520	0.890	0.845	0.807	0.782
RTF [24] ($\sigma = 2.55$)	**32.355**	26.337	21.428	17.328	**0.925**	0.676	0.424	0.301
CSF [25] (non-blind)	29.954	28.126	27.284	26.698	0.881	0.802	0.779	0.750
TNRD [4] (non-blind)	28.882	28.095	27.550	27.118	0.854	0.824	0.800	0.795
TV-L_2 (non-blind)	30.870	28.428	27.594	26.514	0.892	0.811	0.792	0.716
EPLL [37] (non-blind)	32.028	29.789	28.312	27.196	0.920	0.874	0.836	0.803
EPLL [37] + NE [36]	31.857	29.771	28.281	27.157	0.919	0.878	0.839	0.807
EPLL [37] + NA	32.160	**30.248**	**28.957**	**27.851**	0.923	**0.888**	**0.856**	**0.824**
TV-L_2 + NA	31.050	29.135	28.028	27.161	0.894	0.843	0.821	0.798
BD [26]	30.422	28.765	27.908	27.289	0.880	0.833	0.807	0.789
GradNet 1S	25.094	24.700	24.262	23.828	0.754	0.736	0.717	0.698
GradNet 4S	30.286	28.297	27.363	26.753	0.869	0.786	0.751	0.734
GradNet 7S	31.432	28.878	27.551	26.960	0.912	0.841	0.797	0.783

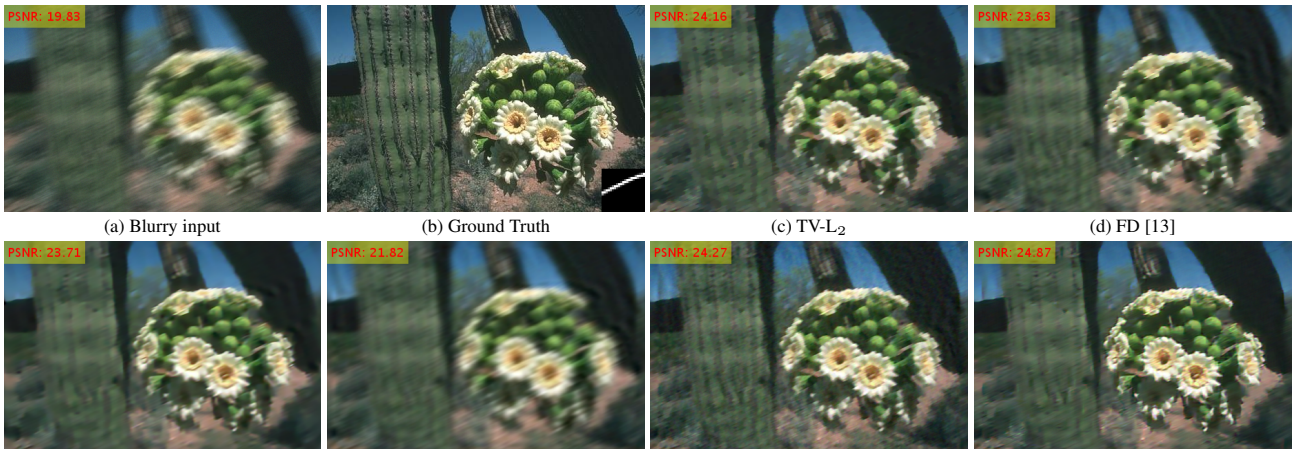
Table 7. Average PSNR (dB) on 32 test images from [16].

Method	PSNR				SSIM			
	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$
FD [13] (non-blind)	30.789	28.898	27.863	27.138	0.851	0.787	0.744	0.714
EPLL [37] (non-blind)	32.049	29.601	28.252	27.338	0.880	0.807	0.758	0.721
CSF [25] (non-blind)	30.875	28.604	27.647	26.969	0.853	0.752	0.718	0.681
TNRD [4] (non-blind)	30.026	28.794	28.040	27.544	0.844	0.790	0.750	0.739
EPLL [37] + NE	32.0.22	29.600	28.249	27.340	0.878	0.807	0.758	0.724
EPLL [37] + NA	**32.182**	**30.077**	**28.770**	**27.806**	**0.882**	**0.826**	**0.775**	**0.736**
TV-L_2 + NA	30.072	28.587	27.600	26.886	0.853	0.793	0.751	0.718
GradNet 1S	27.000	26.459	25.992	25.595	0.731	0.706	0.684	0.665
GradNet 4S	30.430	28.167	27.201	26.675	0.825	0.716	0.671	0.650
GradNet 7S	31.745	29.310	28.044	27.540	0.873	0.798	0.750	0.733

Table 8. Average PSNR (dB) on 640 test images from [29].

Method	PSNR				SSIM			
	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$	$\sigma = 2.55$	$\sigma = 5.10$	$\sigma = 7.65$	$\sigma = 10.2$
FD [13] (non-blind)	24.436	23.240	22.642	22.065	0.664	0.577	0.534	0.492
EPLL [37] (non-blind)	25.377	25.531	22.545	21.905	0.712	0.590	0.521	0.476
RTF [24] ($\sigma = 2.55$)	**25.702**	23.454	19.833	16.939	**0.732**	0.607	0.403	0.280
CSF [25] (non-blind)	24.734	23.603	22.881	22.440	0.693	0.612	0.558	0.521
TNRD [4] (non-blind)	24.174	23.762	23. 270	22.865	0.690	0.631	0.589	0.550
EPLL [37] + NE [36]	25.360	23.532	22.545	21.904	0.708	0.588	0.520	0.478
EPLL [37] + NA	25.570	23.902	22.911	22.271	0.724	0.608	0.537	0.493
TV-L_2 + NA	24.612	23.652	22.896	22.336	0.687	0.607	0.546	0.504
GradNet 1S	21.924	21.695	21.455	21.234	0.506	0.485	0.464	0.447
GradNet 4S	24.665	23.561	22.874	22.391	0.687	0.601	0.548	0.511
GradNet 7S	25.571	**24.227**	**23.464**	**22.942**	0.731	**0.653**	**0.595**	**0.552**

Table 9. Average PSNR (dB) on 50 test images from Berkeley segmentation dataset [1] with large blurs.



(a) Blurry input (b) Ground Truth (c) TV-L_2 (d) FD [13]

(e) EPLL [37] (f) GradNet 1 stage (g) GradNet 4 stage (h) GradNet 7 stage

Figure 6. Results for the 2% noise case. PSNR results are also shown at the top left corner of the estimated images.

(a) Blurry input (b) Ground Truth (c) TV-L_2 (d) FD [13]

(e) EPLL [37] (f) GradNet 1 stage (g) GradNet 4 stage (h) GradNet 7 stage

(i) Difference between EPLL and GT (j) Difference between FD and GT (k) Difference between TV and GT (l) Difference between GradNet and GT

Figure 7. Results for the 1% noise case (best viewed on screen). PSNR results are also shown at the top left corner of the estimated images.