Formant Estimation from a Spectral Slice using Neural Networks

Terry Rooker
B.A., University of Washington, 1979
B.A./B.Sc., The Evergreen State College, 1988

A Thesis submitted to the faculty
of the Oregon Graduate Institute
in partial fulfillment of the
requirements for the degree
Master of Science
in
Computer Science
August, 1990

The thesis "Formant Estimation from a Spectral Slice using Neural Networks" by Terry Rooker has been examined and approved by the following Examination Committe:

Dr. Ronald Cole
Associate Professor
Thesis Supervisor

Dr. Todd Leen
Assistant Professor

Dr. Mark Fanty
Post Doctoral Fellow

Contents

List of Figures

List of Tables

Abstract

Formants are the resonant frequencies of the vocal tract. As the vocal tract is moved to different positions to produce different sounds, there is a corresponding change in the formant frequencies. Estimates of formant frequencies for the lowest three formants can give important information about the phoneme produced. Change in the vocal tract position causes the formant frequency ranges to overlap. We investigate the ability of neural network classifiers to learn important distinctions between the formants, and to assign the appropriate formant labels.

We used both spoken letters of the English alphabet and continuous speech. Our backpropagation network uses conjugate gradient optimization. We first experimently determined the best feature set, influenced by the features used by human labelers. Then we experimentally determined the best representation of those features, and network configuration. Representation questions include feature derivation, and absolute or relative indexing of location. Configuration questions include network size, and presentation and labeling of the feature vectors. We compare the performance to other published algorithms and human performance. This system also compares favorably to both.

1 Introduction

Formants represent the resonant frequencies of the vocal tract. The vocal cavities (including the nasal cavities) can be modeled as series of tubes[5]. The vocal cords vibrate and excite these cavities, which then produce their resonant frequencies. As the articulators (such as the tongue, and lips) change position, the corresponding formant frequencies also change. As the articulators move from one target position to another (for different vowels), the formants may range greatly in frequency. We are interested in the first three formants (F1, F2, F3), since they have the most importance in identifying sonorants.

1.1 Motivation

Formants provide important information about the phoneme produced. Perceptual and analytical studies, such as Peterson and Barney[13], have shown that vowel categories can be well separated by formant frequency locations. In speech synthesis work it has been demonstrated that the frequency locations of the lowest three formants is sufficient to produce intelligible speech[12]. Since formants represent the position of articulators in the vocal tract it follows that the position of the formants is related to the sonorant produced.

A spectogram, of the letter R ([aa] [r]), is included in Figure 1. At the top of the display is the waveform of the acoustical energy. From this waveform,

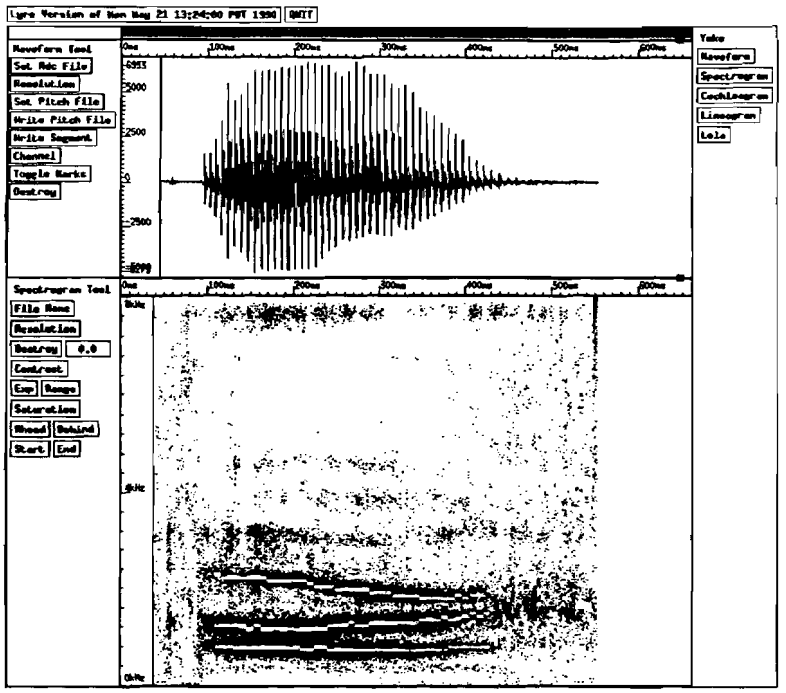

Figure 1: Waveform and pitch-synchronous spectogram of the letter R, male speaker

successive periods are calculated, and this information is used to generate a pitch-synchronous DFT (PSDFT). A PSDFT is a frequency-time display of the energy in the acoustical waveform. The dark bands of energy are the formants. In this utterance we can see F1 steady, F2 rising, and F3 falling. At the very end of the utterance we can see F2 and F3 merging as the energy fades off. Above the dark band of F3 we can see the faint band of F4, F5 and even F6. In this case F4 and F5 are below 4kHz. The white bands superimposed over the formants are the formant peaks found by the formant estimation algorithm. The highlighted formant tracks correspond to the formants visible in the spectogram.

A neural network can be viewed as a graph, with ordered layers of nodes.

Each node is fully connected to the previous and next layers. The connection between nodes is used to transmit the activation of the node to the next layer. There is a weight associated with each connection that modifies the activation sent over that connection. Each node performs some simple calculation, for example summing all the inputs with an output of 1 if the sum is over some threshold value.

One of the great strengths of neural networks has been in classification. We sought to apply the classification ability of neural networks to the formant estimation problem. The ability to generalize individual cases from noisy data would enable a formant estimation algorithm to assign labels to spectral peaks, and then use that label assignment to estimate the formant frequencies.

1.2 Issues

Formant estimation is a difficult problem because of variation in frequency, merged formants, split formants, and fading formants. Formant frequencies vary between speakers because of the different vocal tract sizes. In addition, formant frequencies will vary greatly between different sonorants, even for the same speaker. Since the articulators are in motion the shape of the different vocal tract cavities can become similar, so the formants may merge to form a single peak (F1-2, or F2-3). When air is diverted through

Formant	*Range (kHz)*
F1	0.1-1.5
F2	0.4-3.1
F3	1.4-3.9
F1-2	0.4-2.1
F2-3	1.0-3.7

Table 1: Formant Frequency Range for a Sample Dataset

the nasal cavity an anti-resonance is formed that creates a zero in the spectra of the F1. In a spectogram, this zero appears as white space that splits F1. Finally, as the different vocal tract cavities change shape, different amounts of acoustic energy are produced. This may result in a formant that disappears for a few frames. Coarticulation effects between adjacent vowels can produce even greater formant variance. All of this variance can greatly affect the frequency range of the formants. Table 1 shows the overlap in the first three formant frequencies (from the locally produced ISOLET dataset).

1.3 Goals

Our goal was to use the neural network to assign labels to spectral peaks, and then use those labels to estimate the formant locations. Neural networks have shown their ability to make classifications from noisy data. We expected

the neural network to use this ability and generalize characteristics from the training data. We had a secondary goal to determine whether knowledge-based features, or raw data (spectral coefficients) produced better neural network classification of spectral peaks.

1.4 Previous Work

Our work diverges from previous work in one major aspect. We use the neural network classifier to directly assign formant labels to spectral peaks. Previous work attempts to identify a spectral peak by finding the most probable label using either rule based constraint satisfaction or hidden markov models.

1.4.1 Rule Based Slot Filling

The work of McCandless is an example of a rule based system[11]. McCandless uses Linear Predictive Coding (LPC) for her speech processing. LPC is a model where each coefficient represents a complex pole. The resolution of the analysis is controlled by varying the number of coefficients (the more coefficients, the better the resolution). Candidate peaks are identified in the LPC coefficients, starting at the center of the syllable and working outward. Each LPC frame is viewed as having one slot for each of the first three formants. As each peak is found it is used to fill a formant slot, if the peak meets certain frequency and energy criteria. In the best case, the three strongest

peaks will coincide with the first three formants. Because of the variability in the formants discussed above, three peaks are not always found, or more than three peaks are found. In that case, a series of rules are algorithmically applied to resolve the conflicts. For example, in the case of a merged peak, one slot will go unfilled. The algorithm must identify it as a merged peak, and then fill in the remaining slot according to a predefined rule.

1.4.2 Hidden Markov Models

An example of formant tracking with HMMs is the work of Kopec [7, 8, 9]. Kopec uses Vector Quantization (VQ) for his speech processing. VQ considers each frame of LPC coefficients as a vector. VQ reduces the redundancy in the LPC spectra by mapping similar coefficient vectors onto the same codeword. This reduces the possible encodings of the speech signal to 2048, 256 or even 64 codewords.

A HMM is a finite state machine, where the transitions between states are made based on probabilities determined by the observed input. These probabilities are determined by training the HMM on representative data. As sequences are seen in the training data, the transition probabilities are calculated based upon the observed likelihood of these sequences.

For formant tracking, the states of the HMM represent the possible formant locations, i.e. each state represents a LPC coefficient. The observed sequences of VQ codewords in the training data are presented to the HMM.

The transition probabilities are calculated based on these observations. For a sequence of input frames, the most probable path through the HMM represents the formant track.

1.5 Outline of Thesis

In Chapter 2, we present an overview of the approach and describe the most successful formant estimation algorithm from our experiments. In Chapter 3, we describe the experiments that led to the best algorithm. The performance of the algorithm with different features and network configurations is also discussed. In Chapter 4, we evaluate the performance of the algorithm and it is evaluated against human performance on the same task. In Chapter 5, we discuss future research directions.

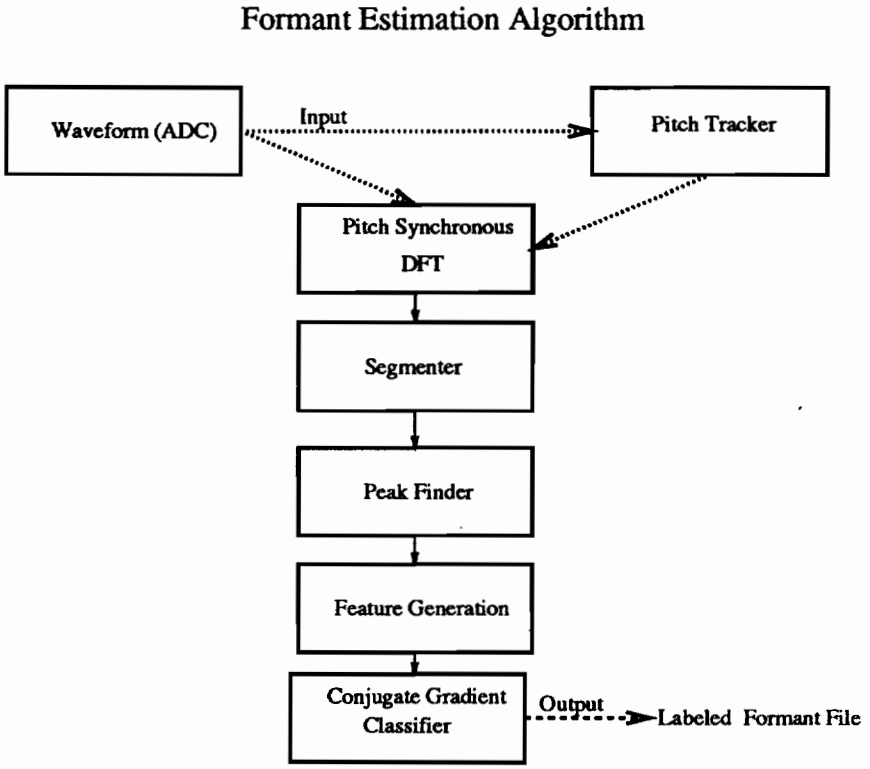

Figure 2: Formant Estimation Algorithm

2 Overview

The processing steps that are used to assign formant labels to spectral peaks in sonorant intervals are shown in Figure 2. We apply a peak-finding algorithm to a Pitch-Synchronous DFT to detect candidate formant peaks. To classify these peaks we generate features that were found to be important for formant labeling. These features are then used as inputs to a neural network classifier which labels that peak as NotF, F1, F2, F3, merged F1-2, or merged F2-3.

