Detection, Classification, and Superposition Resolution of Action Potentials in Multiunit Single-Channel Recordings by an On-Line Real-Time Neural Network

Rishi Chandra and Lance M. Optican*

***Abstract*—Determination of single-unit spike trains from multiunit recordings obtained during extracellular recording has been the focus of many studies over the last two decades. In multiunit recordings, superpositions can occur with high frequency if the firing rates of the neurons are high or correlated, making superposition resolution imperative for accurate spike train determination. In this work, a connectionist neural network (NN) was applied to the spike sorting challenge. A novel training scheme was developed which enabled the NN to resolve some superpositions using single-channel recordings. Simulated multiunit spike trains were constructed from templates and noise segments that were extracted from real extracellular recordings. The simulations were used to determine the performances of the NN and a simple matched template filter (MTF), which was used as a basis for comparison. The network performed as well as the MTF in identifying nonoverlapping spikes, and was significantly better in resolving superpositions and rejecting noise. An on-line, real-time implementation of the NN discriminator, using a high-speed digital signal processor mounted inside an IBM-PC, is now in use in six laboratories.**

***Index Terms*— Multiunit spike sorting, neural networks, on-line, real-time discrimination.**

I. Introduction

THE analysis of simultaneous activity from several neurons can lead to a better understanding of their functional connectivity. Extracellular recordings with low-impedance electrodes are capable of recording such activity from several neurons near the tip of the electrode. The shape manifested in the recording of an action potential, or spike, emitted from a particular neuron is a function of the position of that neuron relative to the electrode. Thus, by sorting the different shapes in an extracellular recording, the interleaved spike trains of the individual neurons can be separated. There are two significant difficulties in analyzing extracellular recordings: high levels of correlated neural noise and superpositions of waveforms. Background noise consists mainly of the activity of distant neurons. These noise levels can be relatively high and since most neurons have similar time constants for potential changes, the spectral contents of the noise and signal overlap considerably. Superpositions occur when two neurons fire within 1 ms of each other, resulting in the addition of their individual waveforms. In the case where firing patterns are not correlated, the frequency of superpositions is determined by the number of neurons and their firing rates. If there are x neurons, each firing with an average frequency of y Hz, and having a spike duration of d s, the average superposition rate is given by the expression $y^2\,dx(x-1)$ [1]. For example, if there are three neurons each firing at 50 Hz, with waveforms of 1-ms duration, then the average number of superpositions would be 15 s^{-1}. Doubling the common firing rate to 100 Hz causes the overlap rate to increase fourfold to 60 s^{-1}. Thus, superposition resolution is an important consideration in spike sorting. When the firing patterns are correlated to each other or a particular stimulus, the rate of superpositions can be much higher. This paper presents a fully automated system for sorting spikes and their superpositions on-line in real-time.

Three criteria governed the development of our spike sorter, 1) good isolation of single units in multiunit single-channel recordings; 2) an on-line system, because details of experiments are often determined by the selectivity of the identified neurons; and 3) resolution of superpositions, because many neurons burst together when a stimulus is presented, which results in many superpositions.

Many signal-processing techniques have been applied to the challenge of sorting multiunit recordings (see the review paper by Schmidt [2], and a comparison of various sorters by Wheeler and Heetderks [3]). One of the most popular methods is matched template filtering, commonly referred to as template matching [4]–[13]. This technique uses templates that represent the typical waveform shape of each neuron to filter the data. Variants of template matching have been implemented in the form of reduced feature sets, such as principal component analysis or simple waveform characteristics (e.g., spike height and width), to provide faster processing [14]–[16]. A learning phase is required for the template generation and determination of classification thresholds for all these sorters. Some investigators have considered the problem of superpositions [8], [13]. The general scheme for superposition resolution in template matching relies on iterative subtraction of all possible template combinations from unidentified waveforms. This approach is computationally

Manuscript received July 24, 1995; revised January 7, 1996. *Asterisk indicates corresponding author*.

R. Chandra is with the Laboratory of Sensorimotor Research, National Eye Institute, NIH, Bethesda, MD 20892 USA.

*L. M. Optican is with the Laboratory of Sensorimotor Research, National Eye Institute, NIH, Bethesda, MD 20892 USA (e-mail: lmo@lsr.nei.nih.gov).

Publisher Item Identifier S 0018-9294(97)02959-5.

intensive, making on-line implementation difficult, especially when the rate of superpositions is high.

Matched template filtering is a time domain technique; frequency domain methods have also been applied to spike sorting, the most common technique being optimal linear filtering [17]–[21]. This approach is based on deriving optimal filters that respond to one unique template and reject the others and the background noise. The learning phase generates templates which are then used to derive their respective optimal linear filters. Although linear optimal filtering can resolve linear superpositions, it is dependent on multichannel recordings. Roberts and Hartline [17] stated that the number of channels should be greater than or equal to the number of units to be identified. The optimal filtering technique is ideal for multichannel recordings where the units appear on at least two of the channels. All the investigators using this technique [17]–[21] concluded that the discrimination performance would deteriorate rapidly if this condition were not met, although this dependency may be reduced by adding nonlinear properties to the filters, e.g., an iterative subtraction scheme that removes the largest remaining units on each iteration.

An off-line Bayesian modeling scheme has been proposed by Atiya [22] and Lewicki [23]. This method uses both the waveform and firing rate information to minimize the probability of error during classification. To resolve superpositions, Atiya implements a comprehensive search of the space of overlapping spike shapes and event times to find the sequence of maximum probability. This is a very computationally demanding approach. Lewicki uses the same principle but restricts the search by using a Poisson model for the firing of the neurons, then uses data structures and dynamic programming to efficiently sort the spikes. Lewicki's technique can be implemented in a pseudo-on-line system by processing one set of data while collecting another. If results of the spike classification are needed before the end of an experimental trial, a true on-line system would still be needed.

Artificial neural networks (ANN's) have been used for the classification of units in a multiunit recording with fully connected, feed-forward networks [24]–[26]. An implementation of the adaptive resonance theory (ART)-2 neural network (NN) algorithm was also applied to this problem [27]. Although these applications have been successful in isolating single units and are better than template matching [25], none were trained to recognize, or were tested on, superpositions of waveforms. A Hopfield-type network has been implemented in real-time to separate units and their superpositions [28], although the investigators concluded that template matching performed better than the Hopfield-type network. A unique pseudo-unsupervised training method based on network relaxation has been implemented off-line [29], [30] that learns the template shapes. This application was developed to resolve motor unit action potentials (MUAP's), which are significantly longer than neuronal action potentials. Superposition resolution was possible when the MUAP peaks were sufficiently separated in time.

An NN was used in the work presented here because it combines linear filtering (by the weights from the input layer to the hidden layer) with nonlinear classification (by the nonlinear hidden units), which should enable it to resolve superpositions of units in a multiunit single-channel recording. Another advantage of using an NN for sorting is that in multidimensional feature space, decision regions are hyper-ellipses, because the background neural noise is colored-Gaussian-type noise (if the noise were white, then the decision regions would be hyper-spheres, i.e., equal variance in all dimensions), and NN's are capable of generating nonspherical decision boundaries. The final advantage of using an NN lies in the ability to implement it on high-speed digital processors, allowing on-line real-time multiunit sorting with superposition resolution. This paper presents a method for training an NN to sort spikes and resolve superpositions. This training method has three novel components. First, the training set consists of both individual templates and superimposed templates. Second, the training set includes simulated noise with the same spectral characteristics as the neuronal noise. Third, the NN is used as a nonrecursive [finite impulse response (FIR)] digital filter, simultaneously accomplishing both detection and classification of spikes. The performance of this NN is compared to that of an MTF on simulated data sets with different templates and signal-to-noise ratios (SNR's).

