Robust acoustic object detection

Yali Amit,[a] Alexey Koloydenko,[b] and Partha Niyogi[c]
Departments of Computer Science and Statistics, The University of Chicago, Hyde Park, Chicago, Illinois 60637

(Received 14 March 2002; revised 19 July 2004; accepted 24 May 2005)

We consider a novel approach to the problem of detecting phonological objects like phonemes, syllables, or words, directly from the speech signal. We begin by defining *local* features in the time-frequency plane with built in robustness to intensity variations and time warping. Global templates of phonological objects correspond to the coincidence in time and frequency of patterns of the local features. These global templates are constructed by using the statistics of the local features in a principled way. The templates have clear phonetic interpretability, are easily adaptable, have built in invariances, and display considerable robustness in the face of additive noise and clutter from competing speakers. We provide a detailed evaluation of the performance of some diphone detectors and a word detector based on this approach. We also perform some phonetic classification experiments based on the edge-based features suggested here. © *2005 Acoustical Society of America.* [DOI: 10.1121/1.2011411]

PACS number(s): 43.72.Ne, 43.72.Ar [DDO] Pages: 2634–2648

I. INTRODUCTION

We consider the problem of detecting phonological objects from the speech signal. Humans are able to accomplish this task reliably and robustly. In spite of significant progress in automatic speech recognition over the years, robustness still appears to be a stumbling block. Current commercial products are quite sensitive to changes in recording device, to acoustic clutter in the form of additional speech signals, and so on. The goal of replicating human performance in a machine remains far from sight.

By detection we are referring to the identification of time points at which a specific predefined object or class of objects is found. Detection is essentially a two class classification problem—object versus background or nonobject. In this sense it is a simpler problem than multiclass classification which requires more complex boundaries in the representation space. In detection the two classes are not treated symmetrically as in classification. Typically one aims for *low* false negative rates (missed detections of the selected class), at the expense of a higher false alarm rate (the rate at which background is labeled as object). Detection may serve a limited purpose such as word spotting. However it can also be viewed as a building block in a speech recognition algorithm. In the context of speech, trying to faithfully classify every time segment as one particular phoneme is recognized as a very difficult problem. Instead each phoneme detector separately flags time points where that phoneme may be present. Some time instants may be labeled with multiple phonemes, and clearly many of the labelings will be wrong. This results in a transformation of the data into a labeled point process which would serve as input to higher level algorithms. The advantage of such an approach is that training detectors, based on very simple and parsimonious statistical models, require much smaller data sets and are much less sensitive to noise not encountered during training. The final disambiguation would be left for the higher level algorithms that employ context, knowledge of vocabulary, and syntax. Assuming the false negative rates are low, the efficiency and accuracy of the next level depends on the false positive rate. The main purpose of this paper is to demonstrate that it is possible to produce detectors that are robust to a variety of degradation, are easily trained, and efficient to compute at the price of a relatively low false positive rate.

Since the pioneering work of Harvey Fletcher [(1995); see a recent interpretation by Allen (1994)] speech perception experiments have suggested that the acoustic correlates of linguistic categories are locally distributed in the time-frequency plane and irrelevant parts may be perturbed leaving recognition intact. Some speech recognition models try to exploit this fact. For example, subband based models of recognition (Tibrewala and Hermansky, 1997; Bourlard and Dupont, 1997; Saul *et al.*, 2001) attempt to construct separate detectors/recognizers in each of several frequency subbands that are then combined to yield a global recognizer. Since the individual recognizers in the ensemble are based on only local frequency subband information, they are naturally poorer and the consequences of having an ensemble of several poor recognizers need to be better understood.

Our approach gives a different computational expression to some of the ideas presented in Fletcher (1995) and Allen (1994) where global templates are made from the coincidence of binary features that are local both in time and frequency. These features are computed through adaptive thresholding of simple difference linear filters with very small local support in the time-frequency plane. Borrowing from vision terminology, we employ local oriented edges in the time frequency plane.

There are several important advantages to constructing models based on simple binary local features. First we obtain

[a]Electronic mail: amit@galton.uchicago.edu
[b]Electronic mail: koloyden@cs.uchicago.edu
[c]Electronic mail: niyogi@cs.uchicago.edu

invariance to local amplitude modulations as well as a simple mechanism to filter out moderate levels of acoustic clutter and noise. Second by "spreading" the detected binary features either along the time axis or the frequency axis, we easily introduce invariance to a considerable range of linear and nonlinear variations in the duration of components of the acoustic object, as well as variations in the basic formant frequencies across individuals. The use of global templates composed of large numbers of these "spread" features introduces robustness to occlusion or degradation of part of the signal. Thus, one of the primary advantages of this approach is having the robustness hardwired in the detection algorithm. Furthermore, since the only quantities estimated in training are the frequency of occurrence of each of the local features, templates produced with very small training sets generalize well. Finally since the actual detection involves a sequence of simple binary template matches it can be computed very efficiently and lends itself easily to parallel implementations with neural network type architectures.

An additional motivation for employing such models has been the success of similar approaches in visual detection tasks, see for example Grimson (1990), Ullman (1996), Amit (2000), Fleuret and Geman (2001), Viola and Jones (2002). Object detectors in gray level images are constructed from templates based on "spread" oriented edges or conjunctions of oriented edges, yielding very efficient detection algorithms, all produced with very small training sets. The idea of importing methods from computer vision to the speech domain can also be found in some earlier work, e.g., Leung and Zue (1986) or Riley (1989), where computer vision techniques are used to detect complex time-frequency features though the details are quite different.

The use of edge based representations in vision has been fueled in part by the seminal work of Hubel and Wiesel (1968) and subsequent neurophysiological investigations. Recent work on the auditory cortex of animal species suggests the existence of neurons that fire selectively when oriented "acoustic edges" in the time-frequency plane are presented (Kowalski *et al.*, 1996; Theunissen and Doupe, 1998; Sen *et al.*, 2001). While it is still unclear what role such edge detectors play in speech perception, it is certainly worthwhile to understand more fully the statistics of speech sounds in representational spaces constructed from these edge maps. There have been very few detailed studies in this direction although Schwartz and Simoncelli (2001) present preliminary steps. Most studies of speech have tended to use vector quantization on the continuous valued vectors of spectral or cepstral coefficients, and it is in these representational spaces that acoustic phonetic insights have primarily been developed.

It is worthwhile to note that formants correspond to local maxima in the time-frequency plane and in this sense constitute a form of local feature that has received considerable attention in speech production and perception studies. As we shall see, the edge based representations considered in this paper are strongly related with the formant structure in the speech signal. For example, the template corresponding to a stop consonant may be interpreted as a series of sharp changes in each of several frequencies simultaneously. The templates corresponding to phonetic categories with strong formant structure (for example, in most sonorant regions of the signal) ultimately "learn" to represent such structure. In Fig. 4, we show how the emergent templates resemble the formant patterns. Indeed the templates derived for the various acoustic objects are strikingly similar to the classical templates shown in phonetics texts. We see this acoustic-phonetic interpretability as a significant strength of our approach. We note that regions of great spectral change seem to have a certain kind of perceptual saliency and play an important role in landmark based approaches to speech recognition (Stevens, 1991; Liu, 1996) as well as the approach to stop detection pursued in Niyogi and Sondhi (2002).

The approach described here is clearly statistical in nature. However it marks a departure from the usual statistics based models of recognition at several levels. For one, an unusually large number of highly local acoustic properties are measured. Second, the global templates that are constructed may be interpreted as a rather sparse representation of the time-frequency plane that are correlated with phonetic content. This sparse nature of the modeling suggests that one does not need to account for the *entire* signal but only informationally significant portions of it. We note that similar types of sparse representations were used in Amit and Murua (2001) for robust recognition of isolated spoken digits using relational decision trees.

The work in Hopfield *et al.* (1998) bears some similarity with our approach to acoustic detection. There the binary local features are defined as local maxima in time of the spectrogram at different frequencies, and invariance to multiplicative time stretching is obtained by taking logarithms of the time coordinates. In the context of syllable detection in bird songs the work in Chi and Margoliash (2001) also makes use of local maxima and invariance is achieved by "spreading." It therefore seems that there is much promise in using predesigned binary local features, both in achieving robustness to noise and clutter and in providing a straightforward way to incorporate invariance to nonlinear warping in time and frequency.

We should emphasize that detection per-se is not a solution to the continuous speech problem. One will ultimately need to recognize words. One approach to this might be to make word detectors—a possibility we briefly describe later in the paper. However, even if we could produce very accurate word detectors it is not computationally feasible to detect each word from even a moderately sized vocabulary in order to parse the entire speech signal. In Sec. VII we outline several possible ways in which detection may be integrated into a continuous speech algorithm including as a higher level entry into an HMM.