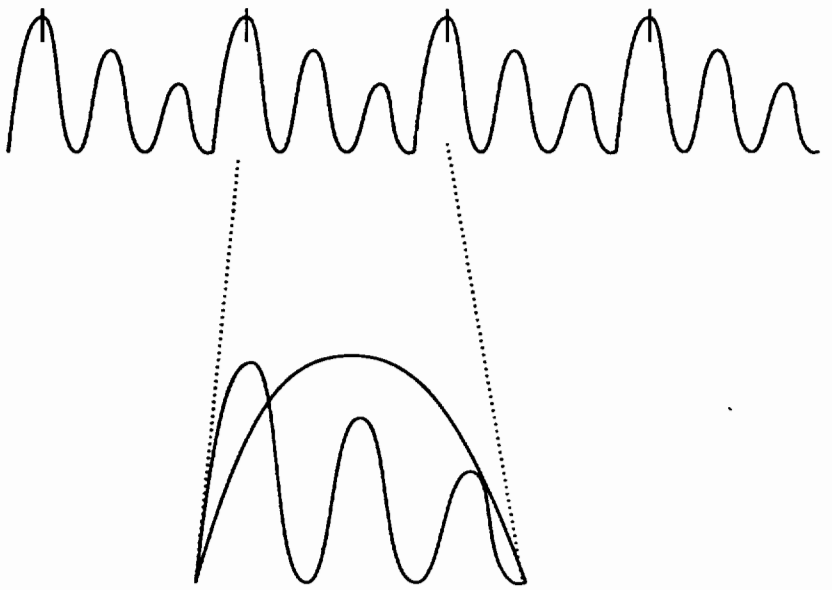

Figure 3: Pitch aligned Hanning Window over the acoustic waveform to generate a pitch-synchronous DFT

2.1 Pitch-Synchronous DFT

We use a pitch-synchronous discrete fourier transform (PSDFT) because it gives better resolution of the spectral peaks. The basis of this transform is the DFT. A pitch synchronous DFT is created by aligning a Hanning window to successive pitch periods (as shown in Figure 3), replacing the fixed window size and window increment normally used. Thus, the DFT is performed every pitch period. If the pitch tracker does not find a pitch period, then a constant increment DFT (10ms window with a 3ms increment) is used until another pitch period is found.

A neural network pitch tracker provides the pitch estimates[1]. The pitch

tracker was trained to discriminate peaks that begin pitch periods from peaks (in the acoustic waveform) that do not begin pitch periods.

2.2 Segmentation

We are interested in the formant frequencies within sonorants. Sonorant intervals were found using a rule-based segmenter that provided segmentation and broad classification of the utterance[4]. For example, a pitch period, also marked by high peak to peak amplitude in the waveform, will indicate a sonorant, or a high zero crossing rate in the waveform indicates frication. This segmenter reliably detects the sonorant onset and offset so it is adequate for the formant estimation research.

2.3 Peak Finding Algorithm

To assign formant labels to spectral peaks we must first find the spectral peaks. We smooth the spectra in both frequency and in time. This smoothing is accomplished by using a weighted average (0.25 0.5 0.25) of each coefficient and the adjacent coefficients. The effect of this smoothing is to remove spurious peaks. A peak finding algorithm was developed at Carnegie Mellon University that locates all peaks below 4kHz. A peak is defined as a local maximum value that has a 3dB fall on both sides. The 3dB fall criteria was chosen empirically. The peak finding algorithm provides the frequency

location and amplitude of each candidate peak, for the six largest candidate peaks in a spectral frame.

2.4 Feature Measurement and Normalization

A neural network requires a basic representation of the information in a spectral slice. Knowledge-based features were determined by experiments described in section 3.1. The feature values were normalized from -1 to 1 by finding the maximum and minimum spectral coefficient values in the spectral frame, and then normalizing all the values by the difference of the maximum and minimum. We present the features of each peak to the network. In this case important information can be explicitly presented to the network, allowing the network to learn the important distinctions in that information.

We hypothesized that the feature-based approach was superior to raw spectral coefficients because of inherent complexity in the formant labeling task. To confirm this hypothesis, our preliminary experiments were designed to investigate the proper feature set, and to compare these features to raw coefficients. The results of these experiments confirmed that a feature-based approach was superior. The features for each peak that we found most useful are:

- Frequency Location of the Peak
- Amplitude of the Peak

- Width of the Peak, measured by the upper and lower falloff of the peak
- Interpeak Minima, Amplitude and Location

2.5 Neural Network Classifier

These features are used to create a feature vector which is then presented to a neural network for classification. The classifier is a fully-connected, feedforward, multi-layer perceptron that was trained using backpropagation with conjugate gradient optimization[2]. This algorithm is a modification of the standard backpropagation (BP) algorithm. A problem with BP is that there are parameters, such as momentum, that must be determined empirically for each data set. Adjusting these additional parameters may slow training further. The conjugate gradient training algorithm replaces these additional variables by using information derived from the error surface. This information is data dependent, and in essence, automatically sets the manual parameters of BP. Since these parameters are automatically determined from the data, training can proceed much more quickly than in BP.

A three layer network is used in the algorithm. There are 77 input nodes, 30 hidden nodes, and 6 output nodes (one for each of the six possible labels). The input vector provides the amplitude, frequency location, and upper and lower width measures for each peak. The interpeak minima are represented by their amplitude and frequency location. Up to 6 peaks in the target frame

are included in the vector to provide context. Because the vector is shifted across the inputs, there are additional input features for this context. A complete description of the network is included in Section 3.3.9.

3 Experiments

A series of experiments were performed to develop and evaluate the feature set. We also tested the performance of raw spectral coefficients against the performance of selected features. The second set of network experiments were conducted to evaluate the best neural network configuration.

3.1 Feature Experiments

The purpose of the initial series of experiments was to investigate the best set of features, and to develop the necessary software support. The initial set of features was established by determining the important information used by human labelers. These features include: peak location, peak amplitude, peak width, interpeak minimum (both location and amplitude), and median pitch.

Peak location is critical in determining the formant label. Each formant has a frequency range. We found that it was the single most important information for classifying the formants. We used the index of the spectral coefficient as a measure of frequency. We used a 256 point PSDFT (128 real-valued coefficients). We were only concerned with information from 0-4kHz, so 64 coefficients covered the range of 4kHz resulting in frequency increments of 62.5Hz.

Peak amplitude is important for distinguishing non-formant peaks from

formant peaks, since formant peaks are stronger. For this feature we used the amplitude of each spectral coefficient measured in decibels.

Peak width is important for distinguishing merged peaks. The merged peaks tend to be wider, especially relative to their amplitude. We first used the location of the 3 dB falloff provided by the peak finder. We also tried using a single number for the width (found by subtracting the index of the width features), which was less successful. We finally used a derivative based measure of width to better capture the shape of the spectral peak. This feature was calculated by using the frequencies with the maximum value for the first derivative of the spectral shape on either side of the peak. Of the basic features, width was the most difficult measure to find a suitable representation.

The interpeak minima are important because they help define the overall shape of the spectral peaks. For example, peaks about to merge have less distinct (the minima is not as low) interpeak minima, where the minimum between fully split peaks tends to be very low.

Median pitch is important because the formant locations will vary with the size of the vocal tract. Generally, the longer the vocal tract the lower the pitch.

3.1.1 Data

We used utterances from the TIMIT database (the locally produced ISO-

Label	Training	Testing
NotF	2812	850
F1	2666	684
F2	2553	586
F3	2389	594
F1-2	2464	582
F2-3	2697	789
Total	15581	4085

Table 2: Number of Labels used from TIMIT Dataset

LET was not ready), a standardized continuous speech database of English language sentences [6, 10]. We used 80 utterances in the training set, and 20 utterances in the test set. The signal processing environment used for both datasets was similar.

If a class in the training set has fewer instances (by an order of magnitude) than the other classes, then the neural network cannot learn that class. To get balanced numbers of training instances for each label, we sampled the input data files. We used the following percentages of each label:

- 5% of NotF labels
- 7% of F1 labels
- 7% of F2 labels

- 7% of F3 labels
- 50% of F1-2 labels
- 50% of F2-3 labels

After sampling, the number of each label in the training and testing sets is presented in Table 2.

3.1.2 Summary of Feature Experiments

The network used in these experiments was the Repeated Target network (Figure 5), it is described in detail in Section 3.3.3. We were interested in the contribution of the various features. There were two reasons for this interest. One, we did not want to use any features that were not helping to distinguish formant labels. Two, we were interested in the relative importance of the features. The remaining preliminary experiments were oriented to those goals. The results of the Feature Experiments are summarized in Table 3.

3.1.3 Basic Approach

Our first experiment consisted of training a network using all of the basic features except for median pitch. In this experiment the locations of the 3 dB falloffs on either side of the peak were used as a measure of width. The network was able to correctly label 87% of the formant peaks in the test set.

Amp	49.60%
Freq	67.96%
Amp & Freq	84.51%
Amp & Valley	62.30%
Freq & Valley	70.80%
Amp, Freq & Valley	85.36%
Amp, Freq & Width	86.50%
Amp, Freq, Width & Valley	86.92%
All & Pitch	89.22%
64 Coefficient	78.46%

Table 3: Summary of Feature Experiments

3.1.4 Amplitude Only

For this experiment we trained a network using only the amplitude values of the peaks. Because the amplitudes were presented in peak order, there was implicit frequency information in the ordering of the peak amplitudes. This network was able to successfully label 49% of the formant peaks.

We found this result interesting. With only the normalized amplitude of the peaks and their relative ordering, the network was still able to successfully classify half of the peaks. That is three times better than chance. We found that a testament to the power of neural network classifiers.

3.1.5 Frequency Only

The next experiment involved training a network using just the frequency coefficients. Because of the formants' frequency range overlap (see Table 1), it would be interesting to see how well a network could distinguish formants with only frequency information. This network was able to successfully label nearly 68% of the formant peaks.

This result was about what we expected. Frequency information is more specific than amplitude with relative ordering.

