Pitch Detection With a Neural-Net Classifier

Etienne Barnard, Ronald A. Cole, M. P. Vea, and Fil Alleva

Oregon Graduate Center
Department of Computer Science
and Engineering
19600 N.W. von Neumann Drive
Beaverton, OR 97006-1999 USA

Technical Report No. CS/E 89-011

Pitch Detection With a Neural-Net Classifier

Etienne Barnard *
Ronald A. Cole**
M. P. Vea*
Fil Alleva***

*Department of Electrical and Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213

**Department of Computer Science and Engineering
Oregon Graduate Center
19600 N.W. Von Neumann Drive
Beaverton, OR 97006

***Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Pitch detection based on neural-net classifiers is investigated. To this end, the extent of generalization attainable with neural nets is first examined, and it is shown that a suitable choice of features is required to utilize this property. Specifically, invariant features should be used whenever possible. For pitch detection, two feature sets, one based on waveform samples and the other based on properties of waveform peaks, are introduced. Experiments with neural classifiers demonstrate that the latter feature set – which has better invariance properties – performs more successfully. It is found that the best neural-net pitch tracker approaches the level of agreement of human labelers on the same data set, and performs competitively in comparison to a sophisticated feature-based tracker. An analysis of the errors committed by the neural net (relative to the hand labels used for training) reveals that they are mostly due to inconsistent hand labeling of ambiguous waveform peaks.

TO APPEAR IN IEEE TRANSACTIONS ON ACOUSTICS, SPEECH & SIGNAL PROCESSING, 1991.

I. Introduction

Of all the new developments in the pattern-recognition literature in the past decade, few have been as important as the growth of interest in classifiers based on neural nets. This development is already influencing approaches to speech recognition significantly – whereas very few researchers studied the applications of neural nets to speech recognition as recently as 1985, there are numerous indications that neural nets are currently seen as an important tool for speech recognition. This new interest is typified by the presence of a session at the Fall 1988 meeting of the Acoustical Society of America entitled "Speech Communication IV: Neural Networks and Other Techniques." ("Other Techniques" included descriptions of the most successful speech recognition systems of that time.) [1] provides an excellent review of the research in this area.

Whereas some fraction of this recent activity is simply attributable to the novelty of the subject, it has become clear that there is indeed a niche that neural-net classifiers fill well. To understand why this is the case, one should consider the decision boundaries in feature space created by various classifiers. [2] The "old-style" neural nets, (perceptron, [3] Widrow-Hoff classifier [4]) are characterized by linear decision boundaries, and therefore have limited discriminatory ability. Multivariate Gaussian classifiers [5] generally form quadratic decision boundaries. This represents some improvement, but is still unsatisfactory for many applications, especially those involving multimodal distributions. In addition, the assumption of normality fundamental to Gaussian classifiers is rarely valid in practice, so that these classifiers often fail even when optimally placed quadratic surfaces would suffice. Neural-net classifiers such as the backpropagation (BP) classifier, [6] on the other hand, use (approximately) piecewise-linear discrimination surfaces, and do not assume a particular parametric form for the underlying probability distributions. Since any "reasonable" function can be approximated to arbitrary accuracy by a piecewise linear surface, such neural nets are in principle much more powerful than both linear classifiers and Gaussian classifiers. This improved discriminatory ability is especially useful for problems such as speech recognition, which are characterized by highly variable signals. (Nearest-neighbor classifiers [5] also have piecewise-linear decision boundaries. Their usefulness is limited, however, by the need to retain a large number of prototypes for classification.)

Although piecewise-linear decision surfaces constitute a useful extension to the capabilities of pattern classifiers, the abilities of neural nets should not be overestimated. For example, the number of classification surfaces required typically grows exponentially with the number of features, in a well-defined sense, so that unrealistically large nets and unrealistically long training times could be required if the feature space has too many dimensions. Such considerations (and others which we detail below) imply that one should exercise some care in formulating a neural-net solution to a given problem: if the input features are not selected appropriately, no current neural net will be able to perform satisfactorily. On the other hand, if it is possible to describe a problem so that discrimination by simple surfaces is sufficient, the power of neural nets is often not required. Only in the intermediate range of complexity will existing neural

We investigate the advantages and problems associated with neural-net classifiers on a particular problem, namely the detection of pitch. This problem is interesting for a number of reasons: it represents an important part of many speech processing systems, [9] it has attracted a wide variety of proposed solutions, [10] it is still considered a difficult task, [11] and it is similar to a variety of other classification problems, such as identification of R-waves in EKG waveforms. A neural-net pitch tracker is sensible from an implementational perspective: as neural-net hardware is becoming increasingly powerful, [12] speech subsystems implemented as neural nets promise to become fast and economical alternatives. Additionally, as we shall see, this problem allows us to demonstrate many of the pitfalls and advantages of neural nets.

Pitch trackers can be classified into three groups: those that employ time information, those using frequency information, and hybrids which use both time and frequency information. Rabiner et al. [10] have reviewed the properties of many popular pitch trackers. They did not find any one group of trackers to be superior in all respects. We employ time-domain signals for the neural-net pitch tracker, since the time waveforms lead straightforwardly to a classifier paradigm; however, a neural-net tracker based on frequency-domain or hybrid inputs is also conceivable.

In Section II we investigate some general properties of the types of features which can be used as input to a neural classifier, with particular attention to the invariance required if satisfactory performance is to be obtained on real problems. The conclusions reached in Section II are relevant for most applications of neural-net classifiers, as will become clear. In Section III our problem formulation and feature spaces are presented in more detail, and general experimental procedures are described. Various experiments pertaining to the details of a neural-net pitch detector are described in Section IV, and results are presented. Section V contains an analysis of the errors which our best pitch tracker commits. This system does not include any post-processing; although it is clear that a number of simple procedures (e.g. median filtering) can improve the performance of a pitch tracker considerably, [13, 9] post-processing is logically separate from the classification stage. To evaluate the performance of the classifier, we therefore do not include such a post-processing step, even though it would be used in practice. Section VI summarizes the lessons learned from this research.

II. Neural nets for invariant recognition

The power of neural-net classifiers has led many researchers to employ them in ways which would have been unthinkable with conventional classifiers. For example, whereas spectral coefficients would generally not have been considered sufficient for statistical classification of phonetic categories before the new wave of interest in neural nets, precisely these features have been used as the input to various neural net classifiers – sometimes with much success. [14, 15] This attests to the ability of classifiers based on neural nets. However, it is important to understand the limitations of neural nets. This will enable us to decide what input descriptions are appropriate,

---

It should be stressed that we concentrate on neural-net classifiers, because of their discriminatory power. There are many other functions which neural nets can perform (such as optimization [7] or hierarchical clustering [8]), which might be useful for other reasons.

and what problems are simply too complicated for current neural-net solutions.

Let us first investigate what is meant by "generalization by neural nets." In Fig. 1(a) we show training samples from 2 classes (denoted by x's and o's, respectively) for a two-dimensional

feature space. In this space, each class is distinguished by a clear pattern: for the x-class, feature x_1 tends to be large when x_2 is small, and vice versa, whereas the o-class is distinguished by larger values of x_1 (irrespective of the x_2-value). A classifier trained on the data of Fig. 1(a) may create a decision boundary (the solid line in Fig. 1(a) and 1(b)) which to some extent captures these relationships. Thus, when new samples are presented (the bold x and o in Fig. 1(b)), they are classified according to these patterns. Since the new samples may never have been seen, "generalization" is said to occur.

This limited generalization property is easy to confuse with a more powerful form of generalization. Consider the following artificial problem: we are to classify the 8 time signals shown in Fig. 2(a) into two classes, as indicated by the solid (class 1) and dashed (class 2) lines. Each signal consists of three non-zero samples, which have been connected by straight lines in Fig. 2 to facilitate interpretation. It is clear that, with this representation, the signals in class 1 form a set of positive peaks (i.e. the intermediate value is consistently larger than either of the end values), whereas the signals in class 2 are all negative peaks. The height, baseline, width and time of onset of these peaks are all variable.

