Pitch Estimation, Voicing Decision, and Noise Spectrum Estimation for Speech Corrupted by High Levels of Additive Noise

by

David A. Krubsack, M.S.E.E

A Dissertation Submitted to the
Marquette University Graduate School
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Milwaukee, Wisconsin
May, 1990

Preface

This dissertation presents two algorithms that extract parameters which are important to speech processing in high levels of noise. The first algorithm determines whether a signal containing noise corrupted human speech is voiced or not and estimates the fundamental frequency (pitch) of voiced speech. The second algorithm produces an estimate of the additive noise which is corrupting the speech.

Previous research related to the voicing decision and pitch estimation has been concentrated at signal-to-noise ratios (SNRs) above 0 dB. Consequently, speech processing requiring the extraction of these parameters in higher levels of noise could not be performed with much success. The research presented in this dissertation concentrates on SNRs around and below 0 dB. Although the algorithm, based on the autocorrelation function, is designed to work well for high levels of noise, good results for the no noise case have been maintained. The idea of a confidence measure for parameter estimation is introduced. Confidence measures are defined and developed for both the voicing decision and the pitch estimation algorithms.

Estimation of noise that is corrupting a speech signal has been motivated by the need to enhance the corrupted speech. Previous research has concentrated on speech which is band limited to about 3500 Hz. Therefore, the estimation of the noise corrupting high frequency speech had not been considered. The noise estimation algorithm presented in this dissertation considers the effects of high frequency speech on the noise estimate in addition to the effects of low frequency speech. A new spectral averaging method is introduced which significantly reduces the corrupting effect of the speech components on the noise estimate for SNRs above 0 dB. The algorithm is tested for stationary white noise, stationary non-white noise, and non-stationary white noise.

Acknowledgments

I would like to thank the members of my committee, Dr. Brown, Dr. Doerr, Dr. Heinen, and Dr. Josse, for their suggestions and insights and especially Dr. Niederjohn who served as the committee chairperson and my advisor without whose support and encouragement this dissertation would not be possible. I would also like to thank the National Science Foundation for its support through grant MIP-8607831.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The production and analysis of human speech sounds are covered in a variety of general texts [1], [2]. The purpose of this chapter is to introduce terms and concepts that are relevant to the research presented in this dissertation.

Speech is a concatenation of fundamental speech sounds called **phonemes**. In written form, phonemes are identified as occurring between two slashes [2]. For example, the word "sea" is formed using two phonemes. The first phoneme is /s/. During the production of the /s/ sound, the vocal cords are not vibrating so the phoneme is classified as **unvoiced**. It is further classified as a **fricative** because there is a constriction in the vocal tract through which rushing air causes a turbulent sound. The second phoneme is labeled as /i/. The vocal cords are vibrating during the production of /i/ so it is classified as a **voiced** phoneme. The fundamental frequency at which the vocal cords are vibrating is called the **pitch**.

To digitally process speech sounds, the acoustic wave is converted to digital form using a microphone and an analog-to-digital (A/D) converter. In this form, a computer can process the speech in any way a programmer chooses. The processing may be for speaker identification or verification [2], speech recognition for voice operated machinery [2], or analysis of voice disorders [3]. Since the speech may be reconstructed using a digital-to-analog (D/A) converter and a speaker, processing may include speech compression for bit rate reduction in transmission and storage [4], speech enhancement of a corrupted signal [4], or voice disguising. The speech may even be synthesized from a typed text for the blind [2], or from statistics in a database for telephone access [2].

For most speech analysis or processing, **voicing** is an important parameter which needs to be extracted from the digital signal [2]. The voicing decision answers the question "Is the speech voiced or not?" The exact point in time where a transition occurs is often difficult to determine because of **coarticulation effects**, that is the sound of one phoneme overlapping another during the production of speech. The decision becomes more difficult when noise is corrupting the speech signal. Such a decision needs to be made when the goal of processing is to enhance noise corrupted speech.

Pitch is another parameter important to speech processing. Pitch is often used to distinguish a question from a statement. Pitch can convey information about the emotional state of the speaker such as anger or joy [5]. Estimation of the **formants** (resonant frequencies of the vocal tract) [6], background noise estimation [7], and LPC analysis [2] may require an estimate of the pitch while glottal wave estimation [2] and comb filtering [4] are possible only if the pitch is available.

Speech enhancement provides the main motivation for the research presented in this dissertation. A speech enhancement system requiring a voicing decision, a pitch estimate, and an estimate of the noise is outlined in Figure 1.1. This algorithm is based on the assumption that intelligibility enhancement can be achieved by making use of information directly related to speech intelligibility [8]. Each estimated parameter is a function of time.

The first block represents the estimation of the pitch and the voicing decision. The pitch estimate is given as $\hat{F}_0$. The variable $\hat{c}_0$ is defined as the pitch confidence and indicates a probabilistic deviation of the pitch. The voicing decision is made in a continuous fashion using the variable $\hat{c}_v$. The magnitude of $\hat{c}_v$ gives an indication of the confidence of the voicing decision.

The second block represents the estimation of the spectrum of the noise which is corrupting the speech. This estimate is a function of frequency as well as time. The third block represents the estimation of the first three

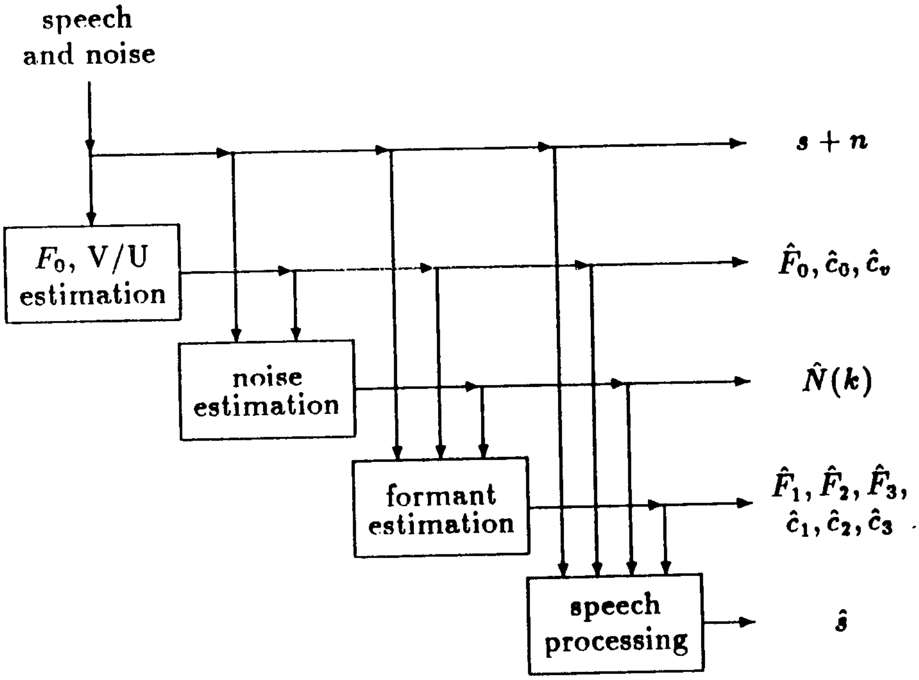

Figure 1.1: Complete block diagram for a speech enhancement system.

formants of voiced speech. Each formant has an associated confidence. The algorithm designated by the final block uses the extracted parameters and processes the speech plus noise signal resulting in what is hoped to be a more intelligible speech signal.

The upper two blocks in Figure 1.1 are the subject of the research in this dissertation. The "formant estimation" and "speech processing" blocks are the subjects of related research by other members of the speech and signal processing group at Marquette University [8], [9], [10], [11], [12].

Chapter 2

An Autocorrelation Pitch Detector and Voicing Decision with Confidence Measures Developed for Noise-Corrupted Speech

This chapter describes an integrated speech feature extraction method [13] consisting of 1) a pitch detector, 2) a voicing decision (V/U), 3) a confidence measure which reflects the probabilistic accuracy of the voicing decision, 4) a confidence measure which reflects the expected deviation of the pitch estimate from the true pitch and the probabilistic accuracy of this deviation, and 5) smoothing techniques for the pitch detector, the voicing decision, and the two confidence measures. The focus of this research is for voiced and unvoiced speech corrupted by high levels of white noise. The voicing decision and the confidence measures were developed by observing the behavior of three features derived from the autocorrelation function and experimentally fitting curves to the data. This integrated set of algorithms is statistically analyzed for speech at seven signal-to-noise ratios (SNRs).

2.1 Introduction

It is frequently necessary to know whether a segment of speech under analysis is voiced or not. If the segment is voiced, the fundamental frequency (pitch) is an important parameter. The extraction of the pitch and voicing decision becomes significantly more complicated in the presence of noise. The goal of the research presented in this chapter is to extract these parameters for application to intelligibility enhancement of speech corrupted by high levels of additive broadband noise. It is intended that the confidence measures will be used to determine the degree to which the pitch estimate and voicing decision will be used.