3.1.6 Frequency and Amplitude

In this experiment we trained a network using both frequency location and amplitude for each of the formant peaks. We expected this network to do better than the individual networks, since the explicit frequency information would help classify the formant peaks, and the amplitude information would help reject non-formant peaks. This network successfully labeled nearly 85% of the formant peaks.

This result was a little surprising. It was performing nearly as well (within 2%) of the network with the full feature set. These two features were accounting for nearly all the performance of the network.

3.1.7 Interpeak Minima

In this experiment we wanted to investigate the utility of the valley features (the interpeak minima's location and amplitude). We trained networks using the last three feature sets (amplitude individually, frequency individually, and both frequency and amplitude) adding the valley features to each. Not surprisingly it helped the amplitude-only network the most, with an improvement of 13%. This improvement was most likely caused by the extra frequency information implicit in the valley frequencies. The valley on either side of an amplitude would put the location of the peak somewhere between the frequencies of the valleys.

The frequency only network was improved only by 3%. This small improvement is probably due to the implicit width information in the valley separation.

The network using both features and valleys was improved by less than 1%. This small improvement is probably because there was very little extra information provided by the valleys. In this case, the only extra information would be implicit width.

3.1.8 Width

We were now interested in the importance of width. The next network used the amplitude, frequency and width features. First we ran a series of sub-experiments to find the best width feature. We empirically determined that a derivative based width feature was better than the 3 dB falloff provided by the peak picker. For these experiments we found the point on either side of the peak where the second derivative of the spectral waveform was 0. This change improved the performance of the width only network by 3%.

The width feature improved the frequency and amplitude combination by 2%, which was within 0.5% of the network performance using all the features. Both width and valley features improved the network's performance. There was much overlap in the improvements so they, expectedly, are providing similar information. They provide a slight improvement in combination, so they are not providing exactly the same information.

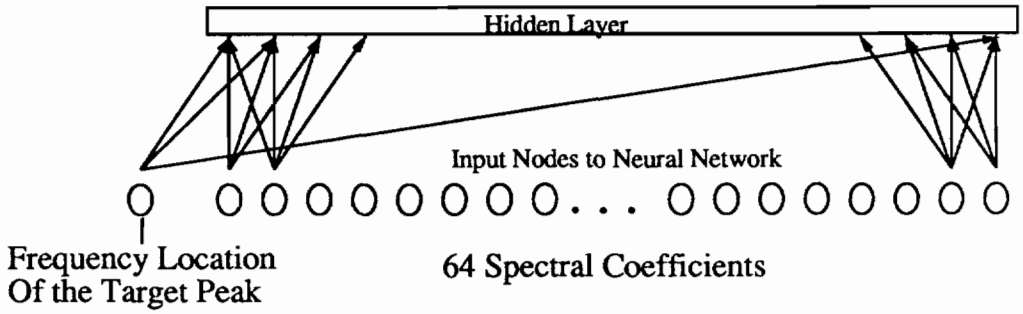

Figure 4: Spectral Coefficient Network (the input to the neural network is the 64 Spectral Coefficients and the Frequency Location of the peak)

3.1.9 Pitch

This was the final experiment in our exploration of the feature set. We took the full set of features and added pitch. Because much of the variation in formant location is due to differences in vocal tract size which is related to pitch, we expected this feature to significantly help the network. With pitch added the network successfully labeled over 89% of the formant peaks. Initially this result appears disappointing. It is only a 2.5% improvement. But it is actually reducing the error by 18%.

3.1.10 Spectral Coefficients

In the ongoing debate about neural networks, a key issue is the amount of processing that should be done to information before it is presented to the network. Our feature-based approach obviously requires much processing of the raw data. To test the validity of this approach we trained a network that used the 64 raw coefficients(Figure 4). They were normalized from 0-1, by subtracting the minimum amplitude in the frame from all values, and then dividing these modified values by the modified maximum value in the frame. Then the location of the peak found by the peak picker was used to designate the peak location for the network. This network was able to successfully label 78% of the formant peaks.

3.2 Discussion of Feature Experiments

The initial feature selection was determined by the information human labelers use to track formants in spectograms. The interesting result of our feature set experiments was that that initial features was also the final set of features, and that all of them provide some information to the network, that is, they improved the performance of the network.

3.2.1 Frequency

Frequency is obviously important for formant labeling. It is probably the single most important feature, which our experiments confirm. There is some overlap in the frequency range of formants, and for human labelers, the order of formants is usually sufficient to resolve formants that fall into the frequency range overlap.

Visual inspection of the errors indicates that the network has learned some internal representation of this ordering. In cases where the peak finder misses F1, the network still tries to assign an F1 label even if the next peak is well above the normal range of F1.

3.2.2 Amplitude

That the network learned ordering information was apparent from the amplitude-only experiments. In these experiments, the amplitude of the 6 peaks in a frame, and their relative ordering were provided to the network. The network still labeled nearly 50% of the peaks correctly. The only information that amplitude directly supplies, is the energy contained in a peak which should help in detecting formant peaks, not labeling them. With only amplitude information, the network still assigned labels at a rate 3 times better than chance. The only information available to distinguish formants in this representation was the ordering of the peaks. It seems that the network learned

that the first candidate peak was F1. That the network did no better, is indicative that spurious peaks can have formant-like characteristics.

3.2.3 Width

The peak finding algorithm used a 3 dB fall on either side of a maxima to define a peak. Although this definition was adequate for peak finding, preliminary experiments revealed that the 3dB fall was not a good feature for classification. We then tried several derivative-based methods to find a better approximation of the peak width. The best measure was the location where the second derivative of the spectral peak was a maximum. This put the width measure well out on the shoulder of the peak. Visual inspection revealed that this measure was also less susceptible to minor variations in the spectral coefficients.

Width had a minor effect on the performance of the classifier. Considering the other characteristics that the network learned (i.e. ordering, 3 peaks per frame), this is not a surprising result. The difference in performance by adding width was so small it is difficult to attribute the improvement to a specific classification. Width appears to help discriminate merged peaks, because there are significant variations in width between merged, and non-merged formants.

3.2.4 Interpeak Minima (Valleys)

Since we used a width feature, it did not seem that the valleys were helping define the size of the peak. They do provide some information about the shape of the spectral curve. Actual formant peaks tend to have distinct low valleys between them, except for formants that are about to merge. Even then, the valleys are more distinct then valleys around spurious peaks. Visual inspection of errors revealed no pattern to the classifications the valleys helped. That they helped implies that the network found some useful information. Unfortunately neural networks do not always use the same classification features that humans use. They sometimes develop a unique perspective, and that is apparent in the case of valleys.

3.2.5 Combinations of Features

There are some subtle interactions among these features. Due to small differences in performance, it is not always possible to analyze which features are acting in concert with other features. For width, we found that the frequency location of the peak shoulders performed better than a simple value representing the difference of those frequencies. The shoulder location also gives the network information about the skew of the peak, and the shape of the slopes. It appears that the network found useful information in the shape of the spectral curve as represented by the features. Since that information

is also available in the raw coefficients and they did not perform as well, it seems that the raw coefficient network was unable to extract all of the important information from the coefficients, at least with the size of networks and amount of training data used in these experiments.

3.3 Network Experiments

The preliminary experiments established the most useful feature set. The purpose of the next set of experiments was to determine the most useful network configuration. The problem was how to best correlate the target peak with the other values in the input vector. That is, the network must be able to distinguish the target peak values from the other values in the input vector representing context.

3.3.1 Data

Except for some initial experiments to ensure continuity, all of these experiments were conducted on the ISOLET (Isolated Letter) Database[3]. The training set had 7 utterances from 20 speakers (140 utterances total), and the test set had 7 utterances from 10 speakers (70 utterances total). For each speaker there was an utterance for each of the sonorants found in the spoken English alphabet; [iy],ey],[eh],[aa],[u],[o], and two sonorants in the letter W. To reduce the number of vectors presented to the neural network, this data

Label	Training	Testing
NotF	913	252
F1	849	214
F2	708	183
F3	815	205
F1-2	815	194
F2-3	1076	286
Total	5176	1334

Table 4: Number of Each Label used from ISOLET Dataset

set was also sampled and the number of each label is presented in Table 4.

For all of these experiments, the same features were used. The goal of these experiments was to test the network configuration, and we needed a constant feature set to determine if changing the network configuration was affecting the performance. The sole exception was an additional experiment to try a new width feature using the new network configuration. Table 5 is a summary of the network experimental results.

3.3.2 Repeated Target Network

The feature vector was always presented to the same input neurons; however, as the target peak changed the input neurons would have a different function. In Figure 5 for the first peak in the frame the square neurons receive the

Network Configuration	*Performance*
Repeated Target with pitch	87%
Shifted Vector with pitch	90%
Individual Formant Specialist	88%
Individual Peak Specialist	84%
Column Activations	82%
SV with pitch and new width	91%
Smoothed Spectra	92%

Table 5: Summary of Network Experiment Results

target peak features. For the second peak, these same neurons now receive the lower context peak features. As each new peak in the frame is presented, the function served by these neurons changes. By the sixth and last peak, these neurons now serve the relatively minor function of distant context. This changing function inhibits the neurons' ability to generalize.

For this experiment the target peak was indicated by repeating that peak's features at the beginning of the feature vector (Figure 5). This resulted in a feature vector with 38 elements that was used for the preliminary experiments. This network consisted of 38 input units, 15 hidden units, and 6 output units. The network successfully labeled 87% of the formant peaks.

There were three classes of error noticed in the labeled output of this

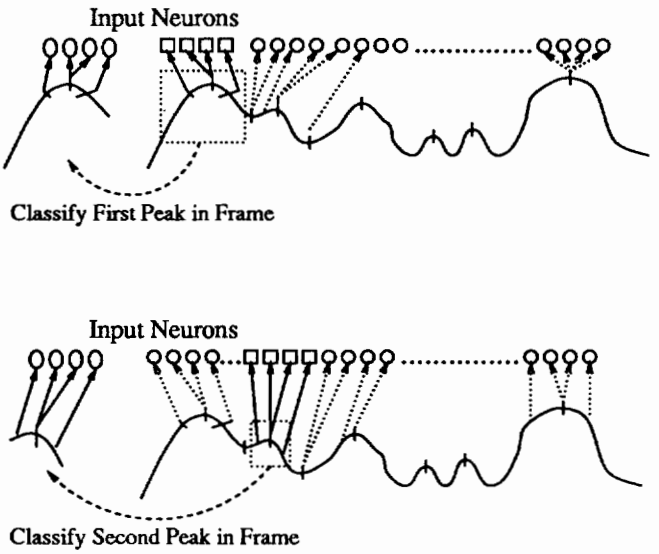

Figure 5: Target peak features repeated in front of the feature vector network. There were:

- Duplicate labels in each frame, for example two F2 labels.
- A low F4 was mislabeled as F3, which also caused some duplicate labels within a frame.
- Inconsistent labelings, either within a frame or between frames. For example, a frame with a F1 label and a merged F1-2 label.

The Repeated Target Network did not present the target peak features to the same input neurons. This appears to have been interfering with the ability of the network to generalize.

