An Anisotropic Noise Reduction Technique Using Semi-Local Edge Detection, and a Target Application

Mark Roden, MS Biomedical Engineering, University of California at Los Angeles, mmroden@gmail.com

August 11, 2008

Abstract

A practical application of anisotropic noise reduction via the Poisson equation is presented. Anisotropic coefficients are obtained through a semi-local determination of edge strength, finely tuning the gradient operator to favor consistent edges over purely local edges. For some, but not all, applications, this approach is comparable or preferable to Total Variation and Non-Local Means noise reduction techniques. Specifically, in the field of x-ray mammography, this algorithm produces results which are preferred by radiologists over both the original data and other algorithms, and also produces results of potentially greater clinical significance.

1 Introduction

A noisy image is represented with the following equation:

$$u^0 = u + n \tag{1}$$

where u represents the original, non-noisy image, n is the noise, and u^0 is the 'initial image' of the combined image and noise. The goal of denoising is to remove the n term while still preserving the discontinuities resulting from sharp edges in u. The Gaussian Blur method is essentially the replacement of each pixel with a local average; these filters are called 'low-pass filters', as they allow low frequency components to remain in the image but remove the high frequency components, including both noise and edge information. The formula for the Gaussian Kernel is:

$$J(x, y, \sigma) = \frac{1}{2\pi\sigma} e^{(x^2 + y^2)/2\sigma^2}$$
(2)

and the application of the Gaussian is done via the convolution operation, which has the form

$$u(x, y, \sigma) = \int \int J(x - x', y - y', \sigma) u_0(x', y') dx', dy' = J * u^0.$$
(3)

This technique is the same as applying the heat equation to the original image. That is, if the intensity of individual pixels in the image were considered to be local temperatures, and the heat were allowed to flow evenly throughout the domain, then the image would be blurred as the various different temperatures melded together. The steady-state version of this equation is:

$$\nabla^2 u = 0. \tag{4}$$

In other words, with constant application of the Gaussian filter to a steady state, then the image will become completely blurred with no preserved features. To preserve any features, a fidelity term will be added, to create a variant of the Poisson equation:

$$\nabla^2 u = \lambda (u - u^0) \tag{5}$$

where λ represents the inverse of the radius of the denoising. As λ would increase, the feature preservation would take precedence, and as λ decreases, the blurring term becomes more prevalent.

In order to use Eq. (5) on a digital image, it will have to be discretized. Discretization is the process of converting a continuous equation into an equation on a grid. The grid is assumed to be isotropic; that is, pixels are the same size in both the x and y dimensions. A second-order discretization of this equation is:

$$u_{l+1,m} + u_{l-1,m} + u_{l,m+1} + u_{l,m-1} - 4u_{l,m} = \lambda(u_{l,m} - u_{l,m}^{0})$$
(6)

where $u_{l,m}^0$ is the initial value of a pixel at a given point, and the value of a pixel at point (l, m) is replaced by the value of its neighbors [6]. Solving the above equation for $u_{l,m}$ yields:

$$u_{l,m} = \frac{\lambda u_{l,m}^0 + u_{l+1,m} + u_{l-1,m} + u_{l,m+1} + u_{l,m-1}}{4 + \lambda}.$$
(7)

However, this version of the equation, while a traditional method for regularized noise reduction, does not allow for the equation to be tailored to local conditions in the image. To do so, the coefficients of the equation will be addressed. In Eq. (7), the coefficients for all the neighboring terms are 1, and the center pixel $u_{l,m}$ is multiplied by the summation of those coefficients. In order to properly address local conditions, those coefficients will instead represent the strength of the boundary between pixels, a problem whose solution will be addressed shortly. Starting from the beginning, then, the original equation as shown in Eq (5) will become:

$$u_{yy} + u_{xx} = \lambda(u - u^0) \tag{8}$$

which is the same equation as Eq. (5), but considering the gradients in each of the different directions on the grid. When using separate coefficients for each direction, the equation becomes:

$$(au_y)_y + (bu_x)_x = \lambda(u - u^0) \tag{9}$$

which has an eight-point discretization of:

$$a_{1}u_{l,m+1} + a_{2}u_{l,m-1} + b_{1}u_{l+1,m} + b_{2}u_{l-1,m} + c_{1}u_{l+1,m+1} + c_{2}u_{l-1,m-1} + d_{1}u_{l-1,m+1} + d_{2}u_{l+1,m-1} - u_{l,m}(a_{1} + a_{2} + b_{1} + b_{2} + c_{1} + c_{2} + d_{1} + d_{2}) = \lambda(u_{l,m} - u_{l,m}^{0}).$$

$$(10)$$

After solving for $u_{l,m}$, Eq. (10) becomes:

$$u_{l,m} = \frac{\lambda u_{l,m}^{0} + a_1 u_{l,m+1} + a_2 u_{l,m-1} + b_1 u_{l+1,m} + b_2 u_{l-1,m} + c_1 u_{l+1,m+1} + c_2 u_{l-1,m-1} + d_1 u_{l-1,m+1} + d_2 u_{l+1,m-1}}{a_1 + a_2 + b_1 + b_2 + c_1 + c_2 + d_1 + d_2 + \lambda}$$
(11)

which will be the equation that will be solved. The construction of the local coefficients a, b, c, d, and λ will be discussed in the next sections. Briefly, a lower value for a coefficient results in a lower contribution of the neighboring pixel to the resulting value of the center pixel. Thus, an edge should be represented by low coefficient values, and a region with no significant edges should be represented with high coefficient values. By constructing the coefficients in this manner, 'heat' can spread over weak edges (noise) and not flow over strong edges. As shall be discussed with stopping criteria, this property of the equation can be used to result in an image with piecewise constant regions of intensity, or with regions with much of the original detail preserved.