The rest of the paper is organized as follows. In Sec. II, we describe the binary features, and formulate a statistical model for the features on object and on background, yielding a classifier for object versus background. In Sec. III we describe an efficient two stage detection algorithm for implementing the classifier at every time instant. The training procedure is presented in Sec. IV. The experimental results on a number of acoustic objects—a phoneme, some diphones, and a word—are presented in Sec. V. Here we study different

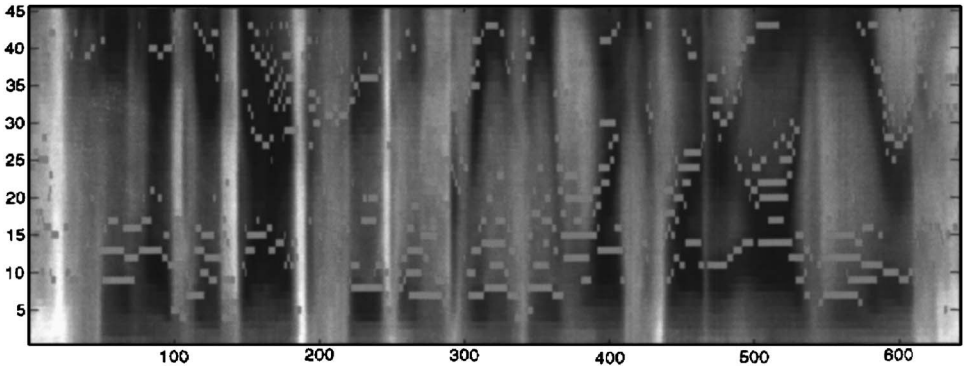

FIG. 1. Edge maps. Locations of edges $Y_{(0,-1)}$ on a spectrogram, detecting transitions from high to low energy in the frequency dimension.

aspects of the detectors constructed. We examine their accuracy in terms of ROC curves as well as where in the duration of each phonological segment the detector peak is usually obtained. We examine the robustness of the detector by considering the effect of various kinds of noise and clutter. We analyze the confusion caused by some of the diphone detectors and try to understand if there are any phonetic regularities in such confusion. We also compare the proposed detector to an idealized, nearest neighbor baseline classifier. We show that our detector is less sensitive to training set sizes and noise than the baseline. The experiments are conducted on TIMIT—an acoustic phonetic database of 630 different speakers. In Sec. VI we perform some preliminary phonetic classification experiments to give the reader a sense of what one might expect when these edge-based features are used for such a task. Finally, in Sec. VII we discuss further directions of investigation within the proposed framework.

II. MODELS FOR ACOUSTIC OBJECTS

A. Robust local binary features

Let $W(t,f)$ denote the windowed Fourier transform of the acoustic signal, at time t and frequency f. In our experiments we use a 20 ms window moved every 5 ms. These values were empirically chosen to provide an appropriate trade-off between fine spectral detail and smoothing on which the local edge detection process worked reliably. Preprocessing consists of taking the log of $|W(t,f)|$, and smoothing the result both in time and in frequency. The smoothing kernel K_G is a 7-pixel-long discrete Gaussian with standard deviation of 1 pixel. (One pixel corresponds to 31.25 Hz in the frequency dimension and 5 ms in the temporal dimension.) The result is subsampled at every other frequency unit, followed by a 3 kHz frequency cutoff since most of the formant activity resides in this region. Thus, we define the *spectrogram* as $S(t,f)=2((\log(|W|))*K_G)(t,2f)$, $f=1,\ldots,F$. A fragment of $S(t,f)$ ($F=45\approx 2.8$ kHz) is shown in the left image of Fig. 2 and corresponds to the diphone "aa-r."

The detection algorithm is not based on the continuous valued spectrogram, but rather on local binary features extracted from the spectrogram, that capture transitions, and are analogous to oriented edge detectors used in computer vision. As shown in the following, such features offer a straightforward framework for incorporating invariance or robustness to amplitude variations and time warping. We also note that neurons in the primary auditory cortex are known to respond to transitions in the energy in the time-frequency plane (Kowalski *et al.*, 1996; Sen *et al.*, 2001; Theunissen and Doupe, 1998).

Eight orientations $v=(\delta t,\delta f)$ where

$$(\delta t,\delta f)\in\{(1,0),(1,1),(0,1),(-1,1),(-1,0),(-1,-1),\times(0,-1),(1,-1)\}$$

are defined, corresponding to each multiple of 45°. An edge of orientation v at point $x=(t,f)$ is a local maximum of the derivative of the spectrogram in the direction of v,

$$S(x+v)-S(x)\geqslant \max\{S(x)-S(x-v),S(x+2v)-S(x+v),\tau_\alpha\},\quad(1)$$

where τ_α is an adaptable local threshold, used to eliminate very small transitions. Specifically, all local differences $S(x+v)-S(x)$ in a region are computed and the αth percentile τ_α of the positive subsample is determined. (If no positive difference is found then $\tau_\alpha=+\infty$. In our experiments $\alpha=70\%$.) We write $Y_v(x)=Y_v(t,f)=1$ if condition (1) is satisfied, otherwise $Y_v(x)=0$. In Fig. 1 we show the locations of all edges in the frequency direction $v=(0,-1)$. Note how the features pick up part of the formant structure.

B. Invariance

By definition, since the inequalities are preserved, these local features are invariant to affine transformations of S and are robust to a wide range of smooth monotone transformations.

Robustness to time warping and frequency variations is obtained by "spreading" each detected feature to a neighborhood of the original location. For example, an edge of type v with $v=(\pm 1,0)$, corresponding to a local maximum of the time derivative, will be assigned to all locations (s,f) in the strip $(t-\Delta,t+\Delta)$. Other edges are spread the same way in the direction of v. Thus we define

$$\tilde{Y}_v(x)=\max_{-\Delta\leqslant j\leqslant\Delta} Y_v(x+jv).\quad(2)$$

The right panel in Fig. 2 is a spread version of the edge map shown in the middle panel, corresponding to transitions from high to low amplitude along the frequency axis. Note how the raw edge map before the spreading operation (middle panel) captures local structure such as pitch. This is smoothed out and the edge map in the right panel captures the overall spectral resonance profile (formant profile). In

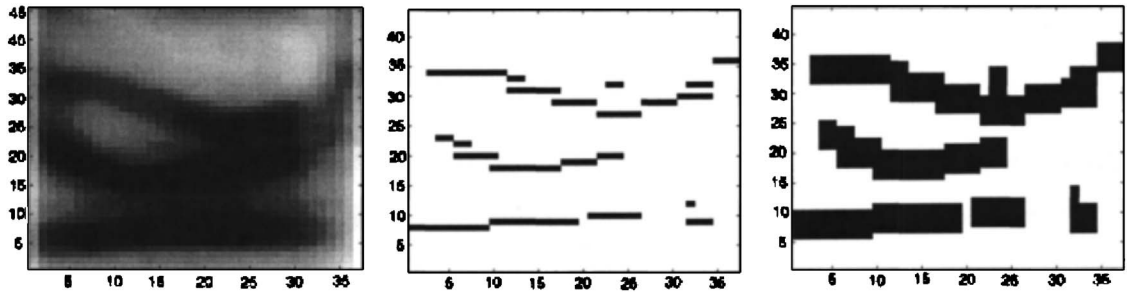

FIG. 2. The spreading operation. Left panel: the spectrogram $S(t,f)$. Middle panel: the extracted edges $Y_{(0,-1)}$ (same orientation as in Fig. 1). Right panel: the result of spreading $\tilde{Y}_{(0,-1)}, \Delta=2$. The x axis corresponds to time (in milliseconds) and the y axis to frequency (0–3 kHz quantized into 45 bins).

general, for all speech signals with clear formant structure, the edge maps capture the formant patterns in a similar way.

Spreading can be motivated as follows. Assume a certain transition in time (an edge of type $Y_{(1,0)}$) is characteristic of the object population at approximately t time units from the beginning of the signal, and at frequency f. The probability of finding this transition at precisely (t,f) may be quite small, but with high probability the transition will be found at frequency f somewhere in a small time interval $(t-\Delta, t+\Delta)$. This is equivalent to saying that the spread feature $\tilde{Y}_{(1,0)}(t,f)=1$ with high probability. Depending on the degree of variability of the object population larger degrees of spreading can be used. However larger spreading increases the background frequency of the features possibly reducing their discriminatory power. In our experiments we use $\Delta=2$.

Since the variables are all binary, spreading, which is defined through a local maximization, becomes a simple OR-ing operation. This has proved to be a crucial element in constructing efficient invariant detectors and classifiers in the visual domain [see Amit and Geman (1999), Amit (2002)]. The original continuous valued spectrogram is now reduced to a set of binary maps indexed by frequency and time, one map for each binary feature. Alternatively one can view this as one eight dimensional vector map in time and frequency, where the vector at each point (t,f) is binary valued and indicates which features are present at that point. We note that one of the main attractions of using local features is the ability to *adapt* their parameters online as discussed in Sec. VII A.

C. The statistical model

Let T denote the average duration of the acoustic object we seek to detect and let F denote the highest frequency in the (preprocessed) spectrogram. For a given time t, let $\mathbf{Y}(\mathbf{t})$ denote the vector of binary variables $\tilde{Y}_v(t+s,f), 0 \leq s < T, 1 \leq f \leq F$ (s corresponds to time and f to frequency) for all eight directions v. Conditional on the object being present at time t (its starting time is t) we assume these variables are independent, and with marginal probability

$$p_{s,f,v,o} = P(\tilde{Y}_v(t+s,f)=1 | \textit{object at } t),$$

yielding a joint distribution

$$P(\mathbf{Y}(\mathbf{t})|\textit{object at } t) = \prod_{s,f,v} p_{s,f,v,o}^{\tilde{Y}_v(t+s,f)} \cdot (1-p_{s,f,v,o})^{(1-\tilde{Y}_v(t+s,f))}. \tag{3}$$

Conditional on the object *not* being present at time (i.e., background) t the features are again assumed independent with a *different* marginal probability

$$p_{f,v,b} = P(\tilde{Y}_v(t+s,f)=1 | \textit{background at } t),$$

that does not depend on s, assuming stationarity of the background population. This yields a background joint distribution

$$P(\mathbf{Y}(\mathbf{t})|\textit{background at } t) = \prod_{s,f,v} p_{f,v,b}^{\tilde{Y}_v(t+s,f)} \cdot (1-p_{f,v,b})^{(1-\tilde{Y}_v(t+s,f))}. \tag{4}$$

This is a very simplistic model. Clearly the features are dependent due to the spreading operation both on object and on background, and on object there are strong correlations due to variability in the time duration of the object. We ignore these aspects in the current model.

