Median Filtering for Removal of Low-Frequency Background Drift

Alvin W. Moore, Jr.,* and James W. Jorgenson

CB 3290, Department of Chemistry, University of North Carolina—Chapel Hill, Chapel Hill, North Carolina 27599

INTRODUCTION

We have recently discovered a very simple method of removing low-frequency baseline drift in chromatographic data. The method involves digitally filtering the chromatographic data using a moving median filter. This is a nonlinear filter which gives results quite different from the more common moving average filter. The moving median filter removes impulse characteristics (such as noise spikes) in a signal but preserves large sudden changes of level (i.e. edges) such as baseline shifts and low-frequency changes such as baseline drift.

Median filtering was suggested as a tool for data analyses by Tukey in 1971 and later came to be used in image processing.[1] Typical uses of the moving median filter include removal of noise in scanned images,[2,3] removal of cosmic ray spikes from image data collected with charge-coupled device (CCD) detectors,[4] and cleaning pitches from noise in speech processing.[5,6]

We are using the moving median filter in a manner parallel to an image-processing technique called object extraction.[1] In this case, the median filter is used to help distinguish objects in an image with a varying background. For our purposes, the objects of interest are the chromatographic peaks, while the varying background is the undesirable baseline drift. We first median filter the chromatogram, actually removing the peaks of interest and leaving only the baseline drift. We then subtract this filtered data from the original raw data to give the difference data. The difference data reproduces our original peaks of interest, but now on a flat baseline, with little or no distortion of the original peak shape.

In this note we will explain the nonlinear moving median filtering method and its application in removal of background drift. We will also identify some of the factors controlling its effective use.

EXPERIMENTAL SECTION

All data processing was done on a Macintosh II personal computer, running LabVIEW 2 software (National Instruments Corp., Austin, TX). LabVIEW 2 is a high-level scientific programming language for the Macintosh. It has a graphical user interface and comes with a number of preprogrammed ("canned") routines for digital signal processing. All of the digital filtering described here is done with the prewritten LabVIEW 2 routines. However, implementation of the moving median filter in other programming languages should not be difficult (see mathematical description of filter, below).

Data taken from the literature were scanned into the computer as a bitmap using a Microtek MSF 300GS image scanner and then digitized into numeric data using Flexitrace software (Tree Star, Inc., Campbell, CA).

FILTERING

In image processing, the moving median is used to suppress impulse noise. Here, impulse refers to locally large positive or negative values of short duration. Moving median filters suppress such noise provided the filtering window is at least twice the width of the impulses. Then impulses which are sufficiently separated will be completely deleted by the filter. Impulses lying close to each other may remain, though reduced in intensity. In our case, the "impulses" are not noise but are the chromatographic peaks, actually the part of the signal we wish to keep. The filter rank value controls the width of the window and determines whether peaks will pass through or be eliminated.

The moving median filter processes the input data such that each point in the output data is the median of a subset (window) of points centered on the corresponding point in the input data. The median is the middle value of the subset, after sorting it from high to low values. Chromatographic peaks in the data will always be sorted toward the high end of the subset, while baseline values in the data will always be sorted toward the low end. Thus peaks which span less than half the points in the subset never reach the middle (median) value and are effectively removed from the output data.

Mathematically, the median filter can be explained as follows. The median filter routine accepts as inputs the data array to be filtered, X, of size n, and a value for the filter rank, r, where $n > r \geq 0$. The output of the median filter routine is a filtered data array, Y, of the same dimensions as X. Each point in the output array is the median of a subset of $2r + 1$ points centered on the corresponding point in the input array, that is

X = input array, data to be filtered

Y = output array, filtered data

n = size of input array

r = median filter rank, where $n > r \geq 0$

J_i = subset of input array X,centered about the i*th* element of X, such that

$$J_i = \{x_{i-r}, x_{i-r+1}, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_{i+r-1}, x_{i+r}\}$$

and indexed elements outside the range of X are set equal to zero. The elements of Y are calculated by $y_i = \text{median}(J_i)$, for $i = 0, 1, 2, \ldots, n - 1$.

As mentioned earlier, peaks which make up less than 50% of the points in a subset are removed. The median filter preferentially removes sharper peaks and passes broader features, and its discrimination between sharp and broad is controlled by the value of the filter rank, r. Lower values of r give smaller windows and only remove the sharpest peaks, while higher values of r give larger filtering windows and can result in the removal of even relatively broad peaks from the input data.

