

A Robust and Fast Non-Local Means Algorithm for Image Denoising

Yan-Li Liu^{1,2} (刘艳丽), Jin Wang¹ (王 进), Xi Chen¹ (陈 曦), Yan-Wen Guo³ (郭延文)
and Qun-Sheng Peng^{1,2} (彭群生)

¹State Key Lab of CAD& CG, Zhejiang University, Hangzhou 310058, China

²Department of Mathematics, Zhejiang University, Hangzhou 310058, China

³State Key Lab of Novel Software Technology, Nanjing University, Nanjing 210000, China

E-mail: liuyanli@cad.zju.edu.cn

Revised January 7, 2008.

Abstract In the paper, we propose a robust and fast image denoising method. The approach integrates both Non-Local means algorithm and Laplacian Pyramid. Given an image to be denoised, we first decompose it into Laplacian pyramid. Exploiting the redundancy property of Laplacian pyramid, we then perform non-local means on every level image of Laplacian pyramid. Essentially, we use the similarity of image features in Laplacian pyramid to act as weight to denoise image. Since the features extracted in Laplacian pyramid are localized in spatial position and scale, they are much more able to describe image, and computing the similarity between them is more reasonable and more robust. Also, based on the efficient Summed Square Image (SSI) scheme and Fast Fourier Transform (FFT), we present an accelerating algorithm to break the bottleneck of non-local means algorithm — similarity computation of compare windows. After speedup, our algorithm is fifty times faster than original non-local means algorithm. Experiments demonstrated the effectiveness of our algorithm.

Keywords image denoising, non-local means, Laplacian pyramid, summed square image, FFT

1 Introduction

With the prevalence of digital cameras and scanners, digital images can be easily acquired nowadays. Unfortunately, digital images are often degraded by noise during acquisition or transmission process. Various image-related applications, such as medical image analysis, image segmentation, and object detection, etc., generally require effective noise suppression method to further produce reliable results. Therefore, image denoising has been one of the most important and widely studied problems in image processing and computer vision. So far, image denoising methods can be basically divided into two categories: spatial filtering methods^[1–4] and transform domain filtering methods^[5–9].

In spatial domain, for every pixel of noisy image, many existing noise reduction methods employ the information of its nearby local region to estimate its denoised version. Examples include Gaussian filter^[1], median filter^[3], bilateral filter^[4] and so on. We call this kind of denoising algorithms local-based spatial meth-

ods. Whereas, generally speaking, the information encoded in a natural image is redundant to some extent, that is to say, there may exist some repeat patterns in natural images, particularly in textured or periodic case. Based on this observation, Buades^[10] developed a non-local image denoising algorithm which takes full advantage of image redundancy. For the sake of simplicity, we name it the spatial nl-means in the paper. Like many noise reduction algorithm, the method is also based on weighted average. The essence of the method is: to estimate a certain pixel, the method uses the similarities between it and all the other pixels in image to act as weight, and the similarities are not computed from pixels themselves but from their neighboring window (compare window). The algorithm has demonstrated strong superiority over local-based spatial methods such as Gaussian filter, bilateral filter in terms of both PSNR and visual quality. However, certain critical issues need further investigation. Firstly, the computational cost is high for its pixel-by-pixel window matching, and it limits the method to be widely used in application. Secondly, the quality and compu-

Regular Paper

This work is supported by the National Grand Fundamental Research 973 Program of China (Grant No. 2002CB312101), the National Natural Science Foundation of China (Grant Nos. 60403038 and 60703084) and the Natural Science Foundation of Jiangsu Province (Grant No. BK2007571).

tation cost of the algorithm is closely related to the size of compare window which represents geometrical configuration of pixel's neighboring region. For small compare window, the algorithm is restricted to suppressing the high-frequency noise and cannot remove the low-frequency (large-scale) noise. For large compare window, the algorithm removes the low-frequency noise effectively, but becomes less sensitive to small details and the additional burden on the computational resources becomes unacceptable.

Since the essential goal of denoising is to preserve the image features while reducing noise effectively, a logical extension of spatial denoising is to transform images into a representation that distinctive features such as edges can be extracted from image and perform noise reduction algorithms in this domain. As for denoising in frequency domain, we are more familiar with wavelets^[5-9] which fall into two classes: orthogonal wavelet and non-orthogonal wavelet. Orthogonal wavelet compositions are critically sampled (i.e., non-redundant) image descriptions. It has been successfully used in image compression^[11,12] and threshold-based image denoising^[5,13]. However, the non-redundant property of orthogonal wavelet means they are not suitable for non-local means denoising which builds upon image redundancy.

In this paper, we emphasis on overcomplete non-orthogonal representations of image. Some non-orthogonal wavelet representations are redundant to some extent. It should be noted that wavelet decomposes image into three sub-bands for every level, implying that connected spatial structures analyzed at increasing resolutions are split into separate sub-bands, thereby losing their spatial connectivity^[14]. Another drawback of wavelet representation is that comparing the similarity respectively in three sub-bands will induce heavy computational cost. Based on these considerations, Laplacian pyramid^[15], being a redundant image representation, is selected in the paper. Due to the overcompletion property of Laplacian image, every frequency component image is redundant, thus we can compute the non-local similarity in frequency component. In the paper, we first exploit Laplacian pyramid to decompose the given noisy image and generate a series of manipulatable components, which allows us to develop different strategies on different components. Then we perform non-local means on every level of Laplacian pyramid with different sizes of compare windows.