Now consider using the sample values at times t=0,1,...,7 as input features to a neural-net classifier. The net "learns" that, for class 1, the sample at precisely t=5 must be larger than the samples at t=3 and t=7, and similarly the sample at t=2 must be larger than the samples at t=1 and t=6. For class 2, the sample at t=2 must be smaller than those at t=1 and t=3, etc. The classifier has learned the amplitude

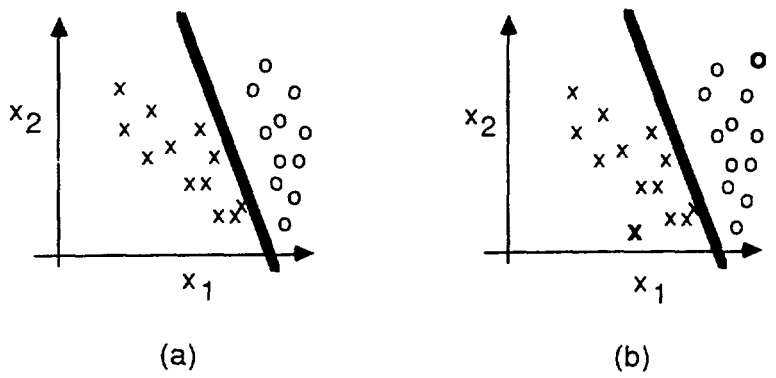

Fig. 1: Graph showing samples from two classes demonstrating the ability of a neural net to generalize: (a) training set; (b) classification of two unseen test samples.

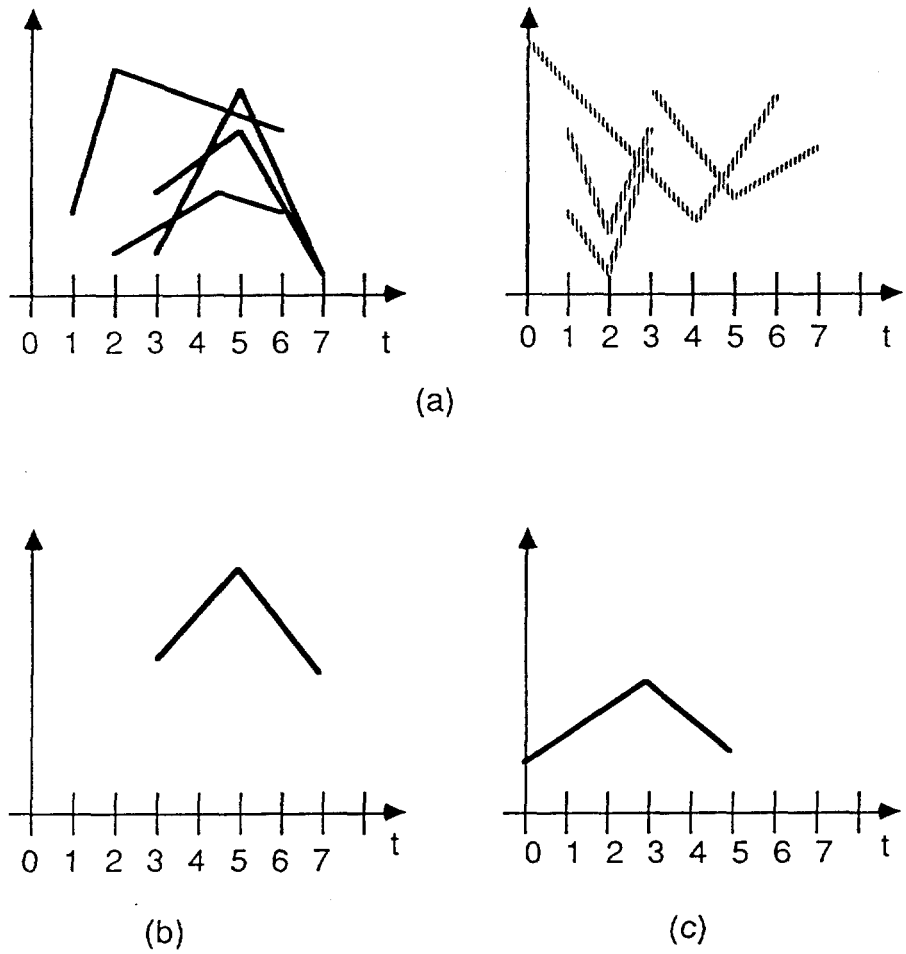

Fig. 2: Representation of classification problem: (a) training set; (b) test sample which is classified correctly; (c) test sample for which classifier generalization is not sufficient.

relationships between a number of specific triplets. As we discussed above, this learning involves some generalization; as long as the middle value of a specific triplet is larger than the flanking values, the classifier will assign the pattern to class 1. Thus, the input in Fig. 2(b) is classified correctly even though it has not been seen before (since the net has learned the class-1 relationships between the values at t=3, t=5, and t=7). However, the net has no basis to classify the input shown in Fig. 2(c), since it has not obtained any information about relationships between samples at t=0, t=3 and t=5. Since the pattern which should be deduced from the training samples does not refer to a particular set of features, but to relationships between different sets of features, the net cannot learn it from the samples shown. Only if positive and negative peaks involving every possible triplet of times are included in the training set will the net be able to discriminate between positive and negative peaks faultlessly.

There are therefore two levels of generalization: a classifier might be able to generalize by detecting a certain pattern among a set of features, without being able to generalize such patterns to **other** sets of features. Whereas humans are able to perform the more general operation, current neural-net classifiers specialize in the more limited domain. Thus, when we speak of generalization by neural nets, we have to keep in mind that we refer to the type of situation shown in Fig. 1, and not the situation of Fig. 2.

Now consider using a spectrogram as input to a neural classifier. This presents us with a problem analogous to that of the latter situation, since the distinctive patterns again involve different sets of features, depending on factors such as phonetic context, speech rate and the vocal tract length of the speaker. This implies that this feature set will only lead to suitable generalization if the classification is simple enough that a large fraction of all possible transformations of the relevant signals are input during training.

A feature set that is generally more appropriate as input for a classifier is suggested by the problem of Fig. 2: for that problem, we can use a three-dimensional feature space, with the three features being the three non-zero samples, ordered with respect to their time of occurrence. In this case, the class 1 feature vectors would be represented by sets of three numbers, with the middle number larger than the other two. With these features, generalization of the type shown in Fig. 1 is sufficient to learn the correct classification of positive and negative peaks from samples such as those shown. The critical property of these features is that they have an **invariant** meaning for this problem. Similarly, when speech recognition is performed with neural nets, one should try to capture the important features of the desired output classes by features with invariant meaning. This will often require considerable knowledge of the speech problem, since appropriate invariant features are highly problem-dependent.

In conclusion, neural-net classifiers are capable of only a limited form of generalization. If the problem under consideration is sufficiently complex, an intelligent choice of features is required in conjunction with neural classifiers, since such a choice can ensure that this limited generalization is sufficient. It is possible that neural nets which do not function as conventional classifiers might be able to overcome this limitation; however, we are not aware of any realistic model which has been demonstrated to be able to do so.

We now give a more detailed description of the problem we wish to solve. Thus we will be able to elaborate on the extent to which our problem requires a quasi-invariant input description.

III. Problem description and experimental method

To understand the fundamental issues involved in the time-domain estimation of pitch, we consider the waveforms in Fig. 3. In this figure (and all similar figures below) the waveform is delimited by two horizontal bars, and (3 msec.) frame and sample-point marks and labels are shown above the top line. The frame labels, which are the smaller topmost numbers in Fig. 3, will be used to identify particular portions of the waveform. All waveforms we show have been low-pass filtered by a zero-phase filter with cut-off frequency of 700 Hz. (The filter was designed using the Remez exchange algorithm to have 48 dB per octave rolloff in the transition band and 0.48 dB ripple in the passband.)

