Jin *et al. EURASIP Journal on Image and Video Processing* (2015) 2015:26
DOI 10.1186/s13640-015-0082-5

EURASIP Journal on Image and Video Processing
a SpringerOpen Journal

RESEARCH Open Access

Bilateral image denoising in the Laplacian subbands

Bora Jin[1], Su Jeong You[2] and Nam Ik Cho[1*]

Abstract

This paper presents an image denoising algorithm, which applies bilateral filtering (BLF) in the Laplacian subbands. It is noted that the subband images have wider area of photometric similarity than the original, and hence, they can be more benefited by the BLF than the original. Specifically, an image is Gaussian filtered to obtain a low band image, and the low band image is subtracted from the original to have the high band signal, which forms the Laplacian subbands. For the high band image denoising, we derive an adaptive kernel that is dependent on the edge intensity and photometric similarity of subband images. The high band image is convolved with this kernel and then added to the denoised low band signal, which produces the denoised image. We also propose to process the denoised high band signal by the gradient histogram preservation method, for sharpening the edges with less noise amplification. Experimental results show that the proposed denoising method provides higher PSNR than the original BLF and other multi-resolution denoising algorithms. Since the high band image is also effectively denoised in this process, the sharpened image by high band modification is also visually more pleasing when compared with the results of the conventional sharpening methods.

Keywords: Bilateral filter; Denoising; Image enhancement

1 Introduction

Image denoising is a fundamental process in image formation, transmission and display systems, and thus a huge number of methods have been developed. The overview of classical linear filtering and some of recently developed nonlinear methods can be found in [1], where the relations of different nonlinear methods are also well explained. For suppressing the noise while keeping the edges, the state-of-the-art methods use the similarities of pixels locally or globally. For example, a simple yet effective local-similarity method is the bilateral filtering (BLF) [2, 3], and the representatives of global similarity methods are non-local means (NLM) filtering [4] and block matching 3D (BM3D) algorithm [5].

The edge-sharpening is also an important topic in image processing, which enhances the visual quality of images [6–9]. One of the classical edge enhancement methods is to use the "unsharpening filter", where an image is low pass filtered and subtracted from the original, which leaves the high band signal that contains the edges. The high band signal is then amplified and added to the low pass filtered image, which is the edge-sharpened result. When the Gaussian filter is used for the low pass filtering, its subtraction from the original is the Laplacian of Gaussian, and thus the subband images so formed are called the Laplacian Pyramids [6]. In this process, since the noise in the high band can also be amplified, it is necessary to denoise all the subband images in the Laplacian pyramid.

In this paper, we modify the BLF for denoising the Laplacian subband images, which is aimed as a new denoising algorithm that works better than the original BLF and also as an efficient method of suppressing high band noise when sharpening the edges in the Laplacian pyramids. The idea of applying the BLF to the Laplacian subband images is based on the observation that the BLF works better when there are more photometric similarities in the images, and the subband images have wider area of photometric similarity than the original. However, since the properties of subband images are different from the original, we need to design a new filtering kernel, which is one of the modifications proposed in this

*Correspondence: nicho@snu.ac.kr
[1] Department of Electrical and Computer Engineering, INMC, Seoul National University, 1 Gwanangno, Gwanak-gu, 151-742 Seoul, Korea
Full list of author information is available at the end of the article

paper. Also, for the edge enhancement with noise suppression, we propose a new enhancement technique which restores the strength of edges that are smoothed by filtering and then adds the restored edges to the high band signal.

Experiments on the images corrupted by pseudo white Gaussian noise, shot noise, and mixture noise are performed, and it is shown that the proposed method improves PSNR than the original BLF and other local neighbor methods based on the subband decomposition. For evaluating the results for the real camera noises, we capture noisy images under low light conditions and compare the visual qualities. Also, considering that the addition of multiple tripod captured images as a reference, we compare the PSNR. Comparisons on real noisy images also show that the BLF in the Laplacian subbands improves the denoising performance than the original BLF. Also, modification of the edge coefficients in the high band gives sharpened images with less noise amplification than the conventional edge-sharpening method in the Laplacian pyramids. When compared with the nonlocal approaches such as NLM [4] and BM3D [5], the proposed method shows lower or similar PSNR for the white Gaussian noise like any other local adaptive filters. However, for the real noise and mixed noise, the proposed method shows comparable or sometimes higher PSNR than the nonlocal methods while requiring much less computations due to the nature of local filtering. In summary, the proposed method shows better results than the conventional BLF and other subband filtering schemes such as [10, 11], which are the representatives of local adaptive filtering methods, and shows comparable results to the nonlocal methods for the nonstationary noises while requiring less computations. Hence, the Laplacian subband BLF can be a reasonable choice for denoising and enhancing the images when fast or real-time implementation is needed.

2 Related works

2.1 Laplacian subbands

When a signal is low pass filtered and the filtered output is subtracted from the original, then we obtain a high band signal. If this process is repeated for the low pass filtered signal, then we obtain a set of subband signals. The Laplacian pyramid for an image is constructed in this manner, where the low pass filter is a Gaussian filter with appropriate kernel bandwidth. More specifically, for a given image I, the Gaussian filter is applied iteratively with downsampling at every step. This process can be described as

$$\begin{aligned} G_0 &= I, \\ G_{k+1} &= \downarrow_2 (Gaussian(G_k)) \quad for \quad k = 0, \ldots, n-1 \end{aligned} \tag{1}$$

where $\downarrow_2 (\cdot)$ denotes the downsampling by 2 and $Gaussian(\cdot)$ is the Gaussian filtering. Then, the Laplacian subbands are defined as

$$\begin{aligned} L_{k+1} &= G_k - \uparrow_2 (G_{k+1}) \qquad for \quad k = 0, \ldots, n-1 \\ L_{n+1} &= G_n \end{aligned} \tag{2}$$

where $\uparrow_2 (\cdot)$ denotes the upsampling by 2 and n is the level of pyramid. In this paper, we use just two levels of Laplacian subband ($n = 1$), where L_1 denotes the high-frequency subband and $L_2 = G_1$ represents the low-frequency subband.

