# Image deblurring and denoising with non-local regularization constraint

Peter van Beek,[a] Junlan Yang,[b] Shuhei Yamamoto[c] and Yasuhiro Ueda[c]

[a]Sharp Labs of America, Camas, WA, USA
[b]University of Illinois at Chicago, Chicago, IL, USA
[c]Sharp Corporation, Production Technology Development Group, Tenri, Nara, Japan

## ABSTRACT

In this paper, we investigate the use of the non-local means (NLM) denoising approach in the context of image deblurring and restoration. We propose a novel deblurring approach that utilizes a non-local regularization constraint. Our interest in the NLM principle is its potential to suppress noise while effectively preserving edges and other texture detail. Our approach leads to an iterative cost function minimization algorithm, similar to common deblurring methods, but incorporating update terms due to the non-local regularization constraint. The data-adaptive noise suppression weights in the regularization term are updated and improved at each iteration, based on the partially denoised and deblurred result. We compare our proposed algorithm to conventional deblurring methods, including deblurring with total variation (TV) regularization. We also compare our algorithm to combinations of the NLM-based filter followed by conventional deblurring methods. Our initial experimental results demonstrate that the use of NLM-based filtering and regularization seems beneficial in the context of image deblurring, reducing the risk of over-smoothing or suppression of texture detail, while suppressing noise. Furthermore, the proposed deblurring algorithm with non-local regularization outperforms other methods, such as deblurring with TV regularization or separate NLM-based denoising followed by deblurring.

**Keywords:** image restoration, deblurring, denoising, regularization

## 1. INTRODUCTION

In this paper, we propose an image deblurring and denoising algorithm using a non-local regularization constraint. We compare the proposed iterative reconstruction algorithm with algorithms using a total variation (TV) regularization constraint, and with algorithms consisting of separate denoising and deblurring.

We consider images that are degraded by blur and noise inherent in the image capturing process. Blurring involves suppression of high-frequency detail in the image and may be due to the camera optics and imaging sensor, as well as other factors. Random noise may be caused by the sensor (e.g. photon noise), by the electronics, by analog-to-digital conversion, and other factors. We consider the following observation model for the degraded image data $\mathbf{y}$: $\mathbf{y} = \mathbf{H}\,\mathbf{x} + \mathbf{v}$, where $\mathbf{x}$ is the "ideal" image, $\mathbf{H}$ is the combined lens and sensor blur p.s.f. and $\mathbf{v}$ is additive noise. Our goal is to estimate $\mathbf{x}$, given $\mathbf{y}$ and assuming we approximately know $\mathbf{H}$. Hence, we aim to reduce the blur and noise in $\mathbf{y}$, while avoiding introduction of artifacts or further loss of detail. A common image restoration approach[1,2] is to minimize a cost function such as:

$$\Phi(\mathbf{x}) = \|\mathbf{y} - \mathbf{H}\,\mathbf{x}\|^2 + \lambda\, C(\mathbf{x}). \tag{1}$$

The first term in $\Phi(\mathbf{x})$ ensures fidelity of the estimate to the data $\mathbf{y}$. The second term is a *regularizing* or *stabilizing constraint* $C(\mathbf{x})$, which may exploit prior knowledge of the underlying image and can be used to avoid noise amplification during deblurring. A regularization parameter $\lambda$ controls the trade-off between the two terms.

A common regularization constraint takes the form $C(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|^2$, combining a linear high-pass filter $\mathbf{L}$ with the $L_2$ norm. This approach enforces smoothness of the solution and suppresses noise by penalizing high-frequency components. More recently, regularization based on the total variation (TV) norm has been introduced. Total variation minimization was originally introduced for noise reduction[3,4] and has also been used for image deblurring[5] and super-resolution image reconstruction.[6] The TV constraint is computed as the $L_1$ norm of the gradient magnitude: $C(\mathbf{x}) = \|\nabla\mathbf{x}\|_1$. TV minimization is better able to preserve sharp edges and fine detail

in the image. Both approaches lead to regularization constraints defined in terms of pixel values in a local neighborhood, and have the potential for oversmoothing and re-introducing some blur.

A wide range of image denoising methods has been proposed in the past that could be applied in a separate noise suppression stage before or after deblurring, to alleviate the concerns of noise amplification. The *non-local means* (NLM) filter[7] is a recent approach, that is able to remove additive noise while preserving sharp edges and fine texture details. The NLM filter, described in the next section, has been shown to perform very well, especially on textured images. An iterative version of the NLM filter has been proposed in Ref. 8. A very general framework that includes local and non-local regularization as well as the NLM filter as special cases has been proposed in Ref. 9. However, the NLM approach has sofar been applied to image denoising only.

We propose to use the NLM principle in the form of a regularization constraint in image deblurring. The NLM-based regularization constraint has the potential to suppress noise while preserving edges and other detail, more effectively than conventional regularization constraints. Compared to a scheme with a denoising stage separate from deblurring, this method has the advantage of lower computational cost, as well as better performance.

This paper is organized as follows. In Section 2, we briefly review existing deblurring methods, and describe NLM-based denoising in more detail. These deblurring and denoising methods will be used in experimental comparisons with our proposed method. In Section 3, we introduce the proposed deblurring method with non-local regularization. In Section 4, we provide experimental results.