The log-likelihood ratio of object to background at time t is then

$$\begin{aligned} J(t) &= \log \frac{P(\mathbf{Y}(\mathbf{t})|\textit{object at } t)}{P(\mathbf{Y}(\mathbf{t})|\textit{background at } t)} \\ &= \sum_{s,f,v} \tilde{Y}_v(t+s,f) \log \frac{p_{s,f,v,o}(1-p_{f,v,b})}{(1-p_{s,f,v,o})p_{f,v,b}} + C \\ &= \sum_{s,v,f} \tilde{Y}_v(t+s,f) \cdot w_{s,f,v} + C, \end{aligned} \tag{5}$$

where C is a constant that does not depend on the data.

Detection proceeds by evaluating the log-likelihood ratio $J(t)$, and identifying those locations where $J(t)>\rho$ for some predetermined threshold $\rho>0$. *No time warping is performed.* This test is useful only if there are many locations s for which $p_{s,f,v,o}/p_{f,v,b}$ is either much larger or much smaller than 1. Only the location t of the object is specified in the model, whereas the length of the object can vary. Since the probabilities are attached to a specific location $t+s$, as explained earlier, they will tend to be close to the background probabilities, unless spreading is performed.

Assume for example that given the object starts at time t, for each of the five points in the interval $[t+s-2, t+s+2]$,

one-fifth of the population has an edge of type $E_{(1,0)}$ at that point. It seems reasonable however that these edges all correspond to the same "event" on the object. The *entire* population would have an edge $\tilde{Y}_{(0,1)}$ at $t+s$ if spreading with $\Delta=2$ is performed. Thus the spreading operation is a mechanism for increasing on-object probabilities. Although background probabilities increase as well, there is a range of spreading that provides significant gains in the ratio, for those cases where it is greater than 1. Moreover those locations where the ratio remains significantly less than 1, are now much more reliable.

The threshold ρ can be determined in several ways. If the conditional independence assumption were indeed appropriate, under the null hypothesis of an object at t, $J(t)$ in Eq. (5) can be approximated as a normal random variable with mean $\mu=\Sigma_{s,f,v}^{N} p_{s,f,v,o} w_{s,f,v}+C$ and variance $\sigma^2=\Sigma_{s,f,v} p_{s,f,v,o}(1-p_{s,f,v,o}) w_{s,f,v}^2$. If our goal is to minimize *false negatives*, i.e., the proportion of object missed by the detector, we could keep all instances of time for which

$$J(t) > \mu - k\sigma \equiv \rho,$$

for some choice of k. Clearly conditional independence is not a valid assumption; nearby edges are highly correlated. However as the distance between two features increases it is safe to assume that conditionally they are weakly dependent. Thus $J(t)$ still may be approximated as a normal variable but the variance would have to include a term involving covariances of pairs of variables. Finally it is possible to choose ρ from the training set, for a particular predetermined false negative rate.

Note that the detector can be viewed as a simple linear classifier (perceptron) between object and background. The weights are obtained from the probability estimates of the individual features. Directly training a perceptron for object against background data could produce more powerful weights, however there is always the danger of overfitting and we recall that one of our goals is to use small training sets. This is even more so if nonlinear classifiers are used. However under the same conditional independence assumptions as noted earlier the Fisher linear discriminant analysis yields an alternative linear classifier of the form

$$w_{s,f,v} = \frac{p_{s,f,v,o} - p_{f,v,b}}{p_{s,f,v,o}(1-p_{s,f,v,o}) + p_{f,v,b}(1-p_{f,v,b})}. \tag{6}$$

In our experiments we find that the outcome of these two classifiers is essentially the same.

It is also possible to construct more complex features, in terms of the elementary ones in such a way that p_b decreases much faster than p_o and independence becomes a more credible assumption, this has been implemented in the context of visual detection [see Amit (2000)].

III. DETECTION

Computing $J(t)$ for all t amounts to nothing more than a convolution of a filter composed of the chosen weights $w_{s,f,v}$ with the binary output $\mathbf{Y}$ obtained from computing the local features [see Eq. (5)]. However this is quite a large filter given that T can be on the order of several tens of time units and F several tens of frequency units. The computation can be quite intensive. Our solution is to perform the computation in a two stage manner.

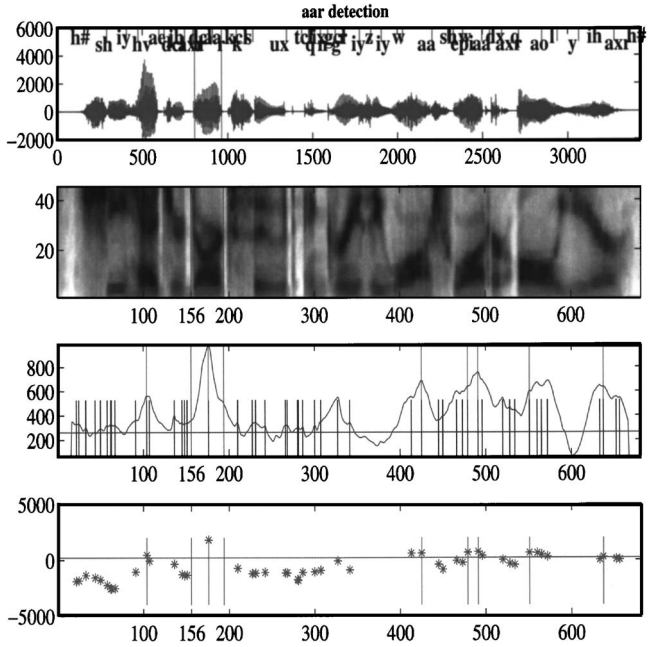

FIG. 3. Detection of **aar**. The target segment is marked with green stems. Top panel: Original wave form. Second panel: preprocessed spectrogram. Third panel: continuous J_0 response (blue curve), J_0 peaks with $J_0 \geqslant \rho_0$ (black stems). Fourth panel: J responses (blue stars), cluster centers (red stems), and empirical 0.99 threshold ρ (blue).

A. A two stage detector

First a simpler model is defined in which the weights $w_{s,f,v}$ are nonzero only on a feature set I for which $p_{s,f,v,o} \geqslant \lambda$, where λ is chosen to be higher than all the background probabilities, so that $p_{s,f,v,o} \gg p_{f,v,b}$. Furthermore we use equal weights for all the features in I yielding a sum

$$J_0(t) = \sum_{(s,f,v)\in I} \tilde{Y}_v(t+s,f), \tag{7}$$

which again will be compared to a predetermined threshold ρ_0. This sum is over a much smaller set I and only involves counts, with no multiplications, and is hence much faster to compute. Only at times t for which $J_0(t) > \rho_0$ do we compute the full model $J(t)$. Thus, our overall detection rule is $J_0(t) \geqslant \rho_0$ *and* $J(t) \geqslant \rho$ (we denote this conjunction by $J \circ J_0$) combined with a *clustering* procedure described in the following section.

In Fig. 3 we show the original wave form, the preprocessed spectrogram, and the detections obtained by $J_0(t)$ and $J(t)$.

B. Clustering

The statistics J_0 and J are essentially convolutions and therefore are inherently smooth in t. Consequently, a match between the template and a realization of the object will result in a cluster of time points $\{t_0 < t_1 < \ldots < t_L\}$ at which the detection statistic will stay above the prespecified threshold. It is then necessary to choose a small number of discrete

times to represent the cluster. We do this in two steps, treating the J_0 and J responses separately. At the J_0 stage, we simply declare $J_0(t)$ a detection if and only if $J_0(t) \geqslant \rho_0$ *and* $J_0(t)$ is a local maximum: $J_0(t-1) \leqslant J_0(t) \geqslant J_0(t+1)$. Examples of such detections are represented by the black stems in Fig. 3, the third panel from top. Note that instead of computing J at all 700 time points it is computed at only about 50 points.

Next, $J(t)$ is evaluated at all the points selected in the previous step, and we keep only those points where $J(t) \geqslant \rho$. These we cluster as follows.[1] Initially, the set of the detection points is partitioned into chains $\{t_0 < t_1 < \ldots < t_L\}$ with links $t_l - t_{l-1} < C_0$. As these resulting clusters may still extend across several object lengths, we split them further, successively cutting their longest links (i.e., t_{k-1} and t_k move apart to different clusters if $t_k - t_{k-1} \geqslant t_l - t_{l-1} l = 1, \ldots, L$) until all resulting clusters satisfy $t_L - t_0 \leqslant C_1$, for some prescribed constant C_1. This gives us clusters no longer than C_1 time units. Finally, one response $[t_k, J(t_k)]$ is reported for the kth cluster, corresponding to the maximum of J within that cluster. The two bottom images of Fig. 3 mark the clustered $J \circ J_0$ responses by red vertical lines. Presently, our constants $C_0 \approx T$ and $C_1 \approx 1.5T$ are determined empirically and thus depend on the average duration of the object (i.e., the length of the template). We also note that simple coarsening of the time scale might prove to be a sufficiently effective alternative to our clustering.

Thus after applying $J \circ J_0$ and clustering, the algorithm postulates a set of times at which the target object is thought to occur. Our experiments are conducted on a labeled data set (TIMIT) for which a phonetic segmentation is available. A detection time t_i is deemed a *correct detection* if it lies anywhere within the segmentation provided by the TIMIT labeling, otherwise it is deemed a *false alarm*. If the object class was present over a time interval and no detection occurred within that interval, then this event was deemed a *false negative*. Thus the number of false negatives and false alarms are calculated for any detection scheme. In this paper, we compare our approach with a baseline nearest neighbor classifier using the same criterion for scoring false alarms and false negatives.