Also, an acceleration method to our algorithm is proposed in the paper. Exploiting the Summed Squares

Image (SSI) scheme and Fast Fourier Transform (FFT), the per-pixel neighborhood matching in Laplacian image is converted into the SSI pre-computing and efficient FFT. Computational complexity analysis and experiments indicate that our accelerated algorithm is more than 50 times faster than the original non-local algorithm. Using our algorithm, it normally takes less than 1 second to denoise a 640×480 image, and less than 11 seconds to denoise a 500 Megabyte photograph, which enables the algorithm applicable to practical situations.

The main contributions of our paper are as follows.

1) We propose a robust non-local denoising method based on Laplacian pyramid. Combining Laplacian pyramid and non-local means for image denoising has the following advantages. (i) Laplacian pyramid decompose noisy image into easily manipulatable image components. By performing non-local means (nl-means) on different levels of Laplacian pyramid with different sizes of compare windows, we not only reduce high-frequency noise but also reduce low-frequency noise while preserving the image details (edges, textures) well. (ii) Since the features extracted in Laplacian pyramid are localized in spatial position and scale, they are much more able to describe image, and computing the similarity between them is more reasonable and more robust. (iii) As the level of Laplacian pyramid increases, the amount of noise in Laplacian image decreases rapidly, therefore, denoising in Laplacian image is less noise-sensitive than in spatial domain.

2) We also present an accelerating algorithm to speed up similarity computation of compare windows. Based on efficient SSI scheme and FFT, the accelerated algorithm is fifty times faster than the spatial nl-means.

The result of spatial nl-means leaves a trail of smear. When coming across heavy noise, the spatial nl-means introduces artifacts such as canvas effect which is pronounced in flat regions and soft edges of denoised image. In contrast, our method reduces noise effectively and preserves the image details and weak edges well. As noise level increases, the difference of results between our method and the spatial nl-means is even more striking.

The rest of this paper is organized as follows. Section 2 gives background and related work about nl-means algorithm. Laplacian pyramid and how our algorithm integrates both nl-means algorithm and Laplacian pyramid as well as our acceleration scheme are detailed in Section 3. Experimental results are demonstrated in Section 4. Section 5 gives the conclusion of the whole paper.

2 Background and Related Work

2.1 Background

In spatial nl-means, when modifying a pixel, the algorithm first computes the similarity between the window centered around it and the windows centered around the other pixels in the whole image, then it takes the similarities as weight to adjust this pixel. It also can be interpreted that the method defines as “neighborhood of a pixel i ” any set of pixels j in the image such that a window around j looks like a window around i , and all pixels in that neighborhood are used for predicting the value at i . In essence, the method uses two windows’ similarity other than two pixels’ to estimate pixels and these windows slide on the whole image.

If N^2 is the number of pixels in the image, M^2 is the size of compare window, the complexity of this algorithm is $M^2 \times N^4$. For computational purposes, the simplified algorithm restricts the search of similar windows in a “search window” of size $S \times S$ pixels. By fixing the search window at the size of 21×21 pixels and the neighborhood size 7×7 , the final complexity of the algorithm is $49 \times 441 \times N^2$. However, even the simplified algorithm still takes about 1 minute to denoise a 640×480 image on a common PC. It can be seen that the high computational complexity makes it unfeasible to tackle with practical issues.

2.2 Related Work

To our knowledge, up to now, there is not much work about nl-means denoising. Lately, Lukin proposed a multiresolution-based approach to improve spatial nl-means algorithm^[16]. The algorithm first performs nl-means on noisy image in spatial domain with different block sizes and search ranges. Afterward, it transforms the resulting images (generally two images) into frequency domain, and then combines the coefficients by another filter bank, finally inverses the filter bank to get the denoising result. It can be seen that this method only considers getting rid of different frequency noise, but it does not consider the redundancy of image features itself and comparing similarity in transform domain. Conversely, we compute similarity between image features at different scales, taking advantage of redundancy of different spatial frequency components of image. The methods can not only reduce different frequency noise efficiently but also produce robust results. That is where our approach diverges from Lukin’s algorithm.

An algorithm for accelerating spatial nl-means denoising was proposed by Mahmoudi and Sapiro^[17].

In order to accelerate the algorithm, they introduced filters that eliminate unrelated neighborhoods from weighted average. These filters are based on local average gray values and gradients, preclassifying neighborhoods and reducing the influence of less-related areas when denoising a given pixel. Note that the times of acceleration of Mahmoudi’s algorithm is related to image content, for example, it is 7 times faster than original spatial nl-means for some images, whereas, for some other images it is more than 20 times faster than the original nl-means. In our method, after speedup, the method is fifty times faster than the original nl-means for all the images.

3 Our Robust and Fast Laplacian Pyramid Based nl-Means

3.1 Laplacian Pyramid

Gaussian-Laplacian pyramid has been widely used in texture classification^[18,19], feature detection^[20] and so on. Given an image $I(x, y)$, the Gaussian pyramid of I is computed as follows:

$$\begin{cases} G_0(x, y) = I, \\ G_{k+1}(x, y) = \text{REDUCE}(G_k(x, y)), \quad k = 0, 1, \dots, N. \end{cases} \quad (1)$$

The REDUCE operation is carried out by convolving the image with a Gaussian low pass filter and then sub-sampling the filtered version at a factor of two. Repeating the REDUCE operation generates Gaussian pyramid. The filter mask is designed such that the center pixel gets more weight than the neighboring ones and the remaining terms are chosen so that their sum is 1. The separable Gaussian kernel is given by:

$$w(r, c) = w(r)w(c). \quad (2)$$

As in the general case, in the paper, the kernel we used is of size 5×5 , and $w(r) = (1/16, 1/4, 3/8, 1/4, 1/16)$. Thus, the variance of the Gaussian kernel approximates 1.0.

