UCRL-CONF-232852

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Local Area Signal-to-Noise Ratio (LASNR) algorithm for Image Segmentation

Laura Kegelmeyer, Philip Fong, Steven Glenn, Judy Liebman

July 19, 2007

SPIE Symposium on Optical Engineering & Applications - Applications of Digital Image Processing
San Diego, CA, United States
August 26, 2007 through August 30, 2007

Disclaimer

Local Area Signal-to-Noise Ratio (LASNR) algorithm for Image Segmentation[Ψ]

Laura Mascio Kegelmeyer[*a], Philip W. Fong[a] , Steven M. Glenn[a], Judy A. Liebman[a]
[a]Lawrence Livermore National Laboratory, PO Box 808, L-592, Livermore, CA USA 94551

ABSTRACT

Many automated image-based applications have need of finding small spots in a variably noisy image. For humans, it is relatively easy to distinguish objects from local surroundings no matter what else may be in the image. We attempt to capture this distinguishing capability computationally by calculating a measurement that estimates the strength of signal within an object versus the noise in its local neighborhood. First, we hypothesize various sizes for the object and corresponding background areas. Then, we compute the Local Area Signal to Noise Ratio (LASNR) at every pixel in the image, resulting in a new image with LASNR values for each pixel. All pixels exceeding a pre-selected LASNR value become seed pixels, or initiation points, and are grown to include the full area extent of the object. Since growing the seed is a separate operation from finding the seed, each object can be any size and shape. Thus, the overall process is a 2-stage segmentation method that first finds object seeds and then grows them to find the full extent of the object.

This algorithm was designed, optimized and is in daily use for the accurate and rapid inspection of optics from a large laser system (National Ignition Facility (NIF), Lawrence Livermore National Laboratory, Livermore, CA), which includes images with background noise, ghost reflections, different illumination and other sources of variation.

1. INTRODUCTION

A Local Area Signal-to-Noise Ratio (LASNR) segmentation algorithm was developed to attempt what the human eye does readily: select object pixels that stand out from the local surroundings, independent of what content may be present elsewhere in an image. (Figure 1 indicates such an object of interest in a typical side-illuminated image with complicating reflections.) This segmentation is a first step in our analysis pipeline. It is important to detect all possible candidate objects at this step, so we can apply additional measurements and information later and pare down to the most likely candidates when all of the discriminating information is available.

Segmentation methods are typically strongly application-specific and even within the intended application, still fragile. The LASNR segmentation algorithm has been remarkable in that it has been successfully used for more than five years to inspect a variety of image types for over a dozen optics inspection systems for the National Ignition Facility at Lawrence Livermore National Laboratory.

[Ψ] This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48. UCRL-CONF-232852.

[*] kegelmeyer1@llnl.gov; phone 925-422-0924; fax 925-423-6692

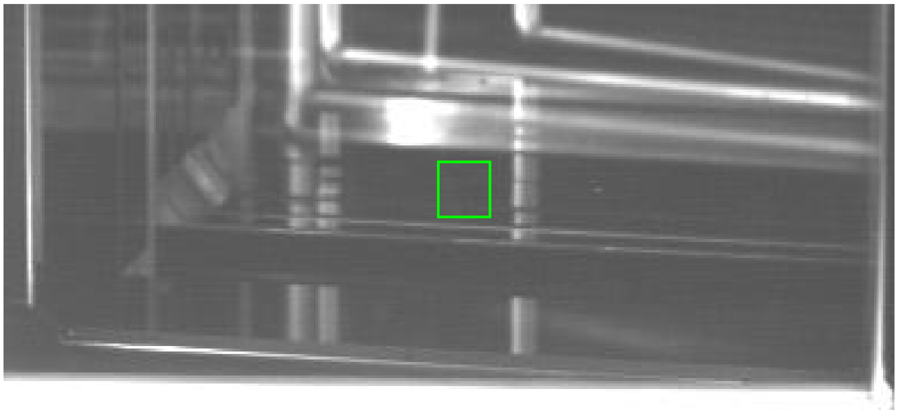

Figure 1: A portion of one type of inspection image (side-illuminated), indicating a small flaw in a noisy background containing hardware reflections.