## 2. DEBLURRING AND NLM-BASED DENOISING

### 2.1 Iterative Regularized Deblurring

Common iterative minimization methods[1,2] can be employed to compute estimates of $\mathbf{x}$ based on Eq. 1. In the steepest descent approach, the estimate $\hat{\mathbf{x}}^k$ at the $k^{th}$ iteration is updated in the (opposite) direction of the gradient of the cost function.

Without the regularization constraint $C(\mathbf{x})$, the estimate of $\mathbf{x}$ is the unconstrained least-squares (LS) solution. Iterative minimization based on the gradient descent approach is as follows:

$$\hat{\mathbf{x}}^{k+1} = \hat{\mathbf{x}}^k - \beta\,[\mathbf{H}^T(\mathbf{H}\,\hat{\mathbf{x}}^k - \mathbf{y})], \tag{2}$$

where $\beta$ is the step-size and $\hat{\mathbf{x}}^0 = \mathbf{y}$. In our experiments, $\beta$ is held constant. At the cost of additional computation, an optimal value of $\beta$ can be computed at each iteration. The method of conjugate gradients and other more advanced minimization methods can be used alternatively.

An iterative scheme for deblurring with TV regularization can be defined as follows:

$$\begin{aligned}\hat{x}^{k+1}(i,j) = \hat{x}^k(i,j) &- \beta\,[H(-i,-j) * (H(i,j) * \hat{x}^k(i,j) - y(i,j))] \\ &- 0.5\,\beta\,\lambda\;[\sum_{l,m\in S} \alpha_{i,j,l,m}(\hat{\mathbf{x}}^k)\;\;(\hat{x}^k(i,j) - \hat{x}^k(i+l,j+m))],\end{aligned} \tag{3}$$

where $\hat{x}^k(i,j)$ is the estimate at a pixel location $(i,j)$ at iteration $k$, and $S = \{(1,0),(0,1),(-1,0),(0,-1)\}$ is a local neighborhood. The $\alpha_{i,j,l,m}$ values can be seen as data-adaptive weights for the local neighbors in $S$, and are given by:

$$\alpha_{i,j,l,m}(\mathbf{x}) = \frac{1}{D_{i,j}(\mathbf{x})} + \frac{1}{D_{l,m}(\mathbf{x})}. \tag{4}$$

Here, $D_{i,j}$ is the so-called *local variation* at pixel $(i,j)$ and is given by:

$$D_{i,j}(\mathbf{x}) \equiv \sqrt{\sum_{r,s\in S} [x(i,j) - x(i+r,j+s)]^2 + \epsilon}. \tag{5}$$

This approach is based on a discrete version of the TV constraint[4] defined by $C(\mathbf{x}) = \sum_{i,j} D_{i,j}(\mathbf{x})$. The above iteration is a so-called fixed-point iteration scheme, where computation of the TV term lags one iteration behind, i.e. the weights $\alpha_{i,j,l,m}$ are computed on the basis of the previous estimate $\hat{\mathbf{x}}^k$ and are updated after each iteration.

### 2.2 Denoising with Non-Local Means Filter and Iterative Non-Local Means

The discrete NLM filter[7] is a form of weighted averaging over a set $N$ of non-local *neighbor pixels*, which may include pixels from a large window or even the entire image. The $n^{th}$ neighboring pixel is offset from the center pixel vertically by $L_n$ pixels and horizontally by $M_n$ pixels in the image. The NLM filter is then given by:

$$\hat{x}(i,j) = \frac{1}{\sum_{n\in N} w_{i,j,n}(\mathbf{y})} \sum_{n\in N} w_{i,j,n}(\mathbf{y})\; y(i+L_n, j+M_n). \tag{6}$$

The process includes the definition of a patch of pixels centered on the pixel of interest at $(i,j)$. This patch of pixels is compared to other patches centered on the neighboring pixels used for averaging. The size of each patch is $(2A+1)\times(2B+1)$ pixels. The data-adaptive denoising weight for the $n^{th}$ neighbor of pixel $(i,j)$ is then defined by:

$$w_{i,j,n}(\mathbf{y}) = exp\{-\frac{1}{h^2}\sum_{u=-A}^{A}\sum_{v=-B}^{B} g(u,v)\;[y(i+u,j+v) - y(i+u+L_n, j+v+M_n)]^2\}, \tag{7}$$

where $h$ is a given parameter that controls the denoising strength. The larger the value of $h$ the higher the strength of denoising. Each weight $w_{i,j,n}$ is based on the pixel differences between the center patch and a neighboring patch, weighted by a (Gaussian) kernel $g(u,v)$. Note that the denoising weights are computed from the noisy image $\mathbf{y}$.

The basic NLM filter can be extended by applying it in an iterative manner (as also proposed in Ref. 8):

$$\hat{x}^{k+1}(i,j) = \hat{x}^k(i,j) - \beta[\hat{x}^k(i,j) - \frac{1}{\sum_n w_{i,j,n}(\hat{\mathbf{x}}^k)} \sum_n w_{i,j,n}(\hat{\mathbf{x}}^k)\; y(i+L_n, j+M_n)], \tag{8}$$

with $\hat{\mathbf{x}}^0 = \mathbf{y}$. Note that the denoising weights $w_{i,j,n}(\hat{\mathbf{x}})$ are defined similarly as in Eq. 7, except that they are computed on the previous solution at iteration $k$, i.e. the (partially) denoised image $\hat{\mathbf{x}}^k$. Hence, the denoising weights are continuously improved as the iterative scheme progresses. At the same time, the noisy input image $\mathbf{y}$ is always used as the data for the actual denoising step in each iteration.