In order to generate Laplacian pyramid, the second level of Gaussian image G_1 is up-sampled by inserting zeros after each pixel, smoothed with the small kernel as defined in Gaussian pyramid. These interpolation and filtering process is referred to as EXPAND operation. The difference between the previous level of Gaussian image G_0 and expanded image \widehat{G}_1 is named as Laplacian image. This process is continued to obtain a set of band-pass filtered images. Expressed by formulation,

$$\begin{aligned} L_k(x, y) &= G_k(x, y) - \text{EXPAND}(G_{k+1}(x, y)), \\ k &= 0, 1, \dots, K, \end{aligned} \quad (3)$$

where N is the decomposition level and the EXPAND operation is defined as follows.

$$G_{k+1}(x, y) = 4 \sum_{m=2}^2 \sum_{n=2}^2 w(m, n) G_k\left(\frac{x-m}{2}, \frac{y-n}{2}\right),$$

$$k = 0, 1, \dots, K. \quad (4)$$

Only terms for which $(x-m)/2$ and $(y-n)/2$ are integers can be included in the sum.

The full reconstruction process is implemented by first expanding the final level of Gaussian pyramid G_{K+1} , then adding to L_k , $k = K, K-1, \dots, 0$, thus reconstructing the Gaussian pyramid, level by level, up to the original input image, G_0 . This is a recursive process, as shown in (5):

$$G_k = L_k + \text{EXPAND}(G_{k+1}), \quad k = K, K-1, \dots, 0. \quad (5)$$

From the above formulation, it can be seen that Laplacian pyramid is a set of bandpass images. It contains most of the image's important textural features, at different scales. The top level of the pyramid contains just the highest spatial frequency components such as the sharp edges, textures, high-frequency noise etc. The bottom level contains the lowest spatial frequency components. The intermediate levels contain features gradually decreasing in spatial frequency from high to low.

It can be seen that the Laplacian pyramid offers an overcomplete representation of the image. Hence, different spatial frequency component, e.g., every level of Laplacian pyramid is redundant, which makes performing nl-means on Laplacian image possible. Since the features extracted are localized in spatial position and scale, they are much more able to express the image, and computing the similarity between them is more reasonable and more robust. Meanwhile, Laplacian pyramid is translation-invariant, meaning that there is no aliasing effect such as pseudo-Gibbs artifacts in reconstruction.

3.2 Our Robust nl-Means

Given a noisy image, we first decompose it into a K -level Laplacian pyramid. For convenience, we denote as L_k the level k image of Laplacian pyramid. For a coefficient $L_k(x_i, y_i)$ of L_k ($k = 0, 1, \dots, K$), when estimating its denoised version $L'_k(x_i, y_i)$, we not only consider the similarities between $L_k(x_i, y_i)$ and its nearby coefficients, but also take the similarities between it and all the coefficients in L_k into account. That is to say, we exploit the similarity between $L_k(x_i, y_i)$ and all the

coefficients in the L_k as a weight to estimate $L'_k(x_i, y_i)$. To express as a formula, $L'_k(x_i, y_i)$ is:

$$L'_k(x_i, y_i) = \sum_{(x_j, y_j) \in L_k} w_k(i, j) L_k(x_j, y_j) \quad (6)$$

where the weight $w_k(i, j)$ of two coefficients $L_k(x_i, y_i)$ and $L_k(x_j, y_j)$ depending on their similarity is defined as,

$$w_k(i, j) = \frac{1}{z(i)} e^{-\frac{s(i, j)}{h_k^2}}. \quad (7)$$

Here, $z(i)$ is the normalized constant,

$$z(i) = \sum_{(x_j, y_j) \in L_k} e^{-\frac{s(i, j)}{h_k^2}}. \quad (8)$$

Here, h_k is a constant proportional to σ_L^k denoting the noise standard deviation of level k image of Laplacian pyramid,

$$h_k = c_k \sigma_L^k, \quad (9)$$

$s(i, j)$ is calculated by the Euclidean distance of the two coefficients' neighboring region N_i and N_j with equal size (M, M) as:

$$s(i, j) = \|N_i - N_j\|^2 \quad (10)$$

where N_i is a vector formed by concatenating all the coefficients in the neighborhood of (x_i, y_i) .

Since we relate the filter parameters with noise standard deviation of every level of Laplacian pyramid, we need to investigate the noise characteristics of Gaussian-Laplacian image pyramid. If the standard deviation of the original noisy image is σ_0^2 , the noise variance σ_s^2 of the smoothed image is given by^[21]:

$$\sigma_s^2 = \sigma_0^2 \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} G_\sigma^2(r, c) dr dc = \frac{1}{4\pi\sigma^2} \sigma_0^2. \quad (11)$$

Here, σ is the standard deviation of Gaussian kernel. In our paper, $\sigma=1.0$.

Thus, if σ_0^2 denotes the noise variance of the level 0 image of an Gaussian pyramid, the noise variance of the level k image in Gaussian pyramid is given by

$$(\sigma_G^k)^2 = \frac{1}{(4\pi\sigma^2)^k} \sigma_0^2. \quad (12)$$

The noise variance of every L_k can be computed as follows.

$$(\sigma_L^k)^2 = \frac{1}{(4\pi\sigma^2)^k} \sigma_0^2 - \frac{1}{(4\pi\sigma^2)^{k+1}} \sigma_0^2,$$

$$k = 1, 2, \dots, N-1. \quad (13)$$

Obviously, as the level of Laplacian pyramid increases, the amount of noise of Laplacian image decreases rapidly.