The key concept behind the LASNR algorithm is to calculate the signal-to-noise ratio at each pixel, with respect to its immediate surroundings. It must calculate the magnitude of the signal among pixels that belong to the object and calculate noise among the pixels belong to the surroundings.

Since we don't know the locations of the objects, a priori, we hypothesize both the size (usually a range of sizes, based on the application) and the location (centered at every pixel in the image). For each object size hypothesis, we pre-define a corresponding background area for computing the surrounding noise. The signal-to-noise ratio is computed as if an object of a certain size is centered, sequentially, over each pixel in the image, resulting in a new image of equal size that contains the local area signal-to-noise ratio value at each pixel. This may be repeated for different size hypotheses. Applying pre-selected signal-to-noise cutoffs to the resulting images at each scale yields at least one seed (member) pixel within each object. The results from different scales are combined and the seed pixels indicate the approximate location of objects of interest. These serve as initiation points from which the object can be grown (expanded) to its full extent. The overall LASNR segmentation algorithm is thus a 2-stage process that first finds object seeds and then grows them to find the full extent of the object, effectively separating the detection and area estimation steps.

The original concept was developed and implemented using simple square neighborhoods for the signal and noise calculations. The size of the neighborhoods was determined empirically depending on how noisy the image is and how close other objects may be.

Years later, when converting the code to MATLAB (The Mathworks, Natick, MA), the implementation was changed to take advantage of the speed with which MATLAB does full-image operations. The size and shape of the hypothesized objects and the corresponding background neighborhoods were modified by Gaussian filters, as described in detail in section 2.3 below.

2. Methods

2.1 Background subtraction

For both of the seed detection methods described below (sections 2.2 and 2.3), the necessary first step is to subtract the large-trend intensity information from the image. An effective and efficient method is morphological "lower envelope" subtraction as described by Verbeek, et al.[1] and implemented with DIPimage.[2] In the original seed detection (section 2.2), we use variable sizes for the local neighborhood area, depending on the size of the noise window to be computed. In a newer implementation for the seed detection, the size of the neighborhood is held constant, much larger than the biggest object expected, and a more local background subtraction is built-in to the local signal-to-noise calculation used there.

2.2 Original method illustrating the LASNR concept

The concept underlying the LASNR is well illustrated with its original implementation, first written in C with SciIImage.[3] It computed one value for the local signal and another for the local noise at each pixel in a background-subtracted image (described in section 2.1) and assigned the ratio of signal/noise for that pixel in the resulting image. This produced a new image where every pixel's value was equal to its signal-to-noise ratio as computed from the original, background-subtracted intensities of its local neighborhood,

$$LASNR_{orig} = \frac{\mu_{sig}}{SD_{bg}}, \tag{1}$$

where μ_{sig} is mean of the background-subtracted intensity in the (smaller) signal window and SD_{bg} is the standard deviation, or noise, contained by the background subtracted intensities in the (larger) background window. (The background window includes the pixels in the signal window.)

In the background-subtracted image, the signal was determined by considering a certain number of pixels surrounding the pixel of interest and averaging their intensities. The mean of intensity of a small "neighborhood" or "window" around the pixel of interest became the signal value for that pixel. The standard deviation in a larger neighborhood around the pixel of interest was used to define the noise.

The expected object size and expected noise profiles for the inspection system were used to select the size (and shape) of the signal window and noise window. Windows of different sizes were combined to achieve results at different scales, so a range from large to small objects could be found.

Applying a constant threshold to this resulting signal-to-noise ratio image allowed each pixel to be selected based on its relative elevation over local surroundings, instead of being influenced by absolute intensities throughout the entire original image. (See Figure 1, Figure 6, and Figure 7 for an illustration of this result.)

2.3 Adaptation of original method for efficient implementation

As SciIImage became unsupported, we migrated the LASNR algorithm to a new platform: MATLAB (The Mathworks, Natick, MA), with DIPimage.[2] To avoid the pointer arithmetic that was efficient when written in C (with SciIImage), and to take advantage of the speed of the Matlab/DIPimage full-image operations and linear filters, the first steps of the LASNR algorithm were re-implemented. Instead of computing the signal-to-noise ratio for square neighborhoods around every pixel in the image, we used the speed of DIPimage's cache-optimized implementation of a separable Gaussian filter[4] to compute an entire signal image and also an entire noise image and then divided those two to yield a new image with a signal-to-noise ratio at each pixel as explained below.