3.3.3 Shifted Vector Network

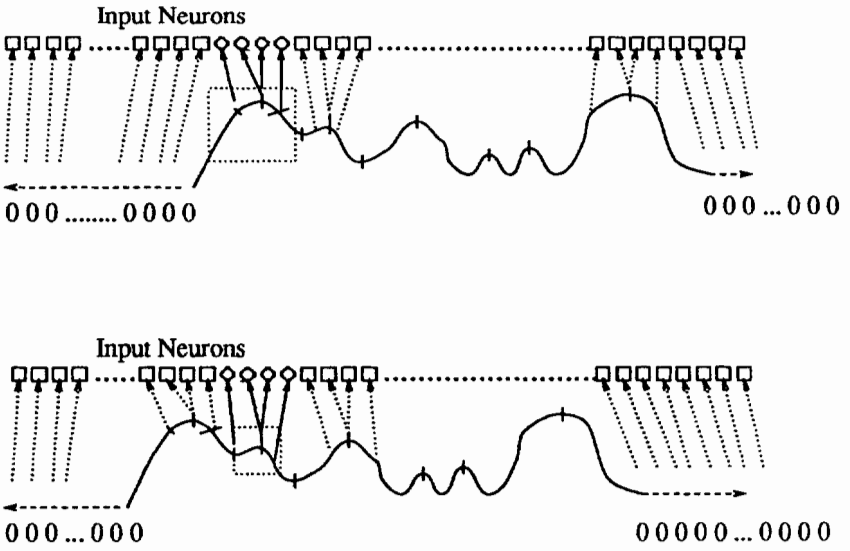

Figure 6: Shift feature vector to keep target peak features under the same inputs

We were not comfortable with repeating the target features as a method for indicating the target peak. To test the assumption that this representation was inhibiting the network, we modified the representation. In the new representation (Figure 6) the target features were not repeated. Rather the feature vector was shifted across the input nodes so that the target features were always aligned under the same nodes. These nodes could then specialize as "target features". The nodes with features from peaks above and below the target could then specialize as context features. This Shifted Vector representation made the relative ordering of peaks explicit. It was felt that this would eliminate some of the duplicate label errors found in the initial representation.

Since backpropagation requires the same number of input nodes, it was then necessary to pad the ends of the feature vector with empty "peak values" to produce the full input vector. As the feature vector was shifted for each successive peak, zeros were added below the feature vector and removed from above the feature vector so the total input vector length was constant. This increased the size of the network to 76 input units, 30 hidden units, and 6 output units.

For both networks (Repeated Target and Shifted Vector) we ran empirical studies on the number of hidden nodes required. Unfortunately, for this critical area of neural network design, there are no established methods. For both networks, the number of hidden nodes was varied from 10-50. For the Repeated Target Network 15 hidden nodes was found to provide the best result. For the Shifted Target Network 30 hidden nodes were found to provide the best result.

This network was able to successfully label 90% of the formant peaks. Although only a 3% improvement, this represents a 25% reduction in error. This representation was superior to the initial representation. A visual inspection of the errors revealed that the occurance of duplicate labels was almost insignificant. In addition, there were fewer occurances of mislabeled F4.

The network's ability to avoid duplicate labels is interesting. It is impor-

tant to remember that when each peak is labeled it is presented in isolation to other labels. That isolation means that the network does not have the information that it had previously labeled a peak as F1 in the same frame. Since it was avoiding duplicate labels when the previous network did not, it seems that the network was developing an internal representation of the entire frame, and at least implicitly labeling the other peaks.

Since the target features were presented to different input neurons, the network had to learn the additional mapping of target location in the input vector. Since the target vector was now shifted under the input neurons, and a back-propagation style network needs a constant number of inputs, the input vector had to be padded to fill in the empty elements. This context on either side of the peak helped the network. We ran experiments by adding context of 1 through 5 adjacent peaks. The network did best when 5 peaks were added. This is not surprising, since only with the context of 5 adjacent peaks is the entire input vector available to all shifted vectors.

This network learned the characteristics mentioned above; ordering, number of peaks. This generalization is a function of having the whole frame available, and knowing the position within the frame explicitly (represented by the amount of input vector on either side of the target).

3.3.4 Shifted Vector Network (with pitch)

The Shifted Vector representation was an improvement over the Repeated Target representation. Since frequency location variance is related to pitch (pitch varies with the size of the vocal tract), we felt that adding pitch as a feature would improve the performance of this representation. We were especially optimistic because the remaining classes of error, low F4 mislabeled as F3 and inconsistent combinations of labels, could be explained at least in part by the frequency overlap of the formants. This increased the size of the input vector by one, so the network configuration was now 77 input units, 30 hidden units, and 6 output units.

Adding pitch to the Shifted Vector representation improved performance, but not by much. The improvement was only 0.3%, compared to 2% improvement with the repeated Target Network, which could also be accounted for by random variation. There was no noticeable change in the class of errors made by this network.

This result is puzzling. The only possible explanation is that the relative ordering of the peaks is as useful as pitch in discriminating formant labels.

3.3.5 Individual Formant Specialist Network

It is possible that the ambiguity and complexity of the labeling task was interfering with the network's ability to generalize. To test this hypothesis

we wanted to reduce the size of the problem. Our first attempt was to train individual networks that specialized on individual formant labels. The same vector configuration was input to the network. The difference was 2 outputs instead of 6 outputs, so the size of the network was reduced to 77 input units, 10 hidden units, and 2 (it is or is not the desired label) output units. Since there are 6 labels, we needed 6 networks in place of the previous single network. The performance of the network was disappointing. The main reason for the poor performance, 88%, was error introduced by arbitrating between the different networks when they had contradictory output. For example, the F1 and F1-2 networks might indicate the same peak. Several methods to resolve the conflicts were attempted, and none were satisfactory.

3.3.6 Individual Spectral Peak Specialist Network

We tried a second approach to providing invariance to the target features. Instead of shifting the feature vector with each successive peak, a single network could be trained for each peak (i.e. lowest peak, second peak, highest peak), therefore 6 networks were required(Figure 7). This representation would reduce the size of each network. The network size was 35 (down from 77) input units, 10 hidden units, and 6 output units. The performance of these networks was disappointing. They successfully labeled only 84% of the formant peaks. This approach suffered from the same problem as the individual formant network: arbitration between labels. There was an

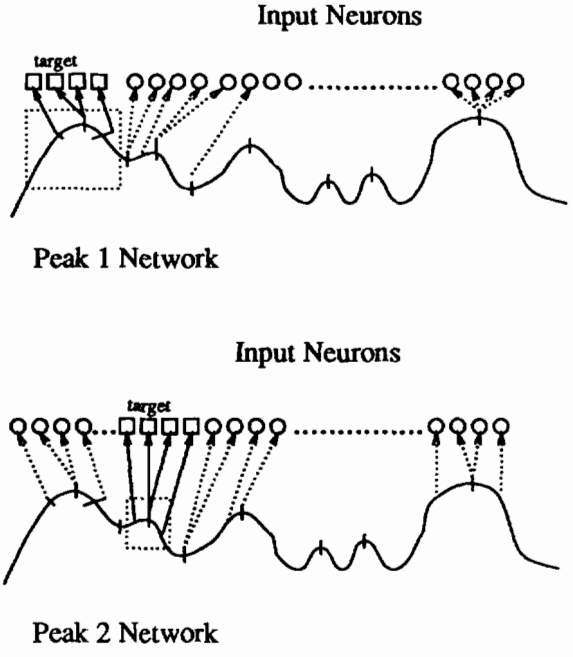

Figure 7: Individual Peak Network (6 networks for each of 6 Peaks)

additional problem caused by an imbalance of training examples. For each peak there would be very few examples of one or two labels in the training set. Their numbers were so small that the networks could never learn to classify them. For example, the second peak training set only had six F2-3 labels compared to several thousand F2 labels. For any reasonably sized training set, there were at least 1% of the labels presented to each Peak Specialist Network that were unbalanced. Therefore the networks could never learn these labels, although increasing the training set size might help.

3.3.7 Column Activation Network

This experiment did not involve training a new network. It involved looking at an old network in a new way. For a given spectral frame the output

Weight Activation Matrix for a Single Frame

Original Method Find Maximum in the Row for each Peak.

	0.1253	0.7936	0.5823	0.0021	0.2491	0.0141	Peak 1
	0.9782	0.2713	0.1987	0.0978	0.2193	0.0762	Peak 2
	0.3462	0.3349	0.7459	0.1826	0.2912	0.1037	Peak 3
	0.9826	0.1428	0.2317	0.2941	0.0893	0.1092	Peak 4
	0.8921	0.0963	0.1535	0.3874	0.0728	0.2312	Peak 5
	0.2194	0.0876	0.1066	0.7903	0.0818	0.2966	Peak 6
	Not-F	F1	F2	F3	F1-2	F2-3	

Column Activation Find Maximum in the Column for each Label.

	0.1253	0.7936	0.5823	0.0021	0.2491	0.0141	Peak 1
	0.9782	0.2713	0.1987	0.0978	0.2193	0.0762	Peak 2
	0.3462	0.3349	0.7459	0.1826	0.2912	0.1037	Peak 3
	0.9826	0.1428	0.2317	0.2941	0.0893	0.1092	Peak 4
	0.8921	0.0963	0.1535	0.3874	0.0728	0.2312	Peak 5
	0.2194	0.0876	0.1066	0.7903	0.0818	0.2966	Peak 6
	Not-F	F1	F2	F3	F1-2	F2-3	

Figure 8: Output activation matrix showing 2 methods to assign labels, choose the best label for each peak or choose the best peak for each label)

activations can be thought of as a matrix (Figure 8) with the peaks along one axis (the Y-axis in this case), and the possible labels along the other axis (X-axis in this case). Originally, the rows were used to select the highest activation for the possible labels for that peak. In this experiment, the columns were used to find the peaks that had the highest F1, F2, and F3 activations. This method ensured that each frame had at most one of each label. In the previous method, using rows associated with each peak, it was possible, and not uncommon, to get two F3 labels. Selecting the best activations by column has successfully labeled 82% of the formant peaks. Visual inspection of the errors reveals that this approach is very promising for spectra without merged peaks. For spectra with merged peaks, this approach encounters a serious problem with resolving conflicts between the merged and non-merged label for a given peak.

3.3.8 Shifted Vector Network with New Width

We made one last attempt at improving the performance of the width feature. We were not satisfied with any of the previous measures. The new feature had 2 changes. First the upper and lower cutoffs (shoulders) were defined as the points marking the middle 80% of the mass of the peak. The mass was found by taking the weighted average of the spectral coefficients. The upper and lower width cutoffs were found by calculating the index where 10% of the peak mass was above or below that index. This measure proved more reliable

since it was independent of the shape of the peak. Any measure based on the shape of the peak would encounter some situation where the curve of the peak would cause erroneous width markings. Second, we originally marked the width by giving the spectral index of the width locations. We felt that that might hide the more important information, namely the relative location of the width to the peak. We tired a method where the index was given relative to the peak location, i.e. +/- the difference in the coefficient index of the peak and of the width mark. Individually they improved the performance by 0.5%. In combination the 2 changes improved the performance of the network by 1% to 91%, which was 10% of the error.