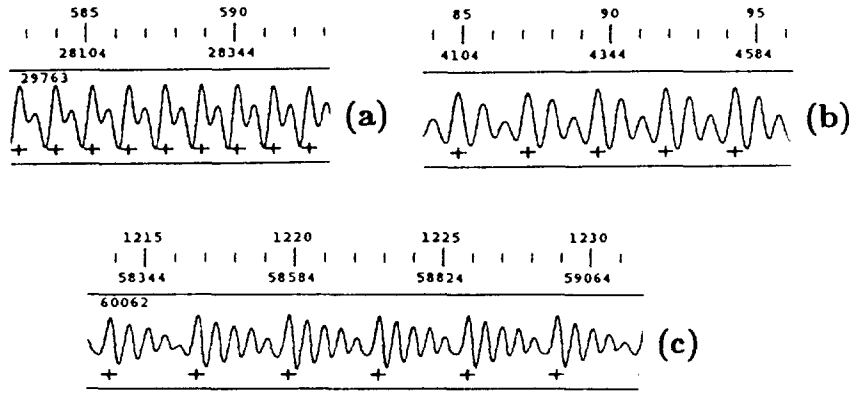

Fig. 3: Low-pass filtered waveforms of three vowels (a-c), demonstrating the time-domain characteristics of pitch excitation.

Figs. 3(a), (b) and (c) are taken from vowels spoken by three different speakers. In all these vowels, two harmonic patterns can be discerned: a quasi-periodic high-frequency structure is modulated by a pattern of lower frequency, so that every n-th period is noticeably larger than the surrounding periods (with n ranging from 2 in Fig. 3(a) to 5 in Fig. 3(c)) - as indicated by the +-signs in Fig. 3. It is well known [9] that the high-frequency pattern is caused by the resonant cavity formed by the speech organs (and thus correlates most strongly with the first formant), and that the lower-frequency pattern corresponds to the periodic excitation due to the vocal cords. That is, the long periods are caused by pitch excitation, and the shorter periods are resonances induced by this excitation. Note that the low-pass filtering enhances this pattern, since it reduces additional structure in the time waveform caused by the high-frequency content of the signal.

The purpose of a time-domain pitch tracker is to isolate this low-frequency periodicity by locating the large-amplitude periods within vocalized speech. The pitch-estimation problem can therefore be stated as a two-class classification problem, namely: given a portion of a waveform, decide whether a specified part of it corresponds to a pitch excitation or not. All the conventional tools of pattern recognition can thus be employed on this problem - in particular, a neural-net classifier can be used to perform the discrimination process. This approach also makes it unnecessary to first isolate the vocalized portions of the waveform, since we can train the classifier so that all non-vocalized parts of the waveform are classified as devoid of pitch. The classifier-based pitch tracker can therefore be used to help locate sonorant portions of speech.

To sensibly employ a neural classifier, we have to decide what features are appropriate for this classification task (as was stressed in Section II). From Fig. 3 it is clear that features based on the peak excursions of each of the periods contain most of the required information, so a feature set based on the waveform peaks is attractive. We therefore rephrase our classification problem in terms of the waveform peaks by asking whether a given peak corresponds to a pitch excitation, and choose features that

describe the to-be-classified peak in relation to its neighborhood. (Since every positive peak is associated with a negative peak, the classification question need only be asked about either the positive or the negative peaks. We chose to work with the positive peaks - they tend to show a more pronounced pitch pattern.)

Waveform Samples as Features. One way of describing the waveform neighboring a given peak is simply to list the amplitudes of a number of waveform samples in a window surrounding the peak. The required sampling frequency can be calculated using the Nyquist criterion and the cut-off frequency of the low-pass filter by which the waveform is preprocessed. The number of samples should be large enough to allow the classifier to extract the typical pitch patterns such as those in Fig. 3.

This feature set is intuitively simple, and straightforward to calculate, but suffers from limited invariance: since classification will always be centered on a waveform peak, the feature set is time-translation invariant, but it is not invariant to changes in frequency, since a fixed sampling rate is used. This is exactly analogous to the situation described in relation to Figure 2. Thus, this feature set is not automatically invariant to changes in speaker pitch, and might suffer from the problems described in Section II. Since we were not able to decide theoretically how detrimental this limited invariance would be, experiments to test the performance of this waveform-based feature set were performed. These experiments are described in Section IV.A.

Peak Descriptors as Features. With the preceding feature set, consisting of waveform samples, we have not utilized the fact that it is the surrounding **peaks** which carry most information about the identity of a given waveform peak. The characteristic features of pitch peaks are that they are larger than neighboring peaks, and that there is a regular decrease in the amplitudes of peaks intermediate to the pitch peaks (see Fig. 3), and that successive peaks tend to be equally spaced. This pattern can be captured by using features such as the following: the amplitude of the peak to be classified; the amplitudes of a certain number of peaks prior and subsequent to this peak; the time difference between each of these peaks, etc. If we use such peak-based features, we obtain significantly enhanced frequency invariance, since the effective sampling rate is now adapted to the dominant waveform frequency. That is, if the waveform is stretched in time (corresponding to a decrease in the frequency at which the utterance is spoken), the **same** set of surrounding peaks will still be used to describe the neighborhood of a given waveform peak. The amplitudes of these peaks will remain unchanged, and their time differences will be increased by a constant factor. These features are therefore conceptually similar to those occurring in Fig. 1 and those recommended for the problem of Fig. 2. The second set of experiments described in Section IV employed such peak-based features.

Experimental Procedures. The experiments used utterances drawn from the TIMIT database, a standardized corpus designed for acoustic phonetic research. [16, 17] The training set consisted of one utterance each from 80 different speakers (approximately 2/3 male), and the test set consisted of one utterance each from a set of 20 different speakers (14 male, 6 female).

The goal of classification is to assign a label of "pitch" or "no pitch" to each candidate peak in the filtered waveform. Candidate peaks were located using a straightforward peak-detection algorithm that locates the largest waveform values between every pair of positive-to-negative transitions of the waveform.

The correct label for each candidate peak – which indicates whether the peak is a pitch peak or not – was produced by a human expert, using the waveform as well as various derived features, including information provided by a zero crossings parameter and the phonetic labels provided with the TIMIT database. Thus, every peak located by the peak-picking algorithm was submitted to the expert for classification, in conjunction with these derived features. Comparison of the expert's labels to those provided by two additional labelers revealed an average agreement between 98% and 99% (see Section V); this level of accuracy is sufficient for the applications (such as speech recognition) that we have in mind.

Network Design. Figure 4 illustrates the structure of the neural-net classifiers that were used in the experiments. The networks were trained based on standard back-propagation (BP). [6] We used layered nets, with adjacent layers completely interconnected. The nets had either three or four layers, where the input layer is included in the counting of layers. To minimize the BP criterion function, we employed conjugate-gradient minimization. [18] This technique has a number of advantages over gradient descent and gradient descent with momentum (which are customarily used with BP) – in particular, it eliminates the search for suitable training parameters such as the learning rate, and minimizes the criterion function fairly rapidly. Its main disadvantage is that it forces us to use the batch mode of updating, [5] which implies that the weight vectors are only updated after all training samples have been processed. These merits and demerits of conjugate-gradient training are discussed in more detail elsewhere. [19]

All experiments used the following procedure: The network was trained for 50 iterations with the conjugate gradient optimizer on the training vectors from the 80 speakers. The trained network was then evaluated on the test vectors from 20 speakers in the test set. This procedure was iterated until no further improvement was observed on the test set following several sets of 50 iterations. We note that, for a set of 20,000 test vectors and a classification accuracy of 98%, a change of less than 0.2% is not statistically significant. This percentage was used as a criterion of significance throughout the research.

Preliminary Experiments. Before starting our main experimental series, we first determined appropriate values for two fundamental parameters, namely (i) the number of speakers to use in our training database and (ii) the number of training samples to use. The optimal values of these parameters depend to some extent on the details of the experiment performed, so that our purpose with these pretests was not to determine these parameters once and for all. Rather, we wished to find what regions of values are suitable to use in comparing different feature sets. Once an optimal feature set is determined, we can then re-estimate optimal values for the number of speakers and training samples, and retrain our best classifier with the optimal values.