Iterations can be performed by running the equation subsequent times; in the case of repeated applications, the update equation would become:

$$u_{l,m}^{n+1} = \frac{\lambda u_{l,m}^{n} + a_1 u_{l,m+1}^{n} + a_2 u_{l,m-1}^{n} + b_1 u_{l+1,m}^{n} + b_2 u_{l-1,m}^{n} + c_1 u_{l+1,m+1}^{n} + c_2 u_{l-1,m-1}^{n} + d_1 u_{l-1,m+1}^{n} + d_2 u_{l+1,m-1}^{n}}{a_1 + a_2 + b_1 + b_2 + c_1 + c_2 + d_1 + d_2 + \lambda}$$
(12)

where $u_{l,m}^n$ now represents the pixel value from the previous iteration.

This paper shall discuss the solution to Eq. (11) in great detail. First, the construction of the particular coefficients a, b, c, and d will be addressed, as these coefficients provide the basis for determining the local characteristics of the image. Second, the other coefficients of the equation, λ , α , σ and the stopping criteria shall be addressed, to provide a complete understanding of the solution. Third, the results of the application of this equation shall be compared with two other well-known noise reduction methods in order to demonstrate the particular utility of the described method. Finally, a discussion shall be presented to provide an explanation for the behavior of the algorithm on projected x-ray images, and why this method is preferable to previous alternatives.

2 Interpixel Coefficient Construction

In the previous section, coefficients were described which determine the flow of heat in each direction in the image. The construction of these coefficients will rely on linear differences derived from linear integrals, with a justification for the approach graphically explained in Figure 1. Lines are constructed from the discrete representation of the image using an iterative approach. In the end, the difference between two pixels will be the maximal difference between two sets of integrals compared one by one; in this way, angular information about the relationships between pixels will be represented by the local difference between pixels. This incorporation

of more information than the simple pixel-by-pixel gradient explains the use of the term 'semi-local' in describing the algorithm. Integrals are constructed using a set of base integrals, where each pixel is averaged with three of its neighbors.

$$S_{i,j,1}^{1} = (u_{i,j} + u_{i+1,j-1})/2$$

$$S_{i,j,2}^{1} = (u_{i,j} + u_{i+1,j})/2$$

$$S_{i,j,3}^{1} = (u_{i,j} + u_{i+1,j+1})/2.$$
(13)

The $S_{i,j,n}^0$ term refers to the *nth* integral of the 0*th* construction iteration at pixel location *i*, *j* where the image is on the domain i = 1...I and j = 1...J. Lines are subsequently constructed by concatenating these segments together. The previous work upon which this work is based[2] performed the concatenation operation by adding segments together, and were able to arrive at a fast implementation of the Radon transform as a result. Rather than add the segments together, these segments will be stored in sequence, such that each segment can be appropriately weighted, as shall be discussed.

Line integrals shall be built iteratively using the following approach. Each iteration shall produce 2n + 1 new concatenated integrals based upon the previous iteration, as explained in Figure 2. Integrals with odd indeces (1, 3, 5, etc) shall be simple concatenations, while integrals with even indeces (2, 4, 6, etc) shall be combined before concatenation. The combination operation is defined in this work as the non-weighted averaging of the underlying two pixel line integrals, but just as concatenation stores the underlying data rather than adding lines together, so too shall the combination operation simply associate edges with one another, and only during the final determination of edge strength resolve to a single number. Continuing with the previously described indexing scheme, the next iteration shall proceed as follows:

$$S_{i,j,1}^{2} = concat\{S_{i,j,1}^{1}; S_{i+1,j-1,1}^{1}\}$$

$$S_{i,j,2}^{2} = concat\{combine(S_{i,j,1}^{1}; S_{i,j,2}^{1}); combine(S_{i+1,j-1,2}^{1}; S_{i+1,j,1}^{1})\}$$

$$S_{i,j,3}^{2} = concat\{S_{i,j,2}^{1}; S_{i+1,j,2}^{1}\}$$

$$S_{i,j,4}^{2} = concat\{combine(S_{i,j,1}^{1}; S_{i,j,2}^{1}); combine(S_{i+1,j,3}^{1}; S_{i+1,j+1,2}^{1})\}$$

$$S_{i,j,5}^{2} = concat\{S_{i,j,3}^{1}; S_{i+1,j+1,3}^{1}\}.$$
(14)

This scheme can be generalized to the following algorithm, where N is the number of line integrals in the previous iteration q - 1, M is the number of integrals in the current iteration q:

$$S_{i,j,1:2:M}^{q} = concat \{S_{i,j,1:N}^{q-1}; S_{i+q-1,j-(1:N-ceil(N/2)),1:N}^{q-1}\}$$

$$S_{i,j,2:2:M}^{q} = concat \{combine(S_{i,j,1:N}^{q-1}; S_{i,j,1:N+1}^{q-1}); combine(S_{i+q-1,j-(1:N-ceil(N/2)),1:N}^{q-1}; S_{i+q-1,j-(1:N-ceil(N/2)+1),1:N-1}^{q-1}\}.$$
(15)