We can derive the above iterative denoising scheme indirectly by introducing a cost function that is subsequently minimized with a gradient descent method. We define our cost function as follows:

$$\Gamma(\hat{\mathbf{x}}) = \sum_{i,j}[\hat{x}(i,j) - \frac{1}{\sum_{n\in N} w_{i,j,n}(\hat{\mathbf{x}})} \sum_{n\in N} w_{i,j,n}(\hat{\mathbf{x}}) y(i+L_n, j+M_n)]^2, \tag{9}$$

with denoising weights $w_{i,j,n}(\hat{\mathbf{x}})$ defined as before. Taking the gradient of $\Gamma(\hat{\mathbf{x}})$ with respect to $\hat{\mathbf{x}}$ and subsequent derivation (involving an approximation step) results in the gradient descent scheme defined in Eq. 8.

## 3. DEBLURRING AND DENOISING WITH NON-LOCAL REGULARIZATION

In this section, we present a regularization constraint based on the Non-Local Means principle and propose a novel deblurring and denoising scheme using NLM-based regularization. We start by formulating our regularization constraint as follows:

$$C(\mathbf{x}) = \sum_{i,j}[x(i,j) - \frac{1}{\sum_{n\in N} w_{i,j,n}(\mathbf{x})} \sum_{n\in N} w_{i,j,n}(\mathbf{x})\; x(i+L_n, j+M_n)]^2. \tag{10}$$

The denoising weights $w_{i,j,n}(\mathbf{x})$ are defined as in Eq. 7, except they are defined on the deblurred and denoised solution image. Comparing Eq. 9 with Eq. 10, it is important to note that the previous cost function involved the observation $y$. This coupling with the observation data was important to prevent iteratively smoothing out all detail of the image (as pointed out in Ref. 8). In the formulation in this section, coupling with the observation data is instead enforced by the data fidelity term in Eq. 10.

Using a gradient descent approach, we arrive at the following iterative minimization scheme:

$$\hat{x}^{k+1}(i,j) = \hat{x}^k(i,j) - \beta[H(-i,-j) * (H(i,j) * \hat{x}^k(i,j) - y(i,j))] \\ - \beta\lambda[\hat{x}^k(i,j) - \frac{1}{\sum_n w_{i,j,n}(\hat{\mathbf{x}}^k)} \sum_n w_{i,j,n}(\hat{\mathbf{x}}^k)\, \hat{x}^k(i+L_n, j+M_n)]. \quad (11)$$

Note that the denoising weights are computed on the (partially) denoised and deblurred image at the previous iteration $\hat{\mathbf{x}}^k$. Hence, the weights are continuously improved as the iterative scheme progresses. Also, $\hat{\mathbf{x}}^k$ is now used as the data for the denoising step in each iteration, unlike Eq. 8. This is because we must compute an estimate of the denoised *and* deblurred image. The data fidelity term of the cost function and proposed algorithm will ensure consistency with the input image data $\mathbf{y}$, as mentioned above. As in the previous section, derivation of this iterative scheme requires an approximation in which specific terms are neglected.

## 4. EXPERIMENTAL RESULTS

We experimentally compared the following deblurring algorithms:

1. Unconstrained least-squares deblurring method (**LS**), see Eq. 2;
2. Basic NLM filter followed by LS deblurring (**NLM+LS**), see Eq. 6 and Eq. 2;
3. Iterative NLM denoising followed by LS deblurring (**INLM+LS**), see Eq. 8 and Eq. 2;
4. Deblurring with total variation regularization (**CTV**), see Eq. 3;
5. Proposed deblurring with non-local regularization (**CNL**), see Eq. 11.

The test images we used in our experiments are shown in Fig. 1. The first two images in our test set are the well-known "Barbara" and "Lena" images. The next three images ("Canvas", "Carpet" and "Wallpaper") are texture images from the Outex Texture Database.[10] The last image ("Panel") is an image obtained by high-resolution imaging of an LCD panel.

A series of experiments was performed, in which we changed the strength of the blur and noise in the degraded images. In all our experiments, we generated simulated degraded images from the original images by blurring with a Gaussian kernel followed by adding white Gaussian noise. We used the following settings:

- **Experiment 1**: Gaussian blur with $\sigma = 1.5$ and noise standard deviation 7.
- **Experiment 2**: Gaussian blur with $\sigma = 1.5$ and noise standard deviation 15.
- **Experiment 3**: Gaussian blur with $\sigma = 2.0$ and noise standard deviation 10.

The **CTV** and **CNL** methods were always applied for 50 iterations. All methods involving unconstrained deblurring, i.e. **LS**, **NLM+LS** and **INLM+LS**, were applied for 10 iterations only. Generally, the **LS** method is sensitive to noise, and a high number of iterations leads to unacceptable noise amplification. In all deblurring iterations, we set $\beta = 0.5$.

We used the following fixed parameter settings for methods involving NLM filtering or non-local regularization. We use a $21 \times 21$ pixel search window, i.e. offsets $L_n$ and $M_n$ range between $-10$ and 10, resulting in a total of 441 neighbors. In **NLM**, we used patches of $11 \times 11$ pixels (i.e. $A = B = 5$); in **INLM** and **CNL**, we used patches of $9 \times 9$ pixels (i.e. $A = B = 4$). The iterative NLM denoising method (**INLM**) is applied for 10 iterations, with $\beta = 0.2$.