3.3 Acceleration of Our *nl*-Means Algorithm

The most time-consuming part of the *nl*-means algorithm is the calculation of the Euclidian distance between compare windows in image. Our method still need to perform similarity computation for pixels of every level L_k ($k = 0, 1, \dots, K$). In order to speedup the algorithm, we exploit SSI and FFT to transform this process into convolution and summation.

3.3.1 Re-Represent the Similarity of Compare Window

For simplicity, we denote L as $L_k, k \in \{0, 1, \dots, N\}$. As (10) indicates, the similarity of the pixel $L(x_i, y_i)$ and $L(x_j, y_j)$ is computed as:

$$\begin{aligned} S(i, j) &= \|N_i - N_j\|^2 \\ &= \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} [L_i(l, m) - L_j(l, m)]^2. \end{aligned} \quad (14)$$

where $L_i(l, m)$ and $L_j(l, m)$ represent the corresponding pixels in N_i and N_j respectively.

In fact, $L_j(l, m)$ in (14) can be represented in the global coordinates on the mirrored image as: $L_j(l - x'_j, m - y'_j)$, with $x'_j = 3M/2 + x_j$, $y'_j = 3M/2 + y_j$ (see Fig.1). So (10) is transformed into:

$$\begin{aligned} S(i, j) &= \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} [L_i(l, m) - L_j(l - x'_j, m - y'_j)]^2 \\ &= N_i^2 + N_j^2 - N_i \times N_j \end{aligned} \quad (15)$$

where

$$\begin{aligned} N_i^2 &= \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} (L_i(l, m))^2, \\ N_j^2 &= \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} (L_j(l - x'_j, m - y'_j))^2, \end{aligned}$$

and

$$N_i \times N_j = 2 \sum_{l=0}^{M-1} \sum_{m=0}^{M-1} (L_i(l, m) \cdot L_j(l - x'_j, m - y'_j))$$

denotes the convolution between N_i and N_j .

In above formula, $N_i \times N_j$ can be figured out quickly with multiplications under the fast Fourier transform,

while N_i^2 and N_j^2 can be fast calculated as well using the *SSI* scheme proposed in Subsection 3.3.2. Note that if the compare window size is $M \times M$, computing the similarity of the two compare window requires M^2 pixel operations, while, in our algorithm, it is figured out once which is achieved by means of FFT.

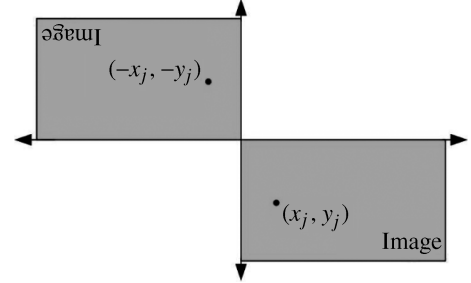


Fig.1. Mirrored image.

3.3.2 SSI

The principle of *SSI* resembles *Integral image* which has been used in face detection^[22]. For each pixel in the image, integral image maintains the summed value of all the pixels in the upper left part of the original image. Here we extend it to our *SSI*. Similar to the definition of integral image, for each pixel (x_0, y_0) , *SSI* stores its sum for the squared values of the upper left pixels,

$$SSI(x_0, y_0) = \sum_{x \leq x_0, y \leq y_0} L^2(x, y). \quad (16)$$

SSI can be obtained in linear time proportional to the image size, we take the following algorithm to calculate it efficiently. For $x_0 = 0, y_0 = 0$,

$$SSI(0, 0) = L^2(0, 0); \quad (17)$$

for $x_0 > 0, y_0 = 0$,

$$SSI(x_0, 0) = SSI(x_0 - 1, 0) + L^2(x_0, 0); \quad (18)$$

for $x_0 = 0, y_0 > 0$,

$$SSI(0, y_0) = SSI(0, y_0 - 1) + L^2(0, y_0); \quad (19)$$

for $x_0 > 0, y_0 > 0$,

$$\begin{aligned} SSI(x_0, y_0) &= SSI(x_0 - 1, y_0) + SSI(x_0, y_0 - 1) \\ &\quad - SSI(x_0 - 1, y_0 - 1) + L^2(x_0, y_0). \end{aligned} \quad (20)$$

Obviously, with above algorithm, each pixel in the original image is processed only once, so the computational complexity for computing *SSI* is $O(N^2)$, in which N^2 is the size of the image.

By means of *SSI*, we can easily get the sum of squares for each pixel in any rectangles of the image within constant time. For example in Fig.2, to calculate the sum of squares in rectangle *D*, only 3 addition operations are required,

$$\begin{aligned} S_D &= S_{A \cup B \cup C \cup D} + S_A - S_{A \cup C} - S_{A \cup B} \\ &= SSI(x_1, y_1) + SSI(x_0, y_0) - SSI(x_0, y_1) \\ &\quad - SSI(x_1, y_0). \end{aligned} \quad (21)$$

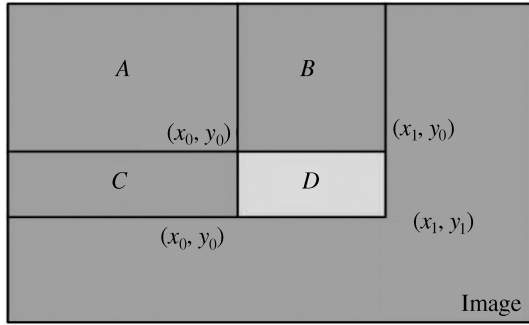


Fig.2. Using *SSI* to compute the summed squared pixels in the rectangle *D*.

