

A Universal Noise Removal Algorithm With an Impulse Detector

Roman Garnett, Timothy Huegerich, Charles Chui, *Fellow, IEEE*, and Wenjie He, *Member, IEEE*

Abstract—We introduce a local image statistic for identifying noise pixels in images corrupted with impulse noise of random values. The statistical values quantify how different in intensity the particular pixels are from their most similar neighbors. We continue to demonstrate how this statistic may be incorporated into a filter designed to remove additive Gaussian noise. The result is a new filter capable of reducing both Gaussian and impulse noises from noisy images effectively, which performs remarkably well, both in terms of quantitative measures of signal restoration and qualitative judgements of image quality. Our approach is extended to automatically remove any mix of Gaussian and impulse noise.

Index Terms—Bilateral filter, denoising, Gaussian noise, image restoration, impulse noise, mixed noise, nonlinear filters.

I. INTRODUCTION

NOISE can be systematically introduced into images during acquisition and transmission. A fundamental problem of image processing is to effectively remove noise from an image while keeping its features intact. The nature of the problem depends on the type of noise added to the image. Fortunately, two noise models can adequately represent most noise added to images: additive Gaussian noise and impulse noise.

Additive Gaussian noise is characterized by adding to each image pixel a value from a zero-mean Gaussian distribution. Such noise is usually introduced during image acquisition. The zero-mean property of the distribution allows such noise to be removed by locally averaging pixel values. Ideally, removing Gaussian noise would involve smoothing inside the distinct regions of an image without degrading the sharpness of their edges. Classical linear filters, such as the Gaussian filter, smooth noise efficiently but blur edges significantly. To solve this problem, nonlinear methods have to be used, most notably the anisotropic diffusion technique of Perona and Malik [1]. Another interesting method is the bilateral filter studied by

Tomasi and Manducchi [2] based on the original idea of Overton and Weymouth [3]. The essence of these methods is to use local measures of an image to quantitatively detect edges and to smooth them less than the rest of the image.

Impulse noise is characterized by replacing a portion of an image's pixel values with random values, leaving the remainder unchanged. Such noise can be introduced due to transmission errors. The most noticeable and least acceptable pixels in the noisy image are then those whose intensities are much different from their neighbors.

The Gaussian noise removal methods mentioned above cannot adequately remove such noise because they interpret the noise pixels as edges to be preserved. For this reason, a separate class of nonlinear filters have been developed specifically for the removal of impulse noise; many are extensions of the median filter [4], [5], or otherwise use rank statistics [6]–[8]. The common idea among these filters is to detect the impulse pixels and replace them with estimated values, while leaving the remaining pixels unchanged. When applied to images corrupted with Gaussian noise; however, such filters are not effective, and in practice leave grainy, visually disappointing results.

There has not been much work carried out on building filters that can effectively remove both Gaussian and impulse noise, or any mixture thereof. Such “mixed noise” could occur, for instance, when sending an already noisy image over faulty communication lines. Peng and Lucke suggested a fuzzy filter designed specifically for mixed noise [9]. Additionally, in 1996 Abreu *et al.*, proposed the median-based SD-ROM filter to remove impulse noise, and their method proved quite effective [10]. They also gave quantitative measures demonstrating the SD-ROM filter's ability to remove Gaussian noise as well as mixed Gaussian and impulse noise. Although their filter has impressive quantitative results, when applied to images with Gaussian or mixed noise it often produces visually disappointing output similar to that of other median-based filters.

In this paper, we introduce a framework for creating a universal noise removal filter that is based on a simple statistic to detect impulse noise pixels in an image. Instead of applying the “detect and replace” methodology of most impulse noise removal techniques, we show how to integrate such a statistic into a filter designed to remove Gaussian noise. The behavior of the filter can be adaptively changed to remove impulses while retaining the ability to smooth Gaussian noise. Additionally, the filter can be easily adapted to remove mixed noise.

This paper is arranged as follows. In Section II, we introduce a local image statistic for detecting impulses. In Section III, we briefly explain the bilateral filter and describe how to incorporate the statistic into the filter to create a universal filter. Finally,

Manuscript received February 14, 2004; revised September 27, 2004. This work was supported in part by the National Science Foundation under Grants CCR-0098331 and CCR-9988289 and in part by the Army Research Office under Grant DAAD 19-00-1-0512. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nicolas Rougon.

R. Garnett is with the Departments of Mathematics and Computer Science, Washington University in Saint Louis, Saint Louis, MO 63130 USA (e-mail: roman@math.wustl.edu).

T. Huegerich is with the Department of Physics, Rice University, Houston, TX 77005 USA (e-mail: hugetim@alumni.rice.edu).

C. Chui is with the Department of Mathematics and Computer Science, University of Missouri–Saint Louis, Saint Louis, MO 63121 USA, and also with the Department of Statistics, Stanford University, Stanford, CA 94305 USA (e-mail: he@cs.ums.edu; ckchui@stanford.edu).

W. He is with the Department of Mathematics and Computer Science, University of Missouri–Saint Louis, Saint Louis, MO 63121 USA (e-mail: he@cs.ums.edu).

Digital Object Identifier 10.1109/TIP.2005.857261

in Section IV, we provide visual examples and numerical results that demonstrate our method's soundness.