The parameter that controls denoising strength in NLM and non-local regularization methods, $h$, was optimized in each experiment to obtain the best PSNR. The regularization parameter in **CTV** and **CNL**, $\lambda$, was also optimized in each experiment to obtain the best PSNR. The optimized parameter settings in experiment 1, 2 and 3 are provided in Table 1, 2 and 3.

The PSNR results for experiments 1, 2 and 3 are provided in Table 4, 5 and 6, the best results indicated in bold. These tables show that the proposed method outperformed other methods in terms of PSNR in all cases. The **LS** method provides deblurring, but no noise control. In the high noise case (experiment 2), the PSNR

deteriorates in several cases. In all cases, deblurring with total variation regularization (**CTV**) outperforms **LS**, the margin being the largest, almost 6 dB, in experiment 2 (high noise condition). Methods **NLM+LS** and **INLM+LS** have similar PSNR results to each other. Both outperform **CTV** in most cases, with a margin up to about 1.5 dB. Using NLM-based denoising prior to **LS** deblurring provides a PSNR increase of up to 7 dB. The proposed deblurring method with non-local regularization (**CNL**) achieves the best PSNR. **CNL** outperforms **CTV**, by up to approximately 2.6 dB in experiment 1. **CNL** outperforms the combinations methods **NLM+LS** and **INLM+LS** by a smaller margin, up to about 1 dB in experiment 2.

A subset of the image results of experiment 2 are provided in Fig. 2, Fig. 3 and Fig. 4. The images show a cropped area to improve visibility. These images illustrate the noise amplification of the LS deblurring method. The figures show the result of denoising by NLM prior to deblurring, as well as the result of the combination of NLM and LS deblurring. Finally, the results of deblurring with total variation regularization and non-local regularization are shown. NLM filtering seems to provide strong noise reduction, but suffers from loss of edge and texture detail. Total variation regularization preserves more detail; however, it suffers from the well-known "staircase" effect, in that the resulting images appear to contain piecewise constant "patches". It appears that non-local regularization may provide a better balance between noise reduction and detail preservation.

In a final experiment (**Experiment 4**), we investigated reducing the computational complexity of the NLM-based methods and CNL. In this experiment, we consider application of these methods to image data with highly regular texture patterns, such as the "Panel" image. This type of texture image contains a pattern that repeats periodically in the horizontal and vertical direction. We assumed prior knowledge of the horizontal period $l_x$ and vertical period $l_y$, e.g. $l_x = l_y = 12$ in the "Panel" image. We restricted the NLM-based methods, including the proposed CNL method, to utilize a very small set of 8 non-local neighbors in addition to the center pixel. This small set is defined by:

$$(M_n, L_n) \in \{(-l_x, -l_y), (0, -l_y), (l_x, -l_y), (-l_x, 0), (0, 0), (l_x, 0), (-l_x, l_y), (0, l_y), (l_x, l_y)\}.$$

We applied a Gaussian blur with $\sigma = 1.5$ to the image and added white Gaussian noise with standard deviation TBD. In this test, iterative deblurring methods were all applied for 100 iterations with $\beta = 0.5$. In **CTV**, we used $\lambda = 0.1$, while in **CNL**, we used $\lambda = 0.3$. In all NLM-based methods, we used a patch-size of 5x5 pixels (i.e. $A = B = 2$). INLM was applied for 20 iterations with $\beta = 0.2$. In **NLM+LS**, **INLM+LS** and **CNL**, we used $h = 25$, $h = 18$ and $h = 25$ respectively.

The PSNR results are listed in Table 7 and selected visual results are shown in Fig. 5. These results again indicate that NLM-based denoising prior to deblurring can be beneficial. Iterative NLM-based denoising improves performance further. The proposed deblurring method with non-local regularization (**CNL**) again performs best.

This experiment suggests that, in certain applications, the computational cost of NLM filtering or non-local regularization can be reduced, while retaining good performance.

## 5. CONCLUSIONS

We have investigated the use of the non-local means (NLM) denoising approach in the context of image deblurring. We have proposed a novel deblurring approach that utilizes a non-local regularization constraint. We compared our proposed algorithm to conventional deblurring methods, including total variation (TV) regularization. We also compared our algorithm to combinations of the NLM-based filter followed by conventional deblurring methods. The use of NLM-based filtering and regularization seems beneficial in the context of deblurring, providing improved noise suppression, while reducing the risk of oversmoothing or detail suppression. Furthermore, in our experiments, the proposed deblurring algorithm with non-local regularization outperformed other methods, including TV regularization, in terms of PSNR.

## REFERENCES

[1] J. Biemond, R. L. Lagendijk and R. M. Merserau, "Iterative methods for image deblurring," *IEEE Proceedings*, vol. 78, no. 5, May 1990.

[2] M. Banham and A. K. Katsaggelos, "Digital Image Restoration," *IEEE Signal Processing Magazine*, vol. 14, no. 2, May 1997, pp. 24-41.

[3] L. I. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, pp. 259-268, 1992.

[4] T. F. Chan, S. Osher and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. on Image Processing*, vol. 10, no. 2, February 2001.

[5] Y. Li and F. Santosa, "A computational algorithm for minimizing total variation in image restoration," *IEEE Trans. on Image Processing*, vol. 5, no. 6, June 1996.

[6] M. K. Ng et al., "A total variation regularization based super-resolution reconstruction algorithm for digital video," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, Article ID 74585.

[7] A. Buades, B. Coll and J.-M. Morel, "A non-local algorithm for image denoising," *IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR 2005)*, San Diego, CA, USA, June 20-25, 2005.