C. Time invariance

Although we are performing a rigid template match, invariance to time warping of the acoustic signal is incorporated both through the spreading of the features in the data, as described earlier, and through the use of conservative thresholds. For short objects such as phonemes or diphones of average duration 80–100 ms, the variance is on the order of 10–15 ms. The spreading in time of each feature is on the order of ±8 ms (for those features corresponding to a time discontinuity in the spectrogram).

IV. TRAINING

The TIMIT set is a phonetically balanced labeled data set consisting of 6300 sentences spoken by 630 speakers, representing various US dialects. The set is partitioned into eight directories, DR1 through DR8, corresponding to the distinct dialects. The entire database is divided into a training and a test portion. Each dialect region (DR1 through DR8) is thus correspondingly split into a training and a test subset. This is the standard split of the TIMIT database and we have used this to separate training and test data for all our experiments.

A training subset of 1460 sentences is used for estimating probabilities and determining thresholds, and a test set (disjoint from the training one) has 4840 sentences on which the performance of the algorithm is evaluated. We use the diphone "aar" (as in the word "dark") to illustrate the training procedure.

All (1046) instances of "aar" are segmented from the training data, using the phonetic transcriptions provided with the TIMIT data set. The binary local feature maps are computed on each training example. A fixed interval time T_{ref} is chosen for the template reference grid, given by the average duration of the training examples. For a training example of duration T' each local feature at a location (t,f) is *registered* (aligned) to location $(tT_{\text{ref}}/T', f)$ on the reference grid. This creates a uniform time of the binary features of the different training examples. At each location on the reference grid we then count the number of local features found in the training set at that location after *registration*. The images in the left panel of Fig. 4 show these counts as brightness maps on the reference grid with highest values given in dark. There are eight images corresponding to the eight orientations of the elementary features.

For the first stage of the detection (computing J_0), a set of feature/location pairs (v_i, s_i, f_i), $i = 1, \ldots, N_I$ is chosen by picking only those with frequency (on the training set) above the threshold $\lambda = 0.5$. This produces the set of features I of Eq. (7), and is shown in the right panel of Fig. 4 as black regions. In the templates shown here there are 3729 of feature location pairs (<20% of a total of all feature location pairs.)

For the second stage the counts at each location are stored and become preliminary estimates $\tilde{p}_{s,f,v,o}$, of the (on object) probabilities $p_{s,f,v,o}$. The background probabilities are estimated similarly but assuming translation invariance. Thus, only one probability is computed for each frequency and each orientation. The background probabilities are computed only once and then used in any new detector. The final estimates $\hat{p}_{f,v,b}$ are found to be in the range 0.1–0.5. In order to keep the weight values $w_{s,f,v}$ bounded, it is important to bound the on-object estimates away from 0 and 1. We thus constrain the final estimators $(\hat{p}_{t,f,v,o})$ to be less than 0.995 and greater than 1% of the background probability of the same frequency and orientation. Specifically

$$\hat{p}_{s,f,v,o} = \min(0.995, \max(\tilde{p}_{s,f,v,o}, \epsilon \cdot \hat{p}_{f,v,b})),$$

where $\epsilon = 0.01$. The upper value of 0.995 and the lower value of $0.01 \cdot \hat{p}_{f,v,b}$ were chosen empirically and have no particular theoretical justification.

Note that learning reduces to local estimation of frequencies of features as opposed to the estimation of complex and high dimensional parameters. Thus training is very fast and requires only small data sets. We view this as a crucial property of our approach which is a major departure from the

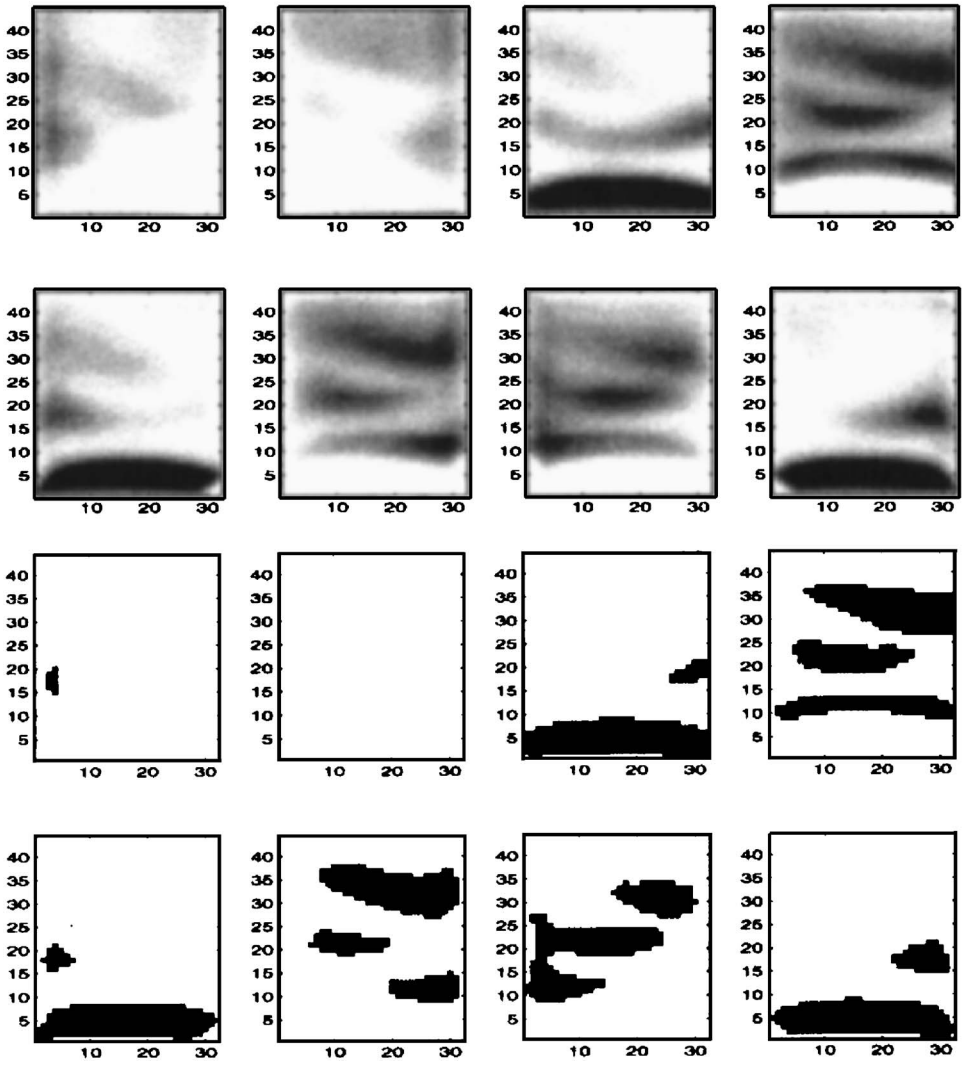

FIG. 4. Templates. Top panel: Probability templates for J—frequencies of local features of eight orientations. Bottom panel: Binary templates after thresholding by $\lambda=0.5$.

existing HMM paradigm. Note also that if templates for all objects which need to be detected are defined in terms of the same local features, the feature extraction step is carried out only once.

Thresholds. The values for the thresholds ρ_0 and ρ (for J_0 and J, respectively) are estimated using the full training sentences; the maximal on-object values of J_0 and J are recorded within the boundaries of the instance of the object in the full sentence, as given by the phonetic transcription. No registration is performed in this step. This yields a histogram of values for the two statistics. For ρ_0 we take $\hat{\mu}_0-4\hat{\sigma}_0$ where $\hat{\mu}_0$ and $\hat{\sigma}_0$ are the mean and standard deviation of J_0 on the training set. The threshold ρ is obtained using a predetermined false negative rate, which in Sec. V B is taken to be 1% on the training set.

V. EXPERIMENTS

We test our algorithm, henceforth called the edge-based detector (EBD), on phonemes, diphones, and words in clean speech and under several degradation conditions, and compare it to an idealized baseline nearest neighbor classifier (see the following), henceforth called the baseline detector (BLD). Our sources of degradation are:

(1) Additive white noise with SNR=5, 10 dB.
(2) Addition of auditory clutter in the form of a second speaker chosen at random with SNR=5, 10 dB.
(3) Addition of Babble noise (ten background speakers chosen at random) at SNR=5, 10 dB.

In all cases, SNR is calculated by using rms ratios. Thus, if $y=x+\eta$ is the corrupted speech where x is the clean speech and η is the added noise, then SNR is computed as

$$\mathrm{SNR}=10\log_{10}\frac{\langle x^2\rangle}{\langle \eta^2\rangle}=20\log_{10}\frac{\sqrt{\langle x^2\rangle}}{\sqrt{\langle \eta^2\rangle}},$$

where $\langle x^2\rangle=(1/T)\Sigma_t x(t)^2$ denotes the average value of the signal energy and $\langle \eta^2\rangle=(1/T)\Sigma_t \eta(t)^2$ denotes the average value of the noise. Table I summarizes the specifications of all the parameters involved in our experiments.

A. A baseline classifier

The baseline detector is a relatively accurate but highly inefficient and idealized nearest neighbor classifier between object and background, using the spectrogram data. Each training example of the object is segmented, using the phonetic transcription provided with the TIMIT data set, and the

TABLE I. Specifications of the model settings and parameters.