3.3.9 Smoothed Spectrum

Visual investigation of the error revealed that there was a problem with distinguishing spurious peaks, especially at the higher frequencies in the F3 range. To reduce the number of spurious peaks we smoothed the spectra in time and in frequency. We used a simple 0.25 0.5 0.25 weighted average of each coefficient with the adjacent coefficients. This made a significant reduction in spurious peaks and enhanced some valid peaks, at the expense of a slight increase in the number of merged peaks. This smoothing improved the performance by 2%,to 92%, which was about 20% of the error. Smoothing the spectra resulted in the network with best performance. Interestingly, the new width measure did not improve the network performance with the

smoothed spectra.

3.3.10 Summary

We tried many different network configurations, although our second attempt, the Shifted Vector Network, performed best with 90% success. Investigation of the errors led us to re-evaluate the features used, and we tried several improved width measures, and only increased the performance by 1%. We then tried to improve the quality of the spectra used as input and applied the smoothing of the spectral coefficients. The smoothing increased performance by 2%, and the improved width measures had little effect on the performance of the network. The Shifted Vector Network with pitch using the smoothed input gave us the best result, 92%.

4 Performance Evaluation

4.1 Performance on Continuous Speech

We were using the isolated letter dataset to develop the network configuration. The initial feature experiments used the TIMIT standardized dataset of continuous speech. When we changed datasets we trained the same network configuration and feature set on both datasets for continuity. We were surprised that the performance on the TIMIT dataset was 2% better. Since continuous speech is more difficult we were interested in why the performance was better. To verify this result we later trained the Shifted Vector Network on the TIMIT dataset, and the results were still 2% better, 92% correctly labeled peaks.

There are 2 possible explanations for the better performance. The recording environment of the TIMIT dataset may have been sufficiently different, and consequently the utterances produce more distinct spectral representations. The other involves training the neural network. To generalize classes, there must be a sufficiently large and varied training set. With letters of the English alphabet, half of the sonorants are [iy] or [ey], both are very similar in their formant locations and transitions. It is possible that with the greater formant variation of continuous speech, the network was better able to generalize the formant labels.

4.2 Human Perception Experiments

The review of the literature reveals one glaring deficiency; human performance on formant estimation is never adequately documented. The reasons for not making such measurements are probably the same as the reasons that make formant estimation difficult (merging, splitting, and disappearing formants). We conducted limited human perception experiments to investigate the difficulty in measuring human performance, and to provide a performance measure for the neural network.

We gave experienced human labelers the same task as the neural network to establish a benchmark for comparison. We did not compare the performance against the hand-labeled files, because of the differences used to assign those labels. The hand-labeled formant files were labeled by consensus among the same three labelers. In that case, the full context of the formant file, and the spectrogram were available. The network is presented only a single spectral slice. To better measure the network's performance, that is what we presented to the human labelers.

For this experiment, each human labeler was given copies of 165 spectral slices, containing 898 possible formant peaks. Their task was to assign a label to each of the peaks, based upon only the information available in a single slice. The criteria used to score these tests were the same as those used to score the network's performance. If the label on a peak did not agree

Labeler	*Score*
# 1	88.9%
# 2	89.4%
# 3	89.9%
Total	89.4%

Table 6: Human Labeler Performance

with that of the reference formant file, then it was marked incorrect. The human labelers were able to correctly label 89% of the peaks (Table 6).

We acknowledge that the task presented to the network and the human labelers is not exactly the same. The network has no knowledge of what labels it already assigned to other peaks in the slice. Discussion with the human labelers indicates that even if asked to only label a single peak, human labelers tend to label the entire slice, and then select the label from the desired peak. This bias is ignored in these experiments for two reasons. First, it is difficult to remove. The whole frame is presented to the human labeler, and he needs that information. Second it was found to occasionally cause errors, as for example when F1 is mislabeled all the labels tend to be wrong. (It was also apparent that human labelers work bottom up in assigning the labels.)

There is some inherent ambiguity in assigning a formant label to a single

Labeler	# 1	# 2	# 3
# 1	NA	88.9%	89.4%
# 2	88.9%	NA	90.3%
# 3	89.4%	90.3%	NA
Total Between All			89.6%

Table 7: Agreement Between Human Labelers

peak in a spectral coefficient array. In some cases, a formant (even F2, or F3) splits, and either split peak can be justified as the correct peak. Two different experiments were performed to gain a feeling for the amount of this ambiguity. When the formant files were hand labeled, randomly selected files were labeled by separate labelers, and then their labelings were compared. In this case, the labelers had the full spectogram to work with in resolving ambiguity. These labeled files had about 98% agreement.

The second experiment consisted of taking the human labeling results of the perception experiment and measuring the agreement between the three human labelers. The results are presented in Table 7. With a full spectogram, experienced labelers have 2% disagreement. With only a single frame the disagreement is 10%. This disagreement establishes the ambiguity in the labeling task, and hence establishes a reference. Disagreement with the reference of more than 10% shows that performance can be improved. Disagreement

of less than 10% shows performance is better than expected. The classifier is finding some method of reducing the ambiguity, i.e. it is doing as well as the human labelers.

4.3 Comparison to Previous Work

In Section 1.4 we described previous methods for formant tracking. The Kopec HMM error rate is about 15%, the McCandless Rule Based system error rate is about 10%, and our neural network error rate is about 8% for the ISOLET data and 6% for the TIMIT data.

The Rule-Based system is tuned to 10 speakers, and could require significant re-tuning of parameters to generalize to new speakers[11]. The HMM and our neural network approach could be expanded by simply increasing the size of the training set. For our neural network classifier, even that may not be needed. In several cases we ran the trained network on new data (without retraining), and the performance decreased by only a few percent.

4.4 Analysis of Error

Most of the errors produced by the algorithm are reasonable. That is, the algorithm does not produce errors such as F2 below F1. Visual inspection reveals that many of the errors are the result of poor resolution on the spectograms. Browsing of errors indicates that when the network has difficulty

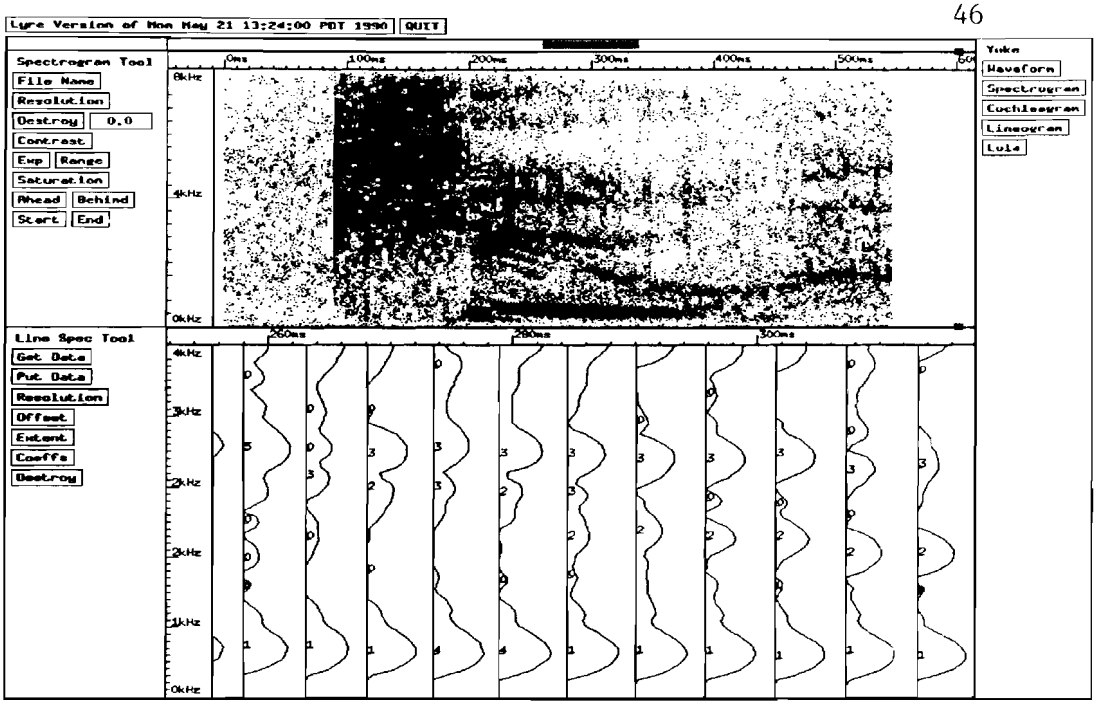

Figure 9: Spectogram 1 of the letter Q spoken by a female speaker

classifying peaks, human labelers also encounter difficulties on the same spectogram.

In the examples that follow we can see typical errors. In most cases the errors are reasonable given the context. The biggest source of error is a pitch-tracking problem that reduces the F3 and causes confusion between F3 and NotF. That problem is discussed in Section 4.6.

4.4.1 Spectogram 1

In Figure 9 we see a spectogram with pitch-synchronous spectra (lineogram) from within the area indicated on the spectogram. The spectra are displayed with the coefficients plotted horizontally. The display is only below 4 kHz since the formants of interest are below 4 kHz. The scale of the 2 displays

are different, so the LineSpecTool is only displaying the portion of the spectogram indicated by the dark bar over the spectogram time display. The displayed part of the spectogram is the middle portion where the F2 appears to fade away. Looking at the Lineogram, we can see that it does not fade away completely. Even with these faint peaks the neural network continues to correctly label those peaks. In the second frame it misses the F2, and the F3 is marked on the wrong peak. In the fourth and fifth frames the broadening of the F1 caused the classifier to mislabel the peak as a merged F1-2 (4 on the display). In the fourth and sixth frames it also inserts two F3 labels. Given the diminished F2 through this section, the errors are reasonable. In addition, towards the end of the lineogram, you can see that with the stronger F2, there is no error in labeling.