The method developed here has four characteristic features: automatic determination of a detection threshold based on noise analysis, automatic clustering of waveforms to obtain templates, supervised training of an NN, and operation of the network as a simultaneous detector/classifier in real-time. This method has been successfully implemented on a PC-host with a digital signal processor (DSP) co-processor system.

II. Methods

A. Data Collection and Spike Detection

Data were acquired during extracellular recording with tungsten microelectrodes from the medial superior temporal area in the cortex of a monkey during a visual experiment with moving stimuli (see Duffy and Wurtz [31] for methods). The system has also performed well on recordings from primate superior colliculus. Data sets were collected from three different recording sessions and stored on analog tape. From each session, a section of the tape with low background noise levels was selected and digitized at a rate of 24 KHz using a 16-b analog–digital (A/D) converter (ADC-DBCS5339-50, Communication Automation and Control, Allentown, PA). The ADC utilizes a front end antialiasing filter, delta-sigma modulation, 64-times oversampling, and a three-stage digital FIR filter to achieve a 94-dB SNR.

Spikes were detected when the amplitude of the recorded signal exceeded a positive or negative threshold. The threshold was determined by breaking the record into pure neuronal noise segments and setting the threshold at three standard deviations of that noise [32]. The neuronal noise was segmented by an algorithm which separates signal from noise based on the Gaussian characteristics of the noise [33]. Detected spikes were clustered together to form noise-free templates using a recently developed simultaneous clustering algorithm [34]–[36]. This algorithm exploits the advantages of simul-

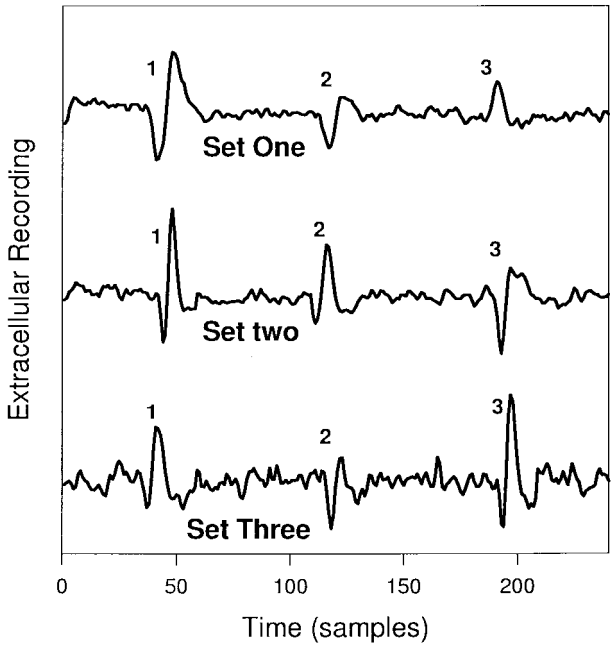


Fig. 1. Artificial data sets based on three separate neurophysiological recording sessions. Three templates that represented the most dense spike clusters with high SNR were chosen from each recording session. The templates are shown embedded in a short segment of their corresponding neural noise. The SNR's for each template in the three sets were: 1) 7.72, 3.46, and 3.26; 2) 7.51, 3.46, and 4.72; and 3) 2.98, 2.40, and 4.93.

taneous clustering and accounts for detection jitter when segmenting waveforms from recordings. The algorithm is described briefly in the Appendix. Three templates, which represented dense spike clusters with high SNR's, and the segmented neural noise, were stored from each of the three data sets. Each template consisted of 24 samples (1.0 ms in duration at 24 KHz). Fig. 1 shows the templates embedded in a short segment of their respective noise traces.

B. Discriminator Details

This study compares the spike discrimination performance of a connectionist NN to that of an MTF. Fig. 2 illustrates the structure of the two discriminators. The MTF [Fig. 2(a)] sets circular decision regions [13] in multidimensional space. The incoming data stream takes the form of vectors in this space, i.e., values for each dimension (feature) are recorded over the length of an analysis window. In our implementation, a 24-sample window was used, where each sample in the window constitutes one feature. The templates are represented in the same manner and the sum squared difference between the incoming data and the templates is calculated. If the error is a local minima and less than threshold, then the incoming pattern is said to match the template. Since the noise is colored, the threshold was determined empirically from a pure noise segment. This was done by building a histogram of the power (sum of squared samples in a 24-sample window) of the noise record. The threshold was set at a value that was greater than 99.9% of the population. This method approximates the squared Euclidean distance threshold for classification and avoids the errors that can be caused by outliers.

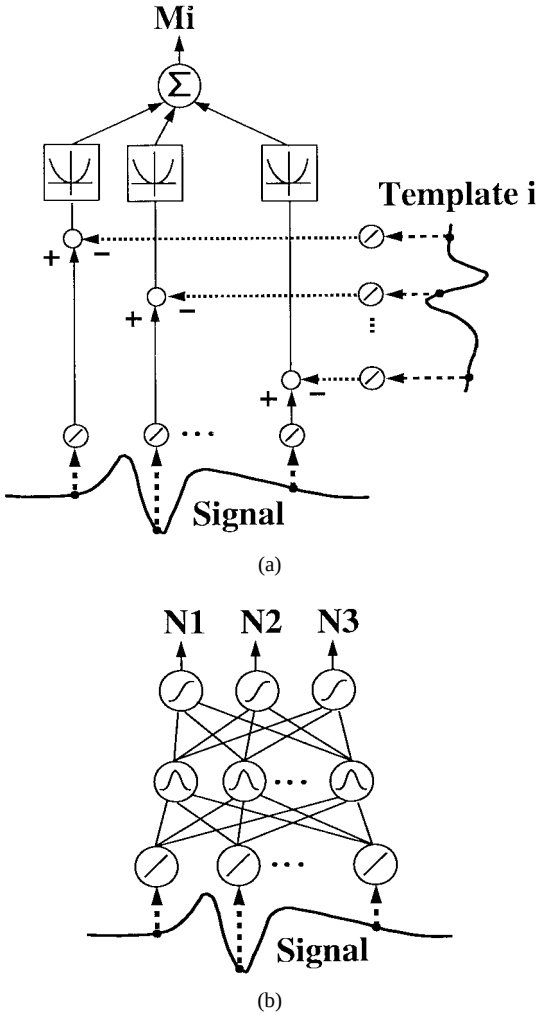


Fig. 2. The structures of the discriminators. The input to each discriminator is the digitized signal, which is shifted one sample at a time. In (a) the MTF, the template is subtracted from the signal, the differences squared and summed. The output of each template filter is compared to a threshold determined from the statistics of the noise. (b) The fully connected feed-forward NN is used as a nonlinear FIR filter. The outputs are compared to a threshold which is empirically determined. Input layer units have a linear function, the hidden layer units have a Gaussian function, and the output layer units have a sigmoid function.