2.2 Bilateral filter

The bilateral filter is a nonlinear filter that considers both of spatial and photometric distances of neighboring pixels. Specifically, the filter output is defined as

$$J(p) = \frac{1}{W} \sum_{q \in N_p} w(p,q) I(q) \tag{3}$$

where p and q denote pixel positions, N_p is the neighbor of p, $I(q)$ is the intensity of input image at a pixel q, W is the normalizing factor $W = \sum_{q \in N_p} w(p,q)$, and $w(p,q)$ is the kernel of the BLF defined as [2]

$$w(p,q) = \exp\left(-\frac{\|p-q\|^2}{2\sigma_d^2}\right) \exp\left(-\frac{\|I(p)-I(q)\|^2}{2\sigma_r^2}\right) \tag{4}$$

where σ_d is the bandwidth for the spatial distance and σ_r for the photometric distance. For successfully reducing noise variance while keeping the edges, it is important to find the balance between σ_d and σ_r, and also to find an appropriate size of the neighbor.

3 Bilateral filtering in the Laplacian subbands

3.1 Example of subband BLF for a 1-D signal

We first show a simple denoising example with a synthetic 1-D signal, which motivates to apply the BLF to Laplacian subbands. Note that the kernel of bilateral filter in Eq. (4) is consisted of two terms, i.e., geometric and photometric terms. From this, we can see that the photometric weights would be kept large for wider area when a pixel p is in the flat area where $I(p)$ and $I(q)$ are similar, and hence many neighboring pixels can contribute for the denoising. On the contrary, when the pixel is in the non-flat area where $\|I(p) - I(q)\|$ is large, the photometric weights diminish and thus the neighboring pixels less contribute for the denoising.

Figure 1 is the illustration for our example, where Fig. 1a is the original noisy signal, and Fig. 1b, c show its Laplacian low and high subbands. It can be seen that the slope area in the original signal becomes a flat area in the high band, and thus more noise reduction can be

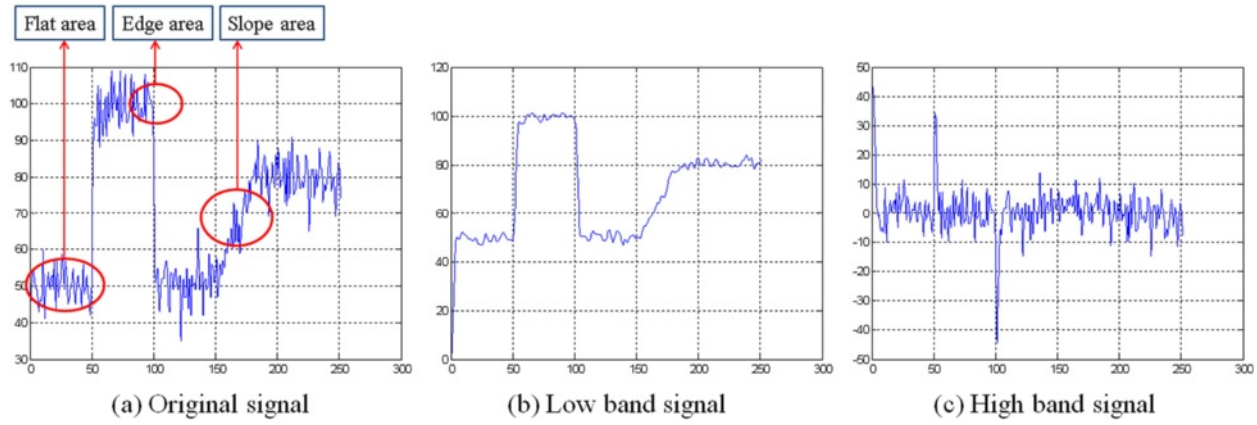

Fig. 1 An example of Laplacian subband decomposition for a 1-D signal. **a** Original signal. **b** Low band signal. **c** High band signal

gained here due to the widened area of photometric similarity. For these three signals, we apply the BLF with the same parameters, $\sigma_d^2 = 4$ and $\sigma_r^2 = 49$, and the denoising results of overall area and specific areas (flat, edge, and slope areas as in Fig. 1a) are summarized in Table 1. In the table, "original" column is the mean square error (MSE) of a noisy signal, "original BLF" represents the BLF result of this signal, and "subband BLF" means to apply the BLF to both of low and high band signals in Fig. 1b, c and then add them. Also, the "overall" means the MSE of the overall region of the signal in [0,255], and flat, edge and slope represent the areas as defined in Fig. 1a. At the first row, it can be seen that the BLF of the signal greatly reduces the noise variance (from 9.12 to 3.24) and the subband BLF reduces the variance further (to 2.11). When comparing the area-wise results, it can be observed that the ratio of the denoising gain is the most significant in the slope area, whereas the subband BLF has worse gain at the edge area. Hence, this supports our motivation that the subband BLF has the possibility of increasing the denoising performance at the slope areas and possibly slowly varying areas which become flat in the high band. From the toy example explained above, it can be seen that the main challenge in applying the BLF to the subbands is to find a method to mitigate the noise in the edge area and also to find appropriate filtering parameters.