II. NEW STATISTIC FOR DETECTING IMPULSES

A. Noise Models

Throughout this paper, we will use standard matrix notation for images. For example, when u is an image, $u_{i,j}$ will represent the intensity value of u at the pixel location (i, j) in the image domain. For the case of additive Gaussian noise, the noisy image u is related to the original image u^0 by

$$u_{i,j} = u_{i,j}^0 + n_{i,j}$$

where each noise value n is drawn from a zero-mean Gaussian distribution.

Impulse noise, again to be denoted by n , is characterized by replacing a portion of the original pixel values of the image with intensity values drawn from some distribution, usually either a uniform or discrete distribution n over the intensity range. Throughout this paper, we consider the uniform noise distribution model, although the methods we discuss could be used without modification for the discrete model.¹ Therefore, for images corrupted with impulse noise, the noisy image u is related to the original image u^0 by

$$u_{i,j} = \begin{cases} n_{i,j}, & \text{with probability } p \\ u_{i,j}^0, & \text{with probability } (1 - p). \end{cases}$$

B. Motivation of the Impulse Detection Scheme

The problem of deciding which pixels in an image are impulses is clearly not well defined. In particular, Lichtenstein or Seurat might be dismayed if he saw one of his paintings "denoised." Therefore, we must be content with detecting pixels that are like impulses, that is, pixels that vary greatly in intensity from most surrounding pixels. Thankfully, uncorrupted natural images rarely contain details isolated to a single pixel and generally have few impulse-like pixels.

Impulse noise removal methods use many different techniques to determine whether a given pixel is an impulse in this sense. These approaches vary in complexity from being relatively simple to highly complex. The most basic impulse detectors are based on two-state methods that attempt to definitively characterize each image pixel as either an impulse or an unaffected pixel. The underlying goal of these two-state methods is to find pixels that are significant outliers when compared to their neighbors. One of the simplest and most intuitive method is to compare a pixel's intensity with the median intensity in its neighborhood, as in [5]. Other methods, such as the two-state SD-ROM filter of Abreu *et al.* [10] and the recent CSAM filter of Pok *et al.* [8], use more complex criteria to judge whether a pixel is an impulse. The advantage of these two-state methods is their simplicity, which makes them easily customizable.

¹In contrast, conditional signal-adaptive median (CSAM) filtering [8], is designed for salt-and-pepper noise, a discrete impulse noise model in which the noisy pixels take only the values 0 and 255. It can remove salt-and-pepper type noise pixels very well, but it cannot perform similarly for uniformly distributed impulse noise and is outperformed by even the median filter.

More complex methods are naturally more successful for detecting impulses in general, but there is a tradeoff for this improvement in detection. The most complicated methods require training procedures to make an optimal classification based on measures of pixels and their neighbors. Methods that require training are bound to be less easily controlled and more unpredictable than simpler methods. If a method is trained on an image with many impulse-like pixels (such as an image with many fine details), its ability to detect and remove actual impulse noise from a different image will be inhibited. On the other hand, if the method is trained on a smooth image with few impulse-like pixels, it will overly smooth an image with many fine details that could otherwise be preserved. Even if care is taken to select an appropriate training image—assuming one is available—the performance of an automatically trained filter is essentially unpredictable.

An additional concern with existing methods arises from the fact that when impulse noise is introduced to an image, a portion of the pixels will be replaced with intensities only slightly different from their original values. Two-state detectors, with or without training, will most likely fail to detect such small impulses since they look exclusively for large outliers. They remove the most conspicuous noise, but the lesser impulses remain, creating a grainy appearance. In response to this problem, we adopt a continuous function to represent how impulse-like a particular pixel may be. By considering the magnitude of this function, we can adapt the behavior of the filter in a straightforward way according to how impulse-like each pixel is.

C. Definition of the Rank-Ordered Absolute Differences (ROAD) Statistic

Let $x = (x_1, x_2)$ be the location of the pixel under consideration, and let

$$\Omega_{\mathbf{x}}(N) := \{\mathbf{x} + (i, j) : -N \leq i, j \leq N\} \quad (1)$$

be the set of points in a $(2N + 1) \times (2N + 1)$ neighborhood centered at \mathbf{x} for some positive integer N . In the following discussion, let us only consider $N = 1$, though the same procedure can be extended to any $N > 1$. Hence

$$\Omega_{\mathbf{x}}^0 = \Omega_{\mathbf{x}}(1) \setminus \{\mathbf{x}\}$$

represents the set of points in a 3×3 deleted neighborhood of \mathbf{x} . For each point $\mathbf{y} \in \Omega_{\mathbf{x}}^0$, define $d_{\mathbf{x},\mathbf{y}}$ as the absolute difference in intensity of the pixels between \mathbf{x} and \mathbf{y} , i.e.,

$$d_{\mathbf{x},\mathbf{y}} = |u_{\mathbf{x}} - u_{\mathbf{y}}|.$$

Finally, sort the $d_{\mathbf{x},\mathbf{y}}$ values in increasing order and define

$$\text{ROAD}_m(\mathbf{x}) = \sum_{i=1}^m r_i(\mathbf{x}) \quad (2)$$

where $2 \leq m \leq 7$ and

$$r_i(\mathbf{x}) = i\text{th smallest } d_{\mathbf{x},\mathbf{y}} \text{ for } \mathbf{y} \in \Omega_{\mathbf{x}}^0.$$

We call the statistic defined in (2) ROAD. In this paper, we will consider $m = 4$ only, and set $\text{ROAD}(\mathbf{x}) = \text{ROAD}_4(\mathbf{x})$.