The other discriminator [Fig. 2(b)] was an NN used as an FIR filter [37], [38]. A fully connected, feed-forward, three-layer NN was used. The network had twenty-four input units, eight hidden units, and three output units. Linear activation functions were used for the input nodes, Gaussian activation functions for the hidden layer, and logistic activation functions (range from zero to one) for the output layer. The last 24 samples of the incoming data formed the input to the network. An event constituted a strong response (>0.95) in any of the output units. In simulations we tested different numbers of hidden units. Below eight hidden units, the classification performance dropped drastically. Although eight hidden units is the maximum we could implement in the real-time system, we tested networks with ten and twelve hidden units and no significant improvement in classification performance was observed.

The network was trained with a modification of the back-propagation algorithm [39]. Modifications that were incorpo-

rated in the training were an adaptive learning rate, weight momentum, weight decay, and weight annealing. The training for each test consisted of each template, with added noise, at all possible shifts along the input buffer of the NN. The weighting for the error at the output layer was an exponential function of the shift. The training input–output set was defined as follows.

If N is the total number of templates, $2M+1$ is the number of possible shifts ($N = 3$, $M = 24$ in this training set), template (i, j) stands for template i with shift j, and $w(j)$ represents the weighting of the output error, then, the inputs are

$$x(i, j) = \text{template}(i, j) + \text{noise}, \qquad \text{for } i = 1, \cdots, N \text{ and } j = -M, \cdots, +M$$

the outputs are

$$o(i, j, m) = 1.0, \qquad \text{for } m = i \text{ and } j = 0;$$
$$o(i, j, m) = 0.0, \qquad \text{otherwise}$$

and the weights for the error are

$$w(0) = 1.0$$
$$w(j) = \exp\left(\frac{|j| - 1}{\tau}\right), \qquad \text{for } |j| \neq 0, \quad \tau = 4.0.$$

The training set with superpositions adds the following inputs:

$$y(k, i, j) = \text{template}(k, 0) + \text{template}(i, j) + \text{noise}, \quad \text{for } i = 1, \cdots, N,\ k = 1, \cdots, N,\ k \neq i, \text{ and } j = -M, \cdots, +M$$

the corresponding outputs and weights are

for $m = k$,

$$o(k, i, j, m) = 1.0 \qquad \text{for all } i \text{ and } j.$$

for $m \neq k$,

$$o(k, i, j, m) = 1.0 \qquad \text{for } m = i \text{ and } j = 0;$$
$$o(k, i, j, m) = 0.0 \qquad \text{otherwise.}$$
$$w(j) = 0.05 \qquad \text{for all } j.$$

This input–output set trains the network to respond only when a spike is placed exactly in the middle of the input buffer. Any shifts cause the network to reject the waveform. In the case of superpositions that are n samples apart, ideally the network would respond to the first spike and then respond again n samples later to the second spike. Fig. 3 shows the output of the two discriminators for a short sample of data. Data is the input, Ni is the corresponding output of the network, and Mi is the squared difference between the waveform and MTF ($i = 1, 2, 3$).

C. Simulations and Testing

Software simulations were used to determine the performance of the MTF and NN discriminators. As a simple model of the recording process, assume that when a neuron fires it always produces the same waveform shape in the recording, but is corrupted by colored Gaussian noise. Then each template, which is an average of many action potentials determined to be from one neuron, represents the "noise-free" shape

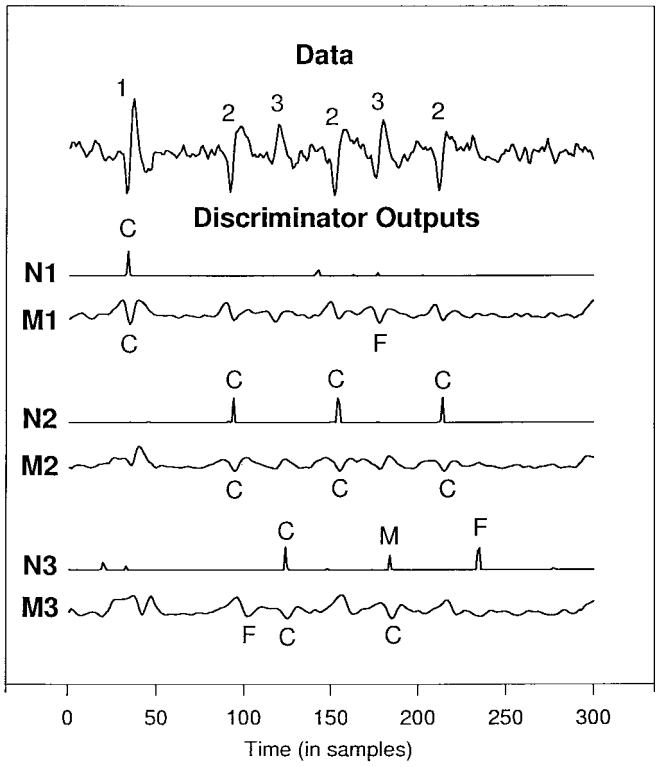
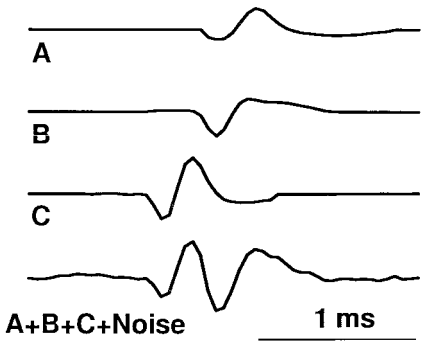


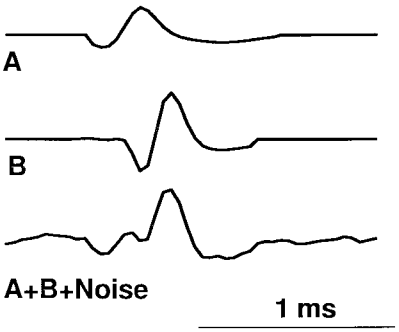
Fig. 3. A 10-ms segment of a simulated spike train with five spikes, and the corresponding outputs of the NN and MTF. The top trace shows the data, which has an SNR for the three templates of 5.74, 3.61, and 2.65. The numbers show the position of each template. The outputs of the two discriminators are shown in the traces below. Ni is the ith output of the NN, Mi is the output of the ith MTF. "C" represents a correct classification, "M" a missed detection, and "F" a false detection. The NN correctly classifies five out of the six waveforms (it misses one of type 3) and has one false positive. The MTF detects every waveform but also has two false positives. Note that the output traces of the two detectors have been shifted forward by 12 samples to align the outputs with the middle of the spikes for illustration purposes.

manifested in a recording by the respective neuron [1]. Thus simulated multispike trains were constructed by embedding the templates at various locations in extended noise traces. Neural noise for each data set was modeled by generating white Gaussian noise [40] and digitally filtering it with a second-order recursive (IIR) filter with three nonrecursive coefficients and two recursive coefficients to match the spectral characteristics of the real neural noise. The coefficients of the recursive filter were determined by fitting a single exponential to the decay of the autocorrelation function [41].