In order to construct integrals perpendicular to these, the image can be rotated and integrals recalculated. Additionally, a set of noise integrals should be calculated as well, using the same iterative formula, but based off of the set of differences:

$$N_{i,j,1}^{1} = |(u_{i,j} - u_{i+1,j-1})/2|$$

$$N_{i,j,2}^{1} = |(u_{i,j} - u_{i+1,j})/2|$$

$$N_{i,j,3}^{1} = |(u_{i,j} - u_{i+1,j+1})/2|.$$
(16)

These noise integrals are grown in a similar manner to the line integrals, via concatenation and combination, until the noise integral is the same length as the line integral. Once lines have been constructed (practically, integrals of length 16 are sufficient, or to S^5 and N^5), the individual segments will be weighted with a gaussian function:

$$LI_{i,j,1:M} = \sum_{k=0}^{l} S_{i,j,1:M,1:k}^{q} * e^{\frac{(k-l/2)^{2}}{2\pi\sigma^{2}}}$$
(17)

and, similarly, for the noise integrals:

-

$$NI_{i,j,1:M} = \sum_{k=0}^{l} N_{i,j,1:M,1:k}^{q} * e^{\frac{(k-l/2)^{2}}{2\pi\sigma^{2}}}$$
(18)

where the fourth index k into the set of lines S enumerates the individual concatenated and combined segments constructed previously, and l represents the length, or the number of 0-level segments (constructed in Eqs. (13) and (16)). This gaussian weighting will ensure that pixels near the center of the segment will be weighted higher than pixels near the edges, similar to a much longer Sobel kernel [3]. Note that this sequence of line integrals will radiate from a single point, as shown in Figure 2– in order to compare one pixel to its neighbor, these integrals will have to be rearranged as shown in Figure 3. Given this set of line integrals, the variation in the idirection will be defined as

$$\phi_{i,j} = \min_{m} e^{-\alpha |LI_{i,j,m}^{1} - LI_{i,j-1,m}^{2}|/|NI_{i,j,m}^{1} + NI_{i,j-1,m}^{2} + \beta|}.$$
(19)

This function is bounded by (0, 1], and determines the strength of the local gradient which will be employed during noise reduction. Specifically, this ϕ describes the semi-local edge strength between pixel $u_{i,j}$ and $u_{i,j-1}$, or, as described in Eq. (11), the constant a_2 , with a_1 representing the difference between $u_{i,j+1}$ and $u_{i,j}$. This entire process is repeated on the image after it has been rotated by 90° in order to obtain a function ψ which describes the difference between $u_{i,j}$ and $u_{i,j-1}$, or the the *b* coefficients. Diagonal coefficients are chosen from half of ϕ and half of ψ , appropriate for the particular direction chosen (see Fig. 3).

3 Construction of Further Parameters and Stopping Criteria

Three parameters, λ , σ , and α , have been introduced but their selection has not been explained. Rather than rely on empirical selection of these parameters, each can be constructed in an adaptive manner for the particular image. Each parameter controls a particular aspect of the noise reduction. λ describes the spread heat from a particular individual pixel as shown in Eq. (11), σ describes the strength of the edge differences in Eq. (17), and α serves a similar purpose as σ in describing a more global determination of heat spread from Eq. (19).

The parameter λ will be the inverse of the summation of the local coefficients, ie,

$$\lambda = \max(a_1 + a_2 + b_1 + b_2 + c_1 + c_2 + d_1 + d_2) - (a_1 + a_2 + b_1 + b_2 + c_1 + c_2 + d_1 + d_2)$$
(20)

where the maximum value will be either 8, if diagonal coefficients are weighted by 1, or $4 + 4/\sqrt{2}$, if diagonal coefficients are distance weighted by $1/\sqrt{2}$. Should the pixel be along a strong edge, then that particular edge pixel will not contribute to the value of λ , but the other pixels which are not involved in the edge will contribute highly to the pixel. Intuitively, λ represents the inverse of the square of the distance of the 'spread' of heat. A higher value of λ preserves more of the value of the central point $u_{l,m}$, while a lower λ places lower emphasis on the original central point $u_{l,m}$ and instead replaces the value with those of its neighbors. Thus, a pixel not near any edges will have high edge coefficients, and a lower λ .

The selection of σ depends on the desired length of the line integrals used to reduce noise in the image. As described, each line will be multiplied by some value derived from a Gaussian distribution; should that distribution be very slender with a σ that is very low, shorter lines will be required to capture enough of the important information in the distribution. A low σ , around 0.25, would require integrals of length 2; while larger values, such as 10, would require integrals of length 16 to sufficiently capture enough of the gaussian distribution along the length of the integral. Practically, a set of candidate values of σ can be chosen, such as 0.25, 1, 3, and 10, each requiring integrals of length 2, 4, 8, and 16, respectively. Each difference can be calculated, and then the minimal difference from Eq. (19) used as the difference coefficient.