A variety of methods have been proposed for pitch determination and the voicing decision. Pitch determination can be carried out in the time domain [14], [15], [16], [17], [18], the frequency domain [19], [20], [21] [22], or using a combination of both [23], [24]. A comparison of some of these methods is available [25]. In earlier work [14], [15], [16], [17], [23], [24], the voicing decision is a direct result of pitch determination. Atal and Rabiner [26] propose that such methods may be inadequate for making the voicing decision. Later work [22], [26], [27], [28], [29] includes additional decision parameters and the resultant voicing algorithms are independent of the pitch detector. Adaptive techniques [30] have also been applied.

In addition to the voicing decision, algorithms for silence [26], [28] and mixed [29] excitation decisions have been developed. Since the work presented in this chapter is concerned with noise contaminated speech, only voiced and unvoiced classifications are made. As noise is added, silence segments tend to appear unvoiced and voiced segments tend to appear mixed. In high levels of noise, weak voiced segments may even appear to be unvoiced. These important facts are discussed later in this chapter.

The purpose of a voicing decision is to correctly partition speech into voiced and unvoiced intervals. The algorithm described in this chapter uses smoothing to correct isolated errors. Consequently, it may occasionally classify V-U-V and U-V-U sequences incorrectly. This concern has been expressed in [26] and [27]. However, it has not been shown to adversely affect intelligibility enhancement of speech in noise.

The voicing decision has previously been madė in a continuous fashion [31]. The application is for speech coding and the "pitch gain" controls the mixture of the pulse and noise sources. The resulting speech has increased in quality and intelligibility relative to the conventional LPC system especially when the speech is corrupted by background noise. For the work presented in this chapter, the voicing decision is also made in a continuous fashion. Rather than controlling a mixture for resynthesis, the voicing confidence will reflect a certainty with which the voicing decision is made. The motivation is for intelligibility enhancement rather than relative improvement over a conventional coding method.

An attempt has been made to minimize the computation required by the method so that it has application in real-time systems. The pitch detector is based on an autocorrelation algorithm found to work well in white noise [32]. The voicing decision and the confidence measures are based on three autocorrelation features that were found robust in noise. The voicing decision makes use of fixed discrimination analysis optimized over a range of signal-to-noise ratios (SNRs) from no noise (∞ dB [1]) to -18 dB. It is optimized in the sense that the critical boundary (Section 2.2.3) is chosen so that the number of voicing errors at 0 dB SNR is a minimum. Voicing decisions for the transitions between sound categories have the greatest chance of being incorrect. In fact, a correct decision category for transition regions may not be possible to determine even under the best of conditions. In the method presented here, this is taken into account through the calculation of a confidence measure to reflect the probability

[1] The term "∞ dB" as used in this chapter implies that no noise was added to the database utterances. The SNR of the database is roughly 45 dB.

that the voicing decision is correct. Similarly, the pitch estimation will have inaccuracies. Thus, a probabilistic measure, the pitch confidence, is developed to reflect the probable accuracy of the pitch detector. The voicing confidence and pitch confidence are derived independently, but make use of the same autocorrelation features. The four integrated algorithms and their associated smoothing techniques have been quantitatively evaluated for seven SNRs. Results are presented in this chapter.

The method was developed using several speech utterances. To ensure that the thresholds and parameters implemented in the method are appropriate, it was tested with a "test" database which consists of an entirely different set of utterances than the "development" database. The results of the test database are presented in Section 2.3.

The method has been designed for situations where the speech signal below 600 Hz is not significantly distorted and the added noise is broadband. The current implementation has been designed as part of a speech in noise enhancement system which easily meets these criteria [8]. The method is also applicable to other practical situations (for example speech feature extraction and speech recognition) where noise corrupted speech is to be processed. The method is not intended for band limited communication channels, impulse noise, or sinusoidal noise which do not meet the specific criteria appropriate for the intended application. A block diagram of the overall method presented in this chapter is given in Figure 2.1. Note that the voicing decision is encoded with the voicing confidence measure.

2.2 Details of the Method

The speech to be analyzed is lowpass filtered with a 6 pole Butterworth filter having a cutoff frequency of 8 kHz and sampled with a 10 bit analog-to-digital converter at a rate of 20 kHz. The (development) database is comprised of 6 speakers, 3 male and 3 female, each speaking 5 sentences. A more complete description of the database is available [22]. Gaussian noise was computer generated [33] to be white to 10 kHz and added to the sampled speech at the appropriate SNR. Each utterance has about a quarter second of silence before and after the speech. The SNR, calculated only over the speech, is determined as ten times the common logarithm of the mean square level of speech to the mean square level of noise.

2.2.1 Pitch Detector

The pitch detector is based on an autocorrelation algorithm described in an earlier paper [32]. The speech is lowpass filtered with a 6 pole Butterworth filter having a cutoff frequency of 600 Hz as suggested by Gold and Rabiner [15]. For speed in calculation, the speech is decimated to 10 kHz as shown in Figure 2.1. (Further decimation, for example to 5 kHz, begins to

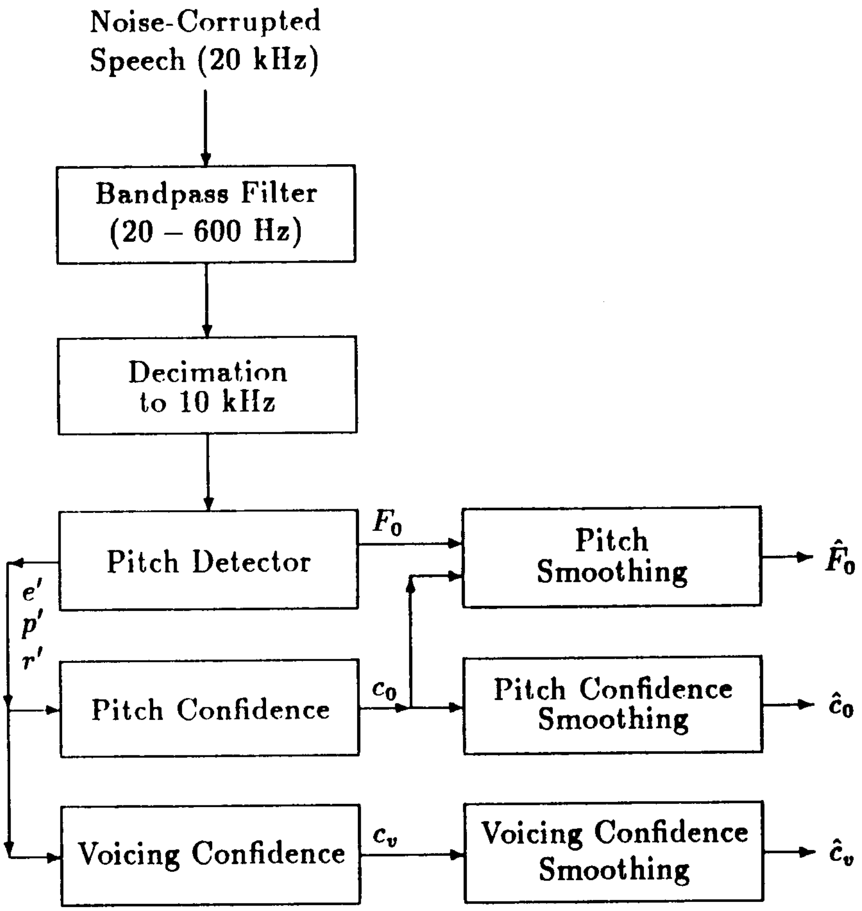

Figure 2.1: Complete block diagram for the pitch detector, voicing decision, and confidence measures.

degrade the algorithm's precision in determining the pitch.) The standard short-time autocorrelation function is calculated for each successive, 51.2 ms speech segment. Successive segments are overlapped by 75 percent.

$$R(l) \;=\; \sum_{n=1}^{512-l} s(n)s(n+l) \qquad l = 0,1,\ldots,511 \tag{2.1}$$

where $R(l)$ is the autocorrelation function and $s(n)$ is the 512 point segment of the signal. (The autocorrelation needs to be calculated only for l=0,30,31,32,...,200 as described in the next paragraph.)

A peak picking algorithm is applied to the autocorrelation function of each segment. This algorithm starts by choosing the maximum peak (largest value) in the pitch range of 50 to 333 Hz (3 to 20 ms, l=30 to 200). This peak has a corresponding lag, L. (If there are two peaks of equal value, the one with the smaller lag is chosen.) The period corresponding to L is the first estimate of the pitch period. Another variable, K, is introduced and set equal to this value of L. The value of K remains constant throughout the peak picking algorithm and is used in Section 2.2.2.