Simulations allowed complete control of test data and exact evaluation of each discriminator's performance. For each simulated spike train, the variables in the record that could be manipulated were: 1) the firing rate of each neuron; 2) superpositions of waveforms; and 3) the background noise level. Three different types of simulated spike trains were constructed for each data set; the first type had an approximate firing rate of 120 Hz for each neuron with superpositions occurring at a rate of 80 Hz, i.e., about 2/3 of the spikes present were the superposition of two or three individual spikes. Fig. 4 shows two examples of superimposed waveforms from the first type of simulations. The second type had a firing rate of 90 Hz for each neuron and no superpositions. For these two



(a)



(b)

Fig. 4. Examples of the superpositions of (a) three and (b) two templates. The bottom traces in each example show the superpositions of the templates shown above with additive noise. The average SNR of the spike train these segments were taken from was 5.0. In (a), the overlap is 0.71 ms between A and B, 0.67 ms between B and C, and 0.38 ms between A and C. The NN was able to resolve the earliest and largest (C) waveform and the matched filter rejected all three. In (b), the overlap between A and B is 0.88 ms. The NN was able to identify both constituent waveforms. The MTF rejected both.

simulation types, the SNR was varied from 1.5 to 10.0 by simply scaling the background noise. It is important to note that the noise had the same autocorrelation characteristics as the real noise from the recording sessions, and only its variance differed. At each SNR, 4 s of data were tested for each data set. The third and last simulation type was designed to test the performance of the two discriminators when the amplitude of the spike changed, because spike amplitude commonly drops during a high-frequency burst. The noise level was fixed to give an average SNR of 5.0 (relative to the original template sizes) and the magnitude of the template was varied between 0.5 and 2.0 times its original size. The original template sizes were used for the matched filter and training of the NN to test the robustness of the two discriminators in the presence of amplitude variation.

D. Real-Time Implementation

The matched filter and NN discriminators were realized using an 80-MHz AT&T DSP32C digital signal processor (Communications Automation & Control) mounted inside an IBM Pentium P5-133 MHz PC (Gateway). The DSP was fast enough to run either discriminator in real-time. The maximum size of the network that could be implemented with a 24-sample input buffer and a 24-KHz sampling rate was eight hidden units and three output units. The output of the chosen discriminator was accumulated and relayed to the host PC every millisecond for storage and on-line display. The learning stage consisted of two phases, template generation, and NN training. Fifteen seconds of data buffers were uploaded to the PC for noise modeling, spike detection, and template generation. The required processing time on an IBM P5-133 is approximately 3–4 s. Network training was performed on the DSP, with a typical training time of 60 s. Thus the total time to acquire new data, generate templates, and train the network is just under 2 min for this system. The code for the host PC was written in the C programming language under MS-DOS; the code for the DSP was written in assembly language.

TABLE I
A SAMPLE OF THE CLASSIFICATION SCHEME. AN ASTERISK INDICATES A SUPERPOSITION, OR OVERLAP OF TWO TEMPLATES. THE LOCATIONS IN THE TABLE ARE THE SAMPLE NUMBERS OF THE WAVEFORMS. WHEN SOMETHING IS DETECTED, IT IS REGISTERED AS A CORRECT CLASSIFICATION IF THE LOCATION IS EXACTLY RIGHT AND THE IDENTIFIED CLASS MATCHES THE TRUE CLASS. ANY OTHER DETECTION IS CLASSIFIED AS A FALSE POSITIVE. ERRONEOUS DETECTIONS OF NOISE AS UNITS ARE ALSO CLASSIFIED AS FALSE POSITIVES

True Class	True Location	Identified Class	Detected Location	Classification
I	350	I	350	Correct
II	475	I	475	Incorrect
III	500	Noise	Not Detected	Incorrect
II	610	II	615	False Positive
Noise	-	I	720	False Positive
I*	800	I	800	Correct
II*	804	II	804	Correct
III*	850	Noise	Not Detected	Incorrect
I*	856	I	856	Correct

III. RESULTS

The simulations described above were designed to compare the discrimination performance of the NN to that of the MTF. Two measures of performance were considered, the percent correctly identified and the rate of false positives. A correct identification means that the waveform was detected at the correct location and was classified correctly. An incorrect identification means either that the waveform was not detected at all, or was detected at the correct location but was misclassified. A false positive is any detection where no waveform had been placed. Table I gives a sample of the evaluation process.

In this scheme, correct classification of both constituent waveforms in a superposition counts as two correct classifications and resolution of two superimposed waveforms. Thus, a superposition resolution of 50% could mean that only one waveform was resolved from every superposition, or that both waveforms from half the superpositions were resolved, or a combination of the two extreme cases. In the examples given in Table I, the superposition resolution is 75%, as three out of four waveforms were correctly resolved. The results presented are the average performances of the two discriminators over the three simulated data sets. The SNR's quoted are the average SNR's of the spike trains. The SNR is defined as the root mean square value of the waveform divided by the standard deviation of the noise. The SNR's for a data set are simply the average SNR's of their constituent templates.