Table 1 Mean squared errors of the BLF results for the original and subband signals

Area	Original	Original BLF	Subband BLF
Overall ([0,255])	9.12	3.24	2.11
Flat ([0,39])	10.8	4.58	2.86
Edge ([80,119])	6.11	1.55	2.64
Slope ([150,189])	8.53	3.45	1.67

3.2 Proposed subband BLF

For a given input image, we first perform subband decomposition as Eqs. (1), (2) to obtain the low band signal L_2 and high band L_1. For the low band image L_2, we apply the conventional BLF with $\sigma_d = 1.8$ and $\sigma_r = \sigma$ as suggested in [10], where σ is the noise variance. As stated above, we concentrate on the filtering scheme for the high band image L_1, especially at the edge area. The basic idea is to give larger weights to the pixels that have similar edge intensities as well as pixel intensities. Also, when it is highly probable that a pixel is on the edge, it needs to be less affected by the neighboring pixels. These ideas are encoded into a new guidance term in addition to Eq.(4) as

$$w(p,q) = \exp\left(-\frac{\|p-q\|^2}{2\sigma_d^2}\right)\exp\left(-\frac{\|I(p)-I(q)\|^2}{2\sigma_r^2}\right)\exp\left(-\frac{\|h(p)-h(q)\|^2}{2\sigma_h^2(p)}\right) \tag{5}$$

where $\sigma_h^2(p)$ is the pixel dependent bandwidth, and $h(p)$ is the intensity of the pixel p in the histogram-equalized image of L_1 which will be explained later in more detail. Comparing this kernel with that of the original BLF in Eq. (4), the third term is our proposal which adaptively controls the weights near the edge areas. The adaptive bandwidth for the BLF has already been considered in [9], where the σ_r is adjusted along with an offset parameter by the optimization method with some training images. Unlike this previous adaptive BLF, our method is quite a simple algorithm which adjusts σ_h in the new kernel depending on whether the pixel is on the edge or not.

In summary, our method employs a new guidance image $h(p)$ in the manner of joint bilateral filtering [3], for reducing the weights on the edge pixels and vice versa. For this, we let the bandwidth in the new term to be

pixel dependent, i.e., the pixel difference in the high band ($\|h(p) - h(q)\|$) is considered in weight control. Precisely, our method adjusts $\sigma_h(p)$ to $2\sqrt{2}\sigma$ or $4\sqrt{2}\sigma$ depending on the edge strength of given image.

For these edge-dependent modifications, we extract edge information from the BLF of low band image L_2, which is denoted as $\hat{L_2}$. For determining whether a pixel is an edge pixel or not, we apply the Laplacian of Gaussian filter and then thresholding. Specifically, we convolve $\hat{L_2}$ with the kernel defined as

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right)\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (6)$$

and then the output pixels larger than 75 % of the mean value are considered the edge pixels. This gives an edge map $E(p)$ which is 1 when the pixel p belongs to edge pixels, and 0 if not. For simplicity, the edge map is obtained from the approximate intensity component (image of $(R+G+B)/3$), and this edge map is applied to all of color components equally. With this edge map, the kernel bandwidth is determined as

$$\sigma_h(p) = \begin{cases} 2\sqrt{2}\sigma & \text{if } E(p) = 1 \\ 4\sqrt{2}\sigma & \text{otherwise.} \end{cases} \quad (7)$$

It can be seen that the kernel bandwidth is small when the pixel is on the edge, so that the neighboring pixels less contribute to the averaging and thus the edge intensities are less changed. Conversely, the pixels in the flat areas are more strongly filtered than the edge pixels. It is worth to mention that we use $\|h(p) - h(q)\|$ (histogram equalized intensities of L_1 into the range [0,255]) instead of $\|L_1(p) - L_1(q)\|$, because $L_1(p)$ can have negative value and its dynamic range is large. Denoting the output of proposed BLF of L_1 as $\hat{L}_1$, the final denoised image is obtained as $\hat{L}_1 + \hat{L}_2$.

Throughout the experiments, it is found that the low-band (L_2) filtering with a variety of parameter changes does not much affect the overall performance. Hence, we apply just the original BLF with $\sigma_r = \sigma$ for the low-band filtering, and we have focused on the kernel design for the high-frequency subband L_1. Also, when comparing the results between the adaptive bandwidth and non-adaptive bandwidth (when σ_h is fixed), the gain by the adaptive scheme is not significant (under 0.1dB PSNR gain) because the edge area is small compared to others.

The overall process is illustrated in Fig. 2, where $\lambda = 1$ corresponds to the proposed Laplacian subband filtering explained above, and $0 \le \lambda < 1$ gives the edge enhanced results to be explained in the next section.

3.3 Image enhancement with the Laplacian subband denoising

As stated in the introduction, edge enhancement can also be easily achieved by manipulating the denoised high band signal in the Laplacian subbands. A straightforward method would be to amplify the denoised high band and then add this signal to the denoised low band image, like the original unsharp mask method. However, since the edge components in the high band have been smoothed in the filtering process, the straightforward method might add the smeared edges. Hence, we try to restore the edge strength of the high band image as strong as the original one, and then add this restored edges. For this, we adopt the idea of gradient histogram preservation (GHP) in [12], which is to impose a constraint that the processed image has the same gradient histogram as the estimated original one. Specifically, for the noisy image model:

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \quad (8)$$

where $\mathbf{x}$ is the original image, $\mathbf{v}$ is the noise, and $\mathbf{y}$ is the observed noisy image, the processed image is constrained to have similar gradient histogram as $\mathbf{x}$. In [12], considering the histogram of gradients of $\mathbf{y}$ as the discretization of the pdf of gradient distribution of $\mathbf{y}$, the gradient histogram of the original image $\mathbf{x}$ is found by solving

$$\arg\min_{H_x}\{\|H_y - H_x \otimes H_v\|^2 + c \cdot R(H_x)\} \quad (9)$$

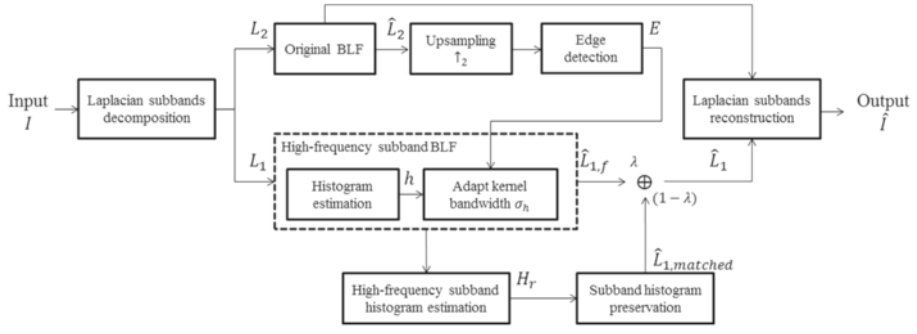

Fig. 2 Block diagram of proposed BLF ($\lambda = 1$) and edge enhancement method. ($0 \le \lambda < 1$)

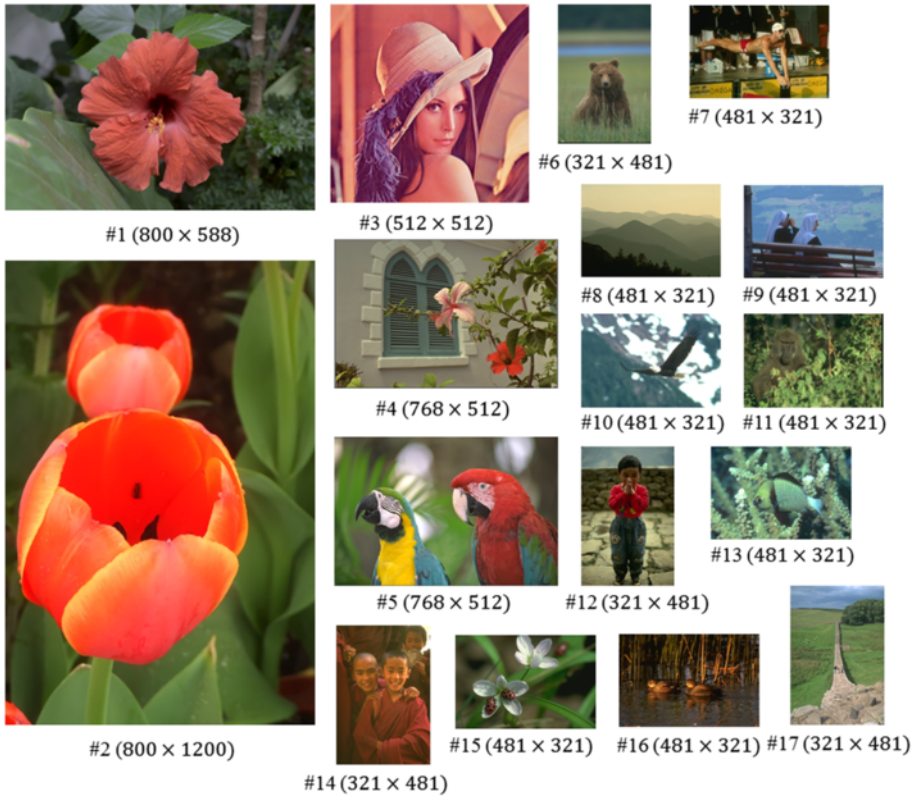

Fig. 3 Test images from Kodak dataset, BSDS 500[13], etc

where H_x, H_y, and H_v are the gradient histograms of $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{v}$ respectively, $\otimes$ is the convolution operator, and $c \cdot R(H_x)$ is a regularization term. For solving this problem, H_y is estimated from the observed data and H_v is modeled as a hyper-Laplacian distribution as [12]:

$$p_x = k \exp(-\kappa |x|^{\gamma}) \tag{10}$$

where k is normalization factor.

Note that the high band image L_1 in our subband BLF scheme is also a kind of gradient image, where the above GHP approach can be applied. Applying the Laplacian subband decomposition to Eq. (8), we have the high band relationship as

$$\mathcal{L}_1(\mathbf{y}) = \mathcal{L}_1(\mathbf{x}) + \mathcal{L}_1(\mathbf{v}) \tag{11}$$

where $\mathcal{L}_1(\cdot)$ is the operator that extracts high band signal of the input image, and hence $\mathcal{L}_1(\mathbf{y}) = L_1$ in our problem. Like the GHP approach, we wish to find the histogram of $\mathcal{L}_1(\mathbf{x})$ so that we match the histogram of $\hat{L_1}$ to this one. Denoting the histogram of $\mathcal{L}_1(\mathbf{x})$ as the "reference histogram" H_r, we obtain it in a similar manner as Eq. (9), except that the positive and negative coefficients are considered separately in order not to diminish the peaks of coefficients that appear around the edges. To be specific, we obtain H_r as

$$H_r = \arg\min_{H} \{\|H_{y,+} - H \otimes H_v\|^2 + \|H_{y,-} + H \otimes H_v\|^2 + c \cdot R(H_r)\}. \tag{12}$$

where $H_{y,+}$ is the histogram of positive values in $\hat{L_1}$, $H_{y,-}$ for the negative values, and H_v is the histogram of $\mathcal{L}_1(\mathbf{v})$ that is modeled as Eq. (10). The range of parameters for solving this problem is set the same as [12], i.e., $\kappa \in [0.001, 3]$ and $\gamma \in [0.02, 1.5]$. Then, the histogram of L_1 is matched to H_r, which is denoted as $\hat{L}_{1,matched}$ in Fig. 2 and the edge enhanced image is obtained as $\hat{L}_1 = \lambda \cdot \hat{L}_{1,f} + (1 - \lambda) \cdot \hat{L}_{1,matched}$.