The selection of α indicates the strength of an edge as it relates to all edges in an image. If α is selected to be a global constant, then the relative size of α will determine the global strength of edge reduction. A high value of α means that all local edges are equally valuable, so even spurious edges from random associations in noise will be preserved along with edges of interest. A low value, in contrast, will reduce 'important' edges while reducing those random noise associations as well. One method for selecting α then comes from basing the selection on global knowledge of edge strength as well as desired edge preservation. To do so, the standard heat equation as described in Eq. (7) can be run forward for an iteration in a particular direction, and then the absolute value of the difference between that image and the original will then provide some idea of edge strength. For instance, in the horizontal direction:

$$HorizontalResidual = |u_{i,j}^{0} - \frac{(u_{i,j}^{0} + u_{i+1,j}^{0})}{2}|.$$
(21)

In general, the lower the value of this residual, the more likely the edge is noise, and the higher the value of the residual, the more likely the edge is of interest. The subsequent values can be rescaled, using a given value of α , like so:

$$\alpha_{i,j} = \frac{(residual_{i,j} - min(residual))}{mean(residual)} * \alpha_{given}.$$
(22)

Practically, this adaptive α has to be further capped at -100, because a standard floating point number usually will not be able to describe the full dynamic range of Eq. 19. A visual description of the effect of using this approach is described in Fig 4.

The selection of the stopping criterion depends entirely the desired result. This equation still describes heat flow, and if run for an infinite amount of time, given that the coefficient equation (19) is non-inclusively bounded by zero, heat will flow across every edge and produce a homogeneous image. The sum of square differences with the original image can describe the extent of that heat flow; the ratio of that sum of the previous iteration divided by the sum of the current iteration provides the key for determining the stopping criterion. Initially, this ratio will be low, as each new iteration modifies the pixels drastically. After a few iterations, the ratio will approach 1, and the closer that ratio is to 1, the more piecewise smooth the image shall be. As described in Fig 5, a choice of 0.975 of the residual ratio preserves most smaller features while still clearing much of the noise, and a choice of 0.99975 produces a very smoothed image. For natural images with little to no texture and sharp edges, a value closer to 1 would be the prudent choice, while for images with quite a bit of local detail a smaller value would be more appropriate.

4 Comparison with Other Algorithms

This Semi-Local (SL) algorithm can best be evaluated when compared to other algorithms commonly used in noise reduction. Two popular candidates are the minimization of the Total Variation (TV)[1], and noise reduction via non-local means (NL)[7].

The first algorithm, TV noise reduction, minimizes the following equation:

$$\min_{u} \lambda \int_{\Omega} |\nabla u| + \frac{1}{2} ||f - u||^2$$
(23)

(where this λ bears no relation to the previously discussed term). This approach to noise reduction has been very well studied, and in this study, the Chambolle ([4]) implementation was used for the sake of comparison. In that particular implementation, the parameter of interest is σ , and should be proportional to the amount of noise in the image. In this case, an adaptive measurement of σ was adopted:

$$\sigma = \frac{\sum_{l=1}^{i} \sum_{m=1}^{j} u_{l,m} - 0.25 * (u_{l-1,m} + u_{l+1,m} + u_{l,m-1} + u_{l,m+1})}{i * j * 2}.$$
(24)

This adaptive σ will describe the noise in the image as the sum of local differences, normalized to the size of the image and a constant factor of 2. If that factor of 2 is increased, then the image remains particularly noisy, and if the factor is decreased, then the image is smoothed aggressively.

The second algorithm, NL means, first calculates the weights of each pixel relative to other pixels in a neighborhood about the current pixel:

$$NL(u)(x) = \frac{1}{C(x)} \int e^{-\frac{G_{\sigma} * |u(x+.)-u(y+.)|^2)(0)}{h^2}} u(y) dy$$
(25)

where G_a is a Gaussian kernel of standard deviation σ , h is a parameter, and $C(x) = \int e^{-\frac{G_{\sigma} * |u(x+.)-u(y+.)|^2)(0)}{h^2}} dz$ is a normalizing factor. The denoised value at pixel x is the mean of all the pixels whose Gaussian neighborhood is similar to x.

The images chosen for evaluation include a simple test case, as in Figure 6, the standard circuitboard image, as in Figure 7, an image of mosquitoes taken with an infrared camera at night, in Figure 8, and mammography data obtained from a Digital Radiography system (Hologic, Inc, MA) in Figures 9 and 10. These images were chosen as representatives of images which are normally considered for the introduction of a new algorithm and also because they will demonstrate the particular strengths and weaknesses of the different algorithms in different ways. The result of each algorithm is presented with the difference of the algorithm result with the original, noisy image. A perfect noise reduction should produce a picture of purely gaussian noise, without any local correlations, but also with no structure added to the noise reduced image. Note that the original was subtracted from the algorithmic result, so any black pixels in a difference image would be bright regions which were reduced to a lower intensity level, and vice versa for bright regions.