4.4.2 Spectogram 2

The next example is more troublesome (Figure 10). In this case, a Y spoken by a male speaker with low pitch, the F4 and even F5 are dropping below 4kHz. The section of spectrum displayed on the Lineogram shows how the extra formants confuse the classifier. Actually, as long as just F4 is below 4kHz, there is little problem. When F5 is a distinct peak (in the first 3 frames), the classifier labels F4 as F3 and consequently has an upward bias in labeling the other peaks (i.e. the network errs in assigning labels to peaks that are high in the frequency range). In the first 3 frames there is an extra

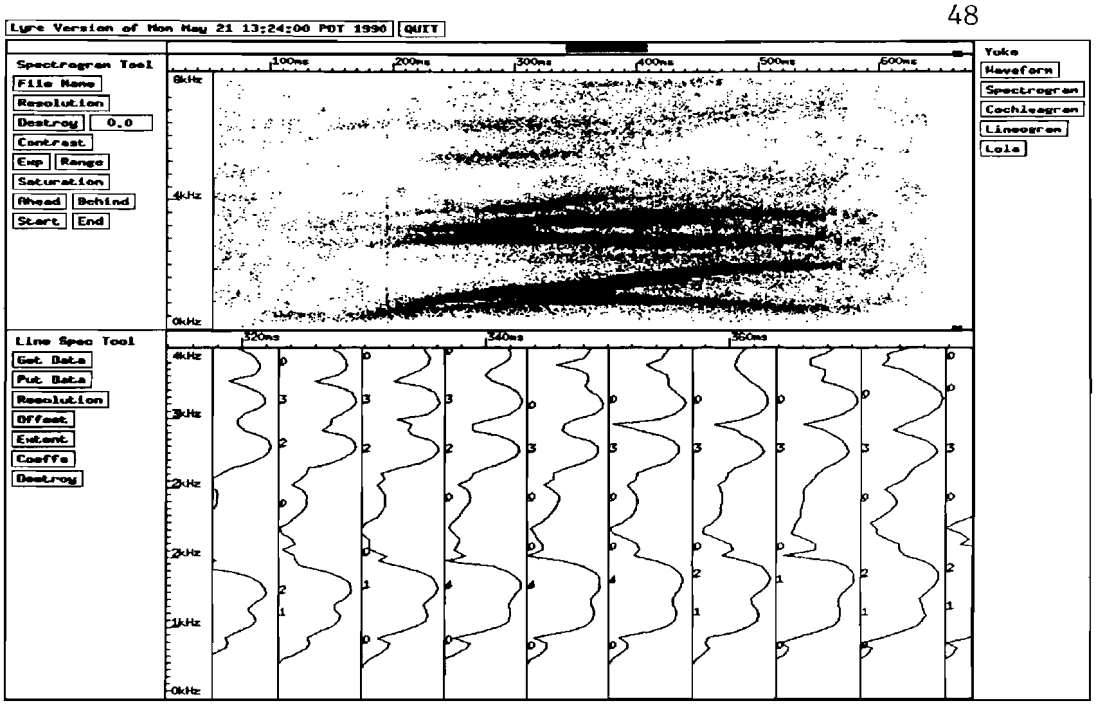

Figure 10: Spectogram 2 of the letter Y spoken by a female speaker

F2 label. In the second frame the merged F1-2 is mislabeled as a F1. The F4 is also causing confusion in the seventh frame since the merged peak is again labeled as F1. In this case it did not confuse the other peaks. Remember that each peak is labeled in isolation from the other labels in the frame. For each peak, the classifier does not know that the label was already assigned, or not assigned. Considering that there are 4 and even 5 formants presented to the network, it has reasonably sorted them out.

4.4.3 Spectogram 3

The next example is a situation where the network correctly labeled the spectogram (Figure 11). In this R, the F3 drops very low, and F4 is faintly visible. Looking at the Lineogram, we can see that there is a significant

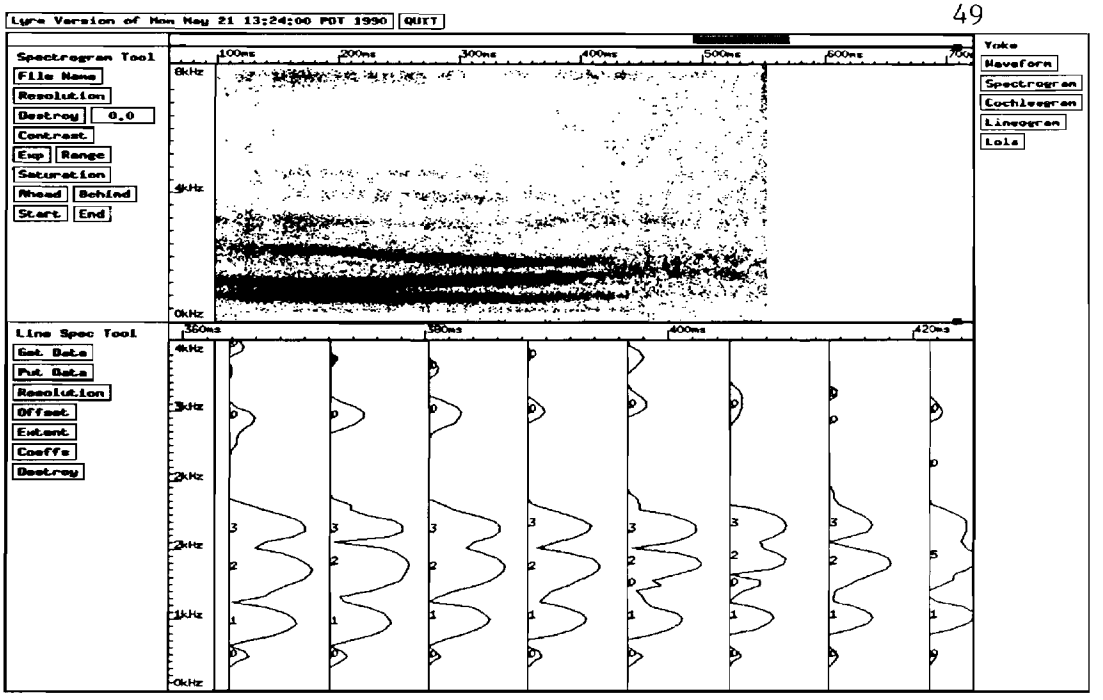

Figure 11: Spectogram 3 of the letter R spoken by a male speaker

difference between F4 and the other formants. The classifier correctly labels all the formants. It catches the merged F2-3 at the end of the Lineogram, and does not mislabel F1 as F1 starts to fade.

4.4.4 Spectogram 4

This is a case (Figure 12) of the formants widening just enough to allow the neural network to assign merged labels (4 and 5) to individual formant peaks. If you look at the center of the spectogram display, you can see the formants through the middle of the sonorant are slightly wider. The mislabeled peaks are isolated so this type of error is easy to correct with almost any kind of post-processing. Since there were only 2 other such errors in the utterance, even uncorrected they are not a major problem.

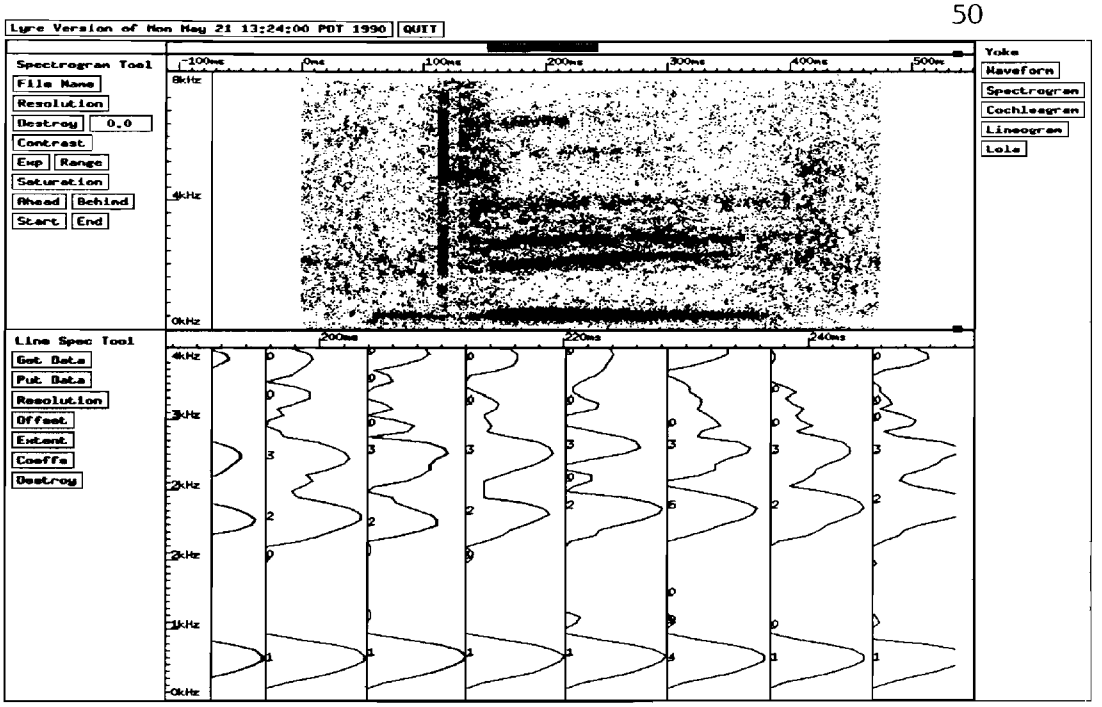

Figure 12: Spectogram 4 of the letter V spoken by a male speaker

4.4.5 Network Output

Table 8 displays the confusion matrix for the Shifted Vector Network. Inspection of this matrix reveals that the confusions are reasonable. F2 and F3, in particular, tend to be confused with No-Formant peaks. Considering the fall off in energy that is seen in spectra, that is only partially offset by pre-emphasis, the higher formant peaks tend to be smaller in the spectra. It is reasonable that these smaller peaks would get confused with No-Formant peaks.

The effect of the smaller peaks in the higher frequencies is also apparent in the confusions for F2 and F3. For the most part they are confused with the merged F2-3 label. Notice that the network as learned to distinguish F2

Label	NotF	F1	F2	F3	F1-2	F2-3
NotF	222	10	2	18	0	0
F1	3	207	0	0	4	0
F2	1	1	162	2	2	15
F3	13	0	3	180	0	9
F1-2	0	9	2	0	183	0
F2-3	2	0	12	14	0	258

Table 8: Confusion Matrix for Output of Best Network

from F1-2 rather well. The confusion of F2-3 is also with F2 and F3. The network does well in detecting the merged peaks, meaning it generally does not assign a No-Formant label to them.

F1 is generally only confused with F1-2. Again this is reasonable, considering the very slight differences between the two labels. The network is very good at detecting the F1 peak, and at least assigning some label to it.

4.5 Weight Magnitudes

The hidden node activations of a neural network can give an indication of the types of distinctions that the network is making. The weight magnitudes to and from the hidden nodes are one way of measuring the contribution of the node to a given classification. We use a tool, called XOPT, that will

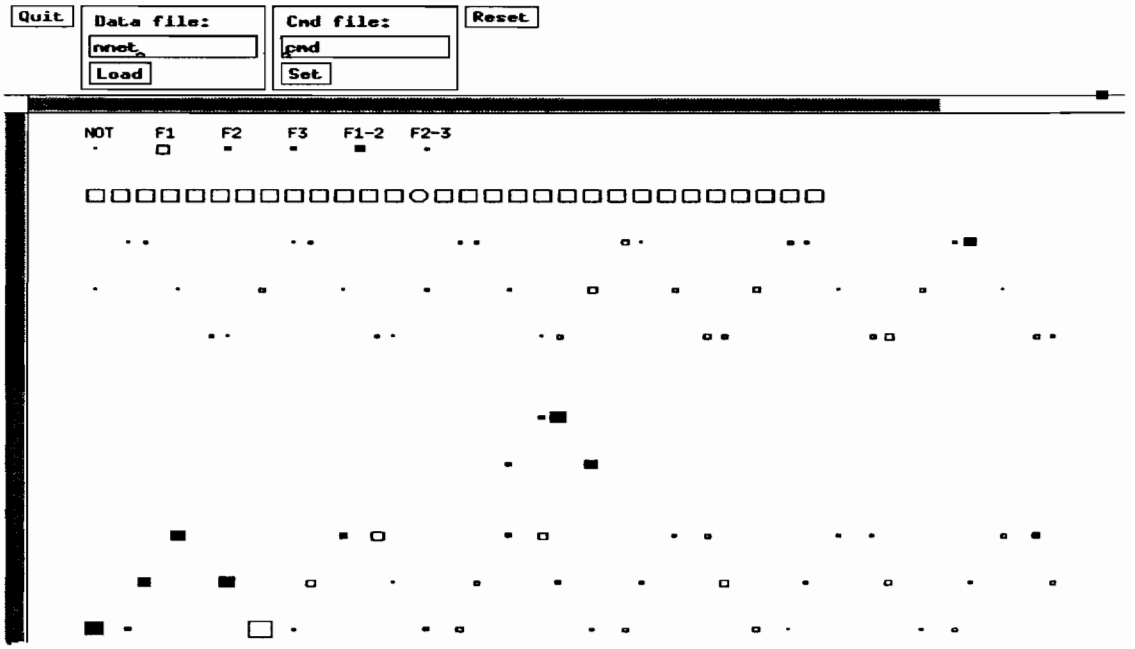

Figure 13: Weight Activations for Hidden Node 14

display the weight magnitudes for networks produced by our simulator. It is not always the case that the distinctions found by the network are the same that people would use in making the same classifications.