Stage	Specifications
Global	Time unit=5 ms, frequency unit=31.25 Hz
Preprocessing	Smoothing: 7×7 Gaussian, $\sigma=1$
	Frequency subsampling: step=2
Feature extraction	Adaptive threshold: $\alpha=70\%$
	Spread: $\Delta=2$ pixels
	Template dimensions: $T\times F\times8$, $F=45\approx2.8$ kHz
Detection Statistics	Estimation: $\hat{p}_{t,f,v,o}=\min(0.995,0.01\hat{p}_{f,v,b})$
	Feature selection for J_0: $I=\{(t,f,v):\hat{p}_{t,f,v,o}\geqslant\lambda=0.5\}$
	J_0-thresholding: $\rho_0=\hat{\mu}_0(J_0\|obj)-4\cdot\hat{\sigma}_0(J_0\|obj)$,
	$J\circ J_0$-clustering: C_0, minimal intercluster distance, $\approx T$
	C_1, maximal cluster length $\approx1.5T$
Sentence parsing	Partition block length $\approx T$
Robustness	White noise: SNR=5, 10; Babble noise: SNR=5, 10
	Clutter: Random speaker SNR=5, 10

spectrogram data (frequencies higher than 3 kHz being always cut off) is scaled in time to $T_{\text{ref}}\times F$, a reference grid of the same dimensions as in the EBD template construction (Sec. II A). A random sample of T_{ref}-long background data segments is obtained, scaled to exactly the same dimensions $T_{\text{ref}}\times F$.

For testing, each instance of the *object* in the test set is *segmented* and the spectrogram on that time segment is rescaled to $T_{\text{ref}}\times F$. This is then compared in sum of squares norm to the training samples of both object and background. Let d_o be the sum of the distances to the three nearest object examples, and d_b the sum of the distances to the three nearest background examples. Classification is given by

$$\frac{d_o}{d_o+d_b}\leqslant\rho_{\text{nn}},$$

in which case the instance is assigned to the "object" class. In order to produce ROC curves for this classifier, we vary $\rho_{\text{nn}}\in(0,1)$. For false alarm rates we apply the same criterion to a sample of full sentences that do not contain the object, at each time instant. Thus, at each time t a segment of duration T_{ref} is extracted, rescaled to $T_{\text{ref}}\times F$ and the above ratio is computed. A clustering mechanism similar to that used for the EBD is then applied to the output.

We reiterate that in addition to the advantage of using nearest neighbors which is known to be a very accurate, albeit inefficient, classifier this baseline also uses a manually determined segmentation and a rescaling for the object samples. This was done to ensure that the performance of the baseline detector was measured in the most favorable circumstances. *The EBD requires no segmentation or rescaling*.

B. Diphone detection

We performed experiments on a number of diphone, phoneme, and word detection tasks. These are too numerous to report so in this paper we will illustrate our general findings by considering the example of a very small number of them. These will provide the reader a sense of the various issues that arise in using this technology for detection tasks.

A detailed analysis is provided for the diphone **aar**

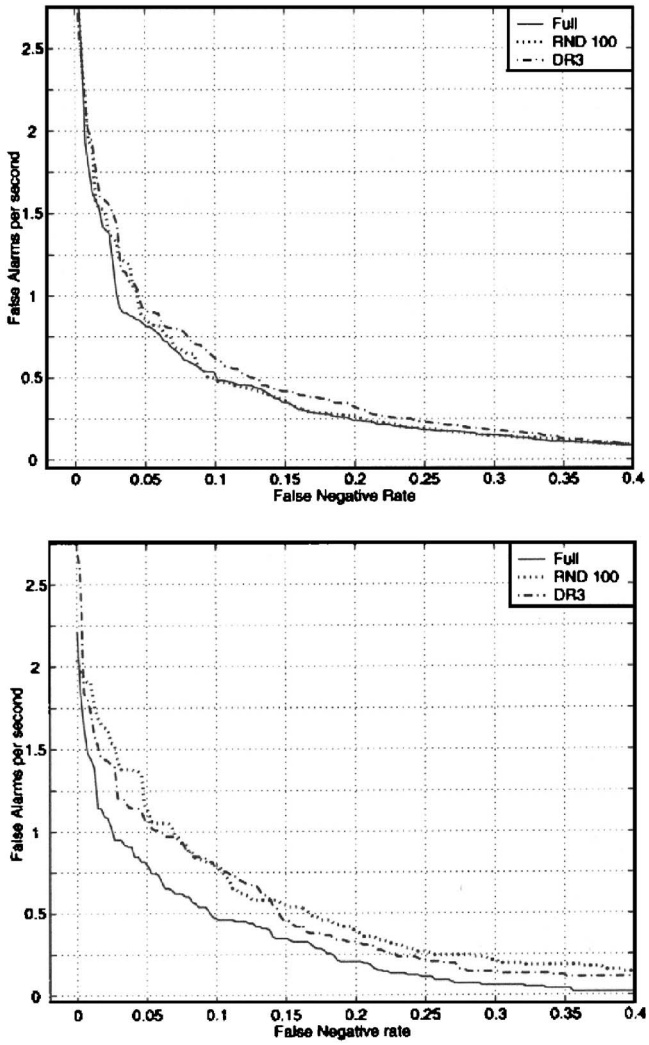

FIG. 5. Performance of **aa-r** detectors as a function of training set size: Top—EBD, Bottom—BLD.

where we show comparisons to the baseline nearest neighbors detector described earlier, and present the full ROC curves. We show that our detector is less sensitive to training set sizes and more robust than the baseline. We also study the confusions made by these detectors in general and try to see if there are any phonetic regularities in these confusions. It is worth noting that ultimately a detector for a particular phonological class does not return a segment but rather returns only a point in time around which it thinks the segment is present. We study where this point in time is located with respect to the actual boundaries of the phonological segment.

1. Dependence on training set size

In Fig. 5, left panel, we show the performance of the EBD as a function of the size of the training set for three choices: all 1046 samples from the training data, 100 random samples from the positive training data, and all 178 samples from the North Midland dialect (DR3). The background probabilities were estimated once and for all from a small sample of sentences and were found to be stable. The horizontal axis measures the false negative rate (proportion of target class phonemes not detected) and the vertical axis

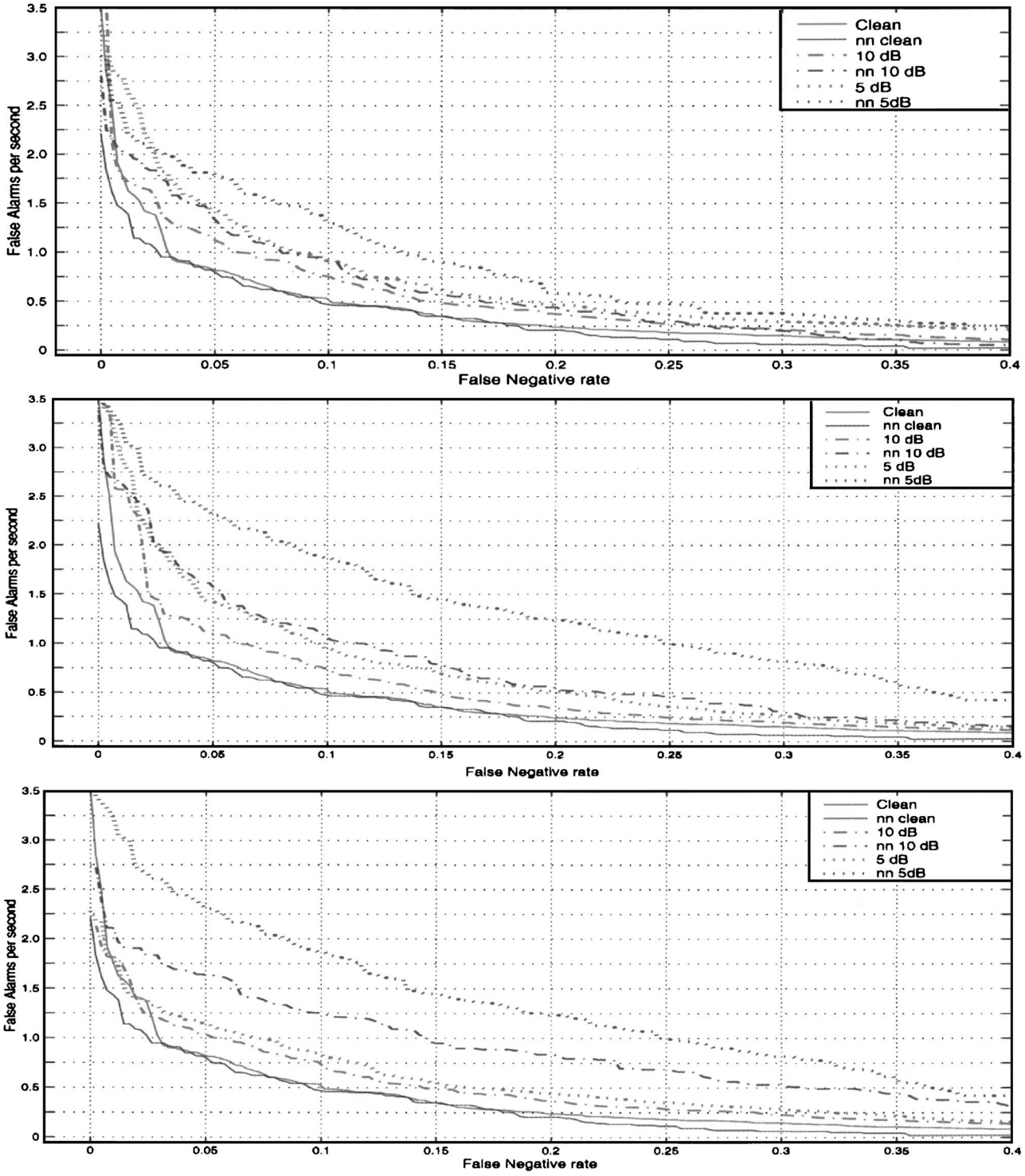

FIG. 6. Sensitivity of **aar** detector to degradation. Top: clean speech and one random background speaker at 10 and 5 dB, red—EBD, blue—BLD. Middle: clean speech and background babble noise at 10 and 5 dB, red—EBD, blue—BLD. Bottom: clean speech and additive white noise at 10 and 5 dB, red—EBD, blue—BLD.

measures the number of *false alarms* (i.e., detections not on the object) *per second*. It is not surprising that the algorithm performs essentially the same with the different training sets, since training only involves the estimation of the probabilities of binary variables, and the algorithm as a whole is not very sensitive to the accuracy of these estimates. For comparison in Fig. 5, right panel, we show the same data for the BLD. There is evidence of some degradation for the smaller training sets.