Therefore, N_i^2 and N_j^2 can be computed quickly with (21) once we get *SSI* of the noised image. Accurate analysis in following subsection will show that our accelerated algorithm is much faster than the original one.

3.4 Time Complexity Analysis

The bottleneck of nl-means is the calculation of the Euclidian distance between compare windows in the image. In simplified spatial nl-means in which an $S \times S$ search window is used instead of the whole image, it takes $M^2 \times S^2$ square calculations for each pixel in the image, where M^2 is the size of compare window.

In our algorithm, this process has been transformed to computing convolution and summation of squares. It is well known that convolution becomes multiplication under Fourier transform, thus only S^2 multiplications need to be taken for each pixel in an image in our algorithm. Furthermore, Fourier transform can be performed quickly by modern hardware FFT accelerator within a neglectable time, convolution can therefore be fast carried out. As for the summations of squares, only addition operations are needed in this process which can also be neglected compared to the multiplications. Thus the computational complexity is $S \times S$, which is M^2 times faster than the original algorithm for pixel of image. In addition, it can be seen that the change of M has no impact on the time complexity of our method.

In theory, the neighborhood size M should be congruent to the size of the repeated patterns in the image. In order to achieve convincing results, M should be set larger in higher resolution images, which leads to a significant slowdown in the original algorithm but no performance changing in ours.

In the paper, we set the decomposition level $K = 2$, namely, we decompose every noisy test image into high, middle and low frequency components. Since there are three images to be processed in our algorithm, at first glance, it seems that our algorithms will be slow. Seeing that the resolution of higher level of pyramid is a quarter of that of the previous level, the window of the same size in higher level Laplacian pyramid represents larger region of image contents than that of original image. Hence, we can restrict the search window in high level of pyramid to a relatively small size. In the paper, we restrict search window in three levels to 21×21 , 11×11 and 3×3 , respectively, from high resolution level to low resolution level. Further, as discussed in Section 1, small compare window is sensitive to image details and effective to suppress high frequency noise. Conversely, large compare window is favorable to remove low frequency noise. Thus we should use small size compare window on the low level of pyramid and large size window on the high level of pyramid. In the paper, the compare windows we used are of size 7×7 , 5×5 , 3×3 respectively, from high resolution level to low resolution level. Considering resolution factor, compare widows of size 5×5 in middle and high level occupy much larger region of original image than that of size 7×7 in low level of Laplacian pyramid. Since the size of compare window has no impact on time complexity, except the decomposition and synthesis of the Laplacian pyramid, the total complexity of our algorithm is $21 \times 21 \times N^2 + 11 \times 11 \times N^2/4 + 3 \times 3 \times N^2/16$, approximately $472 \times N^2$. Hence, if we do not count in the time of implementing decomposition and synthesis of Laplacian pyramid, which is a very fast process due to its low complexity and simple parallel implementation, our algorithm is about fifty times faster than simplified spatial nl-means.

4 Experimental Results and Discussion

We have tested our method on a set of 8-bit grayscale test images. In order to ensure a fair comparison, the test images we used are the same ones used in the denoising experiments reported in [8]. We add different standard deviation of additive Gaussian white noise to the test images, and then make comparisons, in terms of both visual quality and PSNR, between the resulting

images of our algorithm and nl-means algorithm.

In the paper, as mentioned in Subsection 3.4, we make a 3-level Laplacian pyramid for all the images. Theoretically, we should set different c_k for different L_k to achieve perfect results. In pursuit of simplicity, we set the same h ($h = 10$) for all the L_k , and found that there was no obvious decrease in visual quality between two embodiments.

Table 1 summarizes a set of denoising results. As can be seen from it, when noise level is not high, our method is slightly better than spatial nl-means in terms of peak signal to noise ratio (PSNR). As the noise standard deviation increases, the PSNR of our algorithm notably surpasses that of spatial nl-means. For example, when the noise standard deviation σ_0 is 30, the PSNR of our method outstrips that of the spatial nl-means with an excess of 1.0 dB, which can be regarded as a considerable improvement. In order to better visualize the comparison, Fig.3 presents the denoising results as curves, for the image "Lena".

Table 1. Denoising Results in PSNR (in [dB])

σ_0	Lena		Peppers	
	NL	Our Method	NL	Our Method
10	34.17	34.96	33.32	34.38
20	31.45	31.95	31.33	31.87
30	28.68	30.08	28.67	29.88
50	25.41	27.27	26.09	27.36

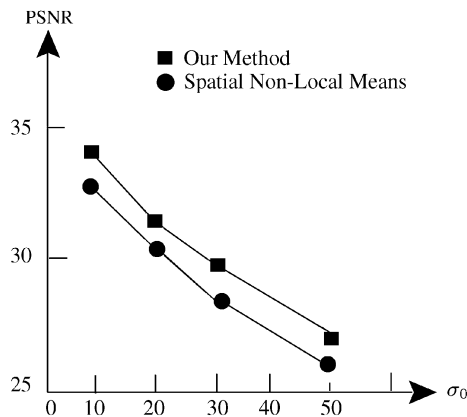


Fig.3. Comparison of PSNR between our method and spatial nl-means.