As shown in Figure 2.2, the algorithm checks for peaks at one-half, one-third, one-fourth, one-fifth, and one-sixth of the first estimate of the pitch period. If $\frac{L}{2}$ (rounded up) is within the pitch range, the maximum value of the autocorrelation within $l = \frac{L}{2} - 5$ to $\frac{L}{2} + 5$ is located. If $\frac{L}{2} - 5$ is less than 30 (3 ms), 30 is chosen as the lower limit instead of $\frac{L}{2} - 5$. If this new peak is greater than one-half of the old peak, the new corresponding lag replaces the old corresponding lag, L. (This new L might not be exactly $\frac{L}{2}$ as shown in Figure 2.2.) We now have a new L which presumably is corrected for the possibility of a pitch period doubling error. This test is performed again to check for double doubling errors (four-fold errors). If this most recent test fails, a similar test is performed for tripling errors of this new L. This test checks for pitch period errors of six-fold. If the original test failed, the original L is tested (in a similar manner) for tripling errors and errors of five-fold. With a sampling rate of 10 kHz (after decimation), the final value of L is used to calculate the pitch estimate, F_0, by the equation

$$F_0 = \frac{10000}{L}. \tag{2.2}$$

The pitch error is defined [34] as

$$F_{0_{err}} = 100\left(\frac{F_0 - F_{0_a}}{F_0}\right) \tag{2.3}$$

where F_{0_a} is the actual pitch. The error is defined relative to F_0 rather than F_{0_a} because processing algorithms which are to make use of F_0 will need a pitch confidence measure which reflects error relative to the F_0. F_{0_a} is calculated as the average period length of all of the pitch periods that are at least 50 percent within the analysis window. The method for determining the actual length of each pitch period makes use of a terminal interactive program developed for this purpose [22].

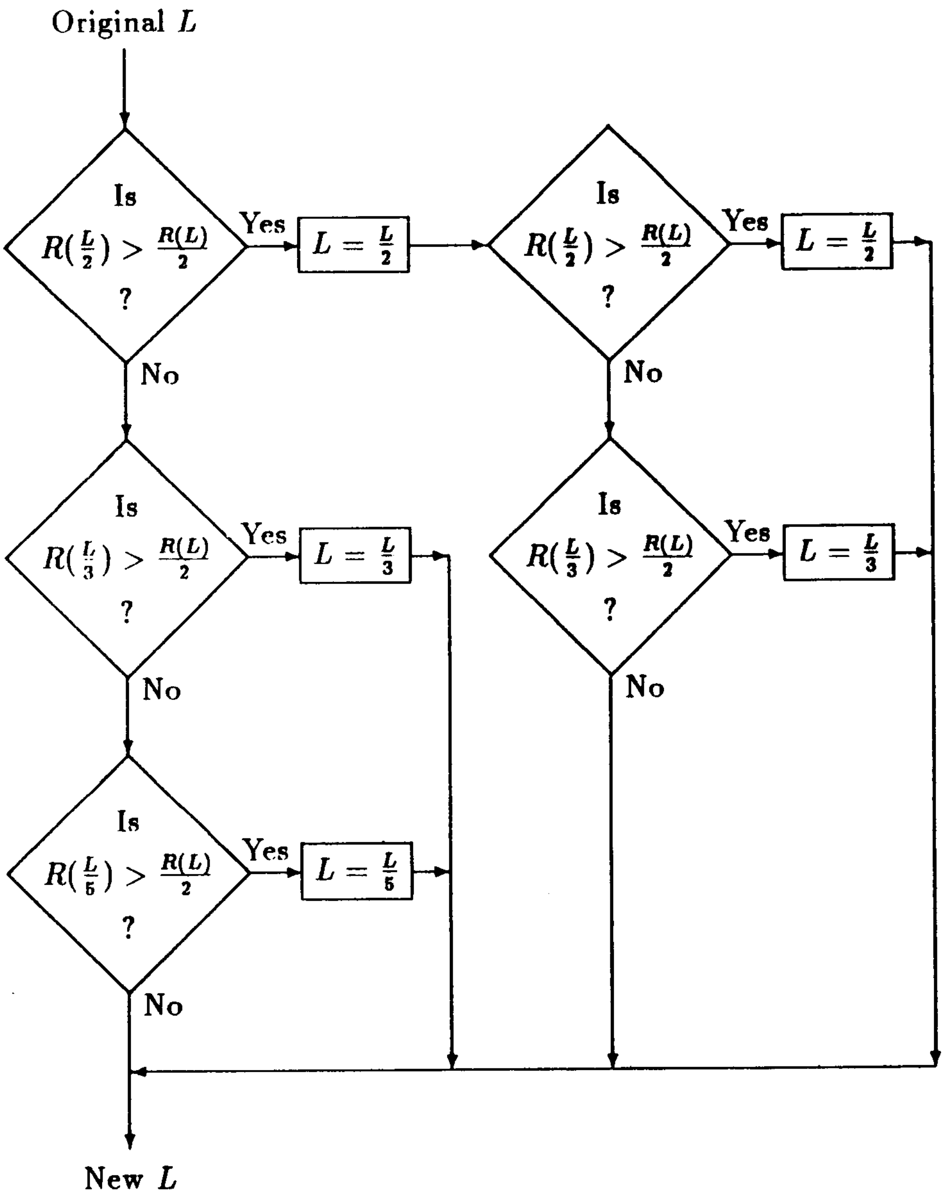
Original L
Is
$R(\frac{L}{2}) > \frac{R(L)}{2}$
?
Yes
$L = \frac{L}{2}$
No
Is
$R(\frac{L}{3}) > \frac{R(L)}{2}$
?
Yes
$L = \frac{L}{3}$
No
Is
$R(\frac{L}{5}) > \frac{R(L)}{2}$
?
Yes
$L = \frac{L}{5}$
No
Is
$R(\frac{L}{2}) > \frac{R(L)}{2}$
?
Yes
$L = \frac{L}{2}$
No
Is
$R(\frac{L}{3}) > \frac{R(L)}{2}$
?
Yes
$L = \frac{L}{3}$
No
New L

Figure 2.2: Decision process for the peak picking algorithm.

	$F_{0_{err}}$		Gross	Fine Error	
		Standard	Error		Standard
SNR	Average	Deviation	(%-age of	Average	Deviation
(dB)	(%)	(%)	Segments)	(%)	(%)
∞	0.837	6.744	0.969	0.476	1.894
12	0.926	6.312	0.969	0.495	1.880
6	1.103	6.442	1.260	0.499	1.972
0	1.184	9.228	1.939	0.548	1.937
−6	1.958	14.070	4.411	0.598	2.328
−12	−0.065	34.619	15.075	0.528	3.585
−18	−12.259	62.223	42.172	0.539	6.098

Table 2.1: Results of the Pitch Detection Algorithm

The results for the pitch detector can be found in Table 2.1 which includes the average and standard deviation of $F_{0_{err}}$. The term gross error was originally introduced by Rabiner, et al. [25]. Table 2.1 uses the 20 percent definition of gross error [34]. This definition is used because it produces results similar to the Rabiner 1 ms definition and the density function for the fine errors is symmetric without favoring high or low pitched speakers. As will be seen later, 20 percent corresponds to the base of the pitch deviation triangle. A 20 percent "gross error" has occurred if the actual pitch differs by more than 20 percent from the estimated pitch, that is, $|F_{0_{err}}| > 20$. The percentage of segments in which a gross error has occurred is listed in the table. Errors that are not gross are considered to be "fine errors" [25]. The average and standard deviation of the fine errors are also included in the table.

The results in Table 2.1 include all voiced segments in the database. All segments that are unvoiced (including silence) or transitional (which include both voiced and unvoiced speech) are not used in the calculation of the results. A pitch is estimated for all segments independent of the voicing decision. This is done because, for high levels of noise, the voicing algorithm may have classified a segment incorrectly, yet the pitch estimate could still be accurate. This estimate is then used for pitch smoothing of adjacent segments that may have been classified correctly.

2.2.2 Features for the Voicing Decision and Confidence Measures

Three features derived from the autocorrelation function are passed to the voicing decision and the confidence measure algorithms. These are 1) e', the rms energy of the segment, 2) p', the maximum value of the autocorrelation function over the pitch range normalized by the value at zero lag, and 3) r', the rms of the normalized autocorrelation function over the pitch range. These features are calculated as follows.

$$e' = \left(\frac{R(0)}{512}\right)^{\frac{1}{2}} \tag{2.4}$$

$$p' = \frac{R(K)}{R(0)} \tag{2.5}$$

$$r' = \left[\frac{1}{171}\sum_{l=30}^{200}\left(\frac{R(l)}{R(0)}\right)^2\right]^{\frac{1}{2}} \tag{2.6}$$

The noise-corrupted speech must have a zero mean for r' to be a valid feature. To assure a zero mean, the speech is highpass filtered with a 6 pole Butterworth filter having a cutoff frequency of 20 Hz. This filter is incorporated with the 600 Hz lowpass filter as shown in Figure 2.1.

Although the voicing decision and the confidence measures are derived from these autocorrelation features, they do not depend on the accuracy of the pitch detector. In fact, these features will be used to determine the pitch confidence measure which reflects the pitch accuracy.