[8] T. Brox, O. Kleinschmid and D. Cremers, "Efficient nonlocal means for denoising of textural patterns," *IEEE Trans. on Image Processing*, vol. 17, no. 7, July 2008.

[9] 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*, vol. 17, no. 7, July 2008.

[10] "Outex Texture Database", University of Oulu, Finland, http://www.outex.oulu.fi/.

Table 1. Parameter settings for experiment 1 (blur $\sigma = 1.5$ and noise st.dev. $= 7$).

| Image | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|
| Barbara | $h = 60$ | $h = 50$ | $\lambda = 1.0$ | $h = 60$, $\lambda = 0.3$ |
| Lena | $h = 60$ | $h = 50$ | $\lambda = 1.0$ | $h = 80$, $\lambda = 0.3$ |
| Wallpaper | $h = 60$ | $h = 60$ | $\lambda = 1.0$ | $h = 70$, $\lambda = 0.3$ |
| Canvas | $h = 60$ | $h = 50$ | $\lambda = 1.0$ | $h = 60$, $\lambda = 0.3$ |
| Carpet | $h = 80$ | $h = 70$ | $\lambda = 1.0$ | $h = 80$, $\lambda = 0.5$ |

Table 2. Parameter settings for experiment 2 (blur $\sigma = 1.5$ and noise st.dev. $= 15$).

| Image | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|
| Barbara | $h = 150$ | $h = 100$ | $\lambda = 3.0$ | $h = 100$, $\lambda = 0.5$ |
| Lena | $h = 150$ | $h = 100$ | $\lambda = 3.0$ | $h = 120$, $\lambda = 0.5$ |
| Wallpaper | $h = 120$ | $h = 100$ | $\lambda = 3.0$ | $h = 110$, $\lambda = 0.3$ |
| Canvas | $h = 110$ | $h = 90$ | $\lambda = 3.0$ | $h = 120$, $\lambda = 0.3$ |
| Carpet | $h = 140$ | $h = 120$ | $\lambda = 3.0$ | $h = 140$, $\lambda = 0.5$ |

Table 3. Parameter settings for experiment 3 (blur $\sigma = 2.0$ and noise st.dev. $= 10$).

| Image | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|
| Barbara | $h = 110$ | $h = 70$ | $\lambda = 2.0$ | $h = 80$, $\lambda = 0.5$ |
| Lena | $h = 110$ | $h = 70$ | $\lambda = 2.0$ | $h = 100$, $\lambda = 0.5$ |
| Wallpaper | $h = 100$ | $h = 70$ | $\lambda = 2.0$ | $h = 80$, $\lambda = 0.3$ |
| Canvas | $h = 90$ | $h = 60$ | $\lambda = 2.0$ | $h = 90$, $\lambda = 0.3$ |
| Carpet | $h = 110$ | $h = 80$ | $\lambda = 2.0$ | $h = 100$, $\lambda = 0.5$ |

Table 4. PSNR results for experiment 1 (blur $\sigma = 1.5$ and noise st.dev. $= 7$).

| Image | Degraded | **LS** | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|---|---|
| Barbara | 23.05 | 23.28 | 24.03 | 24.13 | 23.82 | **24.30** |
| Lena | 26.93 | 27.35 | 30.19 | 30.76 | 29.67 | **31.79** |
| Wallpaper | 25.07 | 27.29 | 29.00 | 29.09 | 27.51 | **30.12** |
| Canvas | 26.86 | 27.75 | 30.00 | 29.85 | 29.44 | **30.60** |
| Carpet | 20.34 | 24.07 | 24.35 | 24.31 | 24.17 | **24.70** |

Table 5. PSNR results for experiment 2 (blur $\sigma = 1.5$ and noise st.dev. $= 15$).

| Image | Degraded | **LS** | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|---|---|
| Barbara | 21.17 | 20.29 | 23.65 | 23.64 | 23.15 | **23.88** |
| Lena | 23.27 | 21.88 | 28.96 | 28.85 | 27.58 | **29.86** |
| Wallpaper | 22.36 | 21.86 | 26.39 | 26.56 | 25.66 | **27.49** |
| Canvas | 23.24 | 21.98 | 26.90 | 26.80 | 27.27 | **28.34** |
| Carpet | 19.23 | 20.89 | 22.47 | 22.47 | 22.45 | **22.86** |

Table 6. PSNR results for experiment 3 (blur $\sigma = 2.0$ and noise st.dev. $= 10$).

| Image | Degraded | **LS** | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|---|---|
| Barbara | 21.90 | 22.13 | 23.59 | 23.56 | 23.16 | **23.75** |
| Lena | 24.66 | 25.11 | 28.93 | 28.84 | 27.76 | **29.39** |
| Wallpaper | 22.49 | 24.46 | 26.26 | 26.18 | 24.78 | **26.62** |
| Canvas | 24.34 | 25.05 | 27.26 | 27.10 | 27.03 | **27.66** |
| Carpet | 17.93 | 21.09 | 21.42 | 21.42 | 21.33 | **22.06** |

Table 7. PSNR results for experiment 4.

| Image | Degraded | **LS** | **NLM+LS** | **INLM+LS** | **CTV** | **CNL** |
|---|---|---|---|---|---|---|
| Panel | 21.33 | 22.41 | 24.10 | 24.55 | 22.96 | 25.03 |