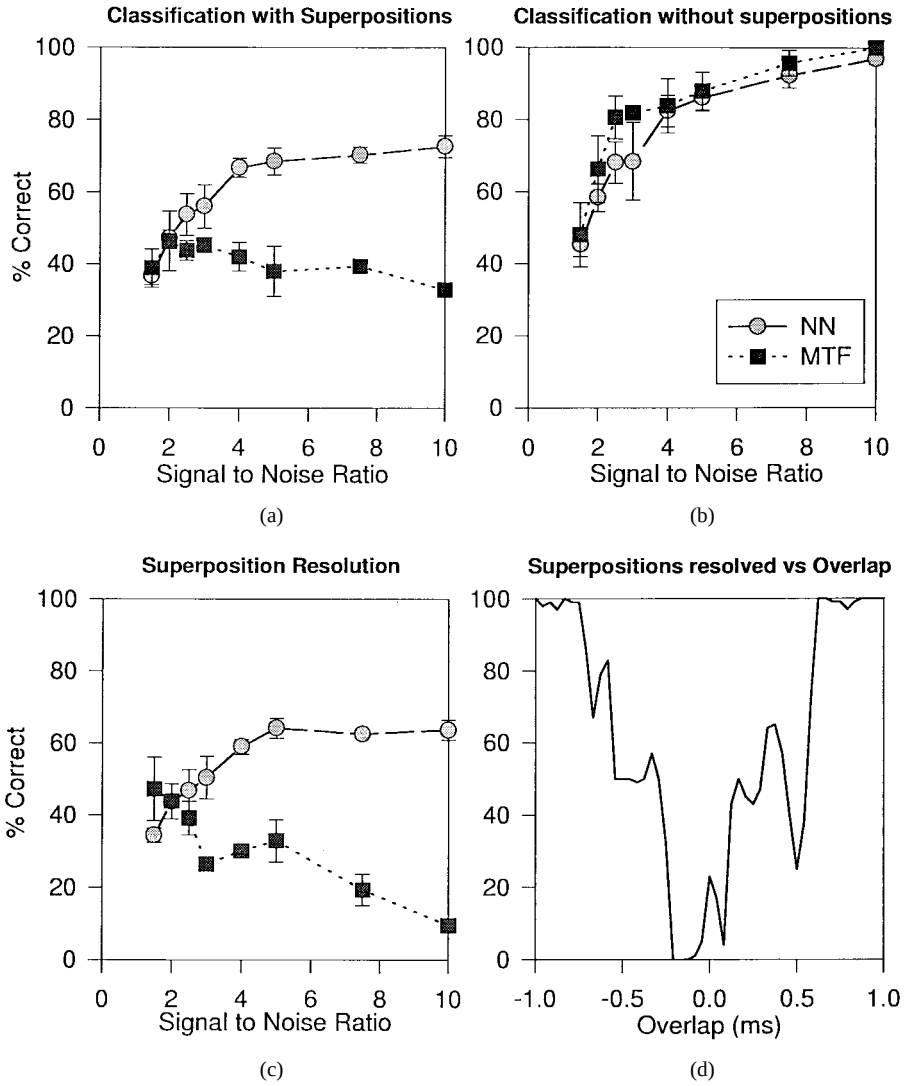


Fig. 5. The performances of the NN and MTF discriminators on simulated spike trains. The performances shown in (a), (b), and (c) are averages, and error bars the standard deviations, over the three data sets. At every SNR, 4-s multiunit spike trains were analyzed for each data set. The SNR's are the average SNR's of the templates. The first class of spike trains (a) had approximately 1440 spikes (a firing rate of 120 Hz for each of the three neurons) with 960 overlapping waveforms. The second class (b) had approximately 1000 spikes (approx. 90 Hz for each neuron) with no overlapping waveforms. The percent of overlapping waveforms resolved for the first class are shown in (c). The percent of superpositions resolved with varying overlap for two particular waveforms is shown in (d); at each overlap, approximately 300 superpositions were analyzed (600 waveforms). The templates used in this example were 2 and 3 from template set two (see Fig. 1). The SNR of each of the constituent waveforms was 3.1 and 4.23, the plot shows the percent when at least one of the two waveforms were resolved. Any performance over 50% means that at least in some instances, both waveforms were resolved.

A. Classification Performance

Fig. 5 shows the percent correctly detected by the MTF and NN in spike trains when superpositions were present [Fig. 5(a)] and in spike trains with no superpositions [Fig. 5(b)]. Fig. 5(c) shows the percent of superpositions resolved in the first case. With the exception of superposition resolution for the matched filter, all the performances saturate after an SNR of 5.0. At this SNR level and above, the separability (defined as the Euclidean distance between cluster centers divided by the standard deviation of noise) between *single* spike cluster centers is greater than six standard deviations of the noise (for our data sets), thus there is no cluster overlap in multidimensional feature space. When there are superpositions present, there is a significant difference in the performances of the two discriminators [Fig. 5(a)]. The performance of the NN saturates at a little over 70%, whereas the MTF only attains a performance of approximately 40%. The difference between the two is due to the inability of the MTF to resolve superpositions. The matched filter also seems to have relatively similar performance across the range of tested SNR's when superpositions are present. This is because at high SNR's superimposed waveforms are rejected and there is limited resolution. At low SNR's the distortion caused by superpositions falls within the classification threshold and the false positive rate also increases significantly as more noise is accepted by the matched filter. The effect of superpositions is confirmed by the next two graphs. When no superpositions are present [Fig. 5(b)], both discriminators attain at least 80% correct classification at SNR's of 5.0 and above. This is because of the good separability of the clusters at low noise levels. It is important to note that the NN detects and classifies single spikes almost as well as the MTF, showing that the difference in Fig. 5(a) is due to classification of superpositions.

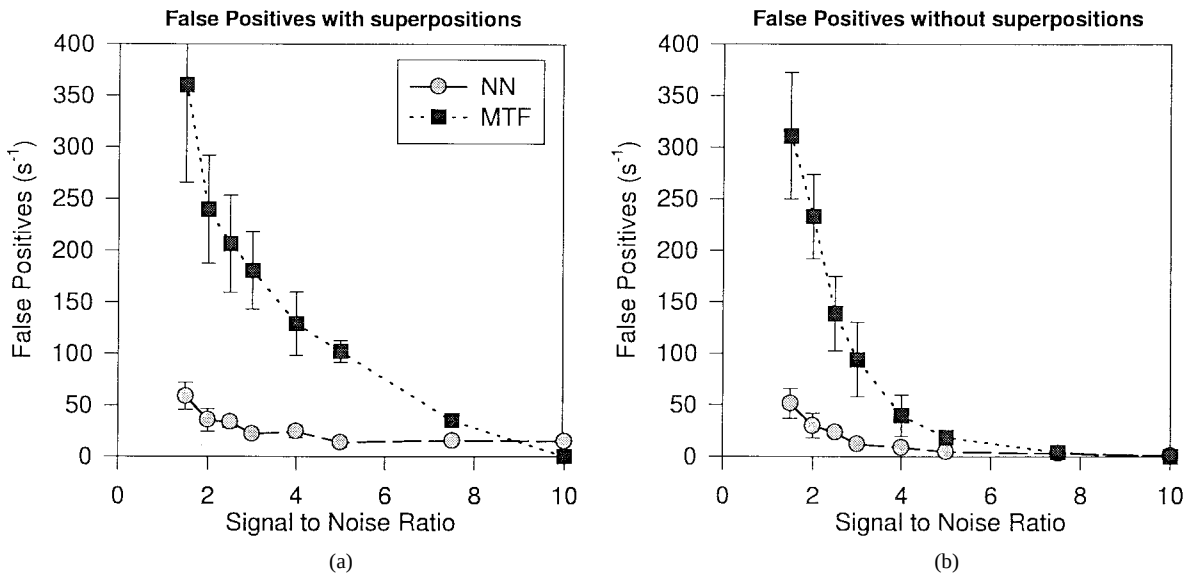


Fig. 6. The false positive rates for the NN and MTF discriminators on the two types of spike trains described in Fig. 4.

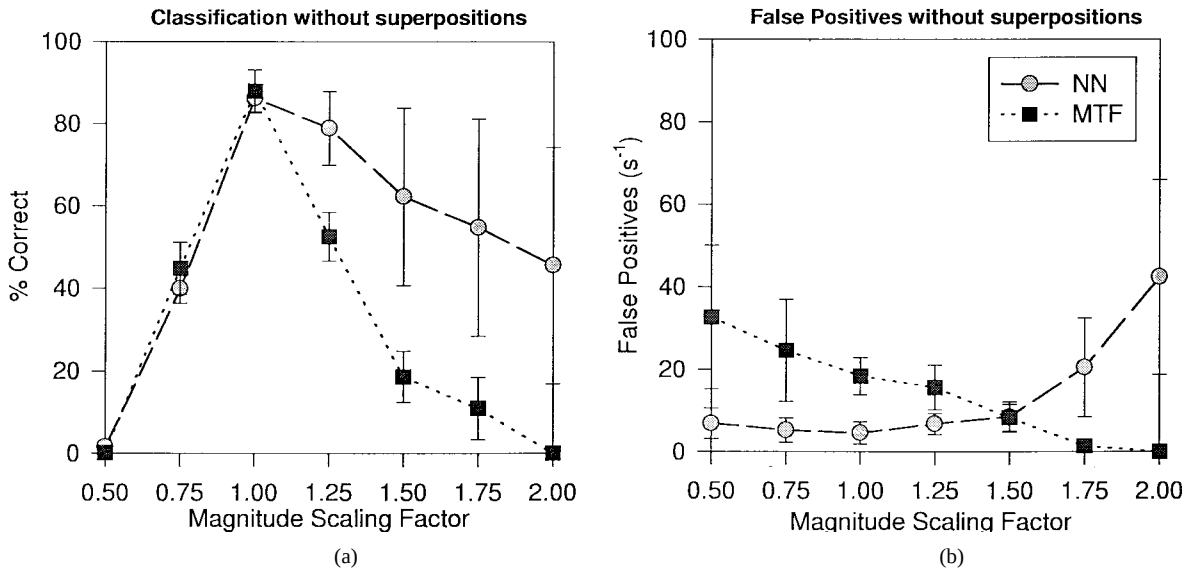
The third graph in the figure shows the difference in the superposition resolution. The NN performance follows that of the previous two graphs. The fact that the network cannot completely resolve more than 50%–60% of all superpositions explains why in Fig. 5(a) the performance of the network only reaches 70%. In the spike trains with superpositions, about 2/3 of the waveforms were superimposed, meaning that if approximately half of those were resolved as shown in Fig. 5(c), the best overall performance one can expect is about 70% correct classification. It was observed that the superposition resolution capabilities of the network varied significantly depending upon the relative shift of the constituent waveforms. For large shifts, generally both waveforms could be classified, as the distortion of each is low. For small shifts, the network was able to resolve some of these superpositions, despite high distortions. Fig. 5(d) shows one example of the superposition resolution for two waveforms at an average SNR of 4.5.