The 2-dimensional Gaussian kernel is defined by

$$g_{2D}(x,y) = \left(\frac{1}{\sqrt{2\pi\sigma}}\right) e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} \tag{2}$$

and the operation of convolving this with an image will hereby be represented by the DIPimage function name, *gaussf*, with its input parameters being the image on which to operate and the numerical value selected for σ:

$$gaussf(I,\ \sigma). \tag{3}$$

The signal image results from a peak-enhancement step whereby the background-subtracted image I_{bs} is further sharpened by subtracting a very locally smoothed version of itself. This form of high-pass filter is designed to punctuate any peak that stands relatively higher than other, noisy peaks in the surrounding area (and thus meets our criterion for conservatively finding all possible candidate objects in this first segmentation stage). The Gaussian weighting provides an efficient approximation of the numerator in Equation (1).

The signal image is defined as:

$$I_{sig} = I_{bs} - gaussf\,(I_{bs}, \sigma_1) \qquad (4)$$

where σ_1 is smaller than the smallest object of interest. In our case it is a fraction of a pixel, so we may enhance and thus detect objects that are as small as 1 pixel in size. This approximates the original LASNR implementation by computing a Gaussian weighted mean, instead of the original unweighted mean around the background-subtracted signal value.

The noise image calculation approximates the original implementation (standard deviation of a local neighborhood) by letting the values in the background-subtracted image (with approximately 0 mean) represent deviation from the mean and by using a Gaussian filter to weight pixels closer to the object more heavily than pixels further from the object:

$$I_{noise} = \sqrt{gaussf\,(I_{bs}^2, \sigma_2)} \qquad (5)$$

where the σ_2 parameter is pre-selected according to the expected size of the noise neighborhood, given a priori knowledge of the image and given that the foreground objects will be approximately 1 to 4 pixels in size. (Larger foreground objects are found with the multi-scale approach described in section 2.4 and shown in Figure 5.)

Now we can divide the signal and noise images to get the resulting LASNR image. For computational efficiency, we avoid the square root operation by squaring the ratio. For the numerator, we square the pixel values in the signal image, but keep the original sign of the peak as shown here:

$$LASNR^2 = \frac{I_{sig} * \left|I_{sig}\right|}{I_{noise}^2} \qquad (6)$$

Using this squared version requires squaring the cutoff value to which it is compared.

$$LASNR^2 > cutoff^2 \qquad (7)$$

This results in a binary segmentation result where every pixel that meets the cutoff criterion gets a value of 1 and the rest are 0. For every cluster of touching “1” pixels, the pixel location with maximum intensity is stored as a seed for the object.

To increase the chances of finding a tiny object with intensity split between two pixels, we smooth the background-subtracted image, I_{bs}, with a 2x2 boxcar filter, scale σ_2 by multiplying it by $\sqrt{2}$ and then repeat the LASNR calculation and combine the seed result with previous seeds gathered so far.

2.4 Applying the LASNR for multi-scale detections

Because the LASNR, as described above, is designed to find small objects about 1-4 pixels in size, we repeatedly reduce the size of the original image in order to find larger objects (see Figure 2). To accommodate possible aliasing effects, we apply the 2x2 boxcar filter before decimating the background-subtracted image by 2 in each dimension. We then scale σ_2 accordingly and repeat the analysis at remaining scales.

The following sequence summarizes the entire seed accumulation process for objects of varying sizes:

a) Accumulate seed locations with the LASNR equation.
b) Blur I_{bs} with a 2x2 boxcar filter (and use $\sqrt{2}\sigma_2$) to find object seeds that may be split among pixels.
c) Combine these seed locations with previous results
d) Decimate blurred image, scale σ_2 and repeat steps a-d until all relevant scales are completed.

Accomplishing these steps requires some empirical knowledge of the nature of the objects and their background noise, in order to determine the LASNR cutoffs as well as σ_1 and σ_2. The result is a binary image with a single seed pixel for every object detected.