Results of the pretests are shown in Fig. 5. We used amplitude and time-difference features (for more details, see Section IV.B) for seven peaks prior and seven peaks subsequent to the peak to be classified. We trained neural classifiers with different numbers of training samples derived from different numbers of speakers, and determined the optimal performance achievable on a test set obtained from a separate set of speakers. In this test, and all experiments reported below, the test set consisted of utterances from 20 different speakers; these utterances contain approximately 19,000 peaks of which approximately 6,000 are pitch peaks. As can be seen in Fig. 5(a) and (b), no

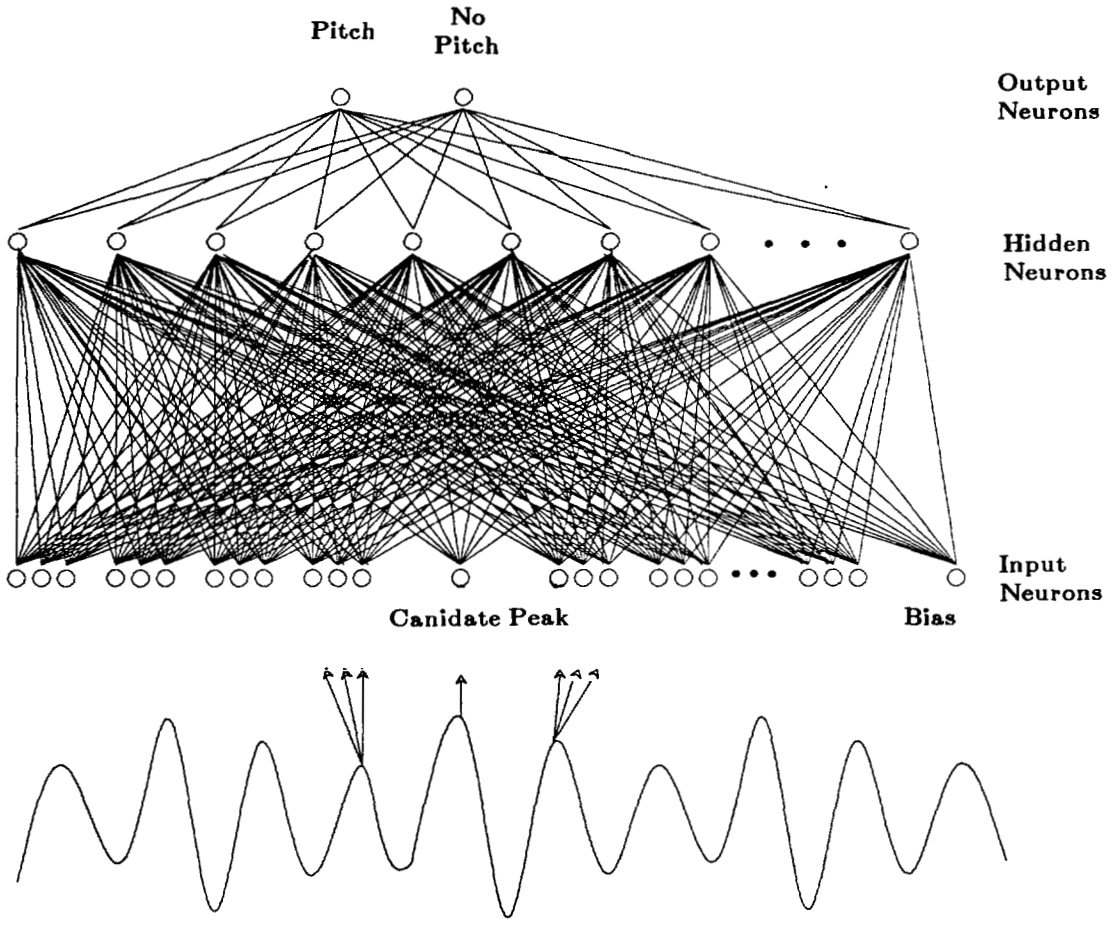

Fig. 4: Typical three-layered feed-forward neural net, with complete interconnects between successive layers.

Formant Tracking

Formants are the resonant frequencies of the vocal tract. The frequencies of the three lowest formants (F1, F2, and F3) provide sufficient information to identify vowels, and formant movements at vowel boundaries provide important information about the identity of adjacent phonemes. Accurate formant tracking provides important information for speech coding, recognition and synthesis.

Formant tracking is a most difficult problem because two formants may merge to form a single peak in the spectrum. For example, in words such as "roar" the second and third formants of the [r] sound typically merge to form a single band of energy. In other vowels, the first and second formant merge to form a single spectral peak. A second problem is that a single formant may split (usually when next to a nasal, as in "mom") and be realized as two distinct peaks.

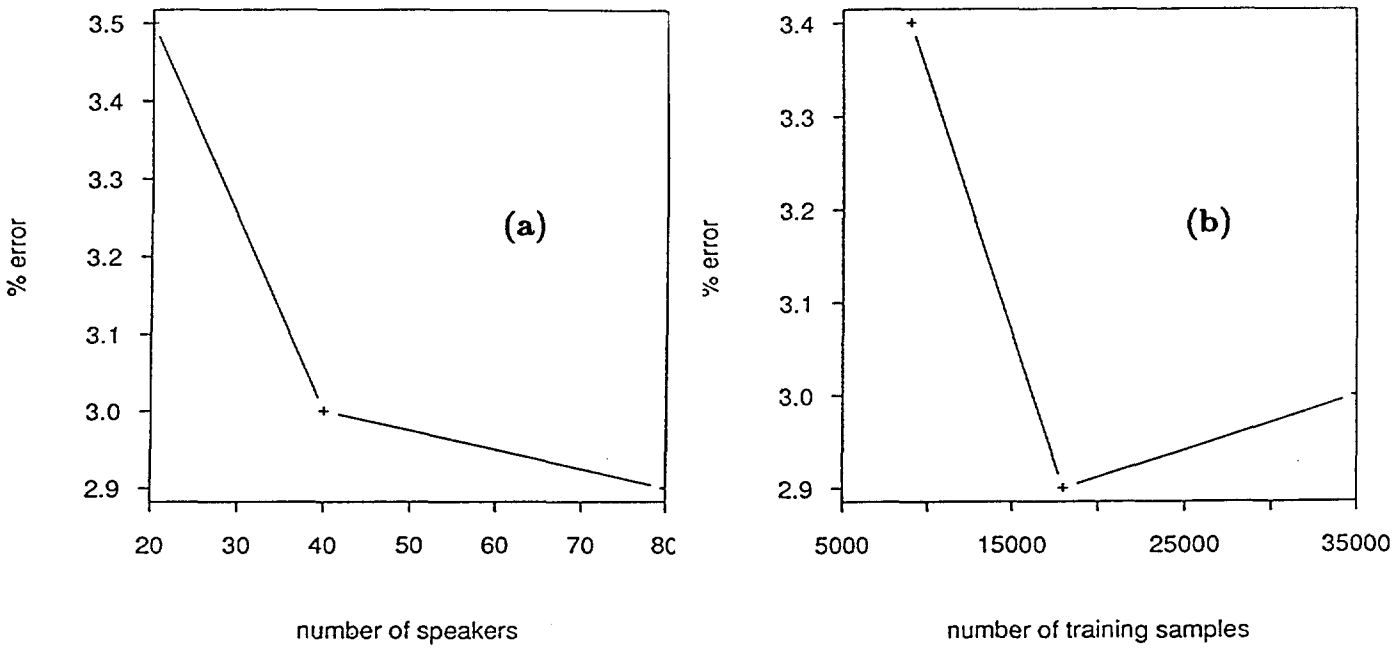

Fig. 5: Graph of (a) number of speakers; and (b) number of samples in training set plotted against error rate of a classifier.