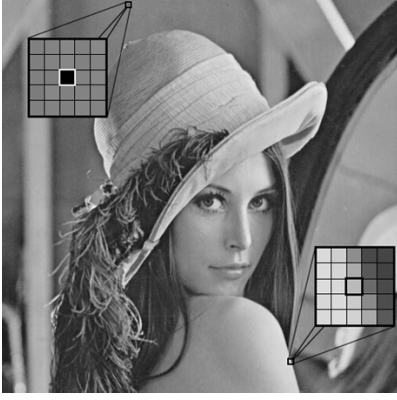


Fig. 1. Closeups of an artificially added impulse (upper left) and a typical edge pixel (lower right). ROAD of impulse: 525; ROAD of edge pixel: 88.

$$\begin{pmatrix} 213 & 171 & 88 \\ 216 & 186 & 107 \\ 218 & 202 & 139 \end{pmatrix} \rightarrow \begin{pmatrix} 27 & 15 & 98 \\ 30 & - & 79 \\ 32 & 16 & 47 \end{pmatrix}$$

Original Neighborhood Absolute Differences

$$\begin{aligned}
 r_1 &= 15, r_2 = 16 \\
 r_3 &= 27, r_4 = 30 \quad \rightarrow \quad 15 + 16 + 27 + 30 = 88
 \end{aligned}$$

The four smallest Final Calculation
absolute differences of ROAD

Fig. 2. Demonstrating how to calculate ROAD.

The ROAD statistic provides a measure of how close a pixel value is to its four most similar neighbors. The logic underlying the statistic is that unwanted impulses will vary greatly in intensity from most or all of their neighboring pixels, whereas most pixels composing the actual image should have at least half of their neighboring pixels of similar intensity, even pixels on an edge. Fig. 1 shows examples from the Lena image comparing a typical impulse noise pixel to an edge pixel. Notice that the edge pixel has neighbors of similar intensity despite forming part of an edge, and, thus, has a significantly lower ROAD value. Fig. 2 demonstrates how the latter value was calculated.

We illustrate numerically that the ROAD statistic is a good indicator of impulse noise. Ideally we want our statistic to be very high for impulse noise pixels and much lower for uncorrupted pixels. Fig. 3 displays quantitative results from the Lena image. The upper dashed line represents the mean ROAD value for noise pixels as a function of the amount of impulse noise added, and the lower dashed line represents the mean ROAD

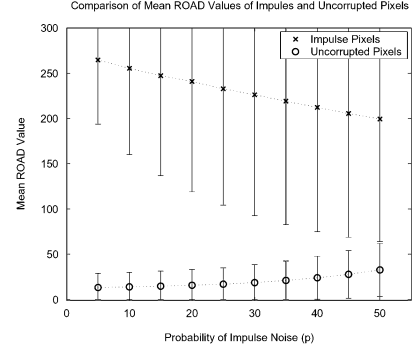


Fig. 3. Mean ROAD values of noise pixels and uncorrupted pixels in the Lena image as a function of the impulse noise probability, with standard deviation error bars demonstrating the significance of the difference.

value for uncorrupted pixels. The noise pixels consistently have much higher mean ROAD values than the uncorrupted pixels, whose mean ROAD values remain nearly constant even with very large amounts of noise.

III. INTRODUCING THE ROAD AND TRILATERAL FILTERS

It would be relatively simple to introduce the ROAD statistic into many existing filtering techniques, allowing them to detect and properly handle impulse-like pixels in a noisy image. For example, one could modify the popular anisotropic diffusion method to utilize the ROAD statistic. Below we describe how one might extend the bilateral filter to create a filter capable of removing both impulse and additive Gaussian noise from images. We begin with a brief introduction to the bilateral filter.

A. Bilateral Filter

The bilateral filter, as described in [2], applies a nonlinear filter to u to remove Gaussian noise while retaining the sharpness of edges. Each pixel is replaced by a weighted average of the intensities in a $(2N + 1) \times (2N + 1)$ neighborhood. The weighting function is designed to smooth in regions of similar intensity while keeping edges intact, by heavily weighting those pixels that are both near the central pixel spatially and similar to the central pixel radiometrically.