Table 2 Averaged PSNRs for AWGN

σ	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
20	30.2919	30.8692	32.4754	32.4435	32.7027	34.6959
30	27.8154	28.3605	30.1490	30.2798	29.9845	32.2624
40	25.9968	26.2451	28.2862	28.5453	27.8337	30.1545
50	24.5750	24.3909	26.7048	27.0307	25.9953	28.6703
Time (s)	0.2788	10.2818	18.1894	0.8729	88.7329	5.7958

Fig. 4 Results of denoising the images corrupted by AWGN with $\sigma = 40$. **a** Original image. **b** Noisy image. **c** BLF, 26.9332 dB. **d** MBLF, 26.6292 dB. **e** BLS-GSM, 29.4673 dB. **f** NLM 29.2290 dB. **g** BM3D 30.2165 dB. **h** Proposed SBLF ($\lambda = 1$ in Fig. 2), 29.0490 dB. **i** Proposed SBLF with high band enhancement ($\lambda = 0.3$), 27.3006 dB

4 Experimental results

4.1 Experiments on pseudo white and Poisson noise

To evaluate the performance of proposed method, images in Fig. 3 are used. Each image is corrupted by the additive white Gaussian noise (AWGN) with variance σ or Poisson noise with parameter Q. We compare our subband BLF (SBLF) algorithm with the original BLF [2], multiresolution bilateral filter (MBLF) [10], BLS-GSM

Table 3 Averaged PSNRs for Poisson noise

Q	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
5	30.4115	30.4539	30.3704	31.8086	30.2686	33.5917
10	28.6911	28.5554	29.0285	30.0963	28.2948	31.7019
15	27.5984	27.3157	28.2461	28.9788	27.2667	30.4961
Time(s)	0.3078	10.8430	16.8132	0.9421	21.5664	6.9882

[11], NLM [4], and BM3D [5] with the authors' source codes. According to [10], we set $\sigma_d = 1.8$ and $\sigma_r = 2\sigma$. Also, 9×9 windows are used for the original BLF, MBLF, and the proposed method. The multiresolution BLF is implemented in MATLAB, and others are implemented in MATLAB and C/C++ through MATLAB MEX functions, and the codes are run on a PC with an Intel Core i5 CPU and 4 GB RAM.

Table 2 shows the averaged denoising results for the images in Fig. 3, each of which are corrupted by 100 different pseudo random sequences of Gaussian distribution with $\sigma = 20, 30, 40, 50$. PSNR for each of the images and other experimental results are available at http://ispl.snu.ac.kr/~idealgod/SBLF , where our source code and full-resolution images of all the figures in this paper are also available. As shown in Table 2, the proposed method

Table 4 PSNRs for mixed noise (20 % impulse noise)

Image	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
1	24.7155	22.3321	24.1887	25.0244	23.6376	24.9905
2	22.7271	22.3321	23.1560	23.5464	22.8560	24.9531
3	24.1931	23.5109	23.2620	25.1243	23.5753	23.2513
4	24.5633	23.5109	24.3265	25.6176	24.3059	24.8292
5	24.2419	23.0465	23.4489	24.9681	23.7099	25.9463
6	26.8504	22.8714	26.7785	27.3136	24.0875	24.9552
7	24.7198	24.5795	23.9426	25.3261	25.9881	27.2638
8	22.8777	24.5795	22.8013	24.2091	24.5127	27.1690
9	21.8124	23.2055	21.4086	22.6783	23.2277	25.4997
10	24.2438	23.2055	22.9440	24.7666	22.3793	23.8574
11	23.4927	22.1039	22.7081	24.1268	21.1000	22.6501
12	24.4074	22.1039	23.8970	25.3700	22.3230	24.5317
13	23.3632	21.2694	22.7738	23.9723	22.3291	24.0320
14	23.5229	22.8539	23.0868	24.0552	22.7604	25.3259
15	25.8834	22.8539	25.2027	26.6016	21.0988	25.2427
16	21.2509	21.9349	20.6449	21.9225	23.5069	23.5867
17	24.0441	24.5466	24.4997	24.7115	22.8812	23.9613
Avg.	23.9359	22.9906	23.4747	24.6667	23.1929	24.8262
Avg. time	0.2767	9.3545	16.8914	0.9955	38.2350	9.0230

Table 5 PSNRs for mixed noise (20 % impulse noise + Gaussian noise $\sigma = 10$)

Image	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
1	25.1272	22.1891	24.1444	24.8254	23.5478	24.9216
2	23.3915	21.0752	22.8482	23.2027	23.3204	22.9711
3	24.3678	23.4815	23.2735	24.9511	24.2775	24.7898
4	24.7087	23.0925	24.3643	25.6365	23.7272	25.9443
5	24.5624	22.8267	23.4414	24.9404	24.0864	24.9208
6	27.1773	24.5983	26.7542	27.3291	25.9194	27.2793
7	25.0360	22.3752	24.0196	25.3808	24.5317	25.5305
8	23.5980	23.2134	22.7675	24.0974	22.3253	23.8019
9	22.3192	19.7029	21.3331	22.6534	21.0883	22.6279
10	24.6008	22.1769	22.9668	24.8366	22.4003	24.5818
11	24.0335	21.3128	22.7450	24.1277	22.4418	24.0698
12	24.5797	22.9088	23.9241	25.4467	22.8593	25.3869
13	24.1103	22.0438	22.8545	24.1076	23.5837	23.7107
14	24.0712	21.5176	23.0955	24.1099	22.8613	24.0113
15	25.9574	24.5380	25.1631	26.5531	24.5009	26.3651
16	21.7938	19.3832	20.7904	21.9789	20.7998	21.8605
17	24.8058	22.4147	24.4745	24.7011	24.3419	24.3968
Avg.	24.3671	22.2853	23.4682	24.6399	23.3302	24.5394
Avg. time	0.2779	10.3789	16.7026	0.9277	32.0858	8.3840