The 'R' test pattern, shown in 6, demonstrates the pecularities of the approaches which are inherent to each. The original image was made in Adobe Photoshop, and then imported to Matlab, where a Gaussian noise pattern with σ =12 was added to the image. The TV noise reduction, using the Chambolle algorithm with a σ defined as previously described, shows significant information is contained in the difference image, suggesting that edges have been smoothed. The SL algorithm ($\alpha_{given} = -7.5$, 0.99975 stopping) performed well on the image, with a slight pattern from the R appearing in the difference image. Finally, the NL means algorithm (7 as a searching radius, 3 as a matching radius, 2,000,000 filter strength) produced an image which is extremely similar to the original image, with very few, if any, patterns in the noise image. Interestingly, while the original presentation of the NL means algorithm suggests that the strength value should be chosen to be about 10 to 15 times larger than σ , in these experiments the strength was varied until a reasonable result was obtained. This image shows that the SL algorithm, for most of the standard images, will produce results in a quality which is roughly between the TV and NL algorithms.

The circuitboard image in Figure 7 presents a different challenge, that of small, linear structures which have been partially obscured by noise. Using the same technique as before, the image has had noise added to it chosen from a Gaussian distribution with σ = 12. This image contains straight lines, so an edge-preserving filter should handle it quite nicely. Additionally, several compression artifacts are present in the image, which could confound noise reducers into preserving these artifacts as well. Each algorithm had some divergence from the original image. The TV approach, using the same protocol as for the R image test, produced the image with the most patterning in the difference image, and most of the small wires and all of the compression artifacts have been smoothed away. The NL Means method, using 7 as a searching radius, 3 as a matching radius, and 40000 as a filter strength, still had some differences with the original image, but for the most part reproduced the original, noiseless image, including compression artifacts. Small round particles tended to get smoothed, and some edges tended to apparently not find similar neighbors in the searching neighborhood, meaning that some pattern in the difference image did appear. The SL algorithm, run with $\alpha_{given} = -7.5$ with a stopping point of 0.975, was somewhat between the two in terms of the final result. While more circuits were preserved than in TV, most of the jpeg compression artifacts were removed, unlike the NL means image. Additionally, small local clusters of noise remained, and several regular structures appeared in the difference image. Small rounded particles were not treated differently than any other regular structure, and so were not apparently smoothed to the extent of other features in the final image, unlike the NL result. The image of mosquitoes in Figure 8 represents a divergence away from the traditional images which are used for algorithmic validation. These images are JPG stills from a commercial infrared video camera used to image mosquitoes at night, with an infrared light providing the light source. The individual dots represent the mosquitoes, but significant structural noise and artifacts, some of it from compression and some from the low photon flux across the detector, has obscured the data from easy visualization. TV noise reduction via Chambolle, using the previously described protocol, produced a very jagged result, which is not much different from the original. The produced difference image does appear to be some, but not all, of the structural nonregularities contained in the image. The NL means noise reduction, operating at a search radius of 7, a matching radius of 3, and a strength of 25, produces a passable result, but along with distinction of mosquitoes in the background comes a very significant structured noise problem (one that was not fixed by changing algorithm strength by any amount). The final algorithm, the SL approach, also has a decent separation of the mosquitoes from the background. Whereas the artifacts in the NL means image tend to follow the structure of the image itself, the SL algorithm tends to reduce the image into patches, with the bright regions representing mosquitoes. For determining the most useful of these images, particle tracking will have to be performed, with the most reasonably tracked particles indicating the best noise reduction.

Both the fourth and fifth set of images were conducted on x-ray mammography data captured using a selenium detector. The presence of noise in these images is a result of lowered photon flux across the imaging detector [8], so increasing the flux should increase the amount of signal. However, the photon energy used in these imaging experiments can be dangerous to the patients, so the amount of flux in these images must be controlled. This flux is represented by two numbers, KVP and mAs. KVP stands for the kilovolts of potential between the anode and the cathode of the x-ray producing device; the higher the number, the more material the photons can penetrate without interacting with any matter. KVP values between 25-35 are typical for mammography, since a lower KVP will be absorbed by the skin, and a higher KVP will pass through the tissue to be imaged. The exposure time and the current strength are represented by the mAs, which is a multiplication of the microamperage of the current which produces the x-rays and the time in milliseconds of exposure to the patient. A smaller number represents a smaller photon flux, which means that a smaller radiation dose was given to the patient. In Figures 9 and 10, both a normal dose and a low dose image was taken of two different patients. Between images, the patient was repositioned, for the comfort of the patient; such repositioning will cause structures within the image to change position relative to one another, but such structures are still visible in both sets of images. The lower dose image was noise reduced, and the structural preservation was compared with that of the normal dose image.

The fourth set of images in Figure 9 show the true strength of the SL algorithm. Both TV and NL are built around the premise that the image must contain edges, and those edges must be preserved. In the case of the image of calcifications in Figure 9, the edge to be preserved is the edge of two calcifications, which indicates the possibility of the presence of cancer in a patient. The TV noise reduction shows, in the difference image, that significant changes were made to the calcification structures, with one calcification almost entirely removed. Furthermore, two radiologists, when asked to rate the images, dismissed the TV image as being too 'fuzzy', and further iterations only reduced the edge distinctions of those calcifications. The application of the NL means algorithm, using the same search radii but a strength of 1000, produced no real reduction of the calcification, but introduced a pattern to the image which actually suggested that the patients condition was worse than the original image, in that associating various pixels together suggested the presence of calcifications where none were present. Increasing the strength only produced stronger patterns. Only a single iteration of the SL approach using an α_{given} of -7.5 and stopping at 0.975 produced results which the radiologists thought were consistently better than both the other algorithmic approaches. The SL means algorithm preserved both calcifications, with minimal loss of other structural information.