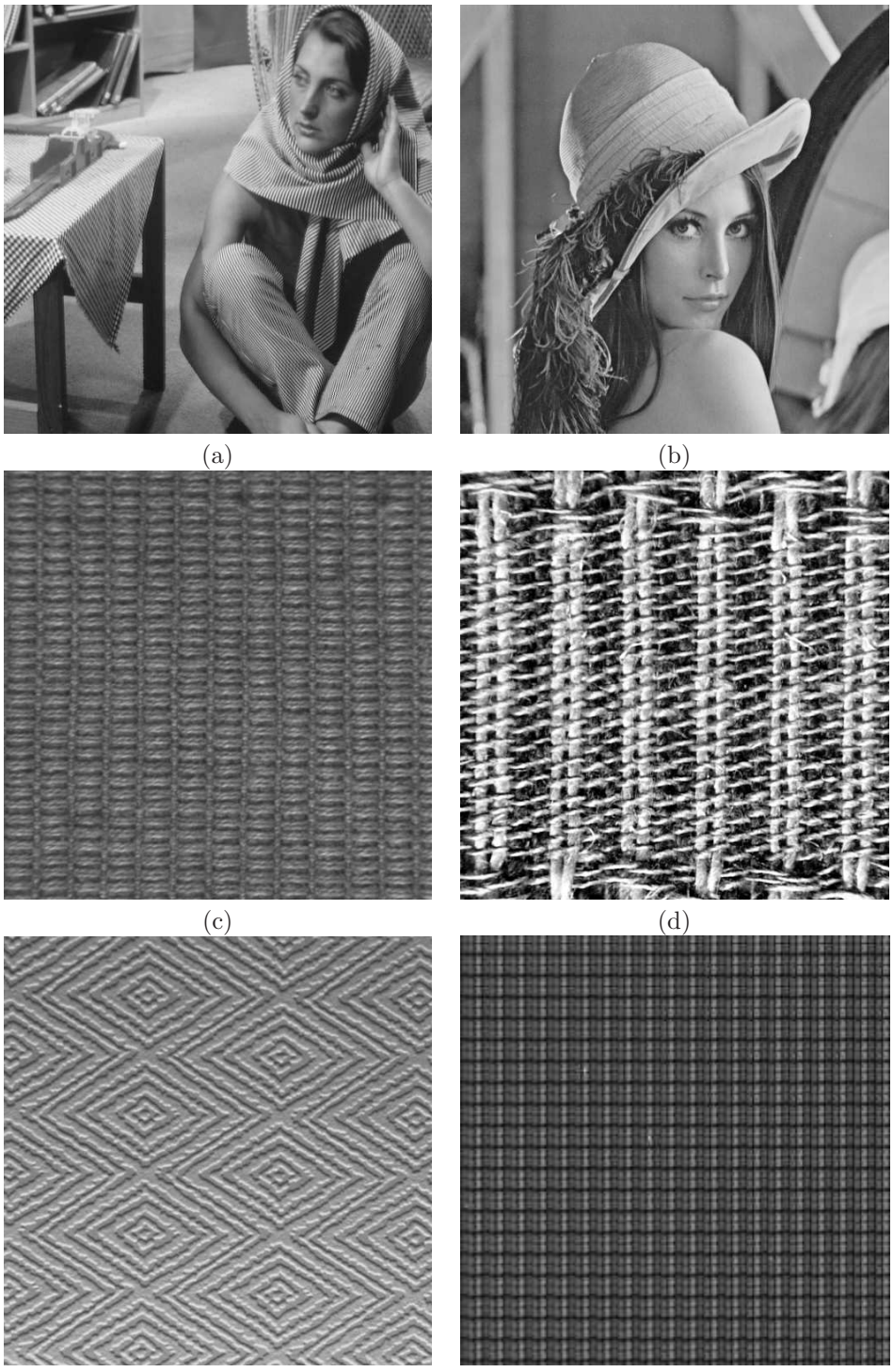

Figure 1. Original images: a) Barbara; b) Lena; c) Canvas; d) Carpet; e) Wallpaper; f) Panel. The size of the "Panel" image is 240x240 pixels; the size of all other images is 512x512.