BASELINE CORRECTION

We would like to remove the baseline drift which is often a problem in chromatographic data and retain the sharp solute peaks that "ride" on top of the drift. Also, we would like to do this without significantly affecting the areas, heights, or shapes of the peaks of interest.

(1) Justusson, B. I. In *Two-Dimensional Digital Signal Processing II, Transforms and Median Filters*; Huang, T. S., Ed.; Springer-Verlag: New York, 1981; pp 161-196.
(2) Franco, M.; Treister, A. *Flexitrace, Automatic Graph to Number Conversion*; Tree Star Inc.: Campbell, CA, 1990.
(3) Wecksung, G. W.; Campbell, K. *Computer* **1974**, *7*, 63-71.
(4) CSMA Software Manual; Princeton Instruments Inc.: Trenton, NJ, 1990.
(5) Rabiner, L. R.; Sambur, M. R.; Schmidt, C. E. *IEEE Trans. Acoust., Speech, Signal Process.* **1975**, *23*, 552-557.
(6) Jayant, N. S. *IEEE Trans. Commun.* **1976**, *24*, 1043-1045.

0003-2700/93/0365-0188$04.00/0

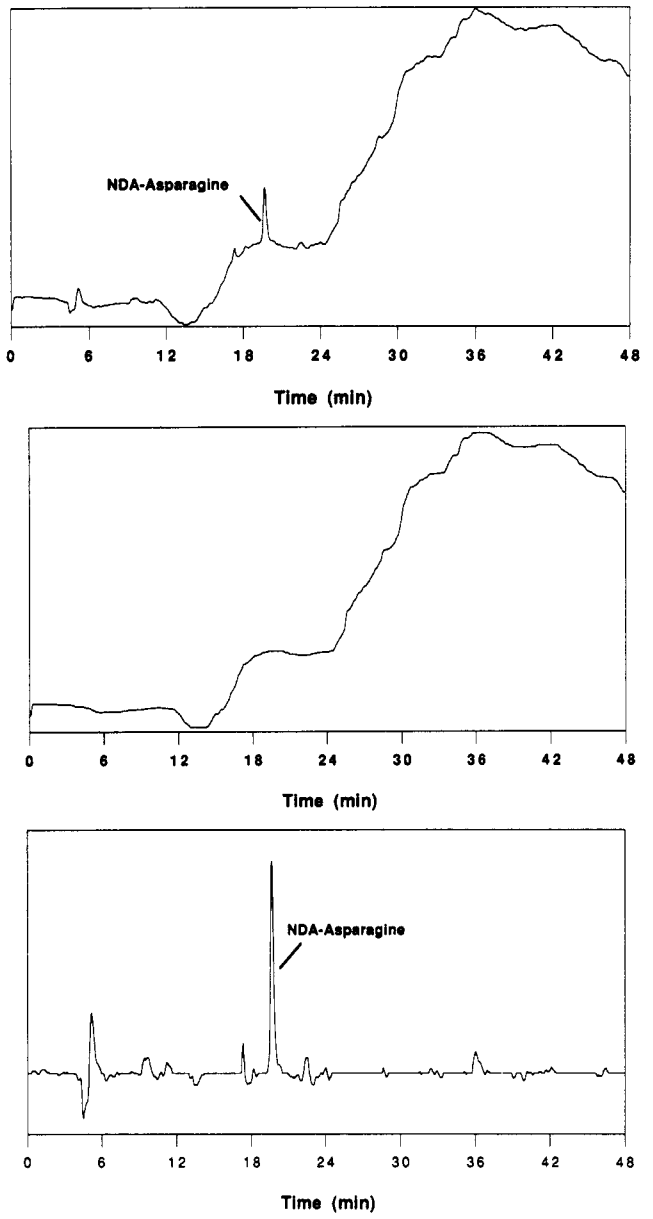

Figure 1. (a, Top) raw data. Chromatographic analysis scanned in from ref 7 and digitized. (b, Middle) median-filtered data. Raw data filtered with a median filter of rank 52. (c, Bottom) raw minus median-filtered data. Background-subtracted (difference) data, raw data from Figure 1a minus median-filtered data from Figure 1b.