More precisely, let \mathbf{x} be the location of the pixel under consideration, and let

$$\Omega = \Omega_{\mathbf{x}}(N) \quad (3)$$

be the pixels in a $(2N + 1) \times (2N + 1)$ neighborhood of \mathbf{x} . The weight of each $\mathbf{y} \in \Omega$ with respect to \mathbf{x} is the product of two components, one spatial and one radiometric

$$w(\mathbf{x}, \mathbf{y}) = w_S(\mathbf{x}, \mathbf{y})w_R(\mathbf{x}, \mathbf{y}) \quad (4)$$

where

$$w_S(\mathbf{x}, \mathbf{y}) = e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{2\sigma_S^2}} \quad (5)$$

and

$$w_R(\mathbf{x}, \mathbf{y}) = e^{-\frac{|u_{\mathbf{x}}-u_{\mathbf{y}}|^2}{2\sigma_R^2}}. \quad (6)$$

The weights must be normalized, so the restored pixel \tilde{u}_x is given by

$$\tilde{u}_x = \frac{\sum_{y \in \Omega} w(x, y) u_y}{\sum_{y \in \Omega} w(x, y)}. \quad (7)$$

The w_S weighting function decreases as the spatial distance between x and y increases, and the w_R weighting function decreases as the radiometric “distance” between the intensities u_x and u_y increases. The spatial component of the weight decreases the influence of pixels far away from x to generally reduce blurring, while the radiometric component diminishes the influence of pixels with significantly different intensities to keep the edges of distinct image regions sharp. Notice that the w_S and w_R weighting functions need not be Gaussians—any suitable nonnegative functions that decrease to zero may be used instead.

In our particular weighting functions, the parameters σ_S and σ_R control the behavior of the weights. They are the values at which the respective Gaussian weighting functions take their maximum derivatives, so they serve as rough thresholds for identifying pixels sufficiently close spatially or radiometrically. Note, in particular, that as $\sigma_R \rightarrow \infty$ and radiometric differences are rendered irrelevant by this high threshold, the bilateral filter approaches a Gaussian filter of standard deviation σ_S . As both $\sigma_R, \sigma_S \rightarrow \infty$ so that all neighboring pixels easily meet both thresholds, the bilateral filter approaches the mean filter.

The idea for bilateral filtering was originally introduced in 1979 by Overton and Weymouth [3] as a simple image preprocessing tool to remove noise. In 1998, Tomasi and Manduchi [2] provided the name “bilateral filtering” and improved the method in several aspects. Specifically, they used Gaussian functions in the weighting functions (instead of the rational functions used in [3]) to improve the filter’s performance, analyzed the interaction between the weighting functions, and proposed metrics to be used for color images. Elad [11] established a connection between bilateral filtering and several other methods in terms of minimizing functionals.

B. New Weighting Function

We incorporate the ROAD statistic into the bilateral filtering framework by introducing a third weighting function influenced by how impulse-like each pixel of the image is. The “impulsive” weight w_I at a point x is given by

$$w_I(x) = e^{-\frac{\text{ROAD}(x)^2}{2\sigma_I^2}}. \quad (8)$$

The σ_I parameter determines the approximate threshold above which to penalize high ROAD values.

We would like to integrate this impulsive component into a nonlinear filter designed to weight pixels based on their spatial, radiometric, and impulsive properties. Unfortunately, the impulsive component is not directly compatible with the radiometric component already present in the bilateral filter. To illustrate this observation, let us consider black impulses on a white background. If the point x under consideration were such an impulse, the radiometric component would weight any other black impulses in the neighborhood of x much more than the white

background pixels we desire. As a result, the black impulses remain black impulses. The radiometric weight works contrary to our goal because it was not designed to remove impulse noise. However, if used selectively, the radiometric weight can still be helpful for removing impulse noise. It can help smooth impulses that are only slightly different from their surrounding pixels without blurring edges, while the impulsive weight works to remove the larger outliers. If we can use the radiometric component for only small impulses, we can improve upon the common two-state methods for impulse noise removal by not only removing the larger outliers, but also smoothing away smaller impulses.

To add the impulsive weight while still retaining the radiometric component of the bilateral filter, we introduce a switch to determine how much to use the radiometric component in the presence of impulse noise. If x is the central pixel under consideration, and $y \in \Omega_x(N)$ is a pixel in the neighborhood of x , we define the “joint impulsivity” J of y with respect to x as

$$J(x, y) = 1 - e^{-(\frac{\text{ROAD}(x) + \text{ROAD}(y)}{2})^2 / 2\sigma_J^2}. \quad (9)$$

The $J(x, y)$ function assumes values in $[0, 1]$. The σ_J parameter controls the shape of the function. Again, any suitably nonnegative function that decreases to zero may be used in place of the Gaussian. If *at least one* of x or y is impulse-like and has a high ROAD value with respect to σ_J , then $J(x, y) \approx 1$. If neither pixel is impulse like, and, thus, neither has a high ROAD value, then $J(x, y) \approx 0$.

We would like to use the radiometric weight more heavily when $J(x, y) \approx 0$ to smooth regions without large impulses and less heavily when $J(x, y) \approx 1$, because if either pixel is an impulse, the radiometric weight fails to function correctly as illustrated above. Conversely, we would like to use the impulsive weight less heavily when $J(x, y) \approx 0$ and more heavily when $J(x, y) \approx 1$, to suppress large impulses. With this in mind, we define the final, “trilateral” weight of y with respect to the central point x as

$$w(x, y) = w_S(x, y) w_R(x, y)^{1-J(x, y)} w_I(y)^{J(x, y)}. \quad (10)$$

Referring to the Gaussian forms of w_R and w_I , we see that raising these functions to the specified exponents has the effect of modifying their effective standard deviations or “thresholds.” When $J(x, y) \approx 1$ so that $1 - J(x, y) \approx 0$, the radiometric threshold becomes very large so that radiometric differences become irrelevant, while the impulsive weight is unaffected. When $J(x, y) \approx 0$, the opposite happens and only the radiometric weight is used to distinguish pixels because the effective impulsive threshold is so high. In this way, the appropriate weighting function is applied on a pixel-by-pixel basis. We will call the nonlinear filter of form (7) with the weighting function $w(x, y)$ given in (10) the “trilateral filter,” since it combines three different measures of neighboring pixels in determining its weights.