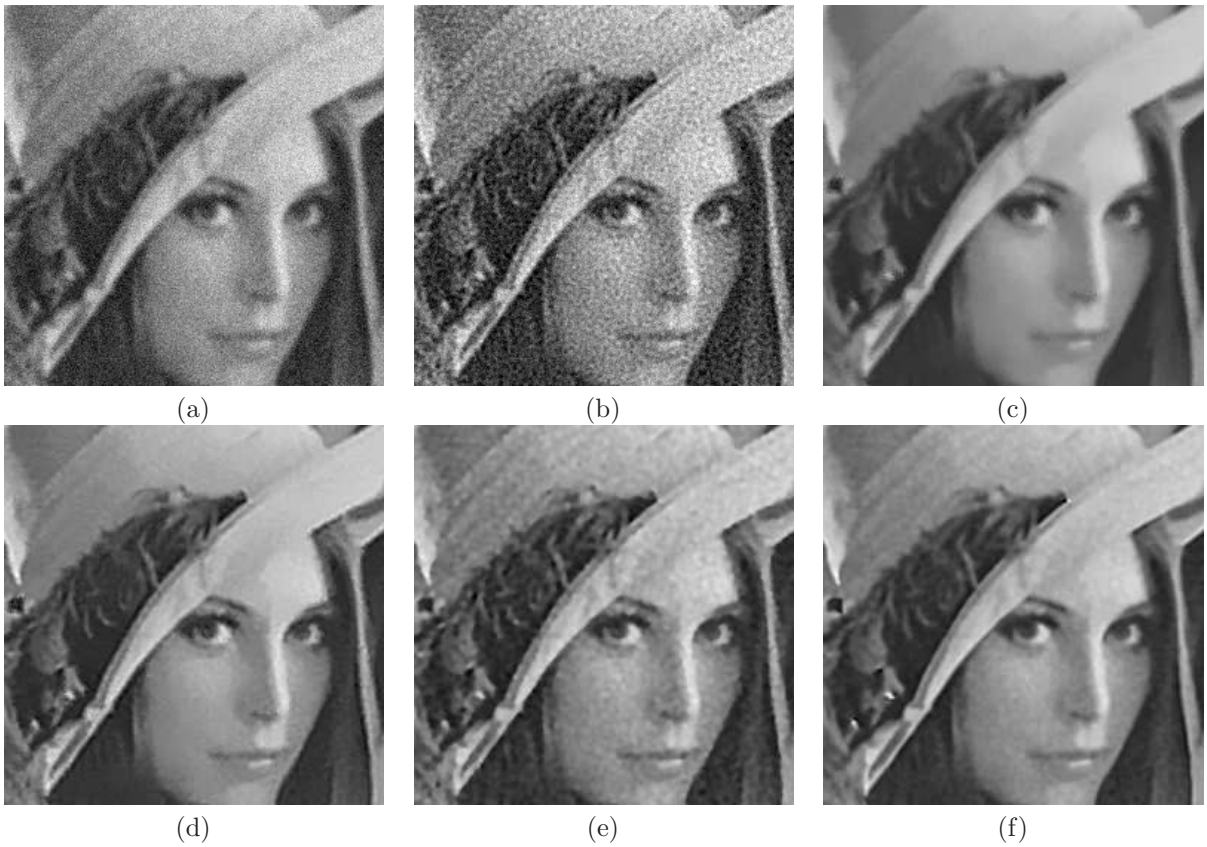

Figure 2. “Lena” image results for experiment 2 (256x256 pixels crop): a) blurry and noisy input; b) LS; c) NLM; d) NLM+LS; e) CTV; f) CNL.

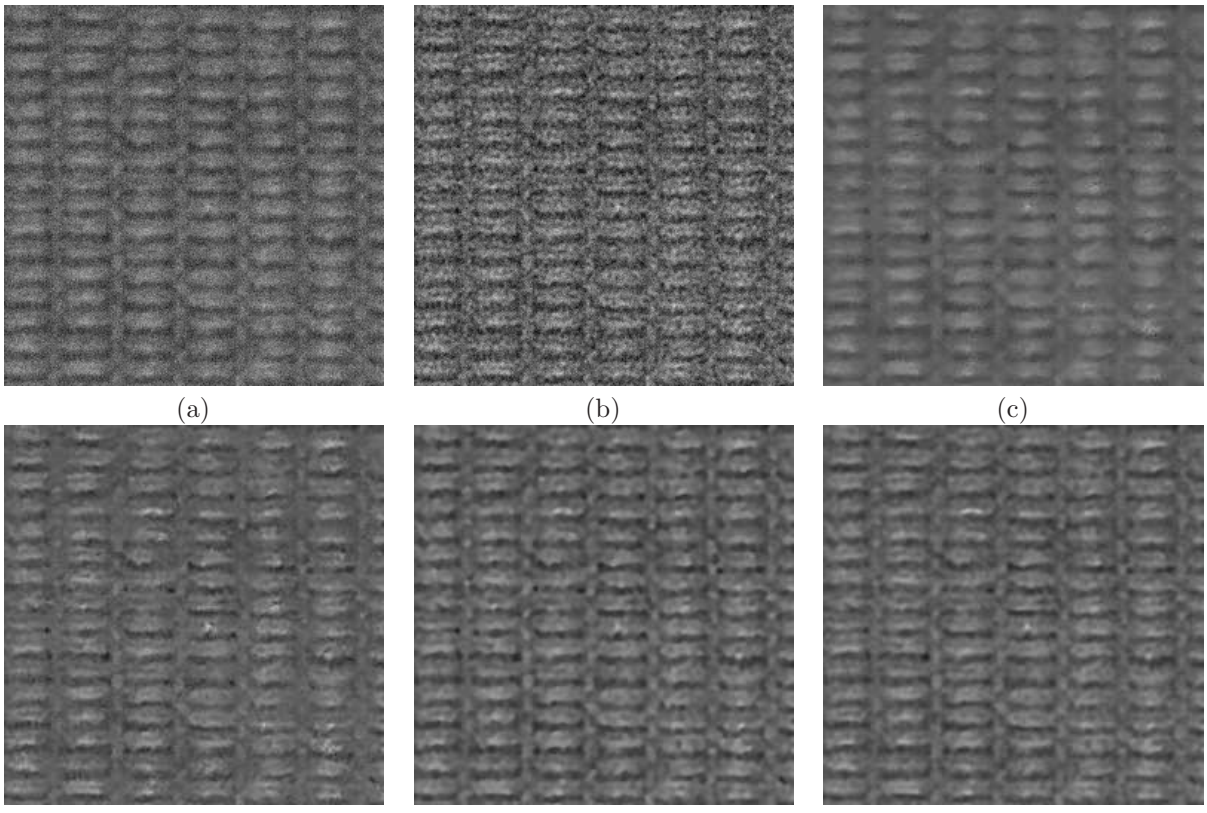

Figure 3. “Canvas” image results for experiment 2 (256x256 pixels crop): a) blurry and noisy input; b) LS; c) NLM; d) NLM+LS; e) CTV; f) CNL.