2.2.3 Voicing Decision

A plot of r' versus p' for the no noise case is shown in Figure 2.3. The voiced and unvoiced regions overlap so that a reliable discrimination is not possible. Upon analysis, it was found that most of the unvoiced points which fall in the voiced region occur when no speech is present and therefore e' is very small. Since e' is the rms energy of the segment, an energy threshold is used to define these points as unvoiced. The threshold was chosen based upon the energy of the segments during no speech and is equal to 14 (quantization levels). This energy threshold was included to correct for peaks that can occur in the normalized autocorrelation function when very low levels of periodic background noise are present. It is important to note that this energy threshold is not a function of the speech level or the

SNR. For any segment with $e' < 14$, p' and r' are set equal to zero. p' and r' are also set equal to zero for any segment with $p' < 0$.

A plot of r' versus p' (SNR $= \infty$ dB) after the thresholds are applied is shown in Figure 2.4. All of the unvoiced points that were in the voiced region have been eliminated such that a reliable discrimination analysis can now be performed. As noise is added (Figure 2.4–Figure 2.6) the cluster of unvoiced points remains fairly constant, but the voiced region moves toward the unvoiced region. This is intuitively appropriate since, in the limit of increasing noise, the voiced segments become entirely noise corrupted and therefore appear as unvoiced segments.

In Figure 2.3–Figure 2.6, only one sentence, "Every salt breeze comes from the sea," spoken by three males and three females is used. In Figure 2.7–Figure 2.8, the entire development database is plotted for seven SNRs ($\infty, 12, 6, 0, -6, -12, -18$ dB). The unvoiced region, Figure 2.7, is virtually independent of SNR. The voiced region, Figure 2.8, has two limiting areas. For low SNRs, the voiced points tend toward the dark area on the lower left (the unvoiced region). For high SNRs, the voiced points tend toward the dark area on the upper right.

Considering the behavior of the voiced and unvoiced regions as described above, it was decided that a fixed discrimination boundary would be used.

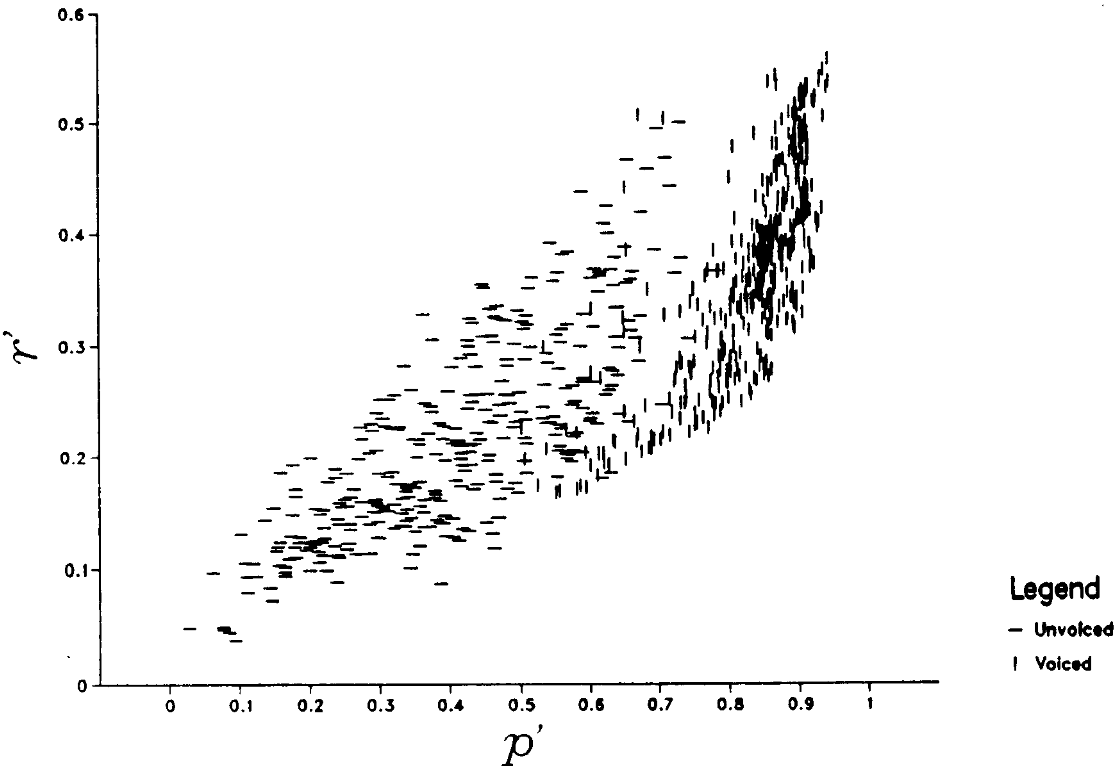

Figure 2.3: r' versus p' for the example sentence "Every salt breeze comes from the sea" spoken by 3 males and 3 females. SNR = ∞ dB, no energy threshold.

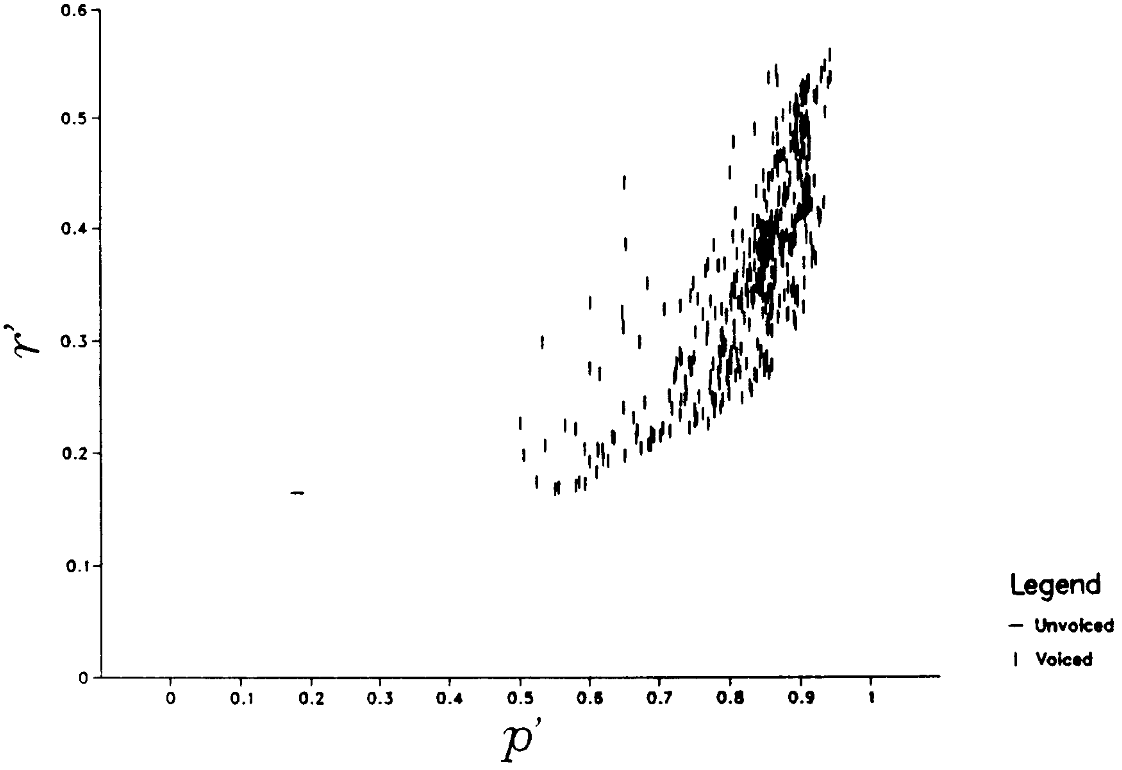

Figure 2.4: r' versus p' for the example sentence "Every salt breeze comes from the sea" spoken by 3 males and 3 females. SNR = ∞ dB.