2. Sensitivity to degradation

The three panels of Fig. 6 show the ROC curves of the algorithm under various degradations. The ROC curve for clean data is shown for comparison. We observe that the performance of the EBD and that of the BLD, with the full training set and on clean data, is essentially the same. We emphasize that without the idealized setting in the baseline algorithm, where the objects are segmented and rescaled, the

TABLE II. Detection results: False negative probabilities and false alarms per second. Threshold ρ is set to first $J \circ J_0$ percentile (i.e., **0.01** false negative probability) on the training set. Each row indicates a different condition. The training set was always clean. N_{train} and N_{test} refer to the total number of instances of each object in the training and test sets, respectively (second row). The first column (for each object) shows the false negative probability and the second column shows the false alarm rate per second.

Object	**aar**		**shiy**		**sux**	
N_{tr} N_{test}	1046	413	812	299	532	191
Clean train set	**0.01**	1.717	**0.01**	1.359	**0.009**	1.451
Clean test set	0.01	1.788	0.03	1.317	0.016	1.373
White N 10 dB	0.024	1.27	0.137	0.714	0.037	1.094
White N 5 dB	0.063	1.064	0.331	0.433	0.084	0.851
Clutter 10 dB	0.005	2.169	0.124	0.902	0.115	1.135
Clutter 5 dB	0.019	2.232	0.224	0.839	0.147	1.093
Babble 10 dB	0.015	2.563	0.428	0.259	0.178	0.601
Babble 5 dB	0.012	2.795	0.659	0.129	0.44	0.373
Object	**oy**		**rae**		**dark**	
N_{tr} N_{test}	684	263	672	243	452	166
Clean train set	**0.01**	1.46	**0.01**	1.394	**0.011**	1.694
Clean test set	0.008	1.548	0.008	1.423	0.012	1.663
White N 10 dB	0.015	1.178	0.012	0.865	0.03	0.455
White N 5 dB	0.038	1.004	0.045	0.667	0.114	0.19
Clutter 10 dB	0.015	1.969	0.004	1.785	0.006	1.461
Clutter 5 dB	0.023	2.045	0.004	1.834	0.018	1.452
Babble 10 dB	0.008	2.294	0.008	1.959	0.036	0.707
Babble 5 dB	0.023	2.432	0.0	2.035	0.048	0.5

baseline algorithm would perform much worse. In all cases: additive noise, clutter and babble noise, the BLD shows significantly greater deterioration. The ROC curve of the BLD at 10 dB is comparable or worse than that of the EBD at 5 dB. This is most pronounced in the case of babble noise. It is clear that the EBD is more robust in the experiments we have performed so far.

Table II summarizes our experiments on all the objects under the various degradations. Here the threshold ρ was predetermined on the training data at a very conservative rate of 1% false negatives. Several conclusions can be drawn from the data presented in Table. II First, at low false negative rates the threshold obtained from training is quite robust and generalizes well to the clean data set as well as to the noisy ones. The main problems are with the diphones starting with a fricative. We feel that the current approach is particularly good when the underlying sounds have a well-developed formant structure that maintains itself robustly against additive noise. Since the fricative regions are noisy and lack such formant structure, the spectrograms in these regions are particularly degraded by noise and clutter. This leads to poorer performance in noisy conditions as can be seen in the examples of **shiy** and **sux**. Interestingly, we find that for sononant sounds, the false negatives go up slightly but the false alarms go up significantly. On the other hand, for fricated sounds, the false negatives go up significantly while the false alarms actually decrease.

On the whole false alarm rates are of the order of two *per second* but less in many cases. Note that the performance of **rae** appears to be particularly good: the false negative rates are always close to the prescribed 1%, while the false alarm rates stay below 2 per second. The detection of **aar**, while also exhibiting well-behaved false negatives, shows, on the other hand, false alarm rates that are consistently higher than the ones of **rae**. However, it is important to realize that the test data have more instances that are confusible with **aar** than those with **rae**.

3. Analysis of confusions

The ROC curves document overall performance for each of the detectors. Additional insight may be obtained by an analysis of the most common confusions. Indeed, the false alarms flagged by the detector are not random but rather appear to have similar phonetic structure to the target object. This is summarized in Table III, where for each of the six detectors we show ten false alarms with the highest probability of detection (skipping the ones that occur less than ten times in the test set). Note that there are between 1500 and 1800 diphone pairs and it is virtually impossible to display the entire confusion matrix but a partial view of it provided in Table III is most instructive.

From examining the confusions made by the diphone detectors (**aa-r, sh-iy, s-ux, r-ae**) we conclude that the most common false alarms are those for which the underlying diphone shares similar broad class features as the target diphone. For example, consider the **aa-r** detector. Most of the false alarms seem to be of the form **low/back-vowel-semi-vowel**. Thus the consonantal part of the diphone is one of **w,l,r,n**. Note that **w,l,r** are all semi-vowels while **n** shares with **r** the property of having an alveolar closure. Perceptually these sounds are also similar. Sometimes, the consonant is not completely released but affects the vowel so some of the false alarms consist of two vowels with a consonantal

TABLE III. Analysis of false alarms: Most frequent confusions. The numbers indicate the proportion of each class detected by a specific detector. Thus the (first row, second column) entry under **aa-r** indicates that the **aa-r** detector fired for 72% of the cases when the underlying object was an **er-ao** transition. Standard TIMIT transcription labels are used.

aa-r		**sh-iy**		**s-ux**		**oy**		**r-ae**		**dcl-d-aa-r-kcl-k**	
er-ao	0.72	sh-ey	0.66	s-ey	0.53	ao	0.81	axr-aa	0.57	l-ay-kcl-k	0.25
aw-axr	0.7	sh-ux	0.62	th-iy	0.5	aa	0.78	er-ao	0.56	dh-ae-tcl-t	0.25
ay-axr	0.69	pau-y	0.6	pau-ih	0.5	ay	0.75	aa-axr	0.55	dcl-d-ey-z	0.24
ao-w	0.67	pau-ih	0.59	f-ey	0.5	aw	0.63	axr-ay	0.54	r-ae-gcl-g	0.23
ay-l	0.66	ch-iy	0.58	sh-er	0.5	ow	0.61	ae-r	0.54	d-ow-n-tcl-t	0.22
axr-ao	0.65	s-iy	0.57	ch-iy	0.5	ah	0.42	axr-ey	0.53	l-iy-r	0.22
ae-l	0.63	s-ux	0.56	f-y	0.48	el	0.41	ay-ae	0.53	dh-ax-s	0.21
axr-ay	0.62	sh-ih	0.56	z-y	0.46	ae	0.4	f-ae	0.52	y-ih-er	0.19
nx-ao	0.6	iy-sh	0.56	sh-ux	0.46	l	0.35	r-aw	0.5	dx-axr-q	0.18
r-aw	0.59	f-iy	0.55	kcl-m	0.45	er	0.32	r-ay	0.53	dx-er-q	0.17

coloring in one of them. Thus **axr**, which is a retroflexed vowel, occurs often in the false alarm list.

Similarly, one may consider the false alarms made by the **sh-iy** detector. These seem to have the structure of **fricative-high/front vowel**. The fricatives **sh,s,f** and the affricate **ch** all share the property of frication. The two cases of **pau-y** and **pau-ih** are unreasonable errors. **pau** refers to a pause in the signal where the energy is very low but has a spectrum resembling background noise and therefore more like the fricatives superficially. This is actually one case where our amplitude invariance may have hurt us since **pau** is discriminated most by total energy rather than spectral detail.

4. Location of detector peak

Recall that the detector for a phonological class is obtained by picking a peak in the output of $J \circ J_0$. Therefore the detector provides a mapping from the speech stream to a labeled point process. Consider a detector for a phonological class X. If there is an instantiation of X in the utterance between times $t=t_1$ and $t=t_2$ (i.e., t_1 and t_2 mark the segmental boundaries of X in the acoustic realization), then the detector is deemed to fire correctly if it fires anywhere in the interval (t_1,t_2). If the detector fires correctly at $t_d \in (t_1,t_2)$, then it may be of interest to have some information about the statistics of t_d with respect to the segment (t_1,t_2).

Before we examine these statistics, it is useful to keep in mind some aspects of our detection framework. First, note that the framework is designed to detect the phonological class and not necessarily the boundaries of that class. A correct firing of the detector would lead to *one* firing somewhere within the segment (time t_d). There is no reason to expect this firing to be an estimate of either t_1 or t_2 (the segmental boundaries). Second, note that we have explicitly incorporated *spreading* of the binary features. This has been done to provide some temporal invariance to variations in speaking rate and other durational properties over the course of the utterance. While this spreading indeed provides such invariance, a consequence of this is also a resulting variability in the precise location of the detector peak. We shall see this variability in the plots that follow.