Table 6 PSNRs for mixed noise (10 % impulse noise + Poisson noise $Q = 10$)

Image	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
Image	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
1	28.5342	25.4233	27.4656	28.3195	27.6470	29.1604
2	25.8080	23.6747	25.0903	25.3725	26.0011	25.4205
3	25.2271	25.3809	24.9301	26.0284	25.6082	26.7212
4	25.2890	24.7387	25.7451	26.8534	26.2451	28.4774
5	25.9956	24.7914	25.1994	26.5095	26.3872	27.7189
6	29.0505	26.8902	28.9912	29.4322	28.8034	30.5975
7	26.5713	24.6991	26.4335	27.3299	27.0361	28.9383
8	23.1844	22.8339	22.4087	23.4954	23.3715	23.5693
9	24.4265	22.6400	24.2504	25.6088	24.9965	26.2820
10	26.3400	24.4213	25.6495	27.1205	25.2780	27.6961
11	26.3146	24.2364	25.2243	26.8433	25.8733	27.6886
12	24.9814	23.8919	24.7310	26.1163	25.0271	27.1157
13	27.3758	24.9263	26.2754	27.4453	27.0570	27.8019
14	26.9052	24.4638	26.1080	27.2045	26.7907	28.0307
15	26.3950	25.4183	25.7355	27.0892	26.1018	27.8844
16	24.0633	22.1646	23.4939	25.2126	24.3040	25.7068
17	27.5090	26.3678	27.7605	27.4656	27.5485	27.8591
Avg.	26.1159	24.5272	25.6172	26.6733	26.1221	27.4511
Avg. time	0.2777	10.3959	16.2441	1.1010	33.1900	9.4430

Table 7 PSNRs for real noise

Image	Local self-similarity				Nonlocal self-similarity	
	BLF	MBLF	BLS-GSM	SBLF	NLM	BM3D
1	32.5736	31.6800	33.0025	33.5443	34.1830	34.4310
2	36.5380	35.5875	37.0479	37.2656	37.8776	38.4951
3	22.6169	22.7788	22.2161	22.8347	22.5071	22.6712
Avg.	30.5762	30.0154	30.7555	31.5482	31.5226	31.8658

yields better results than BLF and MBLF, and comparable results with the BLS-GSM. When compared with the nonlocal methods, the proposed method shows better PSNR than NLM, but lower than BM3D. However, as shown in the last row of Table 2, the proposed method needs much less computation time than the nonlocal methods, as well as other local methods except the original BLF. Figure 4 is a sample set of restored images, which shows that the