The fifth set of images in Figure 10 continues to demonstrate the utility of the SL means algorithm on mammography data. As with Figure 9, the TV algorithm was overly aggressive in smoothing bright structures, and the NL Means algorithm introduced a great deal of structure. In this case, the introduction of structure can lead to a potential misdiagnosis; one indicator of cancer is the level and appearance of structure on the border of a mass **??**, and if such structure were introduced or modified like it has been by the NL Means algorithm, then the image loses its diagnostic ability. The SL means algorithm, by contrast, did not significantly reduce structure while at the same time returning the noise level to either equivalent or further reduced than the high-dose image.

5 Conclusion

A novel approach to anisotropic diffusion using the Poisson equation has been presented, as well as the particular image types where this algorithm does particularly well. Projectional radiography images of soft tissue are particularly well suited to this noise reduction technique, producing results which are more acceptable to radiologists who view them as well as remaining more clinically and diagnostically useful. Previous approaches of Total Variation Minimization and Non-Local Means noise reduction produce reasonable results on typical test data; however, when used on clinical data, neither approach preserved the same level of diagnostic utility. In normal test data, the SL algorithm produced results which appeared to be closer to the desired image than TV minimization, but not as pattern-free in the noise reduction as the NL means algorithm. When compared during the task of mosquito finding, either SL or NL algorithms may be producing a reasonable result, and only further verification by particle tracking can ascertain which produces a more reasonable path.

Given these results, this algorithm appears to be best suited to the particular task of noise reduction in mammography. This success is likely due to the image formation process of mammography images. Unlike 'natural' images, mammography images are not composed of sharp edges. The image is formed through projecting x-rays through the target tissue. Objects closer to the site of x-ray beam production appear to be larger in the resulting image, while objects further from the beam production appear to be smaller. Additionally, as the image depicts x-rays which project through the tissue rather than are absorbed, x-rays which interact with the tissue and are deflected rather than absorbed will contribute to an overall blurring of the final image. Finally, the structures themselves are not sharp edged, but composed of soft tissue. These factors combine to make an image where noise is combined with a smooth underlying structure [8]. Algorithms which depend on the preservation of such sharp edges are not suited to this purpose, as the underlying information does not contain such edges.

Total variation approaches are well suited for images containing sharp edges, but does not work well for images with these soft structures. Modifications to the parameters of the algorithm, as well as implementing the original ROF model [1], yield results which radiologists found to be too blocky or to be extremely similar to the original noisy image. The Total Variation model assumes that the differences in images should be sharp boundaries, rather than smooth transitions, and that initial assumption makes the algorithm potentially unsuitable for this imaging modality.

The non-local means noise reduction algorithm relaxes this shock-based approach to noise reduction to rely on local patterns in the image. Pixels with similar underlying structure near to one another all contribute to reducing the noise in that area. If an image has large amounts of structural noise, as with the mosquito images, then that structure can interfere with the reduction of noise. If the image has very little structure, as with the mammography images, then the algorithm appears to introduce structure. This structural introduction makes this noise reduction approach inappropriate for mammography images.

Given the limitations of these two approaches, the semi-local line integrals approach provides a useful alternative. While the algorithm is based on local characteristics of the image, it is not beholden to local patterns to determine its noise reducing capabilities; rather, the choice of α_{given} and the stopping criteria will each contribute, and so does not have the same limitations as the NL means algorithm. This same choice makes the algorithm less suitable for the natural images where NL means excels; those images can have both sharp edges and local patterns, neither of which the SL algorithm handles as well. On the other hand, the SL algorithm, as it is a modification of the Poisson equation, will handle smooth transitions well in an image, making it ideally suitable for images with smooth underlying information, such as those in mammography.

References

- [1] Rudin LI, Osher S, Fatemi E, Nonlinear total variation based noise removal algorithms, Physica D, 60:259-268, 1992.
- [2] Brandt A, Dym J, Fast Calculation of Multiple Line Integrals SIAM J Sci Comput., Vol 20, No. 4, pp. 1417-1429.
- [3] Gonzales R, Woods R, Digital Image Processing, Prentice Hall, 2002.
- [4] A. Chambolle, Journal of Mathematical Imaging and Vision 20 (1-2): 89-97, January March, 2004.
- [5] Osher S, Fedkiw R, Level Set Methods and Dynamic Implicit Surfaces, Springer-Verlag, 2003.
- [6] Strikwerda, J, Finite Difference Schemes and Partial Differential Equations Wadsworth, 1989.
- [7] A. Buades, B. Coll, J.M Morel, "A review of image denoising algorithms, with a new one", Multiscale Modeling and Simulation (SIAM interdisciplinary journal), Vol 4 (2), pp: 490-530, 2005.
- [8] Jerrold T. Bushberg, J. Anthony Seibert, Edwin M. Leidholdt Jr., and John M. Boone *The Essential Physics of Medical Imaging* Williams & Wilkins, 2001
- [9] American College of Radiology (ACR) Breast Imaging Reporting and Data System Atlas (BI-RADS Atlas). Reston, VA, American College of Radiology, 2003.