A. Waveform Samples

The first set of experiments used the waveform feature set introduced in Section III. To derive these features, we proceeded as follows: all candidate peaks were located (as described above), and for every candidate peak, $2m+1$ samples of the low-pass filtered waveform, evenly spaced 0.375 msec. apart and centered on the candidate peak, were used as features. These samples were normalized by dividing by the maximum amplitude in the low-pass filtered waveform found in the window from the beginning of the utterance to 250 msec. after the candidate peak.

To arrive at the best classifier we had to decide how many samples to use in the feature set, and we also had to determine an appropriate size for the neural net (i.e. the number of layers, and the number of neurons in each of the hidden layers). Two sets of experiments were performed to settle these issues. We first used a neural net with the number of hidden neurons fixed at 10, and trained with different numbers of waveform samples. In Fig. 6(a) we show the optimal performance attained on the test set as a function of the number of waveform samples. It can be seen that performance improves as the number of samples is increased from 11 to 21, but thereafter performance levels off. Since larger neural nets might be able to utilize somewhat more information, we decided to use 41 samples in our next experimental set.

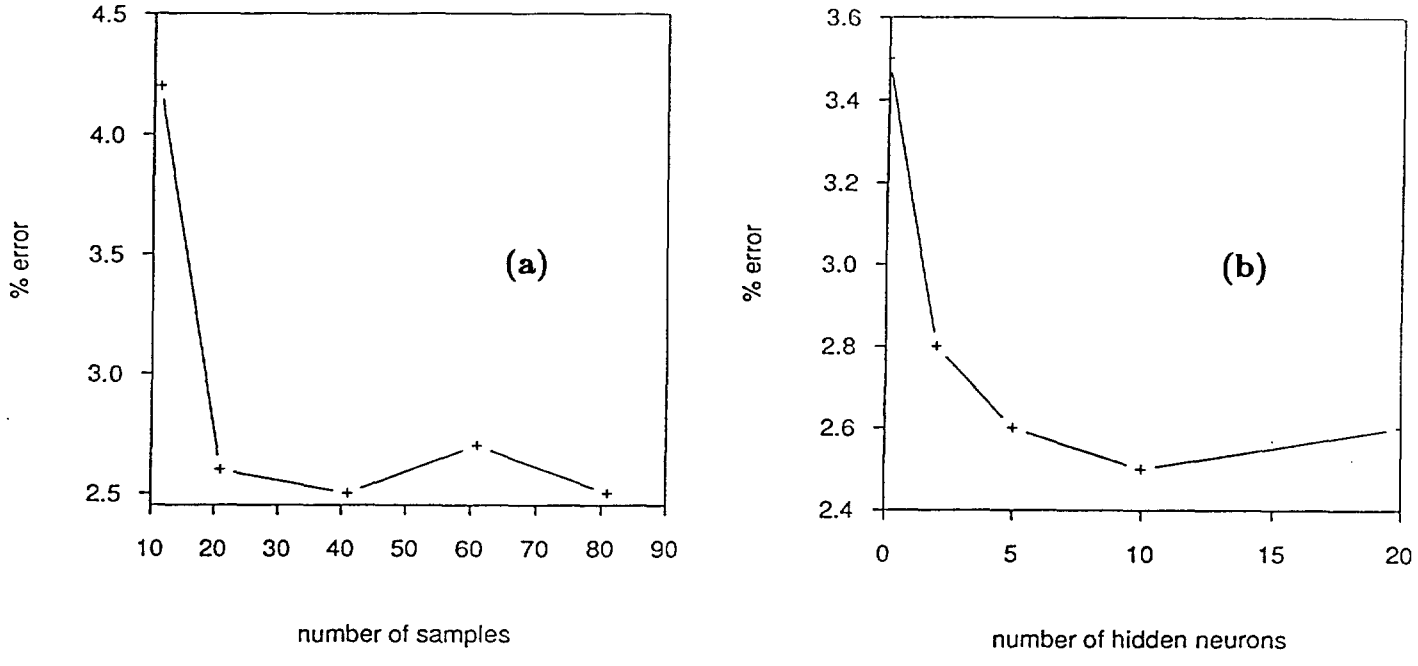

Fig. 6: Performance graph of pitch classifier based on waveform features as a function of (a) the number of waveform samples; and (b) the number of hidden neurons.

Theoretically, one hidden layer is sufficient for any classification task, since any decision boundary can be approximated to arbitrary accuracy by a neural net with one hidden layer and sufficiently many hidden neurons. [20] In practice it is often desirable to use two hidden layers, since a single hidden layer might require unrealistic accuracy in the calculation of neuron activities. We first experimented with one hidden layer, and varied the number of hidden neurons in that layer. As can be seen in Fig. 6(b), there is very little improvement in the performance of the neural net as the size of the hidden layer is increased beyond 5. A separate experiment with two hidden layers, containing 15 and 10 neurons respectively, also did not improve performance beyond that of the neural net with 5 neurons in a single hidden layer.

Since the number of training samples and the number of speakers were determined under different circumstances, we verified that the number of samples used was sufficient. To do this, we compared performance of our best classifier (41 waveform samples as input features, 10 hidden units) on the training set. If the performance of the classifier on the training set were considerably better than the performance on the test set, it would indicate insufficient variability in the training set. However, we found that the error rate on the training set was 2.3%, which is close enough to the performance on the test set (2.5%) to imply that a larger training set would not lead to significant improvement in the performance of the classifier.

We therefore conclude that the lowest error rate attainable with the neural classifier and the waveform feature set is 2.5%. To determine whether the neural net was really needed, we also trained linear and multivariate Gaussian classifiers with the same training set. The linear classifier we used is based on the sigmoid criterion function (see [21]). We obtained an error rate of 3.5% with the linear classifier and 22.4% with the Gaussian classifier. Thus, the decision boundaries required for this task are sufficiently non-linear, and the data are sufficiently non-normal, that a neural classifier is indeed required.

B. Peak Descriptors

We next investigated the performance of the neural net using the peak-based feature set. Since a large variety of features based on the structure and location of the prior and following peaks can be envisaged, we decided to experiment with different combinations of features to obtain optimal performance. Amplitude, negative-amplitude, width, time-difference and correlation features were used, as shown in Figs. 7 (b-f).

Thus, for every peak to be classified (e.g. Fig. 7a), an appropriate combination of these features was calculated for each peak within a window containing n peaks prior to that peak and n peaks subsequent to the candidate peak (with n variable). For each of these $2n+1$ peaks, the peak-based features were calculated as follows:

- **amplitude:** the amplitude of the peak is divided by the amplitude of the largest peak found in a window spanning from the beginning of the utterance to 250 ms after the current candidate peak.

- **time differences:** the time between the peak and the peak to be classified, normalized by a maximum period of 20 ms.

- **correlation:** for this feature, the waveform is segmented; each segment spans the part of the waveform between two successive negative peaks. Thus, one segment is associated with each of the $(2n+1)$ peaks, and the (negative) correlation of this peak with the candidate peak is calculated as

$$C_i = \sum_{r=-R}^{R} (s_i(r) - s(r))^2 \Big/ \sum_{r=-R}^{R} (s_i(r)^2 + s(r)^2),$$

  where C_i is the negative correlation, $s_{i(r)}$ is the r'th sample in the part of the waveform corresponding to segment i, $s(r)$ is the $r'th$ sample in the segment associated with the peak to be classified, and R is the maximum extent of the larger of these two segments (measured from the location of the peak).

- **width:** this equals the time elapsed between the zero-crossing before the peak and the zero-crossing after the peak (normalized by 2 msec).

- **negative amplitude:** similar to the amplitude feature, except that the most **negative** sample value between every pair of positive peaks is used.