In general, the trilateral weighting function works well to remove impulse noise without compromising the bilateral filter’s ability to remove Gaussian noise. For images with no impulse noise—and, thus, few points with high ROAD values—the



Fig. 4. Original Lena image, the image corrupted with impulse noise ($p = 20\%$), and the result after trilateral filtering.

$J(\mathbf{x}, \mathbf{y})$ term in (10) effectively “shuts off” the impulsive component of the weight and only uses the radiometric and spatial weights. Essentially, the trilateral filter reverts to the bilateral filter when processing images without impulse noise.

The $J(\mathbf{x}, \mathbf{y})$ function also allows for future work to further extend the filter. Specifically, we are exploring methods to automatically choose the control parameters σ_S , σ_R , σ_I , and σ_J locally. This would allow us to remove different types of noise from the same image, as well as mixed Gaussian and impulse noise in a single pass.

We have found that even for very high levels of impulse noise, one pass of the trilateral filter will remove almost all of the noise. However, for $p > 25\%$, a few spots of unremoved impulses often remain. This happens because impulses sometimes “clump” together in the original noisy image to form regions of similar intensity so large that they are mistaken for meaningful features. To remove such residual spots, it is often helpful to process the image with an iterative version of the filter. We simply run the image through the trilateral filter several times, using the output of the previous iteration as the input of the next. For low and moderate levels of noise ($p \leq 25\%$), one iteration is sufficient and usually provides the best results. For high levels of noise ($p > 25\%$), applying two to five iterations provides better results.

C. Selection of Parameters

To remove Gaussian noise, we take $\sigma_S = 5$. For suppressing impulse noise, however, smaller values of σ_S , on the order of 0.5, work best. We set σ_R to be about twice the standard deviation of the added Gaussian noise for automatically selecting σ_R .

In [14], Immerkær provides a fast method of estimating the standard deviation of additive Gaussian noise in an image. We modify his method slightly by using a weighted average to discount pixels with high ROAD values. Our estimator for the standard deviation of “quasi-Gaussian noise” exhibited by small impulses, σ_{QGN} , is given by

$$\hat{\sigma}_{\text{QGN}} = \sqrt{\frac{\pi}{2}} \frac{1}{6 \sum_{i,j=1}^{m,n} w_I(i,j)} \sum_{i,j=1}^{m,n} |(u * L)_{i,j}| w_I(i,j)$$

where

$$L = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$

is the Laplacian filter.

We have also found through experimentation that a good initial value of σ_I is 40. In general, σ_I should never be outside the interval [25, 55]. Experimental results have shown that the optimal value of σ_J for most images is around 50, but any value in the interval [30, 80] should work well to remove both impulse and Gaussian noise.

D. Removing Mixed Gaussian and Impulse Noise

The trilateral filter can be easily extended to remove any mixture of Gaussian and impulse noise. The ideal solution would be to locally vary parameters so that they are finely tuned to remove the precise amount and type of noise present in each section of the image. This solution, however, would require a deep statistical study of the ROAD statistic for the automatic selection of parameters, and the best way to do this is not immediately clear.

A simpler, yet still quite effective, solution to restore an image corrupted by mixed noise is to apply the trilateral filter twice with two different values of σ_S —once with a smaller value of σ_S , to remove the impulse noise, and another time with a larger value of σ_S , to smooth the remaining Gaussian noise. A myriad of other options are available for altering the parameters between filtering, but our simple approach produces visually appealing results and only requires changing one parameter.

IV. RESULTS

We have extensively tested the noise removal capabilities of the trilateral filter and compared the results with several existing methods. Our method produced results superior to the other methods we tested in both visual image quality and quantitative measures of signal restoration. Among the many images we tested are the 512×512 8-bit grayscale Lena image, available online from Mike Wakin at Rice University [15], and the 512×512 “bridge” image, available online from the UEA Signal and Image Processing Group [16].

A. Implementation and Testing Procedure

Our implementation of the trilateral filter used a 5×5 window size and performed multiple iterations when it provided better results (for $p > 25\%$). Image boundaries were handled by assuming symmetric boundary conditions.