6 Acknowledgements

I would like to thank Dr. Achi Brandt for many discussions on both the concepts and technical details of the approach. In preparing this manuscript for this publication, Dr. Luminita Vese has been invaluable, providing insight into and helping to correct the flaws in the text. Dr. Lawrence Bassett helped to explain the needs of radiologists and also the particulars of mammographic imaging. Mosquito images are courtesy of Dr. Nick Manoukis of the NIH, and the mammography images are courtesy of Dr. Andrew Smith of Hologic, Inc (Bedford, MA).



Figure 1: Semi-Local Edge Justification: In the noise-free image of a gradient, fig 1(a), three different pixels help to describe three different severity of edges. The A-B pairing is clearly an edge, as can be seen by looking at the edge between the two pixels as well as the surrounding pixels. The C-D pairing is also an edge, but weaker than the A-B coupling, both when comparing the two pixels directly and when using the neighboring pixels as well. Pixels above the C-D pairing are clearly an edge, while pixels below the pair are very close in intensity, so the C-D pair may represent an edge, but one of lesser intensity than the A-B pairing. The E-F pairing is clearly not an edge, either between the pixels directly or by comparing neighboring pixels. Fig 1(b) shows the same image, but with gaussian noise added from a distribution with $\sigma = 10$. The A-B coupling remains a strong edge, but the C-D pair appears to be an even stronger edge. The E-F pair is not as strong an edge as the C-D pairing. When reducing noise, the E-F pairing should be smoothed, the C-D pairing less so, and the A-B pairing only minimally.



Figure 2: Description of the construction of line integrals. During the first iteration, basic lines are constructed by averaging neighboring pixels together, or in the case of noise integrals, the absolute value of the differences are determined. During the second and subsequent iterations, longer integrals are constructed through combination or concatination with shorter integrals from the first iteration. In this case, A and B are averaged, and then concatinated with C and D. This process repeats for each of the lines in the new set; lines on a diagonal or on the horizontal are simply concatinated, and then lines between those are combined and then concatinated. For the third iteration, A and B are combined and concatinated with the combination of C and D, repeating the process from the second iteration. The process continues until lines are considered long enough. The longer the lines, the smaller the angle that can be covered, but the larger the boundary conditions near the edge of the image.



Figure 3: Selection of the set of line integrals used for coefficient determination. This pixel grid shows 8 lines of length 4 which all go through a central point. Of these lines, E, F, G, H, and A are constructed as described in Fig. 2, and A, B, C, D, and E are constructed through the same method, rotated by 90 degrees. Lines A and E are redundantly created in both directions as a consequence of the described process. Of these lines, F, G, and H will provide the set for vertical point comparison, as shown in subfigure 3(b)-the maximal difference, as defined by Eq. 19, of the horizontal lines. That is, if the central point in this figure is $u_{l,m}$, then the difference a_1 between $u_{l,m+1}$ and $u_{l,m}$ will be the maximal difference of the lines B, C, and D. Similarly, B, C, and D will provide the set for horizontal point comparison, the *b* coefficients. Diagonal coefficients will use a similar set of three lines, with A, B, and H providing the set for differences between $u_{l,m}$ and $u_{l-1,m+1}$ and $u_{l+1,m-1}$. Lines D, E, and F provide the set for the opposite diagonal direction, between $u_{l,m}$ and $u_{l-1,m-1}$.



Figure 4: Explanation of selection of α . The original image, 4(a), has had noise added to it from a Gaussian distribution with a value of $\sigma = 25$, as shown in 4(b). When reducing noise using a value of $\alpha = 1$ over the whole image, after 100 iterations, the image has been greatly blurred. Some details remain, but the result is too blurry. If $\alpha = 5$, after 100 iterations, obvious edges are preserved, but so are spurious edges within the noisy domains of the image. Using the adaptive scheme described by Eq. 22 with $\alpha_{given} = 5$ and running the noise reduction for 100 iterations, the resulting image has obvious edges preserved and spurious noise edges smoothed.



Residual Ratio Convergence Behavior

(f) Noisy Lena, $\sigma = 12$

(i) Lena, 0.99975 Stopping Ratio

Figure 5: Explanation of selection of stopping criterion selection. The graph in 5(a) describes the general convergence behavior of the algorithm, quickly and then asymptotically approaching 1. For piecewise smooth images which have had noise added to them, such as the R test image in 5(b), continuing until the residual is very close to 1, as in 5(e), smoothes out local discrepancies due to noise while still preserving edges; the R can be even less blurred with a choice of higher α_{qiven} . For non-piecewise-smooth images, such as Lena (here with a noise from a Gaussian distribution with $\sigma = 12$ added), choice of a less aggressive stopping criterion will more readily preseve features at the expense of noise, but choosing a more aggressive stopping criterion of 0.99975, as in 5(i), produces an extremely blurry image. Such an image might be useful as a segmentation, but is too aggressive for noise reduction. For all noise reduced images, $\alpha_{qiven} = 5$.