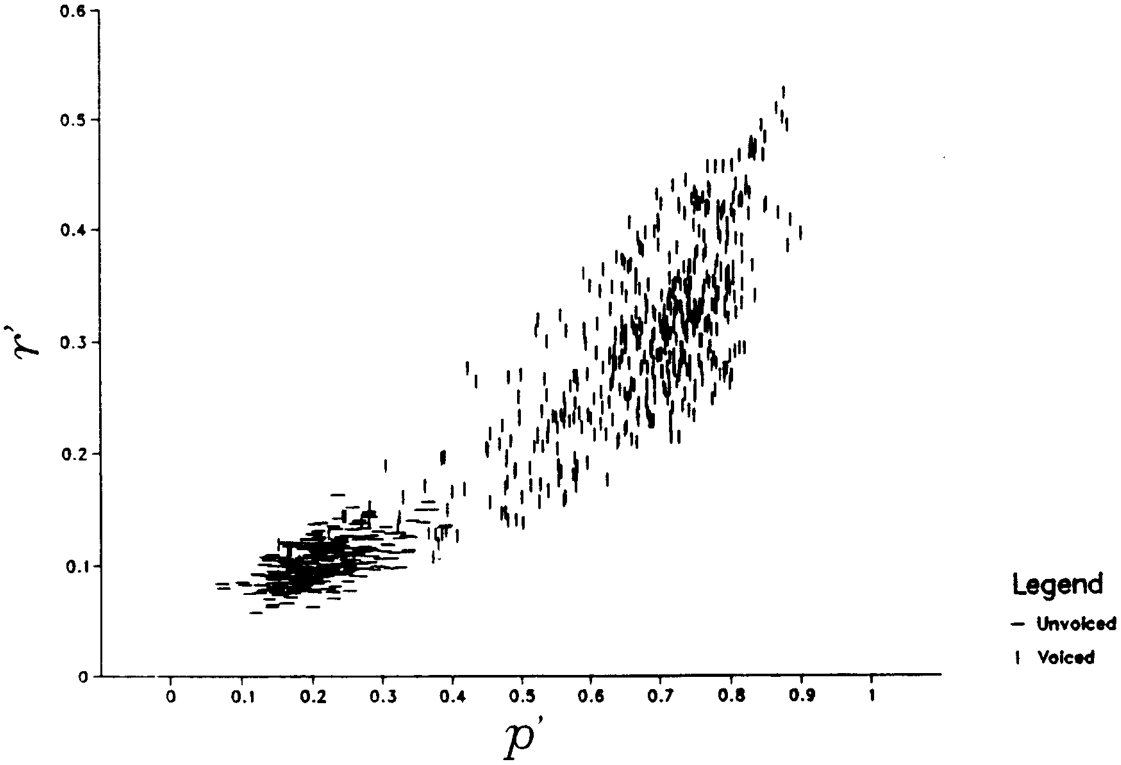

Figure 2.5: r' versus p' for the example sentence "Every salt breeze comes from the sea" spoken by 3 males and 3 females. SNR = −6 dB.

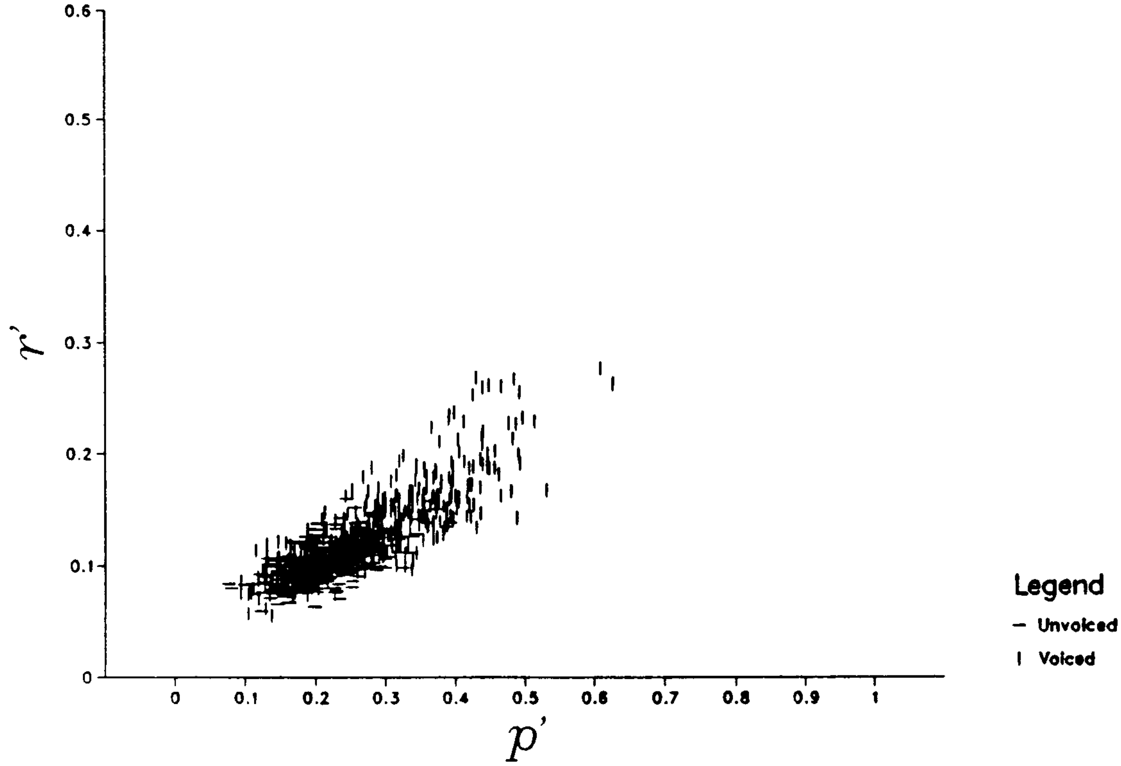

Figure 2.6: r' versus p' for the example sentence "Every salt breeze comes from the sea" spoken by 3 males and 3 females. SNR = −18 dB.

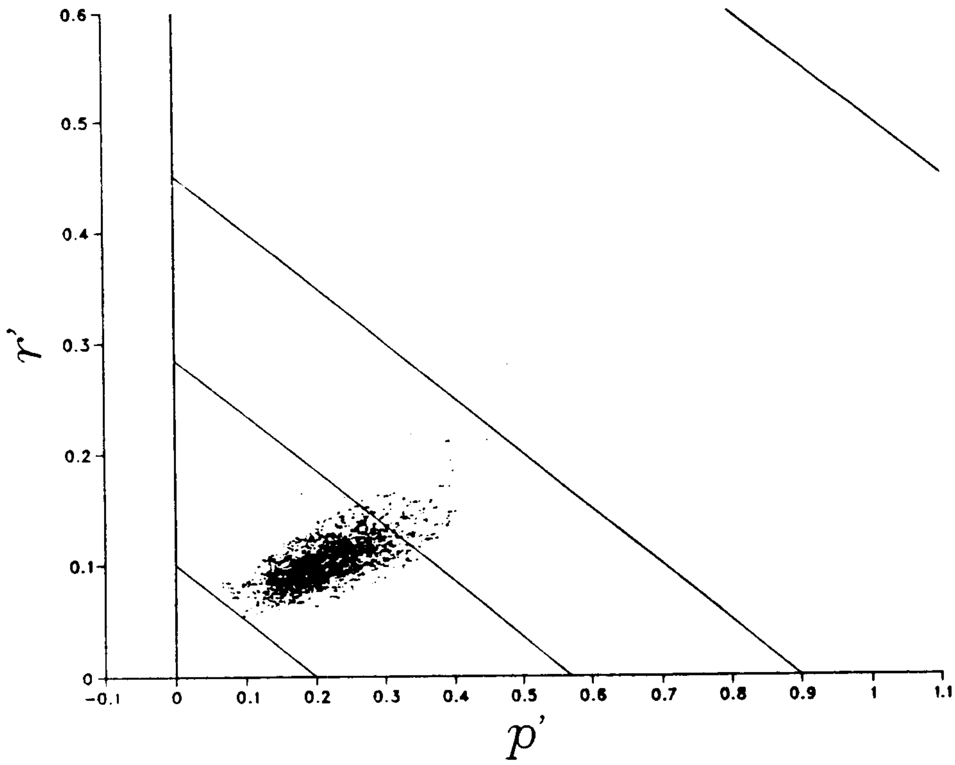

Figure 2.7: r' versus p' for the unvoiced region of the entire development database at seven SNRs. Discrimination boundaries for the voicing decision and voicing confidence are superimposed.

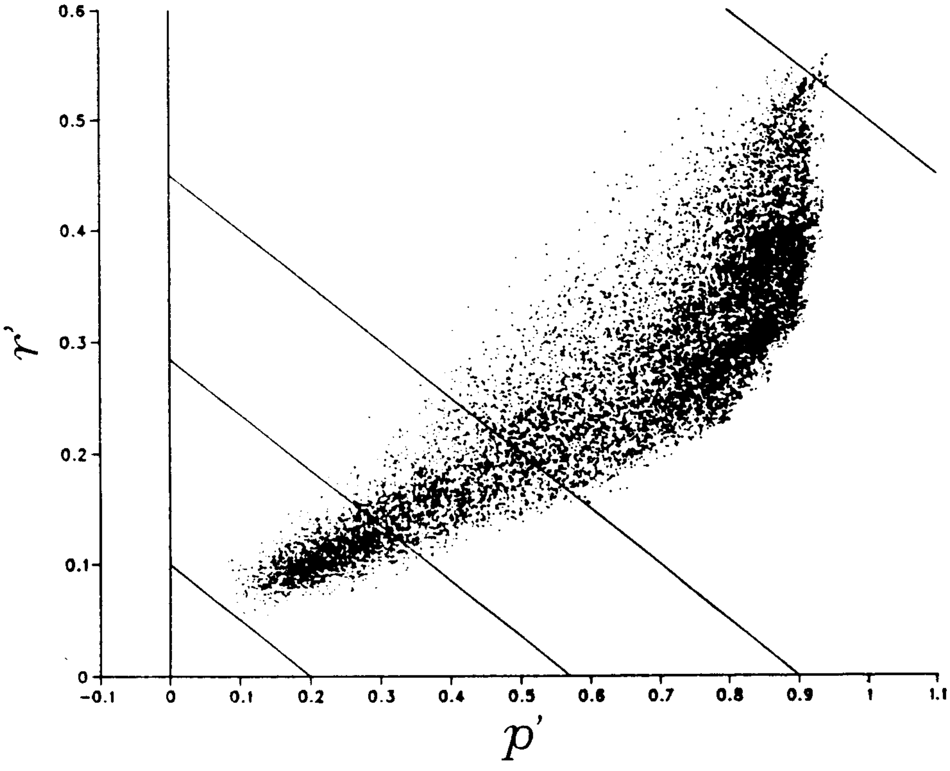

Figure 2.8: r' versus p' for the voiced region of the entire development database at seven SNRs. Discrimination boundaries for the voicing decision and voicing confidence are superimposed.