In each of our experiments, we strove to be impartial while collecting data. For each method tested, we varied its parameters exhaustively [as suggested by its author(s)] to obtain the best possible result. Furthermore, to eliminate the bias created

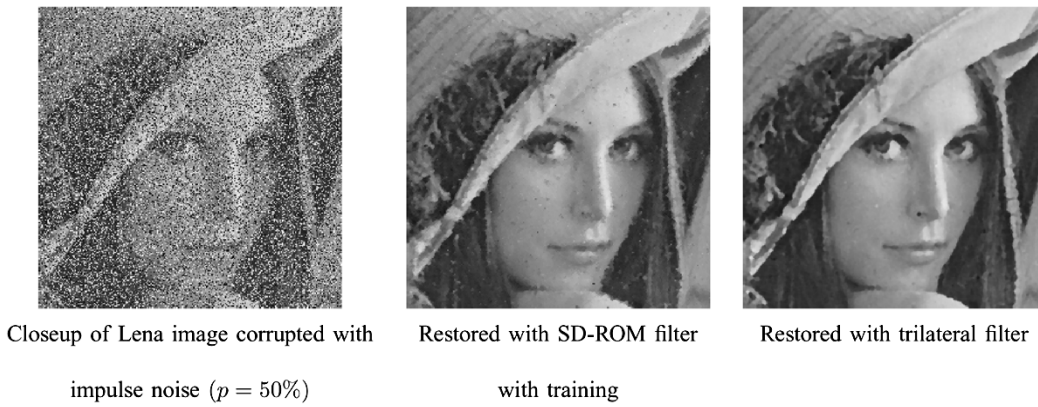


Fig. 5. Lena image corrupted with a high level of impulse noise and the results of applying several filters.

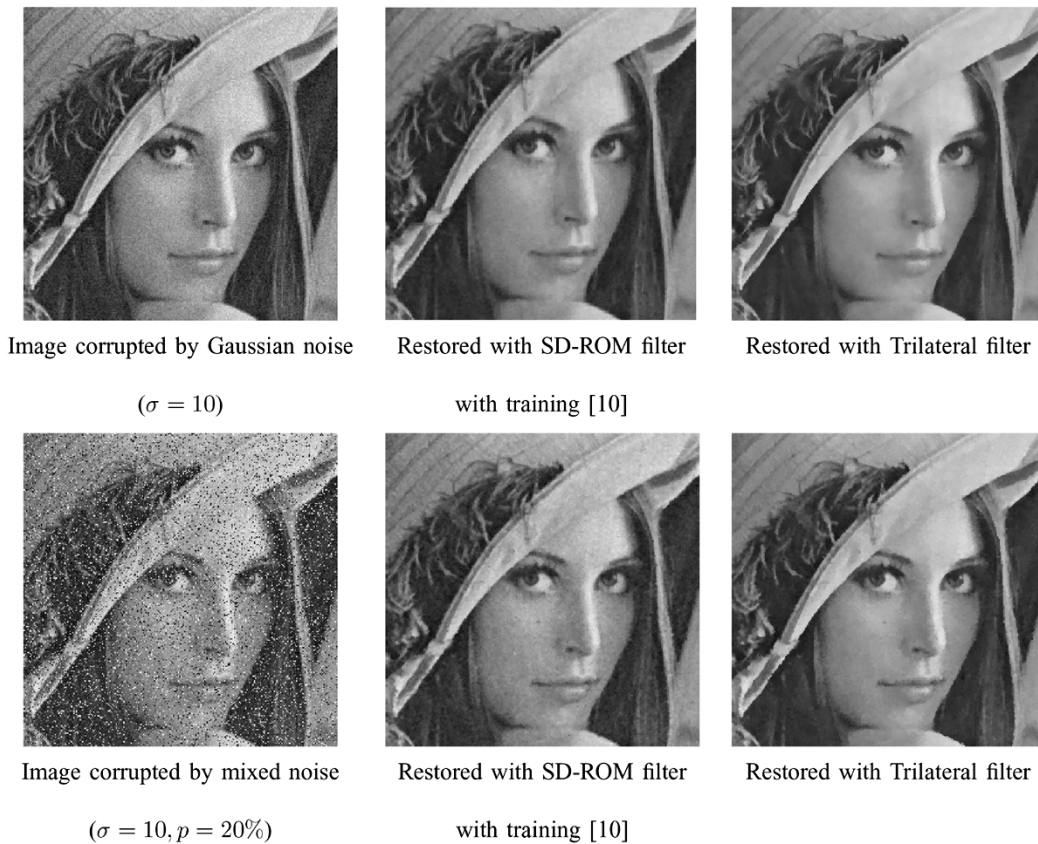


Fig. 6. Comparing the trilateral filter and SD-ROM filter on images with Gaussian noise and mixed Gaussian and impulse noise.

by different manifestations of noise, we created a standard set of noisy images. Five noisy images were created for each test image and noise level and formed the common input to each method. The numerical results shown are the average results for these five images.

B. Image Quality

Our first goal was to ensure that our approach provides visually pleasing output. The trilateral filter can restore images corrupted with low to moderate levels of impulse noise ($p \leq 25\%$) with virtually unblemished results. Fig. 4 shows the Lena image corrupted with 20% impulse noise and the result after trilateral filtering. Comparing with the original, it is clear that the trilateral filter can eliminate a fair amount of noise while preserving edge boundaries and fine details.

Although exceptional for lower impulse noise levels, our filter truly excelled when treating highly corrupted ($p > 25\%$) images. Fig. 5 shows the Lena image with 50% impulse noise and compares the outputs of the trilateral filter with the output of the SD-ROM filter of Abreu *et al.* The trilateral filter's output has fewer spots and other artifacts and is generally more pleasing.