Figure 1a is a chromatogram taken from the literature.[7] The data as plotted in the literature were scanned in, digitized, and replotted to give Figure 1a. A total of 2000 points were taken across the x-axis, so the effective sampling rate for the data shown is 0.7 Hz. This is a liquid chromatographic (LC) analysis of naphthalene-2,3-dicarboxaldehyde (NDA) labeled asparagine using an electrochemical detector in the amperometric mode. The y-axis in the original data is electrochemical oxidation current. Since this is a gradient LC run near the detection limit for NDA–asparagine, baseline drift is severe.

Figure 1b is the result of filtering the raw data in Figure 1a with a median filter of rank 52. That is, each point in Figure 1b is the median of a 105-point ($2r + 1 = 105$) subarray centered on the corresponding point in Figure 1a. Notice that the sharp NDA–asparagine peak and most of the other sharp features in Figure 1a are missing in Figure 1b, but most of the characteristics of the baseline drift remain. The information we are interested in, however, is generally in the sharper features. Next we subtract the data in Figure 1b from that in Figure 1a to obtain Figure 1c, the difference data. In Figure 1c, the NDA–asparagine peak is now the largest peak in the chromatogram, on what is now a flat baseline. The other sharp features from Figure 1a are also accurately carried through into Figure 1c, only much more discernible now after removal of the drift.

(7) Oates, M. D.; Jorgenson, J. W. *Anal. Chem.* **1989**, *61*, 432–435.

QUANTITATION

To quantitatively measure the effects of the filtering procedure, we compared peak height and peak area for the asparagine peak in the raw data and the difference data. Peak areas and heights were calculated from the digitized data using a program written in LabVIEW. The peak start and stop times for the asparagine peak in the raw data were arbitrarily chosen by the user. The program then drew a baseline between these two points, and y-axis values for this baseline were subtracted from the peak's values to get corrected y-values. These corrected values were then used in the peak integration and peak height calculation routines, to avoid any additional area or height due to the baseline drift. The same method and the same start and stop times were used in calculating areas and heights for the raw data and the background-subtracted (difference) data. Results of these peak comparisons as a function of the rank of the median filter are shown in Figure 2.

Figure 2a shows the percent area of the filtered peak relative to the unfiltered peak as a function of the rank of the median filter used. Figure 2b is a similar comparison of peak height. Since the results for peak heights follow the same trend as peak areas, the following discussion will consider only peak areas.

In general, the optimum rank value is greater than or equal to the width at the base (measured in the number of data points) of the widest peak of interest. This will ensure that the filtering window for the moving median is over twice the peak width, and the peaks will be completely removed from the filtered data. Since the width is measured in the number of data points rather than time, the data acquisition rate must be considered. A peak 30 s wide sampled at 1 point/s (Hz) will require a rank of 30 (to give a filtering window width of 61 points). This will ensure that the peak will be completely removed in the filtered data. The same peak sampled at 2 Hz would be 60 points wide and require a rank of 60 (a window 121 points wide).

For a given peak, here NDA–asparagine, peak area in the difference data is much less than in the original data if the rank is less than the optimum value for that peak. If the rank is greater than the optimum value for that peak, peak area in the difference data increases slightly.

The asparagine peak is about 80 s wide at the base, and the effective data acquisition rate for this scanned image is 0.7 Hz. This gives 56 points across the peak and an optimum rank for filtering (as defined above) of 56. In practice we find that peak areas are the same in the original and difference data for ranks of 52–64. As shown in Figure 2a, below rank 52, peak area values drop rapidly in the difference data. Above the optimum, peak area in the difference data has risen only 2% by rank 96. This can be explained as follows.

At rank values below the optimum, significant signal from relatively sharp peaks passes through the median filter and is subtracted out in the second step of the procedure. Thus peak areas in the difference data are lower than in the raw data. At some point (for the asparagine peak, at ranks 52–64), the peak area in the difference data exactly equals that in the raw data. This is the ideal rank value to use with the