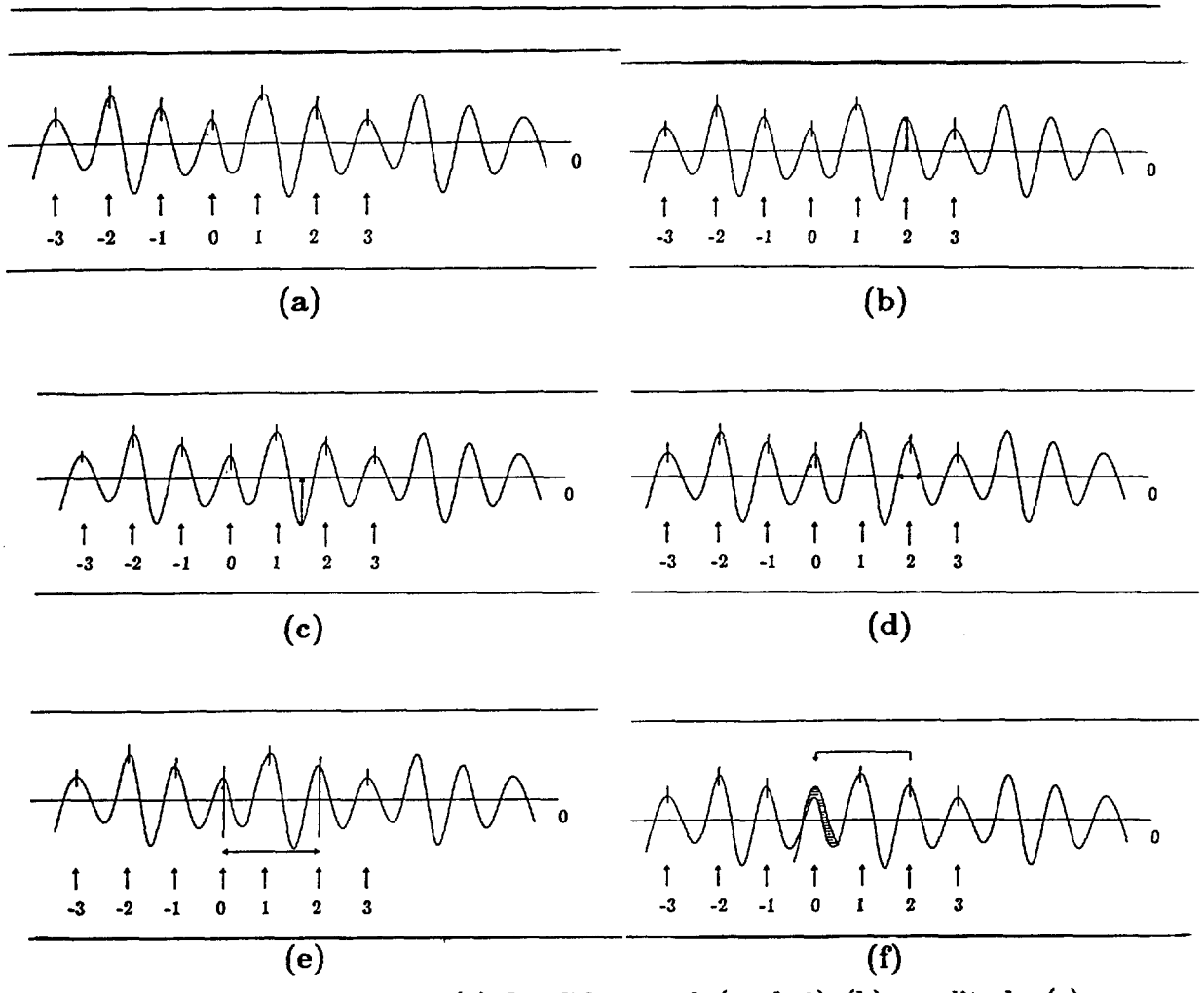

Fig. 7: Peak-based feature set. (a) Candidate peak (peak 0); (b) amplitude; (c) negative amplitude; (d) width; (e) time difference; and (f) correlation.

As with the waveform feature set, we had to determine a suitable number of peaks and a suitable neural-net size. For this purpose we again used only amplitude and time-difference features, and first varied the number of peaks used for a fixed neural-net size. (The net had one hidden layer, with 15 neurons.) The results, shown in Fig. 8(a), indicate that as few as 3 peaks give virtually asymptotic performance with this configuration.

However, we allowed for the possibility that this number may increase somewhat as more features and larger nets are used; we therefore used four prior and following peaks in the further experiments. Fig. 8(b) shows the results obtained when the number of hidden neurons was varied in a net with one hidden layer. In this case, the minimal number of hidden neurons with asymptotic performance is approximately 10. As in Section IV.A, no improvement was obtained by using a net with two hidden layers.

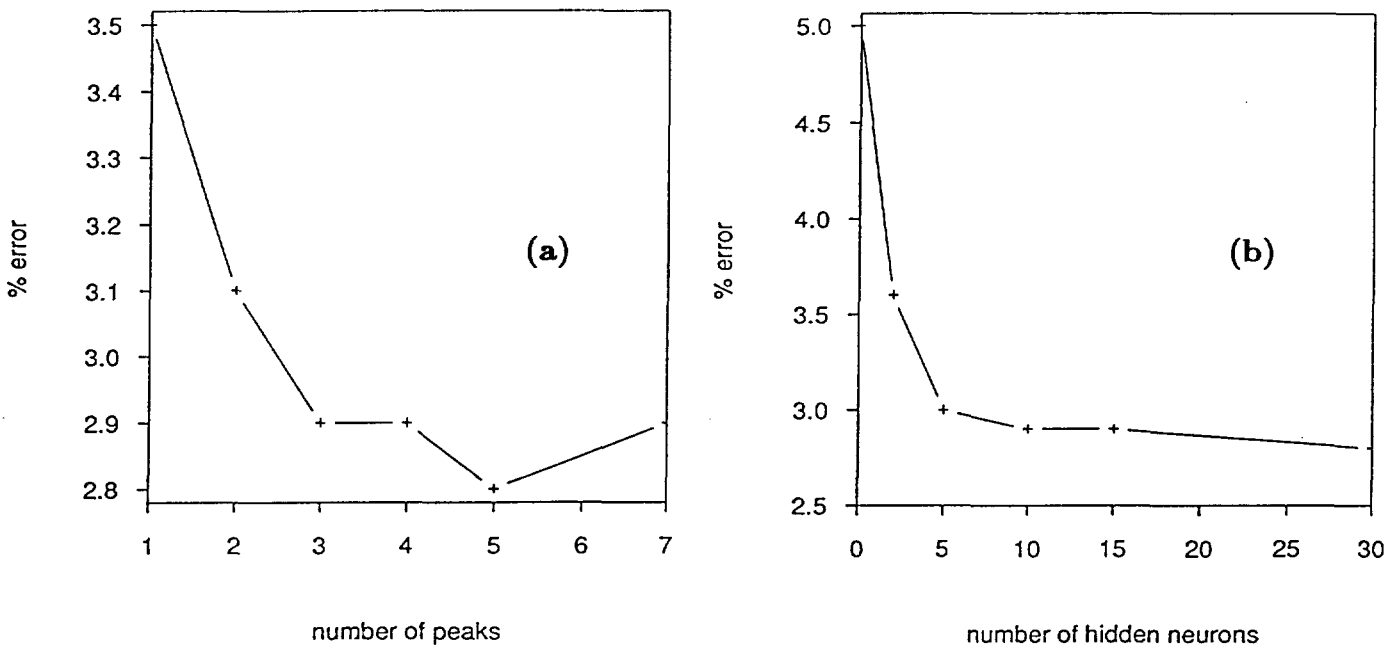

Fig. 8: Graph showing performance of pitch classifier based on peak features as a function of (a) the number of peaks; and (b) the number of hidden neurons.

Having determined the appropriate number of peaks and hidden neurons, we next performed experiments to determine the optimal combination of peak-based features. In Table I the results for various such combinations are listed. With only positive-amplitude features, approximately 4.4% of all peaks were misclassified; adding information about the time difference between successive peaks reduced this number to 2.9%. Of the features that were added to these two, the correlation feature was most useful (leading to an error rate of 2.4%), and a slight further improvement (error rate equals 2.3%) was obtained by adding the negative-amplitude feature to this set.