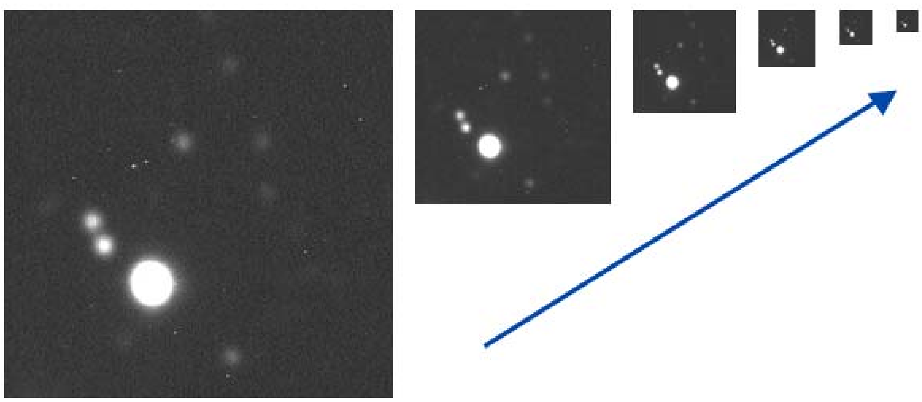

Figure 2: An image of astronomical bodies, shown here, contains objects in a range of sizes in a compact space so as to demonstrate how repeatedly reducing the size of the image allows larger objects to be detected in the same way as smaller ones.

2.5 Adaptive filling from seeds

The final step of the LASNR image segmentation method is to determine the extent of the objects, by starting with the accumulated seeds and growing them. When an object is superimposed on a background feature or noise with elevated intensity, growing the segmented regions to include all connected pixels greater than a pre-selected fraction of the seed pixel can falsely label the background feature as part of the object, as shown on the right side of Figure 3. Instead, an adaptive threshold is determined during the filling process, as described below.

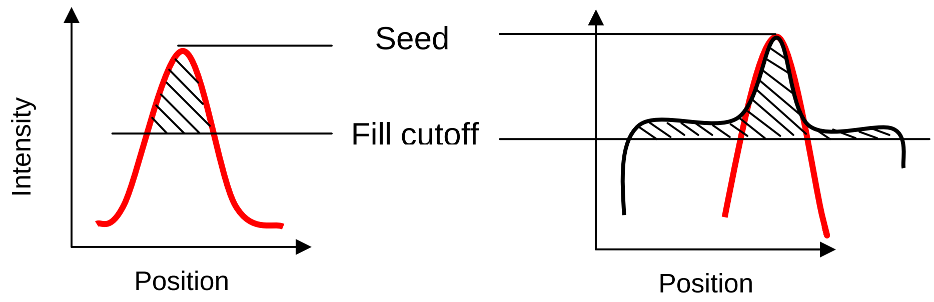

Figure 3: Cross-section of isolated peak (left) and cross-section of peak superimposed on a background feature (right), both shown with cross-hatching to indicate the total area included when a fixed cutoff percentage such as half-max is used to define the extent of the object.

For a peak that is not over a background feature, the area of the object increases smoothly as the cutoff threshold is decreased (Figure 3 left). However, for a peak that overlays a background feature, the area will sharply increase when the threshold drops below the level of the background feature (Figure 3 right).

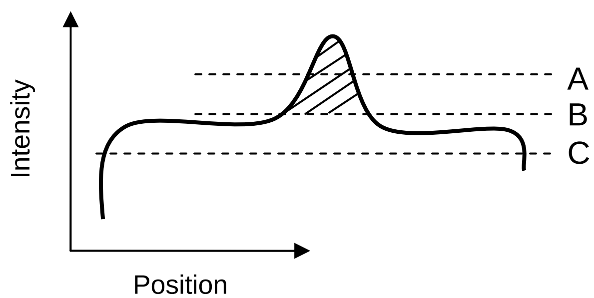

Figure 4: Adaptive filling pixel windows

For each seed pixel, the filling process starts at the seed pixel. Neighboring pixels are added in order of decreasing pixel intensity; the pixel with the largest intensity bordering the region already segmented is added. As the pixels are added, the number of pixels added is tracked over a sliding window of pixel intensity. (Note that the sliding window is with respect to intensity and not position. The window is defined by a fraction of the seed pixel intensity. It includes all of the pixels filled so far that are between the lowest intensity so far and the window fraction above that. These are usually at the perimeter but the width of the "perimeter" will vary and it can include pixels in the middle of the object.) The process stops when the ratio of pixels added in the current window (between B and C in Figure 4) to the number of pixels in the previous window (between A and B in Figure 4) exceeds a threshold or a minimum fraction of seed pixel intensity is reached. It is efficiently implemented using a priority queue to select which pixel to add.