In Fig. 5(c), the matched filter could only resolve superpositions when the overlap was minimal, resulting in 10% resolution at high SNR's. Strangely the performance seems to improve with decreasing SNR. This is caused by a high number of false detections by the MTF discriminator as the SNR drops, thus correct identifications are attained by chance. Thus, it is important to compare the percent correct with the rate of false detections in evaluating a discriminator.

The rate of false positives is shown in Fig. 6. The large difference in the number of false positives between the two discriminators is striking. There is also a large difference between the false positive rate in the matched filter with and without superpositions, but only a minimal difference for the NN. Once again, the improvement in performance for both discriminators saturates at an SNR of 5.0 when no superpositions are present. At high SNR, when there are superpositions present [Fig. 6(a)], the matched filter has no false positives, whereas the network does have a few, which arise from incorrect classification of waveforms in superpositions. This is corroborated by Fig. 6(b), which shows minimal false positives by either discriminator at SNR's greater than 5.0 and none at 10.0 in spike trains with single units only. At low SNR's, where the noise level is high relative to the templates, there will be considerable overlap between the spike clusters and the noise cluster in multidimensional space. This is the cause of high false positives for the matched filter. It is important to note that the number of false positives for the matched filter can be reduced if an additional power detection scheme is used prior to classification. However, since this was a comparative study between the MTF and NN, the discriminators were used for both detection and classification. The network has a low false positive rate even when superpositions are present, indicating that it is much more selective than the MTF.

B. Amplitude Sensitivity

Fig. 7 shows the performance when the amplitude of the templates change at a fixed noise level. This was examined to see how the discriminators perform when the amplitude changes, because a significant problem in real physiological data is the fluctuation in spike amplitude during high-frequency bursts. During such a burst, the electro-chemical gradient that drives the spike drops, which causes the amplitude to decrease. Naturally both discriminators perform best when the scaling factor is 1.0. At scaling factors between 0.5 and 1.0, the percent correctly detected falls off at approximately the same rate for both discriminators. For values greater than 1.0, the NN seems to be more robust in identifying the neurons. When one examines the false positives [Fig. 7(b)], the discriminators behave in opposite fashions. The matched filter has an increasing number of false positives as the scaling factor drops below 1.0, and the NN when the scaling factor rises above 1.0. In the case of the MTF, the false positives result from misclassification of a scaled down big unit as a smaller template. Note that in the data sets used (shown in Fig. 1), some of the units are similar in shape but different in magnitude. The reason there are no false positives for the matched filter when the scaling factor is >1.0 (one might



(a) (b)

Fig. 7. The classification performance for the NN and MTF on simulated spike trains without superpositions when the magnitude of the spikes are varied (by multiplying the waveform by a scalar, the magnitude scaling factor). The performance shown are the averages and the error bars the standard deviations over the three data sets. At every point, 4 s of data were analyzed containing 900 waveforms (a firing rate of approximately 90 Hz for each of the three neurons in every spike train). (a) shows the percent correctly classified and (b) the false positive rate.

expect a smaller spike to be identified as a larger template) is because as the vectors in multidimensional space are extended, the distance between them increases. The false positives for the NN are due to the network confusing the larger waveforms as superpositions. As a result, the percent classified correctly does not drop as rapidly as the MTF does.

C. Asynchronous Sampling

The digitized waveform of a spike includes a variable offset (between $\pm 0.5\times$ sampling interval) because the spikes occur randomly with respect to the periodic sampling, i.e., they are not phase locked to the A/D conversion of the data. This alignment problem can be overcome by sampling at a higher rate, or by interpolating the waveform (provided the sampling rate meets the Nyquist criterion). McGill and Dorfman [42] presented an iterative algorithm which overcomes this problem in the frequency domain. An iterative approach is not suitable for on-line real-time implementations. Thus, we performed tests to investigate the robustness of the NN's ability to detect and classify spikes with variable alignment. The effective sampling rate of the data was increased by up to a factor of ten by Nyquist interpolation (using the sinc function) [43]. The network was trained on the original templates (i.e., no interpolation). The test data sets consisted of templates that had been interpolated, shifted, and then subsampled (back to a rate of 24 KHz) at different phases to simulate asynchronous sampling. This models the real scenario, where one set of templates is used to classify asynchronously sampled data. Under these conditions, the network was tested with the first two simulation types at an SNR of 5.0. The observed performance dropped marginally because the network would occasionally classify the event one sample before or after the actual time of the event (an error of 1/24th of a ms). If we allow for a one sample error, the classification remains unchanged as the correct class was associated with every event, albeit with a possible one sample error.

We also tested the system at higher sampling rates to determine if a sampling rate of 24 KHz is sufficient to overcome the problem of time quantization. The performance of the network improved marginally ($<2.0\%$) for sampling rate of 48 and 72 KHz. This shows that a sampling rate of 24 KHz is sufficient for the NN to overcome the problem of asynchronous sampling. The performance actually deteriorated for sampling rates higher than 72 KHz because with the large increase in training set size, a network with eight hidden units could not converge to a solution. A larger network with more hidden units would be needed to utilize the benefits of such a training set. If the sampling rate and buffer size are simultaneously increased, the effect of asynchronous sampling would be reduced and classification performance should increase. However, only a marginal improvement in performance would be expected for a major increase in the training time and run time of the larger network.

IV. Discussion

Many investigators have applied various signal processing techniques to the challenge of multiunit spike sorting. However, accurate superposition resolution on-line has only been addressed by a few investigators. Discrimination techniques range from template matching to statistical descriptions of multiunit activity. Generally, methods which aim to resolve superpositions require multichannel recordings (linear optimal filters) or are computationally intensive and would be difficult to implement in an on-line real-time system without significant computational resources. Linear optimal filters provide theoretically optimal methods for spike classification and superposition resolution, but are dependent upon multichannel recordings with each unit being active in at least two channels. For superposition resolution from single channel recordings, a nonlinear system is required. The method we present uses a fully connected feed-forward network as a nonlinear FIR filter

which was implemented in an on-line and real-time system. This provides the significant benefit of providing immediate feedback on unit activity to the experiment. Although this class of ANN's has been applied previously, all applications have been used to isolate single units only. We have extended that work to include superposition resolution.