This has the advantage that the voiced and unvoiced regions need not have a definite separation as is the case with clustering algorithms [35]. This is an advantage because in high levels of noise, there is not a separation of the regions, yet it is still possible to identify many of the voiced speech segments.

After trying several non-linear discrimination boundaries, none were found superior to a simple linear boundary. Superimposed on both Figure 2.7 and Figure 2.8 is a plot of the boundaries used in this algorithm. Only five values are necessary to describe these four boundaries. The slope of the lines is $-.5$ and the y-axis intercepts are .1, .285, .45, and 1.0. The boundary at $y = .285$ is used for the voicing decision and will be called the "critical" boundary. This value is chosen to minimize the numbers of voicing errors at 0 dB SNR. The other three boundaries are used for the voicing confidence measure and will be discussed later.

The results of the voicing algorithm are given in Table 2.2. The table includes voiced-to-unvoiced (V/U) and unvoiced-to-voiced (U/V) errors [25]. These errors are tabulated in terms of the number of errors and the percentage of errors. For very high levels of noise, the U/V errors remain relatively low and the V/U errors approach 100 percent. Both of these results are consistent with the discussion above. The transitional segments are not included in the error calculations.

SNR (dB)	V/U Error (#)	V/U Error (%)	U/V Error (#)	U/V Error (%)	Total Error (#)
∞	18	0.87	0	0.00	18
12	19	0.92	0	0.00	19
6	22	1.07	3	0.24	25
0	32	1.55	29	2.31	61
−6	125	6.06	39	3.11	164
−12	481	23.32	34	2.71	515
−18	1484	71.93	31	2.47	1515

Number of Voiced Segments = 2063
Number of Unvoiced Segments = 1256
Number of Transitional Segments = 1059

Table 2.2: Results of the Voicing Decision Algorithm

2.2.4 Voicing Confidence

A voicing confidence measure, c_v, is defined to have a range of $[-1.0, +1.0]$. $c_v < 0.0$ denotes an unvoiced segment while $c_v \geq 0.0$ denotes a voiced segment. The larger the magnitude of the confidence, the more likely the voicing decision is correct.

Referring to Figure 2.8, it is reasonable to expect that the further to the upper right a point is, the more likely it is that the segment is voiced. For any point lying above the upper boundary, $c_v = +1.0$. The area between the critical boundary and the first boundary to its upper right is linearly mapped from 0.0 to +0.9. The area between the two upper right boundaries is linearly mapped from +0.9 to +1.0. This positive confidence measure reflects a normalized level of periodicity. The area between the critical boundary and the lower left boundary is linearly mapped from 0.0 to −1.0. For any point below the lower boundary, $c_v = -1.0$.

The results of the confidence algorithm are given in Table 2.3. This table summarizes the data for all seven SNRs. The V/U errors are tabulated categorically for $c_v < 0.0$ and are also totaled independent of confidence. The U/V errors are similarly tabulated. The voicing error is defined as the percentage of segments in a given confidence range that are classified incorrectly. Column three shows the number of segments in error and column four shows the total number of segments in the given confidence range.

Since the percentage based error depends on the ratio of the number of voiced segments to the number of unvoiced segments in the speech under analysis, an error measure is defined which is independent of the voiced to unvoiced ratio of the speech material. A few variables are introduced which are used to define the normalized error ratio. M_V is the number of voiced segments falling within a given confidence range. M_U is the number of unvoiced segments falling within a given confidence range. T_V is the total number of voiced segments in the entire database. T_U is the total number of unvoiced segments in the entire database. The error ratio depends on the type of error that occurred, V/U or U/V. Consequently, there is a V/U error ratio, $\mathrm{ER}_{V/U}$, which is used for $c_v < 0.0$ and an U/V error ratio,

c_v	Voicing Error (%)	Voicing Error (#)	Total Segments (#)	Normalized Error Ratio
$[-1.0, -0.9)$	0.46	19	4170	0.0028
$[-0.9, -0.8)$	0.00	0	7	0.0000
$[-0.8, -0.7)$	13.64	6	44	0.0961
$[-0.7, -0.6)$	17.78	24	135	0.1316
$[-0.6, -0.5)$	18.89	78	413	0.1418
$[-0.5, -0.4)$	19.12	187	978	0.1439
$[-0.4, -0.3)$	27.55	362	1314	0.2315
$[-0.3, -0.2)$	31.41	480	1528	0.2788
$[-0.2, -0.1)$	38.38	494	1287	0.3793
$[-0.1, +0.0)$	55.25	531	961	0.7518
V/U Errors	20.13	2181	10837	0.1534
$[+0.0, +0.1)$	25.64	80	312	0.5664
$[+0.1, +0.2)$	14.64	41	280	0.2818
$[+0.2, +0.3)$	2.51	6	239	0.0423
$[+0.3, +0.4)$	2.53	5	198	0.0426
$[+0.4, +0.5)$	1.58	3	190	0.0264
$[+0.5, +0.6)$	0.00	0	184	0.0000
$[+0.6, +0.7)$	0.00	0	209	0.0000
$[+0.7, +0.8)$	0.46	1	218	0.0076
$[+0.8, +0.9)$	0.00	0	204	0.0000
$[+0.9, +1.0]$	0.00	0	10362	0.0000
U/V Errors	1.10	136	12396	0.0182
Number of Voiced Segments = 14441 Number of Unvoiced Segments = 8792				

Table 2.3: Results of the Voicing Confidence algorithm

$\mathrm{ER}_{U/V}$, which is used for $c_v \geq 0.0$.

$$\mathrm{ER}_{V/U} = \frac{M_V}{M_U} \qquad c_v < 0.0 \tag{2.7}$$

$$\mathrm{ER}_{U/V} = \frac{M_U}{M_V} \qquad c_v \geq 0.0 \tag{2.8}$$

Equation 2.7 and Equation 2.8 express the ratio of the number of segments incorrectly classified to the number correctly classified for each confidence range.

Since the error ratio will depend on the V/U ratio of the database, T_V/T_U, a normalized error ratio is defined.

$$\mathrm{NER}_{V/U} = \frac{(M_V/T_V)}{(M_U/T_U)} = \frac{\mathrm{ER}_{V/U}}{(T_V/T_U)} \tag{2.9}$$

$$\mathrm{NER}_{U/V} = \frac{(M_U/T_U)}{(M_V/T_V)} = \frac{\mathrm{ER}_{U/V}}{(T_U/T_V)} \tag{2.10}$$

For example, the range [0.2,0.3) has 6 unvoiced segments and $239 - 6 = 233$ voiced segments. There are 14441 voiced segments in the database and 8792 unvoiced segments. The normalized error ratio for this range is $\mathrm{NER}_{U/V} = (6/8792)/(233/14441) = 0.0423$.

The normalized error ratio is included as column five in Table 2.3. As shown by Equation 2.9 and Equation 2.10, this measure has the advantage of being independent of the V/U ratio of the speech material. The voicing confidence algorithm produces a confidence measure, c_v, with the desirable property that the normalized error ratio decreases with increasing magnitude of confidence.

It is interesting to note that the normalized error ratio may be converted to an error ratio for any given database by multiplying $\mathrm{NER}_{V/U}$ (for $c_v < 0.0$) by the V/U ratio of the new database, T_V/T_U, and by multiplying $\mathrm{NER}_{U/V}$ (for $c_v \geq 0.0$) by the U/V ratio, T_U/T_V. If there are more voiced segments in the new database, the error ratio will reflect the increase of V/U errors. If no assumption is made regarding the V/U ratio for a given database, the voicing confidence can only be used qualitatively.

One final note is that Table 2.3 is a compilation of all SNRs. While the normalized error ratio is independent of the V/U ratio of the speech material, it is still dependent on the SNR.

2.2.5 Voicing Decision and Confidence Smoothing

A modified median 9 smoother [36] is applied to c_v resulting in $\hat{c}_v$. Since the voicing decision is inherent in c_v, it also is smoothed. Median 9 was chosen for this 75 percent overlap algorithm so that it would cover the same amount of time as a median 5 smoother used with 50 percent overlap methods [22]. Since voiced points tend toward the unvoiced region as noise is added, it is more likely that a voiced point will be low in confidence (tending toward -1.0) than an unvoiced point will be high in confidence

SNR (dB)	V/U Error (#)	V/U Error (%)	U/V Error (#)	U/V Error (%)	Total Error (#)
∞	5	0.24	0	0.00	5
12	5	0.24	0	0.00	5
6	3	0.15	3	0.24	6
0	15	0.73	29	2.31	44
−6	79	3.83	39	3.11	118
−12	401	19.44	34	2.71	435
−18	1402	67.96	31	2.47	1433
Number of Voiced Segments = 2063 Number of Unvoiced Segments = 1256 Number of Transitional Segments = 1059					

Table 2.4: Results of the Smoothed Voicing Decision Algorithm

(tending toward +1.0). For this reason, the median 9 smoother is modified such that the median value of the confidence is accepted only if it is greater in value than the original confidence.