For the formant classification network, many of the hidden node distinctions are not obvious. As an example of the XOPT tool, we will look at the weight magnitudes for hidden node 14 in the Shifted Vector Network (Figure 13). This node makes one of the more easily understood distinctions, F1 and Not merged F1-2. The display shows the 3 layers of nodes. The top row is the output neurons. The second row is the hidden layer. Because of its length, the input layer has been broken up into 3 rows. The third and fifth rows are the lower and upper context units, each with 6 peaks and six valleys. The fourth row is the nodes representing the target peak. The units

that compose each peak represent the frequency and magnitude. The lower node on either side is the frequency of the shoulder of the peak. Finally, in the context units the two lowest nodes between each peak are the frequency and magnitude of the interpeak valleys.

Each box represents a node. The size of the box indicates the magnitude of the weight. In this case, the size of the input layer boxes indicates the magnitude of the weight going in to the hidden node, and the size of the output layer boxes indicates the magnitude of the weight going from the hidden node to each output node. Since there are no connections between nodes in a layer, the other hidden nodes are a uniform size. The displayed hidden node has rounded corners. Finally, the color (black or white) indicates negative or positive weights.

The F1 node is strongly activated when hidden node 14 is activated. Notice that although all the other output nodes are negatively affected by the activation of hidden node 14, the output for F1-2 is more strongly affected than the others. So, not only does hidden node 14 detect F1, it also distinguishes F1 from the merged F1-2. Notice that all of the features in the target peak NEGATIVELY affect the hidden node. The input features are normalized from -1 to 1, so the lower frequencies are presented as negative numbers. The immediately adjacent upper context nodes appear to have a strong affect. The negative weights on frequency input nodes reflects the

normalization of the input, i.e. lower frequencies are represented by negative numbers. This hidden node is reacting to a target peak with low frequency, and that has a neighboring peak that is low in frequency, which is reasonable for F1.

The other hidden nodes are not making such clear cut distinctions. In many cases, the distinction is negative. That is the decision is Not F1, which may be in part caused by the normalization. Apparently the node that receives the fewest nay votes wins. Other distinctions are not intuitive. For example there are several hidden nodes that distinguish NotF, F1, and F2 from F3, F1-2, and F2-3. In this case it seems that the node is segregating the low amplitude formants from the high amplitude formants. But that cannot be completely true since many of the NotF peaks are also low amplitude.

Visual inspection of the classifier error indicates that the NotF, F3 distinction is difficult for the classifier. This observation is born out in looking at the hidden nodes. 28 of 30 hidden nodes contribute towards distinguishing these labels. For comparison, of the 15 possible label distinction pairs (such as F1 F2, F1-2 F3), 10 of them use between 19-21 hidden nodes. The 4 remaining pairs use less than 19 hidden nodes.

Inspecting the weight magnitudes can be a useful exercise, although for this network, the results of this inspection are not clear cut. The most surprising observation is that most of the magnitudes are small. The network

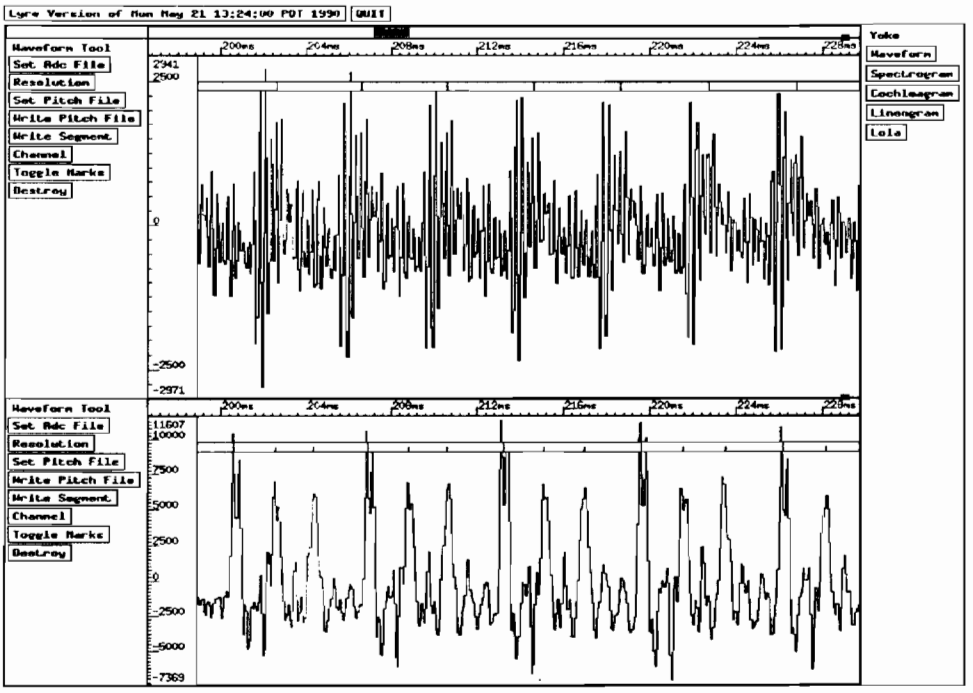

Figure 14: Erroneous and Correct Pitch Marks. In the top picture the pitch marks are not at the peaks of the waveforms, and the bottom picture shows correct pitch mark locations.

as found few definite characteristics that separate the formant classes. The small size of the magnitudes makes it difficult to reach many conclusions about the function of individual hidden nodes. The weight magnitudes do confirm the results of the visual inspection of errors that the network finds the NotF-F3 distinction difficult.

4.6 Pitch Tracker

Close visual inspection of the labeling errors led us to look at the underlying spectral representation. To understand a certain type of problem in the PSDFT we looked at the pitch files. We discovered a very subtle problem.

The problem in the PSDFT is that the pitch marks from the pitch tracker are not aligned with the actual pitch period (Figure 14). The upper display shows the pitch marks are placed after the pitch peak. The lower display shows correctly placed pitch marks for another utterance. This shifts the DFT window, and allows the resolution of harmonics in the PSDFT (Figure 15). The figure shows a Lineogram of the spectogram. In this picture the coefficients are plotted horizontally to better display the values. These harmonics introduce numerous spurious peaks and reduce the valid peaks.

The pitch tracker uses a pitch peak location algorithm to locate candidate peaks. This pitch peak algorithm relies mostly on the low frequency component of the pitch. In this case there is a major component of the pitch that is above 2500 Hz, and it causes the pitch marks to be erroneously placed. The pitch period between marks is the same since all the marks are shifted uniformly. The problem manifests itself when generating a pitch-synchronous DFT. Only in this situation do we need a definite marker at the beginning of the pitch period. Since the pitch tracker places these marks in the wrong place the DFT window is incorrect and that accounts for the harmonic effects visible in the spectogram.

To test this hypothesis we manually moved the pitch marks to more closely correspond to the pitch period (Figure 14). The resulting PSDFT (Figure 16) has an identifiable F2-3 where the uncorrected PSDFT just had

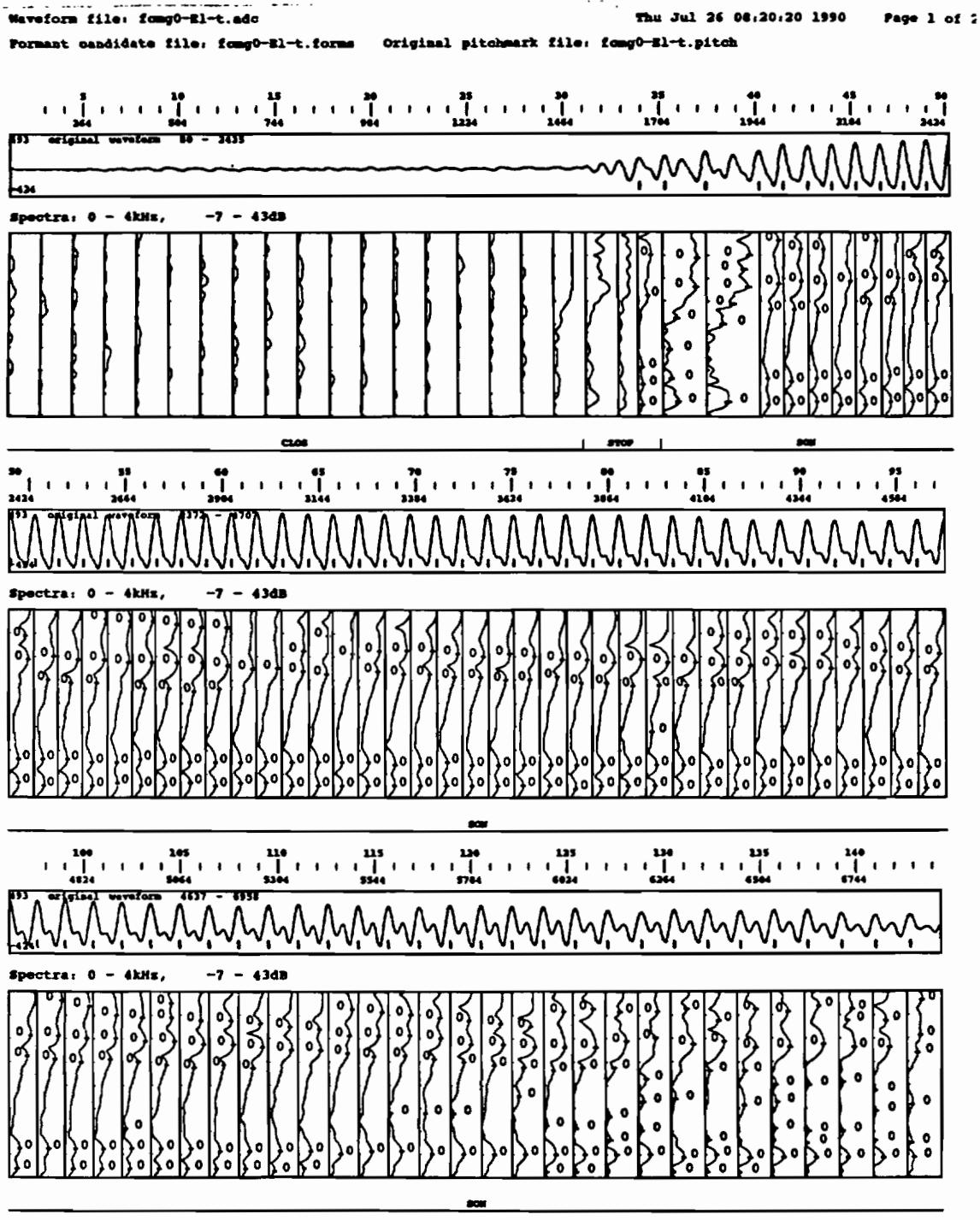

Figure 15: Lineogram of spectra with Bad Pitch Marks, note that there is no identifiable F2 or F3 that continues through the entire utterance

arbitrary peaks. For this utterance the error rate was reduced from 40% to 10%. We feel an improved pitch tracker will provide the greatest improvement. The largest remaining error is distinguishing F3 from NotF. An improved pitch tracker will remove many of the spurious peaks that were confused with F3, aiding both training and classification.