In this method, a novel training scheme was implemented so that the NN would be able to identify single spikes and their superpositions in a multiunit single-channel extracellular recording. The advantage of using an ANN lies in it's ability to generate nonlinear decision boundaries. The three novel components in this implementation of an ANN were: 1) weighted error training on single templates and superimposed templates; 2) neural noise modeling to provide long segments of pure noise; and 3) the use of the network to both detect and classify spikes in real time.

The network was tested on three sets of simulated data under various conditions and it's performance was compared to that of an MTF. The simulated data sets were constructed using physiological components and signal and noise were modeled to match the spectral characteristics of real data. The effectiveness of the NN in classifying single units in a multiunit recording was comparable to the MTF and importantly, the network also resolved most superpositions with overlap less than 0.5 ms. This capability can prove to be especially useful in discrimination during bursts of a population of neurons, when determining correlations among neurons is critical. Our simulations showed that the network was also significantly better at rejecting false positives at low SNR, which is imperative for accurate spike train determination.

With the use of a high-speed digital signal processor and an IBM-PC, an automated real-time system based on the NN was implemented without the need for off-line post-processing. This approach satisfied our three criteria of single unit isolation, superposition resolution and real-time processing. The only off-line processing required for the system was template generation and NN training. This typically is performed in under 2 min. This system has already been installed in six of our neurophysiological laboratories and the training time has been acceptable. The DSP speed in this implementation sets the limit on the sampling rate, network size and the number of units that could be sorted in real time. With our current hardware, the maximum sampling rate that can be used without reducing the size of the network is 24 KHz. It is anticipated that higher sampling rates with a larger input buffer and a larger network will improve the performance of the discriminator; to achieve this, faster processors are required. A significant advantage of NN's that can be exploited by employing multiple DSP's is their highly parallel architecture. With distributed processing, the algorithm can be readily scaled to identify more units from one electrode or to process data from multiple electrodes.

A limitation in our system is that once trained, the network is fixed. If the template shape changes significantly the network must be retrained. Amplitude and shape variation is a problem during high-frequency bursts and during discrimination for extended periods of time. As shown in Fig. 7, the NN is not a robust discriminator in the presence of amplitude variation. Future work will be needed to find ways for the NN to track amplitude changes, and to adapt to slowly changing template shapes caused by electrode movement.

APPENDIX

The *simultaneous clustering algorithm* [34]–[36] used for template generation first finds the best clusters around every waveform, groups these initial clusters together and then selects the best and final clusters. Since this is a simultaneous technique, each detected waveform is a potential initiator waveform for a cluster. Waveforms are clustered with the initiator waveform based on the best alignment and Euclidean distance. This results in M clusters for M waveforms detected and naturally there is cluster overlap in multidimensional feature space. Clusters are automatically selected based on inter cluster distance, cluster density and a cluster scatter measure (mean squared Euclidean distance of waveforms around the cluster centroid). The centroids of the selected clusters are then the templates. The steps of the algorithm are as follows.

1) All M waveforms whose peaks exceed three standard deviations of noise (amplitude detection) are located and are centered about their point of maximum slope in a window of 24 samples.
2) Each waveform, $x(i)\,(i = 1, \cdots, M)$ is represented by $2N + 1$ alternate segmentations, $x(i, j)$ $(j = -N, \cdots, N)$, one sample apart, around the detection location $x(i, 0)$. Each alternate is used for clustering.
3) Each waveform $x(i, 0)$ initiates a cluster i containing other waveforms $x(k, 0)$ that satisfy $d[x(i, 0), x(k, 0)] < d[x(i, 0), x(k, j)]$ for $j = -N, \cdots, N$ and, $d[x(i, 0), x(k, 0)] <$ threshold where $d[a, b]$ is the Euclidean distance between vectors a and b and the threshold is determined from a pure noise segment.
4) The centroid $c(i)$ and the mean squared scatter measure $s(i)$ of each cluster containing at least five waveforms are computed $(i = 1, \cdots, M)$.
5) Cluster centroids are grouped together based on inter cluster distance. Then one centroid is chosen from each group by systematically rejecting the others. A centroid $c(i)$ is discarded if $s(i) > s(k)$ for all k such that $d[c(i), c(k)] <$ threshold.
6) The centroids that are retained $c(i)$ $(i = 1, \cdots, L)$ are used as initial estimates of the templates. Each initial template is updated by averaging with alternates $x(k, j)$ that did not contribute to generating any template if $\min\{d[c(i), x(k, j)]\} <$ threshold.

ACKNOWLEDGMENT

The authors would like to thank M. H. Cohen for his contributions to an earlier implementation of the real-time system, and S. Eifuku and H. Aizawa of the National Eye Institute for help in acquiring the neurophysiological data.

REFERENCES

[1] I. N. Bankman, K. O. Johnson, and W. Schneider, "Optimal recognition of neural waveforms," in *Ann. Int. Conf. IEEE Eng. Med. Bio. Soc.*, 1991, vol. 13, no. 1, pp. 409–410.