Shown in Fig. 7 are histograms of $(t_d-t_1)/(t_2-t_1)$ for a number of detectors. The quantity $(t_d-t_1)/(t_2-t_1)$ represents the distance to the left boundary of the segment normalized by the total duration of the segment. Such a normalized measure allows for more direct comparison across different

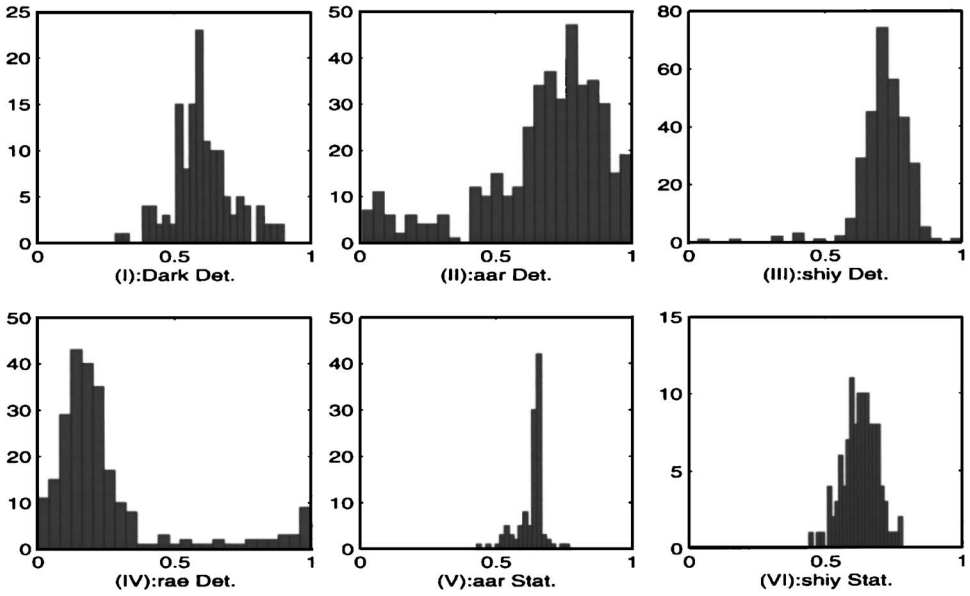

FIG. 7. Histograms indicating the statistics of detector peak locations ($(t_d-t_1)/(t_2-t_1)$-normalized distance to left segmental boundary; see the text) for a variety of detectors (panels I through IV). Panels V and VI are statistics for phonetic boundaries between **aa** and **r** (for the diphone **aar**) and **sh** and **iy** (for the diphone **shiy**), respectively.

acoustic realizations of the same segment. All detectors were operated with a 1% false alarm rate and the statistics are computed for all the correct detections obtained for each detector on the test set. Panel I shows the statistics of the detector output for the detector of the word "Dark." For this case, t_1 represents the beginnng of the **d**-closure and t_2 represents the end of the **k** burst obtained from the phonetic segmentation provided. Notice that the detector is seen to fire at a time that is mostly between 40% and 90% of the duration of the whole segment. Panels II through IV show similar detector statistics for the detectors of the diphones **aa-r, sh-iy**, and **r-ae**, respectively.

Some natural variation is observed. Thus, for example, the detector for **sh-iy** seems to fire mostly at a time around 70% of the duration of the **sh-iy** realization. The spread around this is indicated by the histogram in panel III. The detector for **r-ae** seems to fire mostly at a time around 15% into the duration of the **r-ae** segment. Both the **r-ae** and **sh-iy** detectors have similar variability. On the other hand, the detector for **aa-r** has much more variability as to where in the segment it fires (see panel II).

As we have mentioned before, it is not clear whether t_d ought to be aligned with any phonetic boundary. However, for the case of diphones, it is natural to wonder whether the detector fires usually at the boundary between the first phoneme and the second. Our analysis suggests that this need not be the case. For illustrative purposes, we show in panels V and VI, the statistics for the durations for **aa-r** and **sh-iy**, respectively. If $X=AB$ represents a diphone (phoneme A followed by phoneme B) then one can compute the statistics of duration of$A$/duration of$X$. Shown in panels V and VI are these statistics for **aa-r** and **sh-iy**, respectively. As we can see, the durational statistics for **sh - iy** is tightly clustered around 0.65 suggesting the duration of the fricative **sh** is about 65% of the duration of the whole diphone. This correlates well with the statistics of the corresponding detector output shown in panel III. On the other hand, we see that although there is considerable variation in the statistics of the detector output for the **aa-r** detector (panel II), there is much less variation in the duration of **aa** (normalized by the duration of **aar**) as shown in panel V.

In conclusion, we see that there is some variation as to where in the duration of the segment X, the detector for X will actually fire. It is not obvious whether the statistics of the detector firing will correlate with any natural phonological boundaries if X is made up of multiple phonemes. Thus our detectors should be viewed as detectors for the segment as a whole rather than any boundaries.

As a result of all the above-mentioned experiments, it might be fair to conclude that the detectors are unable to make fine phonetic distinctions but are able to work *robustly* for broad class transitions.

C. Words

In general, the approach presented in this paper can be used to detect phonological objects of arbitrary sizes like syllables, morphemes, or words. Consider for a moment the problem of making a word detector based on these ideas.

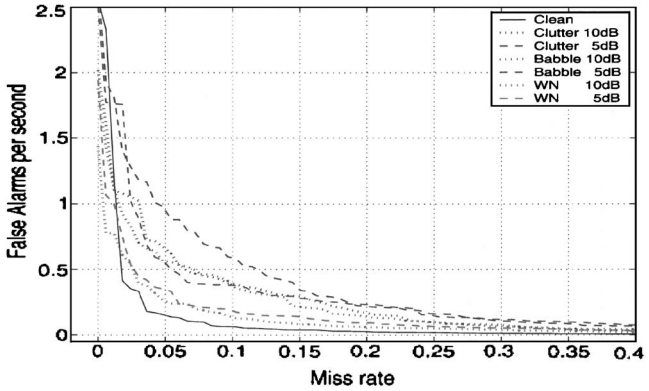

FIG. 8. Sensitivity of **dark** detector to degradation.

There are two basic approaches. One could make the word detector out of several component phoneme (or diphone) detectors and a sequential firing of each of these component detectors in the correct order would trigger the word detection. Alternatively, one could make a whole word template in much the same manner in which the diphone templates were constructed in the earlier section. A fuller investigation of the pros and cons of each of these two approaches is beyond the scope of the current paper and will be the subject of future work. However, to provide the reader with some sense of the performance of word detectors in general, we describe here the behavior of a detector for the word dark based on the whole word template approach.

The six ROC curves for degraded signals are shown in Fig. 8 against the similar curve computed for the clean signal. A detection was deemed correct if it occurred anywhere within the boundary of the word as determined from the transcription labels. If a detection occurred outside the boundary, it was deemed a false alarm. Due to the small amount of data in the test set the ROC curves are not very reliable at the very low false negative rate of under 1%–2%. We do observe that at the range of 5%–10% false negatives, the false alarm rate is several times smaller than that observed for phonemes. For example on the clean data at 5% false negative the false alarm rate for the word is approximately 0.12 per second compared to 0.75 per second for the **arr** detector. At 10% false alarm rate the corresponding values are 0.05 and 0.5 per second. The additional structure contained in the word dark eliminates many of the false alarms detected by the **aar** detector.

VI. EDGE BASED PHONETIC CLASSIFICATION

Thus far we have investigated the issue of efficient detection of phonological objects using the spread edges as the input features. Of interest is the ability to use the spread edges proposed in this paper for more standard classification problems. Do these features capture the necessary information to discriminate between phonetic objects with reasonable accuracy? We do not explore this issue in depth but report some results on the classification of segmented phonemes. Since phonetic classification is a more traditional task (than detection) in the speech community, these experiments

TABLE IV. Confusion matrix of eight broad classes.

	1	2	3	4	5	6	7	8
1	0.82	0.032	0.0039	0.076	0.044	0.0073	0.013	0.003 2
2	0.1	0.69	0	0.028	0.047	0.066	0.066	0.006 3
3	0.36	0.0034	0.43	0.19	0.0067	0.0051	0.0051	0
4	0.12	0.02	0.0018	0.79	0.04	0.012	0.014	0.002 2
5	0.018	0.036	0	0.018	0.82	0.04	0.066	0.001 1
6	0.0042	0.013	0	0.0086	0.028	0.82	0.12	0.000 6
7	0.0021	0.0027	0	0.0049	0.0084	0.068	0.91	0.000 15
8	0.11	0.1	0	0.14	0.074	0.07	0.065	0.44

will provide a sense to the reader as to how this technology might be expected to perform on this more standard task.

We applied a tree based classification procedure we have used in vision problems [see Amit (2002)]. Edges are computed on the continuous signal and the segments are extracted using the attached phonetic transcription. Each edge data segment of size $T \times F$ is placed in a fixed sized grid $T_{\max} \times F$ where $T_{\max}$ is the largest time extent of the segmented data, in our case $T_{\max}=40$ time units corresponding to 200 ms. (A small number of segments may be longer and is simply truncated at $T_{\max}$). We assume the data have all zeros on the remaining subgrid segment $[T, T_{\max}] \times F$. This implicitly conveys information on the length of the segment although this is not explicitly used in the classification procedure. Robustness to local time warping and frequency modulations is obtained by the spreading operation described earlier.