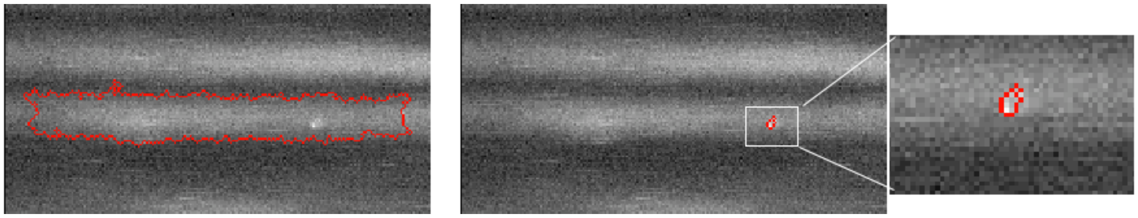

Figure 5: A portion of an inspection image of a NIF debris shield shows a signal that is in a noisy, high intensity background streak. The red-highlighted areas demonstrate the results of (left) filling to a fixed percentage of the peak intensity and (middle) the adaptive filling method. The object of interest is shown magnified (right).

3. DISCUSSION

The general strategy of image segmentation for NIF Optics Inspection is to find all candidate objects and put them in a database. Once they are stored, a host of measurements and meta-information can be applied to winnow down the set of all possible candidates to only the most interesting. It is important that we not discard any candidate objects in the first step. In this way, all relevant information (measurements and meta information) can be applied when it becomes available and before a decision is made for an object.

This means that we must choose our LASNR cutoff and other parameter values appropriately, using all domain knowledge and specific knowledge of the source of the image. For images acquired under tightly controlled conditions, we know that there should be very little noise and therefore desired objects will have large LASNR values. For these, we can choose high cutoff values (e.g. we use a cutoff value of 14 for low-noise systems). In other circumstances, when images are acquired in noisier environments, we use lower cutoff values and put emphasis on the scales of most importance (e.g. we may want to find all large defects but not necessarily tiny ones so we can use lower cutoffs at the coarsest scales, and higher cutoffs at finer scales).

The LASNR segmentation method has no theoretical limitations in object size, but some limitations can be imposed by the selection of algorithm parameters for size and signal-to-noise cutoff values. There are some limitations when dealing with objects of different size that are in close proximity. In that case, these objects lie within each other's noise neighborhood. The larger, brighter object will be a significant deviation in the background of the small object, increasing its local noise. The smaller object will be a less significant source of noise to the bigger, brighter object. The bigger, brighter object is more likely to meet the signal-to-noise ratio cutoff for seed accumulation.

4. CONCLUSION

The LASNR segmentation method has been in daily use for NIF Optics Inspection for several years now, with the two implementations described and has performed well. In meeting with our criterion, it has shown to be much more likely to detect false alarms rather than miss a real object; we apply post-processing to mitigate false alarms. It has been used for inspection systems with edge-injected lighting (Figure 1) as well as laser back-lighting; for images that have a bright field (and dark objects), we invert the image before beginning the LASNR segmentation (Figure 6).