We also verified that the trilateral filter retains the ability to remove Gaussian noise and that it can effectively remove mixed noise. Fig. 6 shows the Lena image corrupted with Gaussian noise and mixed noise and the outputs of the SD-ROM and trilateral filters. The trilateral filter continues to adequately suppress Gaussian noise, even after the introduction of the impulsive weight and the joint impulsivity function.

For both Gaussian and mixed noise, the trilateral filter leaves less noise in the restored image than the SD-ROM filter, and

TABLE I
RESULTS AFTER APPLYING VARIOUS FILTERS TO IMAGES CORRUPTED WITH IMPULSE NOISE

Method	Lena Image				Bridge Image			
	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$	$p = 20\%$	$p = 30\%$	$p = 40\%$	$p = 50\%$
3×3 Median Filter	31.23 dB	28.05 dB	24.61 dB	21.53 dB	24.76 dB	23.14 dB	21.20 dB	19.04 dB
5×5 Median Filter	29.87 dB	28.97 dB	27.61 dB	25.45 dB	23.26 dB	22.73 dB	21.95 dB	20.72 dB
Median Filter with Adaptive Length ^a [4]	31.51 dB	29.49 dB	27.75 dB	26.33 dB	25.14 dB	23.70 dB	22.08 dB	21.03 dB
Sun and Neuvo Switching Scheme I ^{a,b} [5]	32.15 dB	29.69 dB	27.97 dB	25.54 dB	26.16 dB	24.34 dB	22.75 dB	21.02 dB
Rank Conditioned Rank Selection Filter ^d [7]	31.81 dB	30.01 dB	27.72 dB	25.92 dB	24.76 dB	23.68 dB	22.52 dB	20.88 dB
SD-ROM without training ^{a,c} [10]	33.98 dB	31.06 dB	28.30 dB	25.57 dB	26.57 dB	24.82 dB	23.17 dB	21.40 dB
SD-ROM with training ^{a,d} [10]	34.23 dB	31.64 dB	29.19 dB	26.63 dB	26.24 dB	24.98 dB	23.66 dB	22.19 dB
Trilateral Filter ^e	35.03 dB	33.16 dB	31.36 dB	29.44 dB	27.55 dB	25.99 dB	24.55 dB	23.17 dB

^aImplemented recursively.

^bImplemented using a 5×5 window for Lena with $p \geq 30\%$, and using a 3×3 window otherwise

^cUsing the thresholds suggested in [10].

^dThe Lena image was trained on the “bridge” image with 35% noise, and the “bridge” image was trained on the Lena image with 35% noise.

^eImplemented iteratively for $p \geq 30\%$.

its output is generally more visually appealing. However, some fine details are lost in the process. For example, the individual strands of hair in the image are better kept in the SD-ROM filter’s output, at the expense of a less smooth and less pleasing output.

C. Signal Restoration

Once the visual quality of images restored by the trilateral filter had been confirmed, we concentrated on directly comparable, quantitative measures of signal restoration. In particular, we measured the peak signal-to-noise ratio (PSNR). If u^0 is the original $m \times n$ image and \tilde{u} is a restored image of u^0 , the PSNR of \tilde{u} is given by

$$\text{PSNR}(\tilde{u}) = 10 \log_{10} \left(\frac{\sum_{i,j=1}^{m,n} 255^2}{\sum_{i,j=1}^{m,n} (\tilde{u}_{i,j} - u_{i,j}^0)^2} \right). \quad (11)$$

Larger PSNR values signify better signal restoration. We tested each method on impulse noise levels from $p = 0\%$ (that is, no noise) to $p = 50\%$ in steps of 5%. Table I compares the mean PSNR values of the five restored test images for $p \in \{20\%, 30\%, 40\%, 50\%\}$. Again, the trilateral filter provided results with higher PSNR values than the results of the other methods tested, especially for very high levels of noise. In particular, for the Lena image with 50% noise, the trilateral filter produces PSNR values almost a full three decibels higher than the closest competing method.

TABLE II
PSNR VALUES (IN DECIBELS) OF NOISY IMAGES RESTORED WITH VARIOUS FILTERS

Filter	Gaussian Noise $\sigma = 10$		Mixed Noise $p = 20\%, \sigma = 10$	
	Lena	Bridge	Lena	Bridge
3×3 Median	30.92	25.93	29.02	24.20
RCRS ^b [7]	31.00	27.38	29.55	24.46
SD-ROM ^{a,b} [10]	32.20	28.85	29.87	25.10
Trilateral ^c	33.23	29.33	31.64	26.29

^aImplemented recursively.

^bThe Lena image was trained on the “bridge” image, and the “bridge” image was trained on the Lena image with the same level of noise.

^cImplemented iteratively.

We also compared the performance of the trilateral filter with the performance of the previously tested filters on images corrupted with Gaussian noise ($\sigma = 10$) and mixed noise ($\sigma = 10, p = 20\%$). The trilateral filter consistently yielded the highest PSNR for each image and noise level. Table II shows the results for the 3×3 median filter, the trilateral filter, and the previously-tested methods with training procedures: the RCRS and SD-ROM filters. Notice that the other methods generally performed only slightly better than the median filter for mixed noise.