The results of the smoothed voicing algorithm are given in Table 2.4. Compared with Table 2.2, many of the V/U errors are corrected without an increase in the number of U/V errors. There are more U/V errors than V/U errors for 0 dB SNR while the opposite is true for most of the remaining SNRs. The critical discrimination boundary may be raised to balance the errors at, for example, 0 dB, but many V/U errors will result for most SNRs.

The results of the smoothed confidence algorithm are given in Table 2.5.

Many voiced segments now have a higher confidence and many V/U errors have been corrected. This smoothing technique has introduced no errors (at least for the development database). The normalized error ratio for the confidence ranges $[-0.9, -0.8)$ and $[-0.7, -0.6)$ have no meaning because there are no segments in those ranges.

2.2.6 Pitch Confidence

Giordano [7] has proposed the use of a pitch confidence measure, but did not propose a method of obtaining such a measure. This section will assume the need for pitch confidence and propose a method of obtaining it.

A pitch confidence measure, c_0, is defined to have a range of $[0.0, +1.0]$. Pitch error is defined as $F_{0_{err}}$ (Equation 2.3). A confidence of 0.0 implies $|F_{0_{err}}| < 20$. A confidence of $+1.0$ implies $|F_{0_{err}}| = 0$. The confidence range of 0.0 to $+1.0$ is linearly mapped from 20 percent to 0 percent deviation. The maximum deviation of 20 percent is a useful limit because it corresponds to the definition of 20 percent gross error and, as will be seen in this section, corresponds to the base of the pitch deviation triangle.

Figure 2.9 shows the pitch error for all the voiced segments in the database (SNR $= \infty$ dB). The y-axis is $F_{0_{err}}$ and the x-axis is the value

$$x \;=\; p' + 2r'. \tag{2.11}$$

The thresholds for e' and p' in the voicing decision (Section 2.2.3) are also

$\hat{c}_v$	Voicing Error (%)	Voicing Error (#)	Total Segments (#)	Normalized Error Ratio
$[-1.0,-0.9)$	0.24	10	4099	0.0015
$[-0.9,-0.8)$	0.00	0	0	–
$[-0.8,-0.7)$	0.00	0	1	0.0000
$[-0.7,-0.6)$	0.00	0	0	–
$[-0.6,-0.5)$	0.00	0	12	0.0000
$[-0.5,-0.4)$	5.67	20	353	0.0366
$[-0.4,-0.3)$	13.84	197	1423	0.0978
$[-0.3,-0.2)$	24.19	501	2071	0.1943
$[-0.2,-0.1)$	36.52	561	1536	0.3503
$[-0.1,+0.0)$	57.98	621	1071	0.8402
V/U Errors	18.08	1910	10566	0.1343
$[+0.0,+0.1)$	25.56	80	313	0.5640
$[+0.1,+0.2)$	13.71	41	299	0.2610
$[+0.2,+0.3)$	2.71	6	221	0.0458
$[+0.3,+0.4)$	2.51	5	199	0.0423
$[+0.4,+0.5)$	1.61	3	186	0.0269
$[+0.5,+0.6)$	0.00	0	176	0.0000
$[+0.6,+0.7)$	0.00	0	220	0.0000
$[+0.7,+0.8)$	0.47	1	214	0.0077
$[+0.8,+0.9)$	0.00	0	212	0.0000
$[+0.9,+1.0]$	0.00	0	10627	0.0000
U/V Errors	1.07	136	12667	0.0178
Number of Voiced Segments = 14441 Number of Unvoiced Segments = 8792				

Table 2.5: Results of the Smoothed Voicing Confidence Algorithm

applied here. For any fixed x, Equation 2.11 represents a line parallel to the discrimination boundaries of the voicing algorithm. (If the error is added to Figure 2.4 as a third axis, Figure 2.9 can be viewed as a two dimensional perspective of this three dimensional plot.) Therefore, the point $x = .57$ is the threshold for the voicing decision.

As noise is added, Figure 2.9–Figure 2.11, the points move to the left and begin to diverge. The plots suggest that for a given x value, the error density function is independent of the SNR. This is due to the fact that the value of x for a voiced segment depends on the SNR of the segment and not the SNR of the utterance. Therefore, knowledge of the SNR is not needed to determine the error density function for a given value of x.

Figure 2.12 is a plot of the entire development database for seven SNRs. Since all of the points represent voiced segments, any points with $x < .57$ represent V/U errors. There is a distinct triangle within which most of the errors occur. This will be called the "pitch deviation triangle" and it is superimposed on Figure 2.12. This triangle can be mapped to the pitch confidence, c_0, by the following equation.

$$c_0 = \begin{cases} 0.0 & x < 0.5 \\ \frac{x-0.5}{1.7} & 0.5 \leq x < 2.2 \\ 1.0 & 2.2 \leq x \end{cases} \tag{2.12}$$

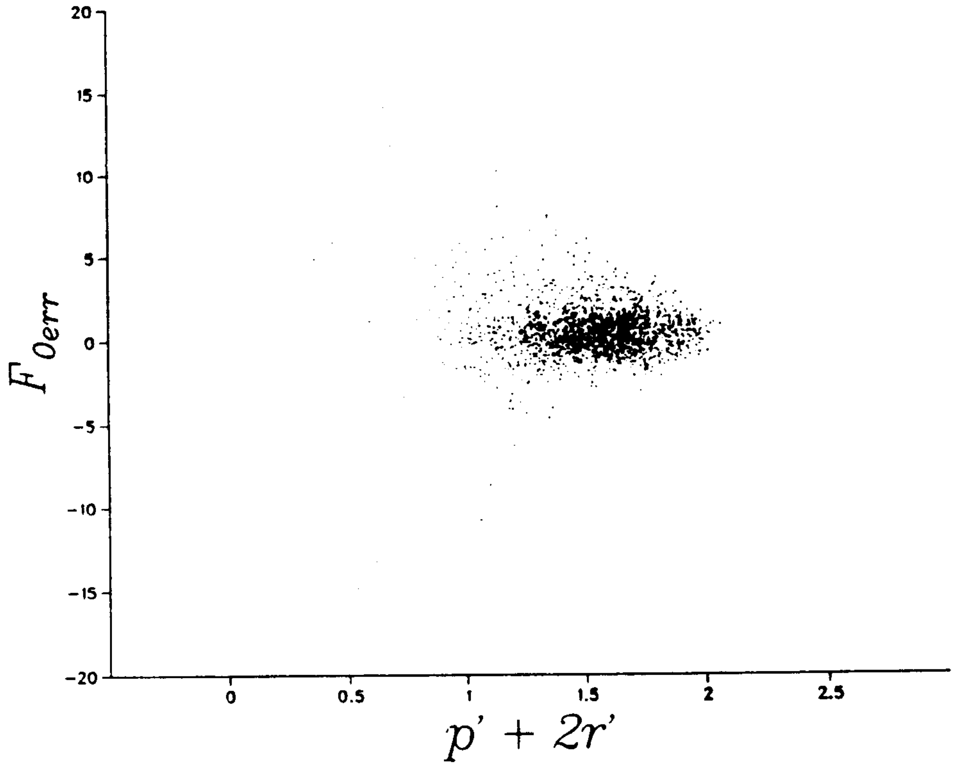

Figure 2.9: $F_{0_{err}}$ versus $p' + 2r'$ for the entire development database. SNR $= \infty$ dB.

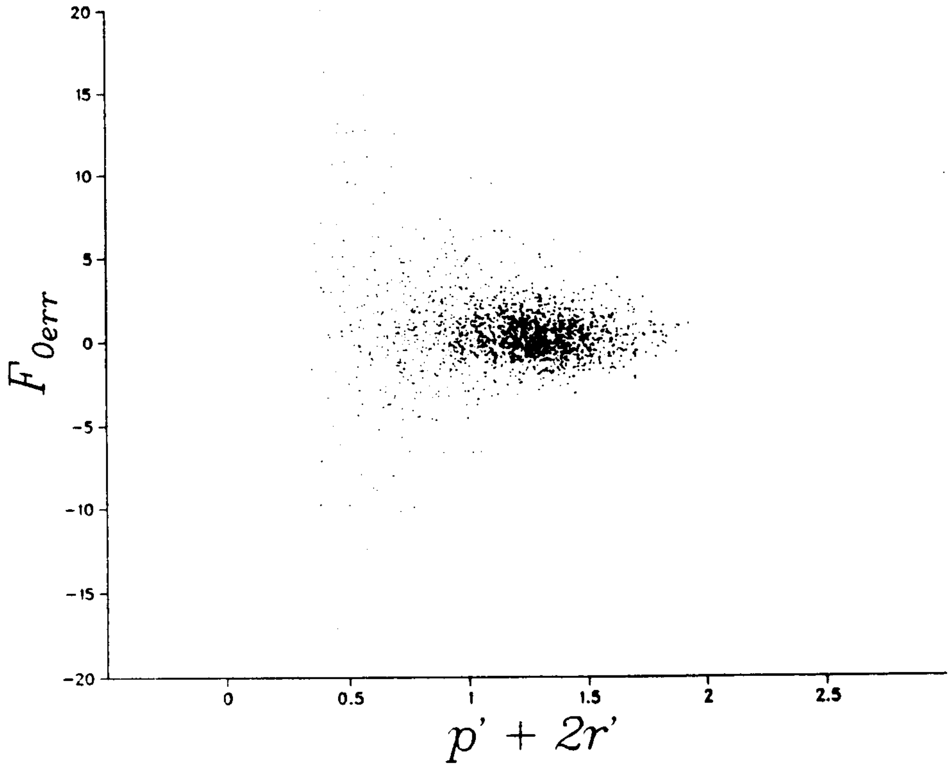

Figure 2.10: $F_{0_{err}}$ versus $p' + 2r'$ for the entire development database. SNR = -6 dB.