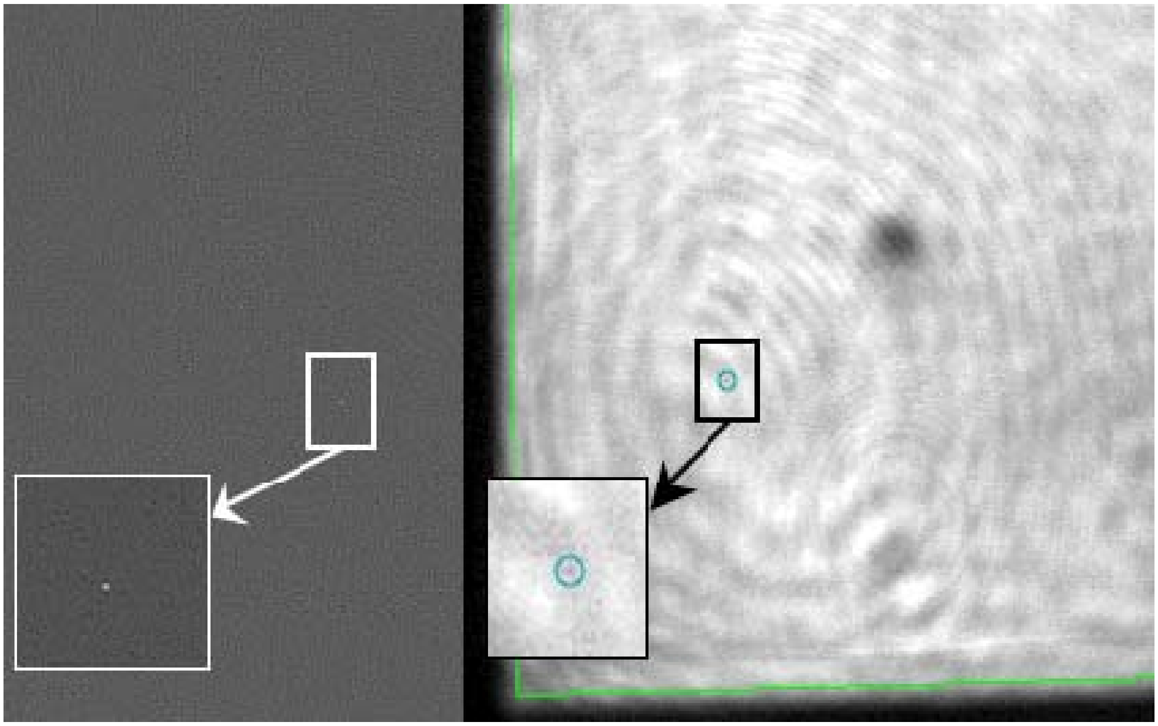

Figure 6: A NIF inspection image illuminated with laser backlighting (right) generates a varying background with diffraction patterns. We invert bright field images before calculating the LASNR image (left) and then proceed with the adaptive filling.

We also applied it to an astronomical image to better demonstrate the multi-scale detection in a single image (Figure 7).

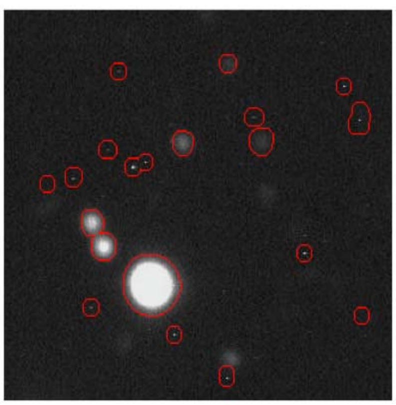

Figure 7: Astronomical bodies, segmented with the LASRN segmentation algorithm. Red highlighting shows a slightly dilated version of the areas resulting from LASNR segmentation.

We required spot detection that is highly sensitive and robust. Through adaptive definition of the extent of each object and careful engineering, the LASNR segmentation algorithm has been very successful in its role for the continual operation of the NIF laser.

Acknowledgments

The authors wish to acknowledge and thank Marijn Bezuijen for the first translation of the original seed detection method during his 2004 internship (from the Delft University of Technology, Delft, The Netherlands) at Lawrence Livermore National Laboratory.

References

1. P.W. Verbeek et al., “Low-level image processing by max- min filters,” Signal Processing 15(3), pp. 249-258, 1988.

2. C.L. Luengo Hendriks, L.J. van Vliet, B. Rieger, M. van Ginkel and R. Ligteringen, DIPimage, software package, Quantitative Imaging Group, Delft University of Technology, 1999-2007. http://www.qi.tnw.tudelft.nl/DIPlib

3. R. Van Balen, D. Koelma, T. K. Ten Kate, B. Mosterd, and A. W. M. Smeulders, “ScilImage: a multi-layered environment for use and development of image processing software.” *Experimental environments for computer vision & image processing,* H. I. Christensen and J. L. Crowley, editors, , pages 107-126. World Scientific Publishing Co. Pte. Ltd., 1994.

4. K. Schutte and G.M.P. van Kempen, “Optimal cache usage for separable image processing algorithms on general purpose workstations,” Signal Processing, **59**, no. 1, 113-122, 1997.