When we tested on the training set it was clear that the number of training samples was not sufficient for this feature set – whereas an error rate of 2.3% was obtained on the test set, the error rate on the training set was only 1.9%. We therefore increased the number of training samples to 35,000, and retrained the net. Now the error rate on the training set was 2.0%, and the test-set error rate remained at 1.9%. Thus, 35,000 training samples suffice, and 2.0%. is the lowest error rate we could obtain with these features.

For this feature set, the linear classifier produced an error rate of 5.0%, and the Gaussian classifier had an error rate of 9.1%. The utility of the neural classifier is again clear.

Table I: Performance of neural-net classifier using various peak-based feature sets.

Neural-net Classifier Performance	
Feature Set	Error Rate
Ampl	4.4%
Ampl, time diff	2.9%
Ampl, time diff, corrln	2.4%
Ampl, time diff, width	2.5%
Ampl, time diff, neg ampl	2.9%
Ampl, time diff, corrln, neg ampl	2.3%

V. Analysis of errors

To analyze the performance of our pitch detector, we have studied its detailed performance on numerous utterances. For this purpose the low-pass filtered waveform was printed out in conjunction with the labels generated by the classifier and the human expert, and the differences were examined. It seems that almost all the discrepancies between the human and automatic labels arise from one or more of the following four causes: (i) ambiguity in the waveform, leading to inconsistent human labeling, (ii) weak signals which are sometimes labeled as containing pitch peaks by the tracker, (iii) signals whose local structure obscures the overall pitch pattern and (iv) places in voiced signals where the pattern of peaks changes, leading to incorrect automatic labeling. These effects will now be described in more detail.

Ambiguous peaks: around half of the differences between the human and machine labels can be attributed to inconsistent labeling (of both training and test data) because of ambiguity in the waveform. This is particularly likely to occur at the end of voiced sections of speech, when it is not clear how far the voiced section extends. In Fig. 9(a) we show a case where a peak was labeled as a pitch peak by the machine but not the human (a short vertical bar below a peak indicates that it was labeled as a pitch peak by the human expert, whereas a horizontal bar indicates that the classifier labeled it as such), and also a case where the opposite occurs. It is clear that these "errors" are really intrinsic to the transient nature of the waveform; consistent labeling of these peaks is probably neither possible nor necessary.

Weak signals: in Fig. 9(b) we show a peak (which occurs within the phoneme "t") which is erroneously labeled as a pitch peak by the tracker. The amplitude of this peak is small, but comparable to the amplitudes of pitch peaks which occur at the end of voiced utterances. Also, the pattern of surrounding peaks happens to be fairly periodic. Thus, it is understandable that mislabelings will occur in such cases. Fortunately, this

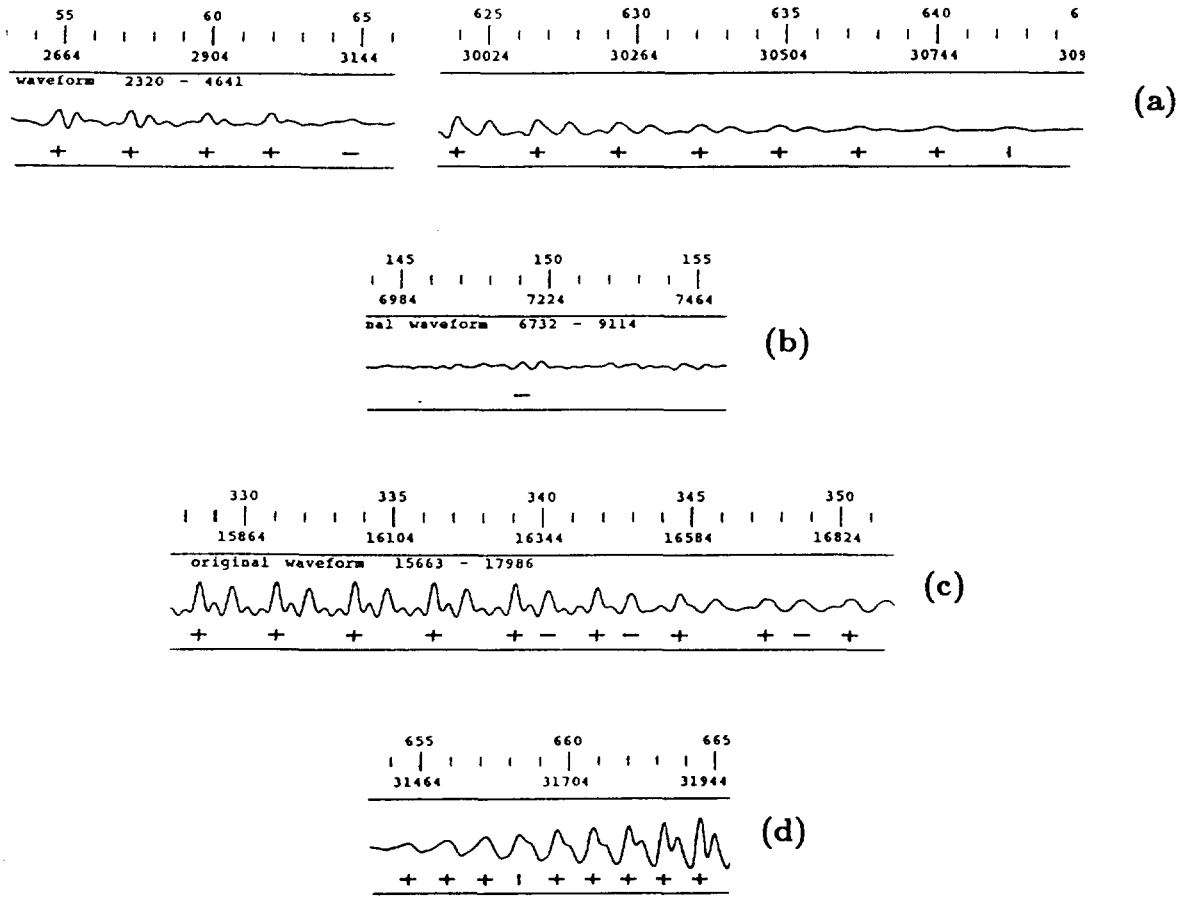

Fig. 9: Examples of the types of errors committed by neural tracker, caused by (a) ambiguous peaks; (b) weak signals; (c) signals with confusing structure; and (d) transitions in peak patterns.

phenomenon is fairly rare, and can almost always be eliminated by suitable post-processing because of the small amplitude of these peaks, their relative isolation and, in this case, the high zero crossing rate of the unfiltered waveform.

Signals with confusing local structure: in the first part of the waveform shown in Fig. 9(c) there are four non-pitch peaks between every pitch peak; the second of these peaks is much larger than the other three. Towards the end of the vocal segment (around time frame 340) and at the beginning of the subsequent nasal (frames 343 and 349) this pattern causes the classifier to insert incorrect pitch markings. Errors in this class may be impossible to correct with post-processing since they can lead to a spurious periodic set of pitch labels which cannot be discerned from the correct labels. Again the set of conditions which lead to this error is fortunately sufficiently rare that this is

not a major concern – we estimate that errors due to this effect occur on the average less than once in every three seconds of speech.

Transitions in pitch patterns: the low-pass filtered waveform sometimes shows small changes in the peak structure which cause a discontinuous change in the features input to the classifier. Consequently, the classifier might mistakenly classify the waveform as though a large change in the input signal has occurred. An example of this occurs in the waveform shown in Fig. 9(d): four consecutive pitch peaks are followed by a small non-pitch peak (and thereafter the pattern changes to one non-pitch peak between every pair of pitch peaks). Because of this transition, the classifier mislabels the last of the initial four consecutive pitch peaks. This phenomenon is generally amenable to correction by median filtering, since it leads to a single insertion/deletion of a pitch peak.

Comparison with Human Labelers. To provide a basis to evaluate these results, two additional human labelers marked each pitch peak on visual displays of the filtered waveforms for the 20 utterances in the test set. The average disagreement between these two labelers was 1.1%. The average disagreement between each of these labelers and the labeler whose hand-marked labels used to evaluate the pitch tracker was 2.0%.