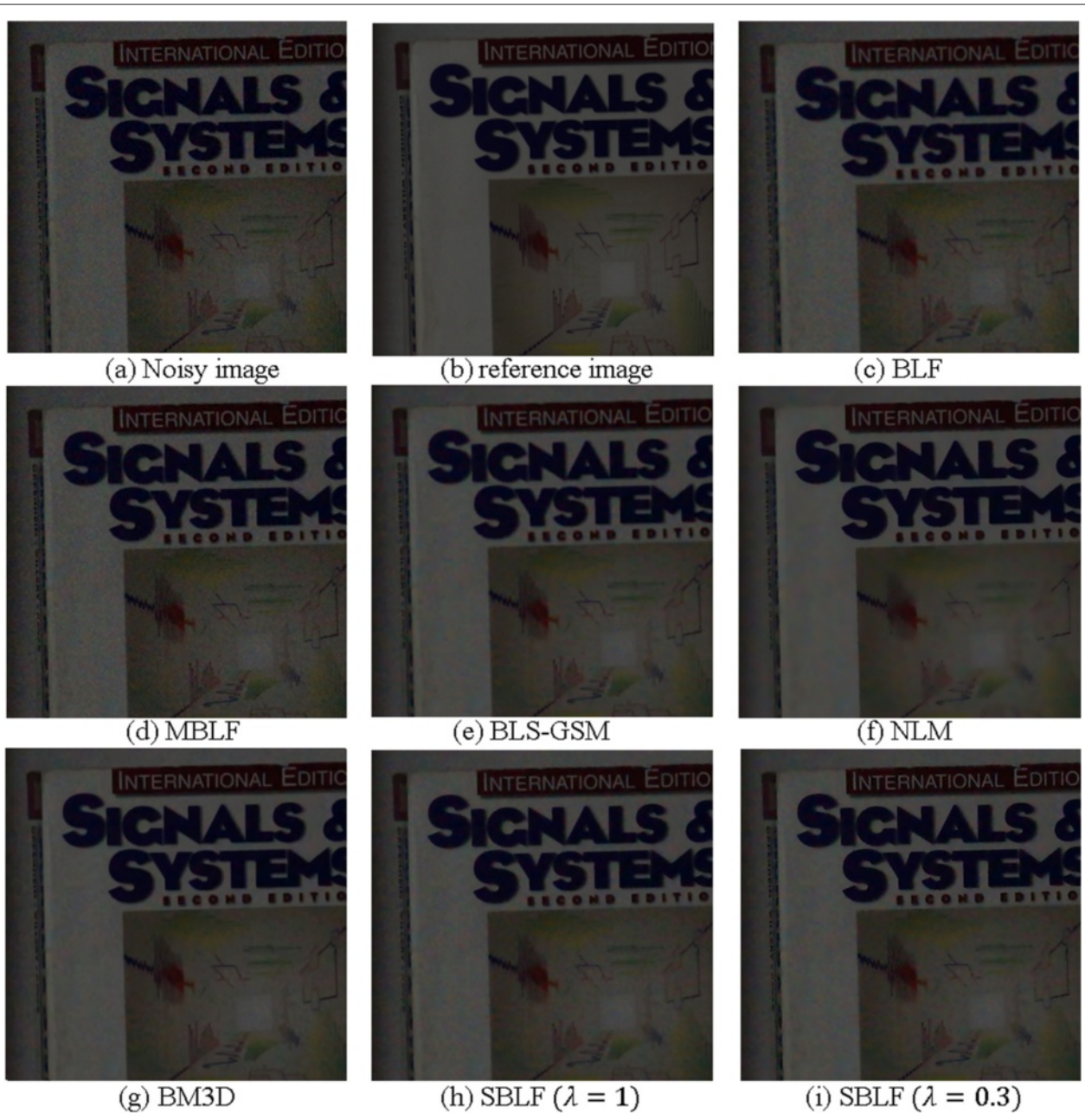

Fig. 5 Denoised results of real image 1. **a** Noisy image. **b** Reference image (average of 30 noisy images). **c** BLF, 36.6848 dB. **d** MBLF 35.5875 dB. **e** BLS-GSM 37.0479. **f** NLM, 37.8776 dB. **g** BM3D, 38.4951 dB. **h** Proposed SBLF ($\lambda = 1$) 37.3870 dB. **i** SBLF ($\lambda = 0.3$), 36.4595 dB

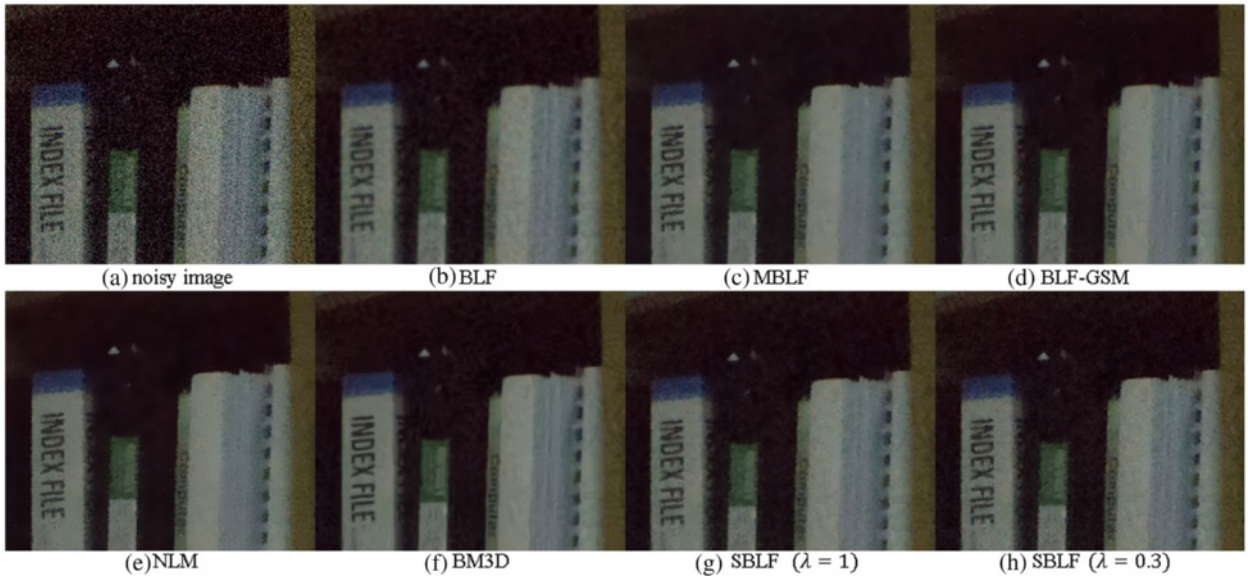

Fig. 6 Denoised results of real image 2. **a** Noisy image. **b** BLF. **c** MBLF. **d** BLS-GSM. **e** NLM. **f** BM3D. **g** SBLF ($\lambda = 1$). **h** SBLF ($\lambda = 0.7$)

proposed SBLF provides better visual quality than other local methods, and comparable quality with BM3D.

Table 3 shows denoising results when the images are corrupted by Poisson noise with $Q \in \{5, 10, 15\}$. It can be seen that the results show similar trends as the Gaussian noise case.

4.2 Experiments on mixed noise

Impulsive and mixed noises are also considered for evaluating the algorithm. In the experiments, images are corrupted by the mixture of impulsive noise with Poisson or Gaussian noise. Table 4 shows the PSNRs of denoised images which were corrupted by 20 % impulse noise. Table 5 shows the results for 20 % impulse noise + Gaussian noise with $\sigma = 10$, and Table 6 presents the results for 10 % impulse noise + Poisson noise with $Q = 10$. It can be seen that the proposed method shows comparable or sometimes better PSNR than the BM3D. The reason for these results seems that there are not much similar patches for the nonlocal methods when there are randomly distributed impulsive noises.

It is worth to note that all the algorithms need (estimated) noise variance as the input, for controlling the filter parameters. In the case of above simulated noises, we know the noise variance and use it for the kernel parameters. However, in the case of mixed noise and the real noise in the following subsection, we cannot know whether the estimate noise variance (by any of estimation methods) is accurate or not. Hence, we try many experiments with the input variance in the range of $[10, 70]$ and choose the best one for each of the algorithms.

4.3 Experiments with real noise

When an image is captured in an insufficient light condition, there appears noticeably strong noise. For the experiments on this kind of "real noise" (not the simulated noises as above), we test the algorithms on the images of indoor scenes. Since there is no ground truth image for the objective comparison in this case, we capture the scene 30 times with tripod and consider the average of these images as the reference image to compute PSNR. Table 7 shows PSNRs for several output images, and Figs. 5 and 6 show the images for subjective comparison. From the objective and subjective comparisons, it can be seen that the proposed method shows better results than the other local self-similarity methods, and comparable quality as the nonlocal similarity methods.