Besides PSNR, we believe that visual quality is also an important criterion in judging various noise reduction methods. Figs. 4~7 provide some visual comparisons of results denoised with these two algorithms. Fig.4(a) is the cropped part of "Lena" and Fig.4(b) is its noisy version with $\sigma_0 = 20$. We see in Fig.4(c) that the resulting image of spatial nl-means is smeary

and the denoised pillar are ragged. In our denoised result, as shown in Fig.4(d), the shape of pillar is well preserved. Fig.5 further shows the denoising results with two methods for a natural image. Fig.6 shows the first level and second level of Laplacian images before and after performing our algorithm for Fig.5(b). Evidently, the denoised Laplacian images, see Fig.5(b) and Fig.6(d) perfectly preserve the shape of pillars. Fig.7 is another comparison of two methods. In the denoising result of spatial nl-means for "Mandrill" with filtering parameters $h = 2.0$, as shown in Fig.7(c), the eyes of mandrill are pale, and on the lower part of image, canvas effect is pronounced. Fig.7(d) is another result of spatial nl-means with large filtering parameter $h = 3.5$ which is used to remove more noise, it can be seen that the resulting image looks more grey than the original image. In contrast, as shown in Fig.7(e), the eyes of mandrill are quite bright and the image is clean with the details well preserved. This is because when noise level is very high, in spatial domain, what a compare window represents is not the true content of image exactly, thus using the similarity of compare window to estimate pixels would result in large error. In our paper, we extract different scale features of image and operate on these features. The method has two advantages: first, it filters out some noise, thus it is noise less-sensitive. Second, image features (lines, edges, details) are more able to represent the image, using it to compute the similarity makes sense and more robust.

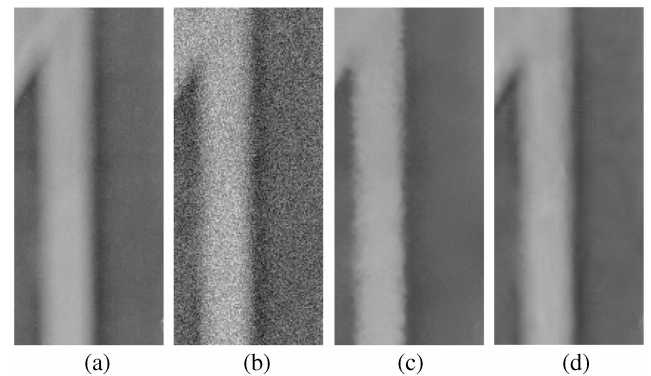


Fig.4. Comparison between the cropped result obtained by spatial nl-means (c) and our method (d) for noisy Lena with $\sigma_0 = 20$ (b). (a) is the cropped original Lena. (Note that the results of spatial nl-means (c) is smeared out and the denoised polar is ragged. In the result of our method (d), the image is clean and the polar is the same as in original image (a).)

As discussed in Subsection 3.4, our algorithm is about fifty times faster than simplified spatial nl-means. Experiment on a PC with a Pentium IV 2.4GHz CPU

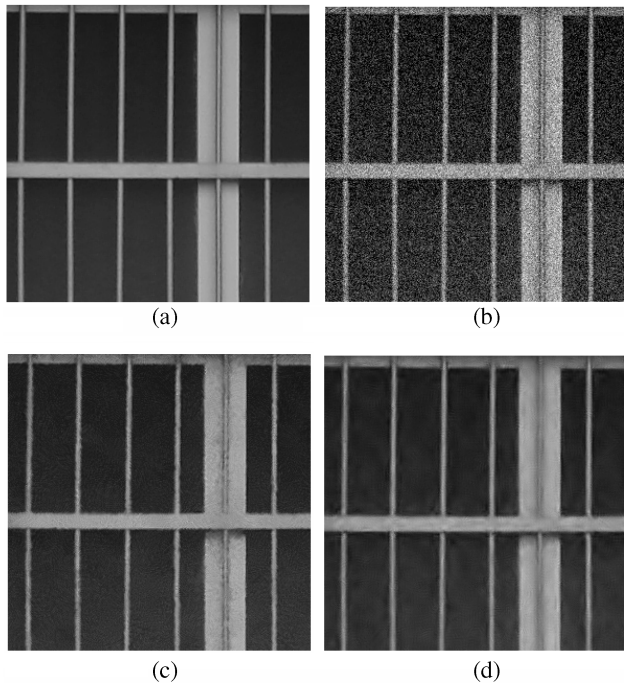


Fig.5. Comparison between the results obtained by spatial nl-means and our method for noisy window with $\sigma_0 = 30$ (b). (a) Original image. (b) Noisy window with $\sigma_0 = 30$. (c) Result by spatial nl-means. (d) Result by our method. It is clearly that our method preserves the shape of pillars well.