⁽g) Lena, 0.75 Stopping Ratio

⁽h) Lena, 0.975 Stopping Ratio



(e) NL Means Denoised

(f) NL Means Denoised Difference

(g) SemiLocal Denoised

(h) SemiLocal Denoised Difference

Figure 6: The R Test Image in 6(a), which tests the ability to preserve the square around the R, the curvature of the R, and the distinction between the different edges of the alternating lines on the left and left top portions of the image. Noise from a Gaussian distribution with σ = 25 was added to produce the image in 6(b)The TV denoised image in 6(c), after 50 iterations of the Chambolle algorithm, shows a bit of haziness over the image, and the difference image in 6(d)suggests that quite a bit of the edge information has been lost. The SL means algorithm, using an α_{given} = 7.5 and a stopping criterion of 0.99975, produced the image seen in 6(g), with a well-preserved square and stripes but a slightly degraded R, further shown in the difference image with the original in 6(h). The NL Means algorithm provides an image in 6(g) which most closely resembles the original image, with a difference image in 6(h) which is almost entirely devoid of pattern. This trend of NL means outperforming both TV and SL, and SL outperforming TV, continues through most other similar test images; only when applied to particularly tricky data, as in Figures 8 and 9, does the need for the SL algorithm become more apparent.



(e) NL Means Denoised

(f) NL Means Denoised Difference

(g) SemiLocal Denoised

(h) SemiLocal Denoised Difference

Figure 7: The standard circuit board image, with different noise reduction algorithms applied. In panel 7(a), the standard circuitboard image is presented, and in panel 7(b), the circuit board has noise added from a gaussian distribution with $\sigma = 12$. The results of total variation denoising, using the Chambolle implementation, are presented in 7(c), with the difference between the noisy image and the denoised image in panel 7(d). Note that this noise reduction method removes quite a bit of edge information. The non-local means noise reduction approach, as shown in 7(e), presents a much better result, with noise reductions generally not following a particular pattern. However, smaller objects can have appearances in the difference image in 7(f), such as the small circular objects in the circuitry. The results of the SL anisotropic noise reduction using $\alpha_{given} = 7.5$ and stopping at 0.975 are in 7(g), with the difference image in 7(h) showing some circuit patterning, but not as severely as the TV reduction. Closer examination will show, for instance, that figures 7(e) and 7(f) have separation of various small wire patterns, and that the small connections in the chip socket at the lower right are preserved by these two algorighms.



(a) Original Mosquitoes



(b) TV Denoised Mosquitoes

(c) TV Denoised Mosquitoes Difference





(f) SemiLocal Denoised

(g) SemiLocal Denoised Difference

Figure 8: Mosquitoes images from a commercial camera for the purpose of determining swarming patterns. The noise (panel 8(a) in this image is very structured, due to jpg artifacts and very low light imaging. For the purposes of finding mosquitoes, the TV denoising (panels 8(b) and 8(c)) is not aggressive enough to remove the noise patterns. The NL means (panels 8(d) and 8(e))and SL methods (panels 8(f) and 8(g)) are roughly equivalent. The NL means algorithm produces many small structures throughout the image which are probably the result of the jpg artifacts themselves and could interfere with object finding routines. The SL algorithm produces a similar image, but without the structured noise pattern, and so might be better suited to find the mosquitoes. Due to the presence of the structured noise, α_{given} was selected to be 2.0, which would produce a more aggressive noise reduction.



(a) Calcifications at high dose

(b) Calcifications at low dose



(e) NL Means Denoised (f) NL Means Denoised Difference

(g) SemiLocal Denoised

(h) SemiLocal Denoised

Figure 9: Calcification recovery after noise reduction. The location of two calcifications are indicated in all panels by the black arrow. Panel 9(a) contains a high dose image (KVP = 29, mAs = 106000). The patient was then repositioned to take a low dose image, shown in panel 9(b) (KVP = 31, mAs=68000). This image contains the same calcifications somewhat obscured by noise, and was the image which was processed by the different algorithms. Panel 9(c) contains the TV denoised reduced result, with the difference in panel 9(d) showing that both calcifications have been severely reduced. Panel 9(e) shows the NL Means result, with the difference in panel 9(f). The calcifications have not been reduced, but the added patterns in the image have reduced the contrast in the region. Panel 9(g) show the SL result ($\alpha_{given} = 2$, stop = 0.975), with the difference in Panel 9(h) showing that the calcifications have been preserved, and contrast enhanced.



(a) Tumor at full dose

(b) Tumor at reduced dose



(f) TV Denoised Difference

(g) NL Means Denoised Difference

(h) SemiLocal Denoised

Figure 10: Noise reduction on a large calcified structure. The image in panel 10(a) contains a large calcified structure imaged at the full recommended dose (KVP = 31, mAs = 96000). The patient was then repositioned to take a low dose image, shown in panel 10(b) (KVP = 32, mAs = 74100). This image contains the same structures as the image in panel 10(a), but patient repositioning has moved the structures relative to one another. This second image is the image which was noise reduced. The TV result is shown in panel 10(c), with the difference image shown in panel 10(f). Quite a bit of the high-intensity structure has been blurred in this image, as can be seen in the difference image; this appearance leads to radiologists preferring other image processing algorithms. The NL Means result is shown in panel 10(e), and the difference image is shown in panel 10(g). Similar to the result in 9(e), no structure has been removed, but structure has appeared in the image, which both lowers the contrast and provides the possibility of introduced artifact causing misdiagnosis. The SL noise reduction algorithm image ($\alpha_{given} = 2$, stop = 0.975) is shown in panel 10(e), and the difference image is in the difference image, more of the original character of the original image is preserved. Additionally, no added structure has appeared, making this algorithm preferred of the three.