4.4 Noisy image enhancement

Finally, we present the results of the proposed image enhancement scheme, specifically the overall scheme of Fig. 2 with $\lambda = 0.3$. For enhancing the noisy images, a plausible method would be to denoise the image first and then apply the conventional image enhancement methods. Since the proposed method is based on the BLF, the comparison is performed with the schemes that apply BLF first and then enhance the image with [6] or [8]. Figures 7 and 8 show these comparisons, where (a) is the original image, (b) is the noisy one, (c) is the result of sequentially applying BLF denoising and high band amplification, (d) is the BLF followed by edge aware local Laplacian filtering [6], (e) is the result of sequentially applying BLF denoising and guided filtering [8], (f) is the

Fig. 7 Enhancement of noisy image ($\sigma = 20$). **a** Original image. **b** Noisy image. **c** Result of sequentially applying BLF and high band amplification. **d** Sequentially applying BLF and edge aware local Laplacian filtering [6]. **e** Sequentially applying BLF and guided filtering [8]. **f** Texture enhanced imaged denoising (TEID) [12]. **g** ABF [9]. **h** Proposed SBLF ($\lambda = 0.3$)

Fig. 8 Enhancement of noisy image ($\sigma = 20$). **a** Original image. **b** Noisy image. **c** Result of sequentially applying BLF and high band amplification. **d** Sequentially applying BLF and edge aware local Laplacian filtering [6]. **e** Sequentially applying BLF and guided filtering [8]. **f** Texture-enhanced imaged denoising (TEID) [12]. **g** ABF [9]. **h** Proposed SBLF ($\lambda = 0.3$)

result of denoising via TEID [12], (g) is the result of ABF [9], and (h) is the output of proposed algorithm. The figures show that the proposed method effectively suppresses the noise while enhancing the texture and edges. In the case of [12] (Fig. 7f and Fig. 8f), it can be seen that the noise is well removed while "preserving" the textures. On the other hand, the results in Fig. 7h and Fig. 8h show that the proposed method enhances the texture area (especially feather areas and patterns around the eyes), because the proposed scheme with $\lambda < 1$ adds the matched high frequency components to the denoised high band.

5 Conclusions

We have proposed an image denoising method based on the Laplacian subband decomposition and BLF. The input image is decomposed into two subbands by the Laplacian of Gaussian, and the BLF is applied to each of the subbands with appropriate filtering kernel and parameter. The experiments show that the proposed method increases PSNR compared to the original BLF and other multi-resolution filtering methods. For the real noisy images, the proposed method also yields comparable results to the non-local similarity methods such as BM3D and NLM, while requiring less computation time. Since the proposed method is based on the Laplacian decomposition, the edge enhancement can also be efficiently achieved along with the denoising.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work was supported in part by Samsung Electronics, and in part by the Ministry of Science, ICT and Future Planning, Korea, through the Information Technology Research Center support Program supervised by the National IT Industry Promotion Agency under Grant NIPA-2014-H0301- 14-1019.

Author details
[1]Department of Electrical and Computer Engineering, INMC, Seoul National University, 1 Gwanangno, Gwanak-gu, 151-742 Seoul, Korea. [2]Robot R&D Group, Korea Institute of Industrial Technology, 143 Hanggaulro, Sangrok-gu, 426-910 Ansan-si Gyeongni-do, Korea.

Received: 23 September 2014 Accepted: 23 July 2015
Published online: 06 August 2015

References

1. P Milanfar, A tour of modern image filtering: new insights and methods, both practical and theoretical. IEEE Signal Process. Mag. **30**(1), 106–128 (2013)
2. C Tomasi, R Manduchi, in *Sixth International Conference on Computer Vision*. Bilateral filtering for gray and color images, (Bombay,1998), pp. 839–846. Stoneham: Butterworth-Heinemann
3. S Paris, P Kornprobst, J Tumblin, F Durand, Bilateral filtering: Theory and applications. Foundations Trends® Comput. Graph. Vis. **4**(1), 1–73 (2008)
4. A Buades, B Coll, J-M Morel, in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference On*. A non-local algorithm for image denoising, vol. 2, (2005), pp. 60–652. doi:10.1109/CVPR.2005.38
5. K Dabov, A Foi, V Katkovnik, K Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering. Image Process. IEEE Trans. **16**(8), 2080–2095 (2007)
6. S Paris, SW Hasinoff, J Kautz, Local Laplacian filters: edge-aware image processing with a laplacian pyramid. ACM Trans. Graph. **30**(4), 68 (2011)
7. G Deng, A generalized unsharp masking algorithm. Image Process. IEEE Trans. **20**(5), 1249–1261 (2011)
8. K He, J Sun, X Tang, Guided image filtering. Pattern Anal. Mach. Intell. IEEE Trans. **35**(6), 1397–1409 (2013)
9. B Zhang, JP Allebach, Adaptive bilateral filter for sharpness enhancement and noise removal. Image Process. IEEE Trans. **17**(5), 664–678 (2008)
10. M Zhang, BK Gunturk, Multiresolution bilateral filtering for image denoising. Image Process. IEEE Trans. **17**(12), 2324–2333 (2008)
11. J Portilla, V Strela, MJ Wainwright, EP Simoncelli, Image denoising using scale mixtures of gaussians in the wavelet domain. Image Process. IEEE Trans. **12**(11), 1338–1351 (2003)
12. W Zuo, L Zhang, C Song, D Zhang, in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference On*. Texture enhanced image denoising via gradient histogram preservation, (Portland, OR, 2013), pp. 1203–1210
13. P Arbelaez, M Maire, C Fowlkes, J Malik, Contour detection and hierarchical image segmentation. Pattern Anal. Mach. Intell. IEEE Trans. **33**(5), 898–916 (2011)