Comparison to Another Pitch Tracker. We also compared performance of our best neural net pitch tracker to performance of a feature-based pitch tracker used extensively in the Carnegie Mellon speech effort in recent years. [15] The latter employs multivariate classifiers and knowledge-based features to assign labels to candidate peaks. Rules are then used to select the final set of classified peaks. The statistical pitch tracker disagreed with the hand-labeled peaks in the test set 4% of the time, compared to 2% for the neural net pitch tracker. Although the results are not directly comparable because of differences in training procedures and design philosophies in the two algorithms, they do indicate that the neural net pitch tracker performs competitively.

VI. Conclusions and Summary

We have found that both the waveform-based and the peak-based feature sets lead to good discrimination of pitch peaks. The best peak-based feature set leads to an error rate that is significantly lower than that of the best waveform set; 2.0% *vs.* 2.5%, a 20% difference in the error rate. This implies that the invariance properties of the former features are fairly useful, though not vital, for this task.

It is somewhat surprising that including more samples or peaks does not lead to improved performance. This is probably due to the relative scarcity of training samples which require the additional information for successful classification. We have noticed that BP is not successful in learning properties of samples with low a-priori probability, since the more likely cases tend to dominate the learning procedure. Several solutions (such as not learning on samples which are classified correctly by a sufficient margin, or subdividing the various classes) have been tried, but none has improved the performance of our pitch tracker. Further research concerning the relationship between BP training and a-priori probabilities might lead to better solutions of this problem.

One important lesson that this research has emphasized is the ability of neural nets to find simple patterns which describe large fractions of the data. Thus, around 96.5% of all peaks can be classified correctly by a net with no hidden neurons, and around 97.3% of all peaks are classified correctly by a classifier which has access to no more than one pitch peak to either side of the candidate peak. Addition of the extra machinery for more powerful classification improves matters by no more than 1% (although this does represent a 30% reduction in the error rate.)

To understand this phenomenon better, we have analyzed the weights occurring in the neural net after training. It turns out that the patterns extracted by the neural net are not the patterns we expected at all. For instance, since an approximately linear increase in the time-difference features (of the peak-based set) is a good indication of voicing, we expected at least some weights from the neurons representing these features to be tuned for such a pattern. In practice, no such behavior was seen. Consider also the weights occurring in the net with no hidden layer when the waveform features were used (Fig. 10): rather than representing a "typical pitch period" – which would occur if the net instantiated a "matched filter" for pitch – the net has discovered an asymmetry between the samples following a pitch peak and those prior to the pitch peak. This discovery enables the net to perform reasonably well (96.5% success) despite variations in pitch frequency, which would drastically degrade the performance of a matched filter.

Finally, we would like to stress the applicability of neural-net subsystems such as the one we have described within larger speech-recognition systems. Because of the wide range of applications for neural nets, the next few years will see the commercial introduction of architectures which implement neural nets with a high degree of parallelism. Many feature-based neural net recognizers, performing various tasks such as pitch detection, formant estimation, segmentation, phoneme classification, etc., can be implemented in parallel on such architectures, leading to very efficient systems for speech recognition.

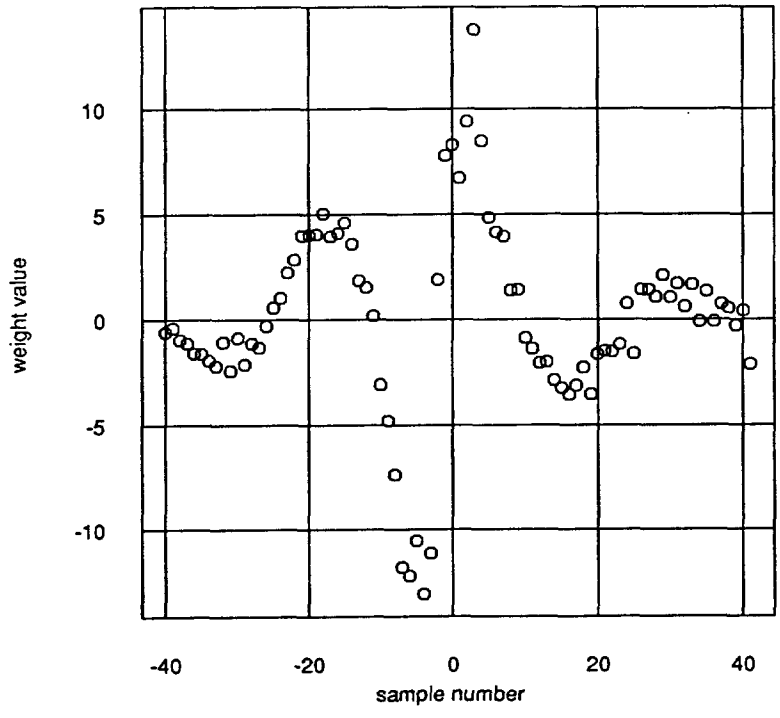

Fig. 10: Weight set for linear classifier produced by training with waveform features.

References

1. R. P. Lippmann, "Review of Neural Networks for Speech Recognition ," *Neural Computation* **1** pp. 1-38 (1989).
2. R. P. Lippman, "An introduction to Computing with neural nets," *IEEE ASSP Magazine*, pp. 4-22 (April, 1987).
3. F. Rosenblatt, *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms,* Spartan Books, Washington, D.C. (1962).
4. B. Widrow and M. E. Hoff, "Adaptive switching circuits," *WESCON Convention Record (Part 4)*, pp. 96-104 (August 1960).
5. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis,* John Wiley and Sons (1973).
6. D. E. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors," *Nature* **323** pp. 533-536 (1986).
7. J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Science USA* **79** pp. 2554-58 (April 1982).
8. G. Lynch, R. Granger, J. Larson, and M. Baudry, "Cortical encoding of memory: Hypothhesis derived from analysis and simulation of physiological learning rules in anatomical structures," pp. 247-289 in *Neural connections, mental computations,* MIT Press, Cambridge, MA (1988).
9. L. R. Rabiner and R. W. Schafer, *Digital processing of speech signals,* Prentice-Hall, Englewood Cliffs, NJ (1978).
10. L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Transactions on Acoustics, Speech, and Signal Processing* **ASSP-24, no.5** pp. 399-418 (October 1976).
11. M. Lahat, R. J. Niederjohn, and D. A. Krubsack, "A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech," *IEEE Transactions on Acoustics, Speech, and Signal Processing* **ASSP-35, no. 6**(June 1987).
12. D. A. Pomerleau, G. L. Gusciora, D. S. Touretzky, and H. T. Kung, "Neural network simulation at warp speed: How we got 17 million connections per second," *Proceedings of the IEEE International Conference on Neural Networks,* pp. 165-172 (July 1988).
13. P. Specker, "A powerful post-processing algorithm for time-domain pitch trackers," pp. 18B.2.1-18B.2.4 in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing,* (1984).
14. A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K.J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech, and Signal Processing* **ASSP-37** pp. 328-339 (March 1989).
15. J. L. Elman and D. Zipser, "Learning the hidden structure of speech," **8701**, Institute for Cognitive Science, University of California, San Diego (1987).

16. W. Fisher, G. Doddington, and K. Goudie-Marshall, "The DARPA speech recognition research database: specification and status," pp. 93-100 in *Proceedings of the DARPA Speech Recognition Workshop*, (February, 1986).

17. L. Lamel, R. Kassel, and S. Seneff, "Speech database development: design and analysis of the acoustic-phonetic corpus," pp. 100-110 in *Proceedings of the DARPA Speech Recognition Workshop*, (February, 1986).

18. M. J. D. Powell, "Restart procedures for the conjugate gradient method," *Mathematical Programming* **12** pp. 241-254 (1977).

19. E. Barnard, "Optimization for training neural nets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (March 1989). Submitted

20. K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Discussion Paper 88-45, Department of Economics, UCSD (February 1989).

21. E. Barnard and D. Casasent, "A comparison between criterion functions for linear classifiers, with an application to neural nets," *IEEE Trans. Syst., Man, Cybern.*, (1988). Submitted