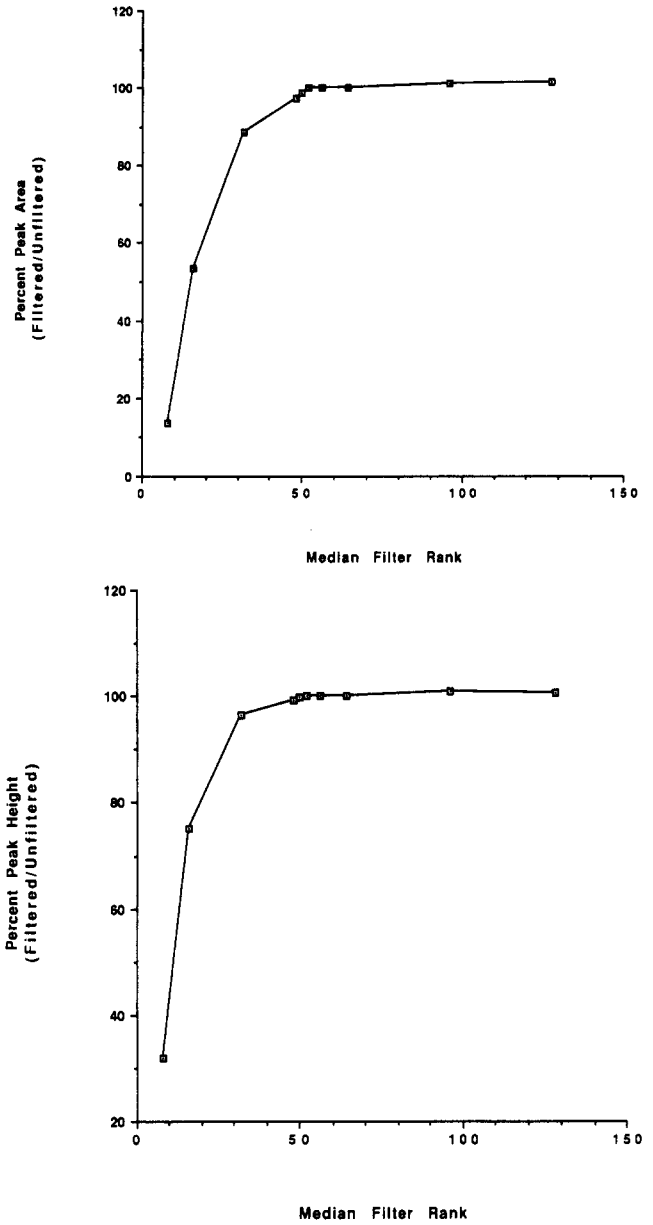

Figure 2. (a, Top) percent peak area vs filter rank. Plot of percent peak area, filtered/unfiltered, as a function of median filter rank. (b, Bottom) percent peak height vs filter rank. Plot of percent peak height, filtered/unfiltered, as a function of median filter rank.

moving median filter. As shown in Figure 2a, the peak area vs rank curve then levels out. At rank values above the optimum, even relatively broad features in the raw data are removed in the first median filtering step. Thus, in the difference data, y-values for the peak and the nearby baseline may be slightly higher, resulting in greater peak area. This is also a function of how close to the peak the start and stop times are chosen, as most of the difference is in the baseline values around the peak.

Choice of the optimum rank value for this method depends in several ways on the particular data being filtered. If there are several peaks of interest, some more broad than others, then the rank value necessary to accurately reproduce the broader peaks should be used. A rank value that is higher than necessary for a sharper peak would result in less error in peak area than a rank value that is too low for a broader peak, if both peaks are of interest.

Another consideration is peak density, the number of peaks within the filter window used. The method works best for well-separated peaks. If the peaks are sharp, such that a lower rank value could be used, but there are enough of them in one window to span more than 50% of the points in that window, they will be filtered incorrectly. Multiple peaks in the raw data will pass through the median filter as one broad peak and will result in low area values and dips below baseline in the difference data. In this case a larger rank value would be needed to give a more accurate treatment of the tight cluster of peaks.

DISCUSSION

Statistically, the median is both robust and nonparametric. No assumptions need be made about the noise distribution in the signal or the population from which the signal data are drawn.[8] The median filter is also resistant to outliers. For a given set of values, a large change in a small number of values would have a large effect on the mean of those values but would have only a small effect on the median. This is in fact the basis of the use of the median filter in this method.

The effects of the moving median filter are quite different from those of other common digital filtering routines. Because it is nonlinear, the moving median filter has no general frequency response function. The response calculated for a particular function will not apply to linear sums of that function or other functions.[1] It is this unique feature which is helpful in our application.