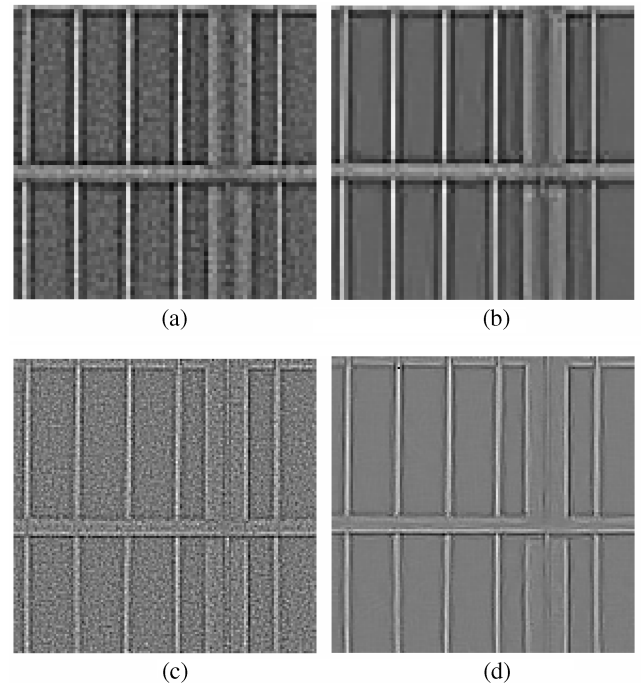


Fig.6. The second and first level of Laplacian pyramid of Fig.5(b) before and after denoising with our method. (a), (c) The second and first level respectively before denoising. (b), (d) Denoised second and first level respectively. It can be seen that the noise has been removed effectively and the shape of pillars are preserved in (b) and (d).

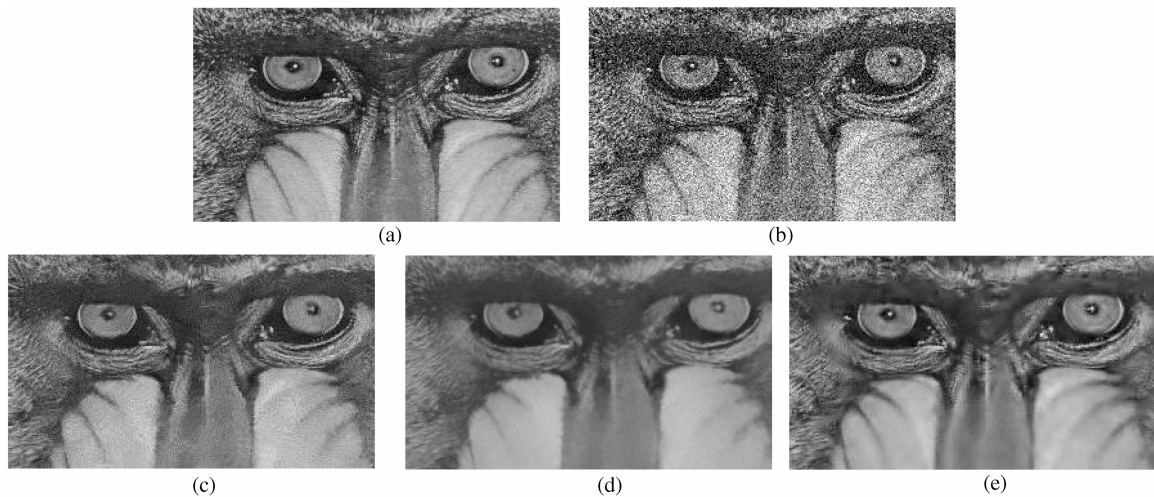


Fig.7. Comparison between the results obtained by spatial nl-means and that of our method for noisy Mandrill with $\sigma_0 = 30$. (a) Original Mandrill image. (b) Noisy Mandrill with $\sigma_0 = 30$. (c) Result obtained by spatial nl-means with filtering parameter $h = 2.0$. (d) Result of the same method with $h = 3.5$. (e) Result by our method. Note that the eyes of Mandrill are pale and the canvas effect appears on the face of Mandrill (c) and (d). In the result of our method (e), the eyes are bright, and there is no artifacts on the face.

and 512MB RAM demonstrates that it takes no more than 11 seconds to denoise a 500 Megabyte photo using our algorithm, while the original nl-means algorithm

requires nearly 10 minutes. This makes the performance of our algorithm acceptable to common users as is demonstrated in Table 2.

Table 2. Performance Results

Image Size	Simplified Spatial NL (s)	Our Fast Method (s)	Ratio
512×512	28.16	0.38	74.1
1024×768	85.45	1.54	55.5
2592×1944	548.1	10.63	51.6

5 Conclusion

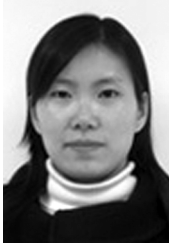
In the paper, we proposed a robust and fast non-local denoising algorithm. The algorithm is based on Laplacian pyramid. Due to the redundancy property of Laplacian pyramid, we can further perform nl-means building upon image redundancy in Laplacian pyramid.

We employ Laplacian pyramid to break up noisy image into bandpassed images. By performing nl-means on different levels of Laplacian pyramid with different sizes of compare windows, we effectively remove high-frequency noise and low-frequency noise while preserving the image details (edges, textures, etc.). We also utilize SSI and FFT to propose a speedup algorithm. After acceleration, our method is nearly fifty times faster than spatial nl-means algorithm. It can be seen that our accelerated algorithm is more feasible to tackle with practical problems.

Compared with spatial nl-means results, the resulting image of our algorithm is neat, while spatial nl-means results are rather blotchy. Our result is a bit more blurry but introduces less artifacts such as canvas effect than spatial nl-means. The limitation of the method, as shown in Table 2, is: when images to be denoised reach to a particularly large size, the ratio between times needed in our method and spatial nl-means algorithms slowly decrease. This is because, for large image, it takes several minutes to decompose it into Laplacian pyramid. We suggest this limitation can be further overcome by using GPU to implement the decomposition process.