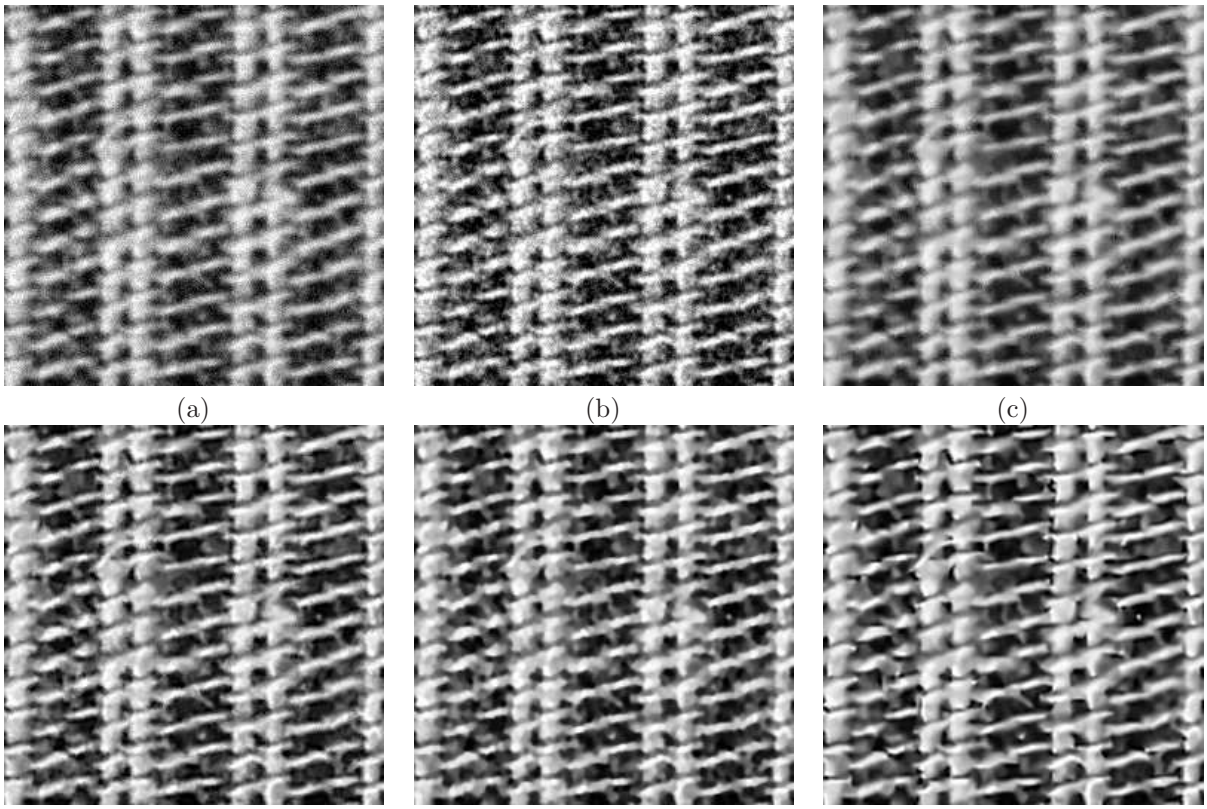


Figure 4. “Carpet” image results for experiment 2 (256x256 pixels crop): a) blurry and noisy input; b) LS; c) NLM; d) NLM+LS; e) CTV; f) CNL.

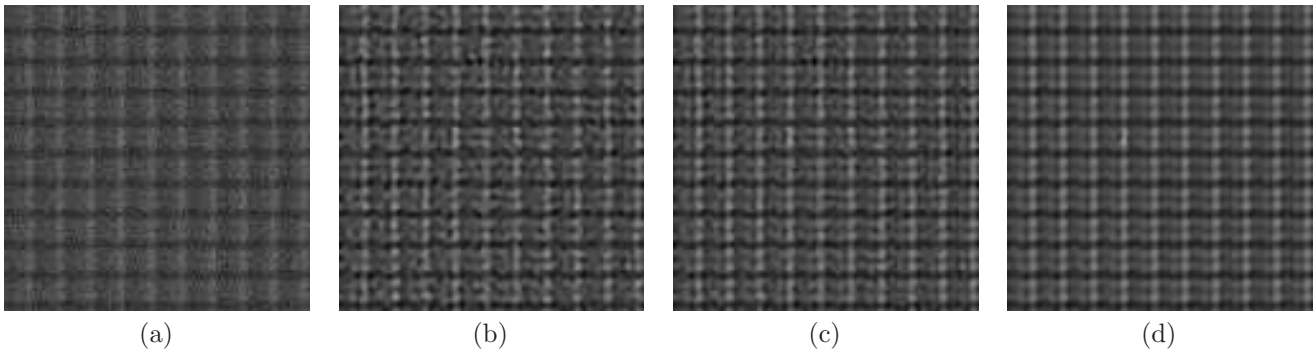


Figure 5. “Panel” image results (120x120 pixels crop): a) blurry and noisy input; b) CTV (22.96 dB); c) NLM+LS (24.10 dB); d) CNL (25.03 dB).