In the first step, we wish to filter out the chromatographic peaks of interest to leave only the baseline drift. The peaks are sharper than the baseline drift and generally contain more high-frequency components, but they also contain some low-frequency components. Thus most linear digital filters which discriminate by frequency are handicapped. High-pass filters lose some of the low-frequency components of a peak as they attempt to block the low-frequency baseline drift. Low-pass filters (used as we use the moving median, in a two-step filter and subtract method) allow low-frequency components of peaks to pass through, but they are then lost in the subtraction step. The moving median has an advantage here because it removes "impulses" relative to a background level, irrespective of frequency. If the filtering window used is of the appropriate size, both high- and low-frequency components of a peak are filtered off together and then restored in the final difference data.

Figure 3 shows the results of some common digital filters applied to the same data seen in Figure 1. These filters are based on the fast Fourier transform (FFT) and as such discriminate on the basis of frequency. The Butterworth filter (Figure 3a) gives a flat response in both the passband and the stopband, and the sharpness of the rolloff (transition between passband and stopband) is determined by the number of filter poles. The minimum number of poles is 2, and the Butterworth and all other filters in this example are two poled. This gives a broader rolloff, but using more than two poles to sharpen the rolloff has the negative effect of increasing ringing in the filtered data.

The Chebyshev filter (Figure 3b) has a flat response in the stopband and gives a sharper rolloff than the Butterworth but allows some ripple in the passband, while the Chebyshev II (or inverse Chebyshev, Figure 3c) allows ripple in the stopband and is flat in the passband.

These filters have a distinct phase response for each frequency present in the input data, which results in a phase shift in the output data. To eliminate this, the filtered data array can be reversed and passed through the filter a second time.[9] All frequencies phase shifted during the first pass will

(8) Snedecor, G. W.; Cochran W. G. *Statistical Methods*, 7th ed.; The Iowa State University Press: Ames, Iowa, 1980.
(9) Hamming, R. W. *Digital Filters*, 2nd ed.; Prentice-Hall, Inc.: Englewood Cliffs, NJ, 1983.

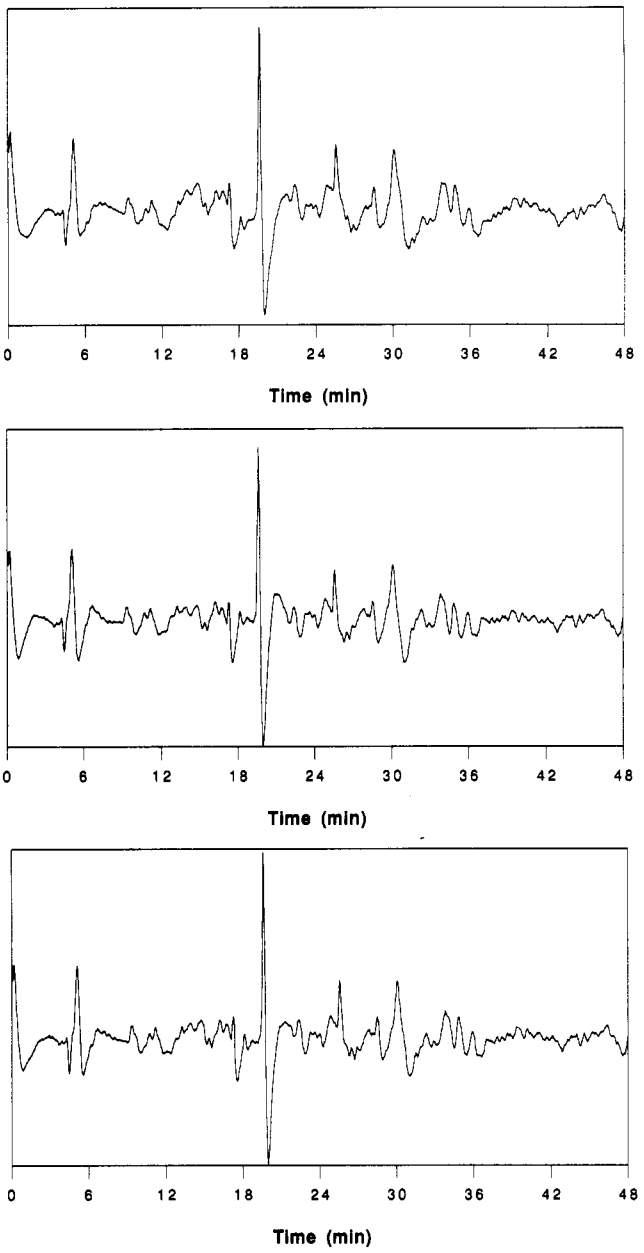

Figure 3. (a, Top) Butterworth high-pass. Raw data from Figure 1a, Butterworth high-pass filtered, cut-on = 0.005 Hz, 2 poles. (b, Middle) Chebyshev high-pass. Raw data from Figure 1a, Chebyshev high-pass filtered, cut-on = 0.008 Hz, 2 poles, ripple = 1 dB. (c, Bottom) Chebyshev II high-pass. Raw data from Figure 1a, Chebyshev II high-pass filtered, cut-on = 0.008 Hz, 2 poles, attenuation = 60 dB.