[2] E. M. Schmidt, "Computer separation of multiunit neuroelectric data: A review," *J. Neurosci. Meth.*, vol. 12, pp. 95–111, 1984.
[3] B. C. Wheeler and W. J. Heetderks, "A comparison of techniques for classification of multiple neural signals," *IEEE Trans. Biomed. Eng.*, vol. BME-29, no. 12, pp. 752–759, 1982.
[4] E. H. D'Hollander and G. A. Orban, "Spike recognition and on-line classification by unsupervised learning system," *IEEE Trans. Biomed. Eng.*, vol. BME-26, no. 5, pp. 279–284, 1979.
[5] M. Okada and N. Maruyama, "Software system for real-time discrimination of multiunit nerve impulses," *Programs Biomed.*, vol. 14, pp. 157–164, 1981.
[6] F. Worgotter, W. J. Daunicht, and R. Eckmiller, "An on-line spike form discriminator for extracellular recordings based on an analog correlation technique," *J. Neurosci. Meth.*, vol. 17, pp. 141–151, 1986.
[7] M. Salganicoff, M. Sarna, L. Sax, and G. L. Gerstein, "Unsupervised waveform classification for multineuron recordings: A real-time software based system, I. Algorithms and implementation," *J. Neurosci. Meth.*, vol. 25, pp. 181–187, 1988.
[8] X. Yang and S. A. Shamma, "A totally automated system for the detection and classification of neural spikes," *IEEE Trans. Biomed. Eng.*, vol. 35, no. 10, pp. 806–816, 1988.
[9] C. Forster and H. O. Handwerker, "Automatic classification and analysis of microneurographic spike data using a PC/AT," *J. Neurosci. Meth.*, vol. 31, pp. 109–118, 1990.
[10] E. V. Goodall and K. W. Horch, "Separation of action potentials in multiunit intrafascicular recordings," *IEEE Trans. Biomed. Eng.*, vol. 39, no. 3, pp. 289–295, 1992.
[11] H. Bergman and M. R. DeLong, "A personal-computer-based spike detector and sorter: Implementation and evaluation," *J. Neurosci. Meth.*, vol. 41, pp. 187–197, 1992.
[12] R. F. Jansen and A. Ter Maat, "Automatic wave form classification of extracellular multineuron recordings," *J. Neurosci. Meth.*, vol. 42, pp. 123–132, 1992.
[13] I. N. Bankman, K. O. Johnshon, and W. Schneider, "Optimal detection, classification, and superposition resolution in neural waveform recordings," *IEEE Trans. Biomed. Eng.*, vol. 40, no. 3, pp. 836–841, 1993.
[14] A. K. Kreiter, Ad. M. H. J. Aertsen, and G. L. Gerstein, "A low-cost single board solution for real-time unsupervised waveform classification of multinueron recordings," *J. Neurosci. Meth.*, vol. 30, pp. 59–69, 1989.
[15] J. F. Vilbert, J. N. Albert, and J. Costa, "Intelligent software for spike separation in multiunit recordings," *Med. Biol. Eng. Comput.*, vol. 25, pp. 366–372, 1987.
[16] M. Abeles and M. H. Goldstein, "Multispike train analysis," *Proc. IEEE*, vol. 65, no. 5, pp. 752–773, 1977.
[17] W. M. Roberts and D. K. Hartline, "Separation of multiunit nerve impulse trains by a multichannel linear filter algorithm," *Brain Res.*, vol. 94, pp. 141–149, 1975.
[18] M. N. Oguztoreli and R. B. Stein, "Optimal filtering of nerve signals," *Biological Cybern.*, vol. 27, pp. 41–48, 1977.
[19] W. M. Roberts, "Optimal recognition of neuronal waveforms," *Biological Cybern.*, vol. 35, pp. 73–80, 1979.
[20] R. B. Stein, S. Andreassen, and M. N. Oguztoreli, "Mathematical analysis of optimal multichannel filtering for nerve signals," *Biological Cybern.*, vol. 32, pp. 19–24, 1979.
[21] S. N. Gozani and J. P. Miller, "Optimal discrimination and classification of neuronal action potential waveforms from multiunit, multichannel recordings using software-based linear filters," *IEEE Trans. Biomed. Eng.*, vol. 41, no. 4, pp. 358–372, 1994.
[22] A. F. Atiya, "Recognition of multiunit neural signals," *IEEE Trans. Biomed. Eng.*, vol. 39, pp. 723–729, 1992.
[23] M. S. Lewicki, "Bayesian modeling and classification of neural signals," *Neural Computation*, vol. 6, pp. 1005–1030, 1994.
[24] R. F. Jansen, "The reconstruction of individual spike trains from extracellular multineuron recordings using a neural network emulation program," *J. Neurosci. Meth.*, vol. 35, pp. 203–213, 1990.
[25] S. Yamada, H. Kage, M. Nakashima, S. Shiono, and M. Maeda, "Data processing for multichannel optical recordings: Action potential detection by neural network," *J. Neurosci. Meth.*, vol. 43, pp. 23–33, 1992.
[26] K. Mirfakhraei and K. Horch, "Classification of action potentials in multiunit intrafascicular recordings using neural network pattern-recognition techniques," *IEEE Trans. Biomed. Eng.*, vol. 41, no. 1, pp. 89–91, 1994.
[27] J. S. Oghalai, W. Nick Street, and W. S. Rhode, "A neural network-based spike discriminator," *J. Neursoci. Meth.*, vol. 54, pp. 9–22, 1994.
[28] Y. Wong, J. Bank, and J. M. Bower, "Neural networks for template matching: Application to real-time classification of the action potentials of real neurons," in *Neural Information Processing Systems*, D. Anderson, Ed. New York: AIP, 1988, pp. 103–113.
[29] M. H. Hassoun, C. Wang, and A. R. Spitzer, "NNERVE: Neural network extraction of repetitive vectors for electromyography—Pt. I: Algorithm," *IEEE Trans. Biomed. Eng.*, vol. 41, no. 11, pp. 1039–1051, 1994.
[30] ———, "NNERVE: Neural network extraction of repetitive vectors for electromyography—Part II: Performance analysis," *IEEE Trans. Biomed. Eng.*, vol. 41, no. 11, pp. 1053–1061, 1994.
[31] C. J. Duffy and R. H. Wurtz, "Sensitivity of MST neurons to optic flow stimuli. I. A continuum of response selectivity to large-field stimuli," *J. Neurophysiol.*, vol. 65, pp. 1329–1345, 1991.
[32] W. J. Heetderks, "Criteria for multiunit spike separation techniques," *Biological Cybern.*, vol. 29, pp. 215–220, 1978.
[33] I. N. Bankman and A. Menkes, "Automated segmentation of neural recordings for optimal recognition of neural waveforms," in *Proc. 14th Annu. Conf. IEEE Eng. Med. Biol. Soc.*, 1992, vol. 14, pp. 2560–2561.
[34] R. Chandra, "Template generation for neural waveforms in extracellular recordings," Masters thesis submitted to the Johns Hopkins Univ, Baltimore, MD, June 1994.
[35] R. Chandra and I. N. Bankman, "Template generation of neural waveforms in extracellular recordings," in *Proc. 13th Southern Biomed. Eng. Conf.*, 1994, vol. I, pp. 893–897.
[36] I. N. Bankman and R. Chandra, "Template generation for neural waveform analysis in extracellular recordings," in *Proc. 16th Annu. Int. Conf., IEEE Eng. Med. Biol. Soc.*, 1994, vol. I, pp. 712–713.
[37] Q. Zhang, "Adaptive equalization using the backpropagation algorithm," *IEEE Trans. Circuits Syst.*, vol. 37, no. 6, pp. 848–849, 1990.
[38] A. D. Back and A. C. Tsoi, "FIR and IIR synapses, a new neural network architecture for time series modeling," *Neural Computation*, vol. 3, pp. 375–385, 1991.
[39] D. E. Rummelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.
[40] W. H. Press, S. A. Teukolsky, W. T. Vettering, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. New York: Cambridge Univ., 1992, pp. 288–290.
[41] R. W. Hamming, *Digital Filters*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1983.
[42] K. C. McGill and L. J. Dorfman, "High-resolution alignment of sampled waveforms," *IEEE Trans. Biomed. Eng.*, vol. BME-31, no. 6, pp. 462–468, 1984.
[43] L. B. Jackson, *Digital Filters and Signal Processing.* Norwell, MA: Kluwer, 1986.

Rishi Chandra was born in Bihar, India, on September 9, 1970. He received the B.S. and M.S.E degrees in biomedical engineering from Johns Hopkins University, Baltimore, MD, in 1992 and 1994, respectively.

Since September 1994, he has been working at the National Eye Institute, Bethesda, MD. His research interests include biological signal processing, neural networks, and adaptive filter theory. His work at the Eye Institute has been primarily focussed on developing computerized instrumentation for automated biological signal processing.

Lance M. Optican was born in Denver, CO, on May 16, 1950. He received the B.Sc. degree in 1972 from Caltech, Pasadena, CA, in physics and biosystems analysis, and the Ph.D. degree in 1978 from Johns Hopkins Medical School, Baltimore, MD, in biomedical engineering.

He has been a Research Biomedical Engineer at the National Eye Institute, Bethesda, MD, since 1982. In 1988 he became Chief of the Section on Neural Modeling in the Laboratory of Sensorimotor Research. His research interests include modeling of oculomotor control systems; study of adaptive brain mechanisms; analysis and modeling of neuronal mechanisms underlying visual perception, pattern recognition, and memory; and the application of information theory, estimation theory, systems theory, and neural networks to the analysis of neurophysiological data.