A related problem is glottalization, the periods when the vocal cords are accelerating or decelerating and the vibrations are irregular. An improved segmenter will assign a glottalization label, so this source of error will simply be avoided. It is a small part of the total error, but it can be a large source of error in a single utterance.

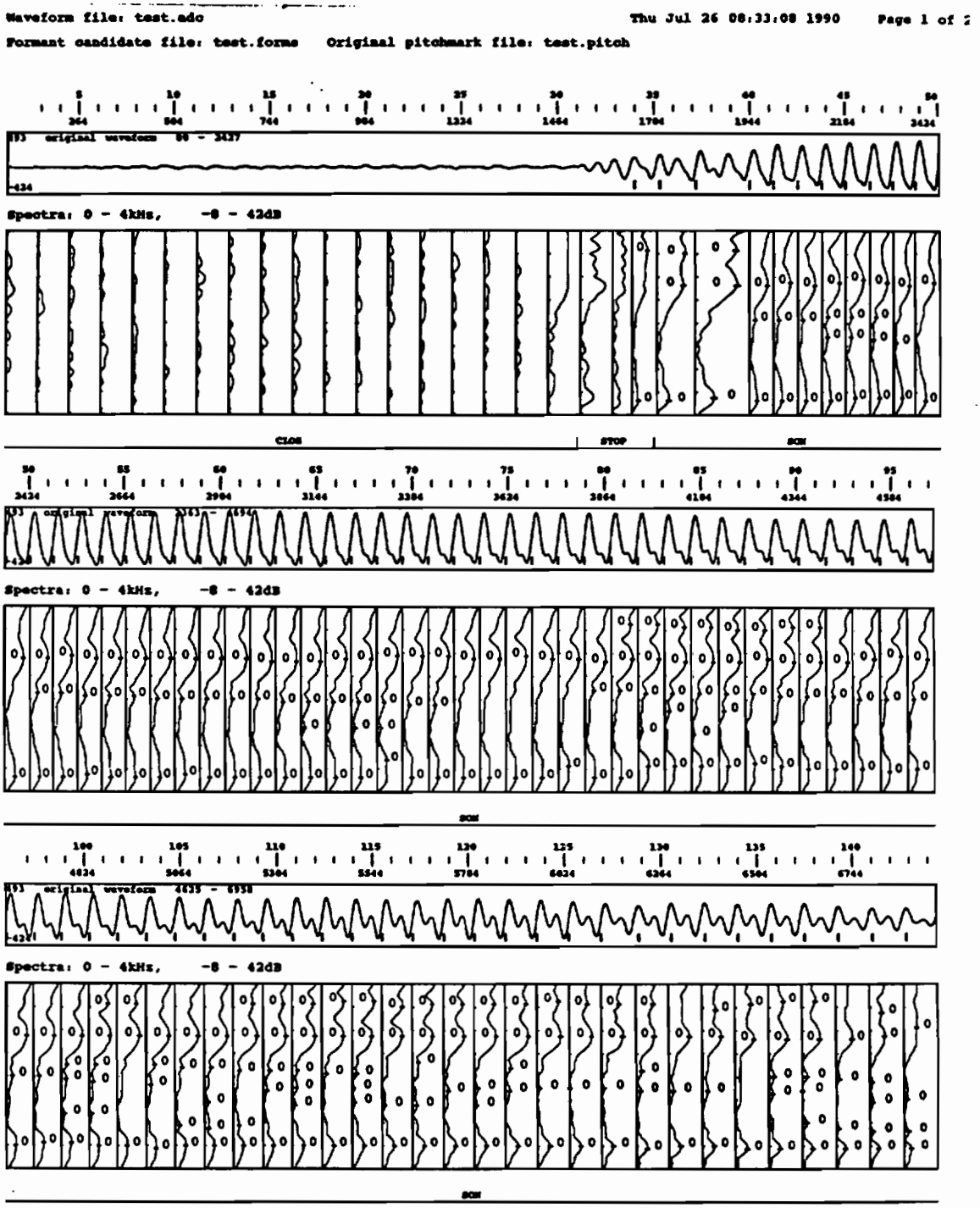

Figure 16: Lineogram of the Spectra after the pitch marks were corrected showing the improved peak resolution, note the identifiable merged F2-3

5 Future Directions

We are pleased with performance of the neural network formant estimation algorithm. With the information available to it, it is performing at human levels of performance. Analysis of the types of errors indicates that the fundamental approach is sound. The errors that occur are reasonable given the information available in a single frame. To resolve these errors information is required about formant labels in adjacent frames. There are several methods to acquire this information: algorithmic (rule-based) post-processing, recurrent neural networks, or constraint satisfaction.

5.1 Algorithmic Post-Processing

This approach would simply apply a series of rules to any frame that had an inconsistent formant labeling (e.g. two F3 labels). These rules would be developed by observation from the datasets, and the classes of errors identified by visual inspection of the algorithm output. This approach would only be able to us information from immediately adjacent frames. It is possible that by running the inconsistent formant labels through the algorithm several times that "correct" information could be propagated to the inconsistent frames. This approach suffers from the same problems as any rule-based approach that the rules are fine-tuned to the existing data, and do not generalized well.

5.2 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) implicitly carry information from previous input vectors forward to the next input vector. They do this by some form of feedback in the internal weight connections between nodes. These connections may feed the output back into the same node, or into a node in a previous layer. This feedback allows the activation from previous inputs to be used in the next input presentation. Superficially this would appear to provide the inter-frame context we sought. The problem is that the information needed to resolve inconsistent labels, may be in later spectral frames. For example, at the beginning of a sonorant, the frames needed to resolve inconsistencies are located after the frame being classified. In this case, the RNN would not provide any advantage. It may be possible to delay a decision for several frames so that the network will have context from both sides of the target frame.

5.3 Constraint Relaxation

This is a method that uses a modified Hopfield network for post-processing. In this network each node represents a label for a peak in the frame. The connections between nodes (within the same peak, frame, or between frames) are inhibitory representing constraints on which labelings are consistent. For example, all the F1 nodes within a frame are inhibitory since there can only

be one F1 label in a frame. The input to the network is the output of the Shifted Vector Network, where the activation of each output neuron is used as the input for each label node in the Hopfield network. Initial results with such a network were encouraging. It was able to correct individual errors (i.e. single errors with the adjacent peaks, and frames correct). In cases where there was a series of inconsistent frames, the network did not do well. Further research is needed to find a method to propagate the inhibitions through several inconsistent frames.

6 Conclusion

Our various experiments with the individual peak classification has been encouraging. In those utterances without spectral representation problems, the basic network does well, correctly labeling 90-98% of the peaks. If this level of performance can be reliably maintained after cleaning up the representation, then simple post-processing methods of resolving incorrect labels will be sufficient.

Even without further developments, our neural network approach seems to have better performance than the 2 major alternatives. Neural networks are robust. Even if variation in data causes the performance to degrade, simply retraining the existing network will recover the performance. Our perception experiments show that the neural network classifier is slightly better than experienced human labelers.

Our research has shown that neural network classifiers can be used successfully for formant estimation. They are better than the major alternatives. They are better than human labelers. In addition for formant estimation with neural networks, we have shown that derived features provide better performance than raw data.

References

[1] E. Barnard, R.A. Cole, M. Vea, and F. Alleva. *Pitch Detection With a Neural Network Classifier*, Oregon Graduate Center Technical Report CSE 89-011, June 1989.

[2] E. Barnard, and R.A. Cole. *A Neural-Net Training Program Based on Conjugate-Gradient Optimization*, Oregon Graduate Center Technical Report CSE 89-014, July, 1989.

[3] R.A. Cole, Y. Muthusamy, and M. Fanty. *The ISOLET Spoken Letter Database*, Oregon Graduate Institute Technical Report CSE 90-004, March 1990.

[4] R.A.Cole, and L. Hou. "Segmentation and Broad Classification of Continuous Speech" in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1988.

[5] G. Fant. *Acoustic Theory of Speech Production.* The Hague: Mouton, 1960.

[6] W. Fisher, G. Doddington, and K. Goudie-Marshall. "The DARPA Speech Recognition Research Database: Specifications and Status" in *Proceedings of the DARPA Speech Recognition Workshop*, 1986.

[7] G.E. Kopec. "Formant Tracking Using Hidden Markov Models" in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1985.

[8] G.E. Kopec. "Formant Tracking Using Hidden Markov Models and Vector Quantization", *IEEE Trans on Acoustics, Speech, and Signal Processing* **34**, No 4, 1986.

[9] G.E. Kopec. "A Family of Formant Trackers Based on Hidden Markov Models", in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1986.

[10] L. Lamel, R. Kassel, and S. Seneff, "Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus" in *Proceedings of the DARPA Speech Recognition Workshop*, 1986.

[11] S.S. McCandless. "An Algorithm for Automatic Formant Extraction Using Linear Prediction Spectra", *IEEE Trans on Acoustics, Speech, and Signal Processing* **22**, No 2, 1974.

[12] D. O'Shaughnessy. *Speech Communication.* Reading, Ma: Addison-Wesley, 1987.

[13] G.E. Peterson, and H.L. Barney. "Control Methods Used in a Study of the Vowels", *The Journal of the Acoustical Society of America* **24**, No 2, 1952.

Biographical Information

Terry Rooker was born on 15 October 1957 in Lancaster, Ohio. Upon graduation from high school he attended the University of Washington on a 4 year Naval ROTC Scholarship, graduating in 1979 with a BA in History. He served 6 years as a commissioned officer in the US Navy. After separation he pursued a BA/BS concentrating in Artificial Intelligence, receiving the degrees from The Evergreen State College in 1988. He then enrolled as a graduate student at the Oregon Graduate Institute.