Classification is achieved by training multiple decision trees, with randomization (Amit and Murua, 2001), at each node only a small subsample of all $T_{\max} \times F \times 8$ binary edge features is inspected to find the most important split. We also implement boosting (Schapire *et al.*, 1998) whereby after each tree the data are reweighted, increasing the weight on misclassified examples. We grow M trees per class that are trained to classify that class against all others (as one negative class), for a total of $M \cdot C$ trees. At each terminal node of a tree grown for class c there is a weight between 0 and 1 corresponding to the proportion of training points of class c that reached that node in training. For testing we drop a data point X down all $M \cdot C$ trees. Let $\mu_{c,m}(X)$ be the weight assigned to c at the terminal node reached by X on the mth tree of that class. We choose the class that accumulates the highest weight:

$$\hat{c} = \text{argmax}_{c=1,\ldots,C} \frac{1}{M} \sum_m \mu_{c,m}(X).$$

We use 100 000 training segments from the TIMIT training data set, and 50 000 testing segments. 200 trees are grown for each class. With the full 52 phonetic classes of TIMIT, we achieve a classification rate of 58.5%. It is common to consider performance on a reduced set of 39 phonemic classes for which we achieve 61.37%. On a set of broad classes (8 in all) we achieve 83.8%. Shown in Table IV is the confusion matrix for the eight broad classes. These correspond to *stops, flaps, affricates, fricatives, nasals, semivowels, vowels*, and *aspiration* (corresponding to **h** in TIMIT), respectively (numbered 1 through 8 in that order in the confusion table). Performance for vowels is best at 91% correct class identification. Performance for all other classes is reasonable except for affricates (43%) and the phoneme **h** (aspiration) (44%), respectively. For affricates, we see that there is significant confusion with stops (36%) and fricatives (19%), respectively. The confusion, though high, is not entirely unreasonable. For the phoneme **h** (class 8), we see that the confusion is a little more random though even here, stops (11%), flaps (10%), and fricatives (14%) dominate. These last three categories often contain a degree of aspiration in their phonetic realizations and we conjecture that this probably leads to the confusion.

VII. DISCUSSION

From the experiments reported here it appears that the statistics of the local features on object and background are sufficiently different to allow for a simple weighted sum to successfully discriminate between object and background. More complex local features may yield even higher power. For example in Amit and Geman (1999) local edge conjunctions are used. Furthermore the simplicity of the models allows for some simple mechanisms of online adaptation which we briefly discuss in the next section, after which we discuss the issue of how such detections may be incorporated into a more comprehensive continuous speech recognition system.

A. Template adaptability

When the acoustic characteristics of the speaker changes, or the acoustic channel changes (from free space to a telephone channel or changes in microphone), when there is reverberation, or noise of various sorts, or auditory clutter, a human displays varying degrees of perceptual constancy that current machines are unable to achieve. It is unreasonable to assume that the stored and learned representation of the acoustic objects are relearned for each new environment. Instead, two aspects come into play. One consists of invariances hardwired in the representation which make them insensitive to certain changes. An example could be limiting the representation for sonorant regions to frequencies under 3000 Hz, so that irrelevant perturbations in higher frequency regions do not affect recognition. Another example is the amplitude invariance in the definition of the local features. However by far more important is the ability to *adapt* pa-

rameters of the representations *on line* as the signal is being processed.

1. Online adaptation of template weights

The templates are defined in terms of binary variables which are functions of local time-frequency information. Detection involves thresholding a weighted sum, where the weights derive from fixed estimates of object and background probabilities. It is possible to have a continuously updated estimate of the background probabilities $p_{b,i}$, performed *online*, over time intervals of larger duration. If there is significant background activity at a particular range of frequencies $[f,f']$, the probability estimates $p_{b,i}$ for the corresponding features would increase, reducing the weight on that feature in the sum in Eq. (5). The threshold ρ would have to be adjusted accordingly. This mechanism for adapting the template to the statistics of the background is simple and transparent in large part due to the simplicity of the original template.

2. Adapting to the individual speaker

Templates of the form shown in Fig. 4 are quite coarse. During training some local features are spread in the frequency direction to accommodate variations in the basic formant frequencies among speakers. However, if the three basic formant frequencies of a particular speaker are known, it is possible to adjust the templates around these frequencies and narrow down their spread. This does not require retraining, rather a simple modification of the list of feature location pairs, based on the estimated baseline formant frequencies. This would enhance the detection rate, especially in terms of false alarm rates.

B. From detection to continuous speech recognition

This paper focuses on the detection of acoustic objects, and demonstrates that robust detection is possible with small training sets. In all experiments described here we experience a certain percentage of false negatives, as well as a certain false alarm rate measured for example as false detections per second. Note, however, that with a relatively small library of detectors, either for all phonemes or perhaps all diphones, one can transform the original spectrogram into a sequence of labeled phonetic features. Since false alarms occur in detection for each feature, it is important to note that the same time point may sometimes be labeled with two different features. These ambiguities are to be resolved at higher processing levels. Thus the approach provides us with a parse of the time-frequency plane in terms of a vocabulary of phonetic feature detectors.

One possible use of these features is to construct word templates. Larger degrees of time invariance can be introduced by more extensive spreading. These are very powerful features in that they are detected with very high probability on the object (word) and very low probability (the false alarm rate) on background. The templates constructed from these features would then provide robust and invariant word detectors. Note that in Sec. V we experimented with a word detector derived directly from the original edge features. This works well for short words but would not provide for sufficient time invariance for longer words.

Another possible use of the new feature map would be as input into an HMM directly trained on the outputs of the phoneme or diphone detectors. Thus higher level knowledge of vocabularies and syntax is incorporated directly at this more symbolic level. We do not advocate resolving all the linguistic content based solely on the new feature map. At some places there will be ambiguities between competing interpretations which may need a more intensive analysis of the original spectral data. However, it is hoped that these will be relatively few, and the high level analysis will provide only a small number of candidate interpretations which require such intensive analysis.

[1]In order to compute ROC curves efficiently (Sec. V), we have to perform the clustering prior to the thresholding. Strictly speaking, this alters the overall detector, but the extent of this effect appears to be insignificant.

Allen, J. (**1994**). "How do humans process and recognize speech," IEEE Trans. Speech Audio Process. **2**, 567–577.

Amit, Y. (**2000**). "A neural network architecture for visual selection," Neural Comput. **12**, 1059–1082.

Amit, Y. (**2002**). *2d Object Detection and Recognition: Models, Algorithms and Networks*, (MIT, Cambridge, MA, in press).

Amit, Y., and Geman, D. (**1999**). "A computational model for visual selection," Neural Comput. **11**, 1691–1715.

Amit, Y., and Murua, A. (**2001**). "Speech recognition using randomized relational decision trees," IEEE Trans. Speech Audio Process. **9**, 333–342.

Bourlard, H., and Dupont, S. (**1997**). "Subband based speech recognition," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Germany, pp. 1251–1254.

Chi, Z., and Margoliash, D. (**2001**). "Feature representation, pattern filtering, and temporal alignment for birdsong detection," Technical report, Department of Statistics, University of Chicago.

Fletcher, H. (**1995**). *Speech and Hearing in Communication* (Acoustical Society of America, New York).

Fleuret, F., and Geman, D. (**2001**). "Coarse-to-fine face detection," Int. J. Comput. Vis. **41**, 85–107.

Grimson, W. E. L. (**1990**). *Object Recognition by Computer: The Role of Geometric Constraints* (MIT, Cambridge, MA).

Hopfield, J., Brody, C., and Roweis, S. (**1998**). "Computing with action potentials," in *Advances in Neural Information Processing Systems*, Vol. **10**.

Hubel, D. H., and Wiesel, T. N. (**1968**). "Receptive fields and functional architecture of monkey striate cortex," J. Physiol. (London) **195**, 215–243.

Kowalski, N., Depireux, D. A., and Shamma, S. A. (**1996**). "Analysis of dynamic spectra in ferret primary auditory cortex: I. Characteristics of single unit responses to moving ripple spectra," J. Neurophysiol. **76**, 3503–3523.

Leung, H., and Zue, V. (**1986**). "Visual characterization of speech signals," in *Proceedings of ICASSP'86*, Tokyo, Japan, pp. 2751–2754.

Liu, S. (**1996**). "Landmark detection for distinctive feature based speech recognition," J. Acoust. Soc. Am. **100**, 3417–3430.

Niyogi, P., and Sondhi, M. M. (**2002**). "Detecting stop consonants in continuous speech," J. Acoust. Soc. Am. **111**, 1063–1076.

Riley, M. (**1989**). *Speech Time Frequency Representations*, (Kluwer Academic, Dordrecht).

Saul, L., Rahim, M., and Allen, J. (**2001**). "A statistical model for robust integration of narrowband cues in speech," Comput. Speech Lang. **15**, 175–194.

Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S. (**1998**). "Bossting the margins: A new explanation for the effectiveness of voting methods," Ann. Stat. **26**, 1651–1686.

Schwartz, O., and Simoncelli, E. (**2001**). "Natural sound statistics and divisive normalization in the auditory system," in *Advances in Neural Information Processing Systems*, Vol. **13**.

Sen, K., Theunissen, F. E., and Doupe, A. J. (**2001**). "Feature analysis of natural sounds in the songbird auditory forebrain," J. Neurophysiol. **86**, 1445–1458.

Stevens, K. N. (**1991**). "Speech perception based on acoustic landmarks: Implications for speech production," in *Perilus XIV: Proceedings of the Symposium on Current Phonetic Research Paradigms: Implications for Speech Motor Control*, Institute of Linguistics, University of Stockholm.

Theunissen, F. E., and Doupe, A. J. (**1998**). "Temporal and spectral sensitivity of complex auditory neurons in the nucleus hvc of male zebra finches," J. Neurosci. **18**, 3786–3802.

Tibrewala, S., and Hermansky, H. (**1997**). "Multiband and adaptation aproaches to robust speech recognition," in *Proceedings of Eurospeech'97*, Rhodes, Germany.

Ullman, S. (**1996**). *High-Level Vision* (MIT, Cambridge, MA).

Viola, P., and Jones, M. J. (2002). "Robust real time object detection," Int. J. Comput. Vis.