V. CONCLUSION

Many noise removal algorithms, such as the bilateral filtering, tend to treat impulse noise as edge pixels, and, hence, end with unsatisfactory results. In order to process impulse pixels and edge pixels differently, we introduce a new statistic based on ROAD in some neighborhood of a pixel. This statistic represents how impulse-like a particular pixel is in the sense that the larger the impulse, the greater the ROAD value.

We then incorporate the ROAD statistic into the bilateral filtering by adding a third component to the weighting function. The new nonlinear filter is called the trilateral filter, whose weighting function contains spatial, radiometric, and impulsive components. The radiometric component combined with the spatial component smooths away Gaussian noise and smaller impulse noise, while the impulsive component removes larger impulses. A switch based on the ROAD statistic is adopted to adjust weight distribution between the radiometric and impulsive components. The resulting trilateral filter performs well in removing Gaussian and mixed noise as well as in removing impulse noise.

REFERENCES

- [1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 5, pp. 629–639, May 1990.
- [2] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Computer Vision*, 1998, pp. 839–846.
- [3] K. J. Overton and T. E. Weymouth, "A noise reducing preprocessing algorithm," in *Proc. IEEE Computer Science Conf. Pattern Recognition and Image Processing*, Chicago, IL, 1979, pp. 498–507.
- [4] H. Lin and A. N. Willson Jr., "Median filters with adaptive length," *IEEE Trans. Circuits Syst.*, vol. 35, no. 6, pp. 675–690, Jun. 1988.
- [5] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing," *Pattern Recognit. Lett.*, vol. 15, pp. 341–347, Apr. 1994.
- [6] A. C. Bovik, T. S. Huang, and D. C. Munson, "A generalization of median filtering using linear combinations of order statistics," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 6, pp. 1342–1350, Dec. 1983.
- [7] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," *IEEE Trans. Image Process.*, vol. 3, no. 3, pp. 192–206, Mar. 1994.
- [8] G. Pok, J. Liu, and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Trans. Image Process.*, vol. 12, no. 1, pp. 85–92, Jan. 2003.
- [9] S. Peng and L. Lucke, "Multi-level adaptive fuzzy filter for mixed noise removal," in *Proc. IEEE Int. Symp. Circuits Systems*, vol. 2, Seattle, WA, Apr. 1995, pp. 1524–1527.
- [10] E. Abreu, M. Lightstone, S. Mitra, and K. Arakawa, "A new efficient approach for the removal of impulse noise from highly corrupted images," *IEEE Trans. Image Process.*, vol. 5, no. 6, pp. 1012–1025, Jun. 1996.
- [11] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. Image Process.*, vol. 11, no. 10, pp. 1141–1151, Oct. 2002.
- [12] R. L. Lagendijk, J. Biemond, and D. E. Boeke, "Regularized iterative image restoration with ringing reduction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 6, pp. 1874–1887, Dec. 1988.
- [13] M. J. Black and G. Sapiro, "Edges as outliers: Anisotropic smoothing using local image statistics," in *Proc. Scale-Space Theories in Comput. Vision Second Int. Conf.*, Corfu, Greece, Sep. 1999, pp. 259–270. LNCS 1682.
- [14] J. Immerkær, "Fast noise variance estimation," *Comput. Vis. Image Understand.*, vol. 64, pp. 300–302, Sep. 1996.
- [15] Standard Test Images—Lena, M. Wakin. [Online]. Available: <http://www-ece.rice.edu/~wakin/images/>
- [16] Standard Image Page [Online]. Available: http://www.sys.uea.ac.uk/Research/researchareas/imagevision/images_ftp/

Roman Garnett received the B.S. degree in mathematics and the M.Sc. degree in computer science from Washington University in Saint Louis, Saint Louis, MO, in May 2004.

The work for this paper was completed while he was a Research Aide at The Institute for Computational Harmonic Analysis, University of Missouri–Saint Louis. His research interests include image and video processing and computer vision.

Timothy Huegerich received the B.S. degree in physics from Rice University, Houston, TX, in May 2004.

The work for this paper was completed while he was a Research Aide at The Institute for Computational Harmonic Analysis, University of Missouri–Saint Louis. His research interests include mathematical ecology and epidemiology.

Charles Chui (F'94) received the B.S., M.S., and Ph.D. degrees from the University of Wisconsin, Madison.

He is currently the Curators' Professor at the University of Missouri–Saint Louis, and a Consulting Professor of statistics at Stanford University, Stanford, CA. He is the Co-Editor-in-Chief of *Applied and Computational Harmonic Analysis* and serves on the editorial boards of seven other journals. In addition, he is the Editor of a three-book series, as well as the Editorial Advisor of the Elsevier series in electrical engineering. His research interests are in the areas of approximation theory, computational harmonic analysis, surface subdivisions, and mathematics of imaging.

Wenjie He (M'04) received the B.S. degree in mathematics from Peking University, Beijing, China, in 1988, and the Ph.D. degree in mathematics from the University of Georgia, Athens, in 1998.

He is currently an Associate Professor of computer science at the University of Missouri–Saint Louis. His research interests include wavelets, wavelet tight frames, computer graphics, and image processing.