be shifted again by the same amount during the second, but in the opposite direction because the array is reversed. The output array from the second pass must then be reversed again before display to give the data in its original time sequence.

Results for the Butterworth no-phase filter are shown in Figure 4, both for the high-pass (Figure 4a) and the low-pass-with-subtraction (Figure 4b). These actually give the best results for the linear digital filters tested, but even here the resulting "smoothed" baseline is much worse than that obtained with the moving median.

CONCLUSIONS

The median filtering method described above gives effective removal of baseline drift while maintaining peak heights and

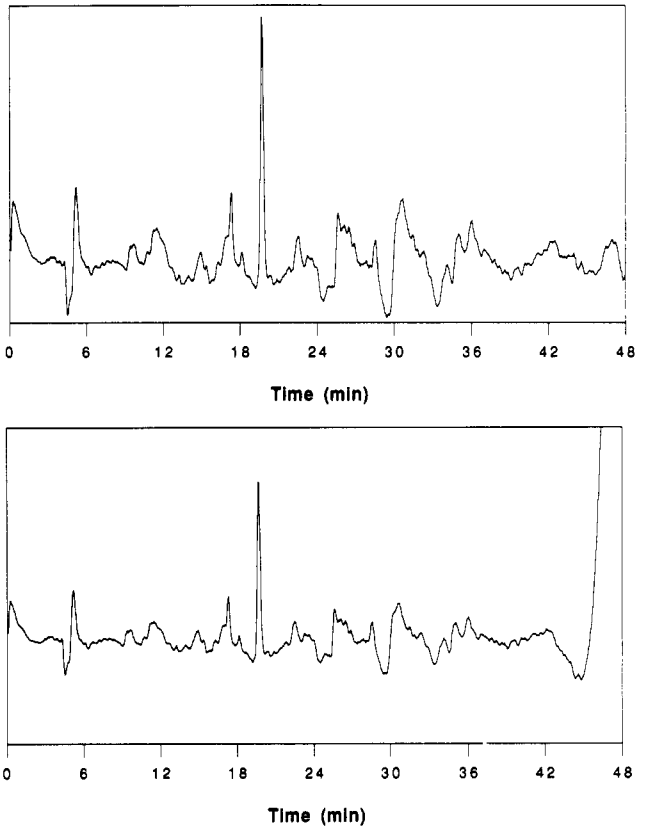

Figure 4. (a, Top) Butterworth no-phase, high-pass. Raw data from Figure 1a, phase-corrected Butterworth high-pass filtered, cut-on = 0.003 Hz, 2 poles. (b, Bottom) Butterworth no-phase, low-pass difference. Raw data from figure 1a, phase-corrected Butterworth low-pass with subtraction of filtered data from raw data. Cut-off = 0.0035 Hz, 2 poles.

areas in the sharper portions of the signal. As with most filtering schemes, it works best when the peaks of interest are on a very different time scale from the undesirable background. In general, sharp peaks on a broad rolling baseline will be handled more easily than broader peaks on the same baseline. Unlike some other filtering methods, it is sensitive to peak density in the input data, and the filtering is more a function of the number of data points across a peak than the particular frequency components of that peak.

The method is easy to program on a personal computer and may find use in a variety of other applications. An appropriate rank value is easily determined using the guidelines above and knowledge of the peaks of interest and, once chosen, generally works on all analyses of that same type. In contrast to the other digital filtering methods we examined, choosing a suitable rank value required much less detailed knowledge of the input data. The other filters tested require in-depth examination of the frequency distribution of the input data or much trial-and-error with the filter coefficients.

ACKNOWLEDGMENT

This research was supported by a grant from the National Institute of Health (GM 39515) and by a gift from Hewlett-Packard. A.W.M. was supported by a fellowship from the Department of Education (Contract No. P200A10047).

RECEIVED for review May 1, 1992. Accepted October 5, 1992.