References

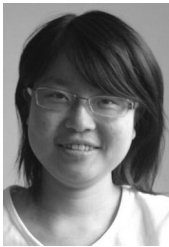
- [1] Lindenbaum M, Fischer M, Bruckstein A M. On Gabor contribution to image enhancement. *Pattern Recognition*, 1994, 27(1): 1–8.
- [2] Alvarez L, Lions P L, Morel J M. Image selective smoothing and edge detection by nonlinear diffusion (ii). *Journal of Numerical Analysis*, 1992, 29(3): 845–866.
- [3] Yin L, Yang R, Gabbouj M, Neuvo Y. Weighted median filters: A tutorial. *IEEE Trans. Circuits and Systems*, 1996, 43(3): 157–192.
- [4] Tomasi C, Manduchi R. Bilateral filtering for gray and color images. In *Proc. the Sixth International Conference on Computer Vision*, Bombay, India, 1998, pp.839–846.
- [5] Donoho D. De-noising by soft-thresholding. *IEEE Trans. Information Theory*, 1995, 41(3): 613–627.
- [6] Chambolle A, DeVore R A, Lee N Y, Lucier B J. Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Trans. Image Processing*, 1998, 7(1): 319–335.
- [7] Cohen I, Raz S, Malah D. Translation invariant denoising using the minimum description length criterion. *Signal Processing*, 1999, 75(3): 201–223.
- [8] Portilla J, Strela V, Wainwright M J, Simoncelli E P. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Processing*, 2003, 12(11): 1338–1351.
- [9] Romberg J, Choi H, Baraniuk R G. Bayesian tree-structured wavelet-domain image modeling using hidden Markov models. *IEEE Trans. Image Processing*, 2001, 10(7): 1056–1068.
- [10] Buades A, Coll B, Morel J M. A non-local algorithm for image denoising. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, USA, 2005, Vol.2, pp.60–65.
- [11] Ahmadian A, Bharath A A. Orthogonal wavelets for image transmission and compression schemes: Implementation and results. In *Proc. SPIE*, 1996, 2825(2): 822–833.
- [12] Kharate G K, Ghatol A A, Rege P P. Image compression using wavelet packet tree based on threshold entropy. In *Proc. the 24th IASTED International Conference on Signal Processing, Pattern Recognition, and Applications*, Innsbruck, Austria, 2006, pp.322–325.
- [13] Jansen M, Bultheel A. Multiple wavelet threshold estimation by generalized crossvalidation for images with correlated noise. *IEEE Trans. Image Processing*, 1999, 8(7): 947–953.
- [14] Aiazzi B, Alparone L, Baronti S, Borri G. Pyramid-based multiresolution adaptive filters for additive multiplicative image noise. *IEEE Trans. Circuits Syst. II*, 1998, 45(8): 1092–1097.
- [15] Burt P J, Adelson E H. The Laplacian pyramid as a compact image code. *IEEE Trans. Communications*, 1983, 31(4): 532–540.
- [16] Alexey L A. Multiresolution approach for improving quality of image denoising algorithms. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Toulouse, France, 2006.
- [17] Mahmoudi M, Sapiro G. Fast image and video denoising via nonlocal means of similar neighborhoods. *IEEE Signal Processing Letters*, 2005, 12(12): 839–842.
- [18] Heeger D J, Bergen J R. Pyramid-based texture analysis/synthesis. In *Proc. SIGGRAPH*, Los Angeles, USA, 1995, pp.229–238.
- [19] Greenspan H, Goodman R, Chellappa R, Anderson C H. Learning texture discrimination rules in a multiresolution system. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1994, 16(9): 894–901.
- [20] Bouzidi A, Baaziz N. Contourlet domain feature extraction for image content authentication. In *Proc. IEEE 8th Workshop on Multimedia Signal Processing*, Victoria, Canada, 2006, pp.202–206.
- [21] Fuchs C. Extraktion polymorpher Bildstrukturen und ihre topologische und geometrische Gruppierung. *DGK, Bayer. Akademie der Wissenschaften, Reihe C, Heft 502*, 1998.
- [22] Viola P, Michael J. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE CVPR*, Hanoi, USA, 2001, pp.511–518.



Yan-Li Liu received her B.S. degree in applied mathematics from Henan University in 2004. Now, she is a Ph.D. candidate in State Key Lab of CAD&CG, Zhejiang University, majoring in applied mathematics. Her research interests include image and video processing, face related research and augmented reality.



Jin Wang received his Ph.D. degree in computer science and technology from Zhejiang University, P.R. China in 2007. Currently he is a researcher in State Key Lab of CAD&CG. His research interests focus on photo/video enhancement, face relevant research, computer vision& computer graphics.



Xi Chen received her B.S. degree in computer science and technology from Zhejiang University, P.R. China in 2007. She is currently pursuing her M.Sc. degree in Department of Computer Science, University of British Columbia, Canada. Her research interests include image and video processing.



Yan-Wen Guo received his Ph.D. degree in applied mathematics from State Key Lab of CAD&CG, Zhejiang University in 2006. He is currently an assistant professor at the National Laboratory of Novel Software Technology, Nanjing University. He worked as a visiting researcher in Department of Computer Science and Engineering, the Chinese University of Hong Kong in 2006. His main research interests include image and video processing, geometry processing, and face related applications.



Qun-Sheng Peng graduated from Beijing Mechanical College in 1970 and received a Ph.D. degree from the Department of Computing Studies, University of East Anglia, UK in 1983. Now he is a professor of computer graphics at Zhejiang University. He serves currently as a member of the editorial boards of several international and Chinese journals. His research interests include realistic image synthesis, computer animation, scientific data visualization, virtual reality, bio-molecule modeling. He is a senior member of CCF.