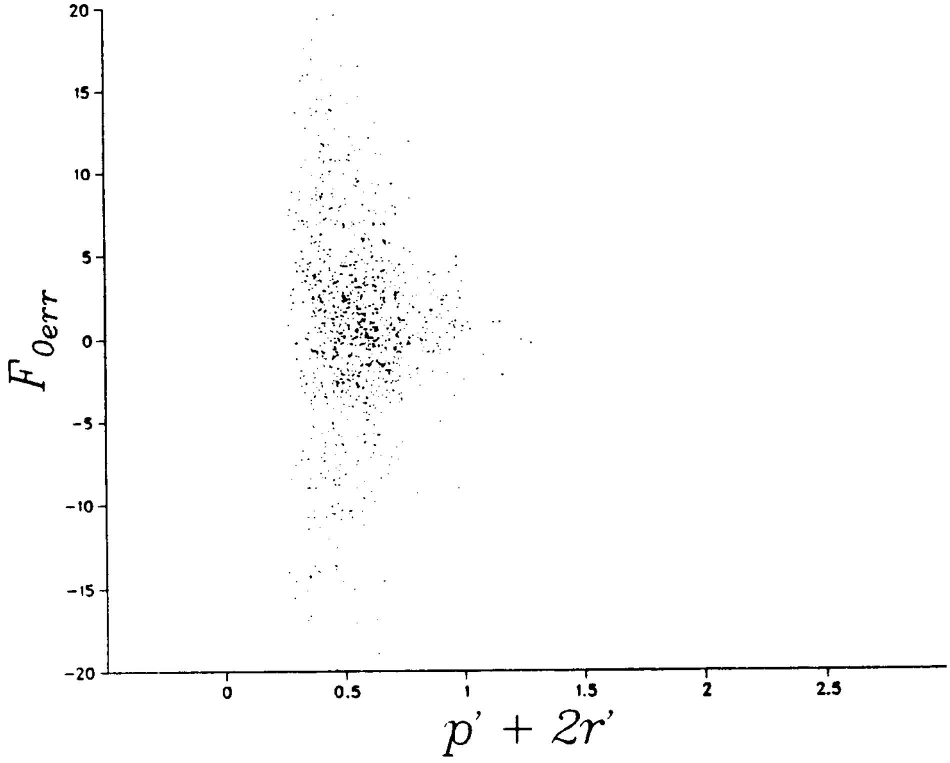

Figure 2.11: $F_{0_{err}}$ versus $p' + 2r'$ for the entire development database. SNR = -18 dB.

Note that the point $x = .57$ maps to $c_0 = 0.04$. The maximum deviation of the pitch in Hz is predicted by F_{0_d} which is calculated using the equation

$$F_{0_d} \quad = \quad .2(1 - c_0)F_0. \tag{2.13}$$

The results of the pitch confidence algorithm are given in Table 2.6. The rightmost column contains a tabulation of the number of segments in each confidence range. Column four contains a tabulation of the number of 20 percent gross errors that occur and column two contains the resulting percentage of errors. Column five contains a tabulation of the errors that occur outside of the deviation predicted by the confidence measure, that is, $|F_{0_{err}}| > 20(1 - c_0)$. These errors will be called "confidence errors." Column three contains the resulting percentage of confidence errors for a given confidence range. Totals are given at the bottom of the table. Recall that these results are for all seven SNRs.

Table 2.6 is arranged so that the pitch confidence, c_0, can be interpreted as providing two items of information, 1) the probabilistic maximum deviation of the pitch and 2) the probability that the pitch is within that deviation. For example, $c_0 = .12$ infers that there is a $100 - 8.84 = 91.16$ percent chance that the actual pitch is within $20(1 - .12) = 17.6$ percent of the estimated pitch.

There exist errors that are greater than 20 percent (gross errors). To

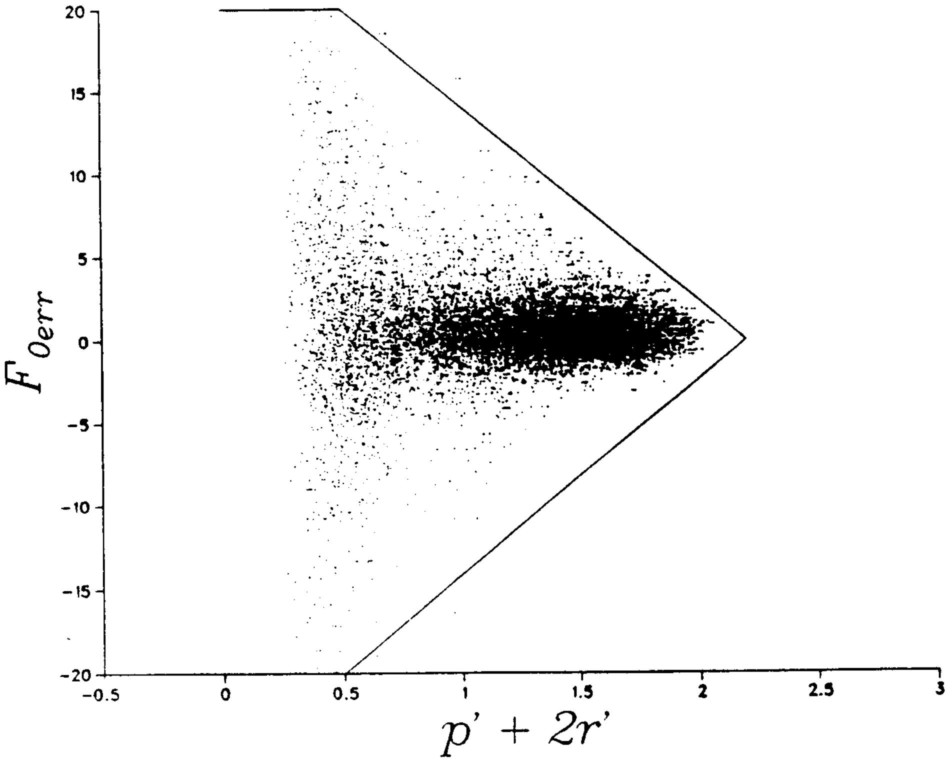

Figure 2.12: $F_{0_{err}}$ versus $p' + 2r'$ for the entire development database at seven SNRs. The pitch deviation triangle is superimposed.

c_0	Gross Error (%)	Confidence Error (%)	Gross Error (#)	Confidence Error (#)	Total Segments (#)
[0.00,0.05)	48.73	48.77	1110	1111	2278
[0.05,0.10)	18.51	18.51	102	102	551
[0.10,0.15)	8.16	8.84	36	39	441
[0.15,0.20)	8.60	8.60	41	41	477
[0.20,0.25)	5.50	5.92	26	28	473
[0.25,0.30)	3.38	4.17	17	21	503
[0.30,0.35)	2.06	2.06	12	12	582
[0.35,0.40)	0.76	0.76	5	5	657
[0.40,0.45)	1.66	1.66	14	14	842
[0.45,0.50)	0.69	0.69	7	7	1011
[0.50,0.55)	0.09	0.09	1	1	1133
[0.55,0.60)	0.36	0.36	5	5	1387
[0.60,0.65)	0.15	0.15	2	2	1333
[0.65,0.70)	0.00	0.00	0	0	1290
[0.70,0.75)	0.00	0.00	0	0	718
[0.75,0.80)	0.00	0.00	0	0	403
[0.80,0.85)	0.00	0.00	0	0	264
[0.85,0.90)	0.00	0.00	0	0	94
[0.90,0.95)	0.00	0.00	0	0	4
[0.95,1.00]	0.00	0.00	0	0	0
Totals	9.54	9.61	1378	1388	14441

Table 2.6: Results of the Pitch Confidence Algorithm

visualize these errors, the entire range of errors is plotted in Figure 2.13. The y-axis still represents error, but a different scale is used. The percent error definition would have produced a plot which was heavily skewed. The octave definition of error [34],

$$F_{0_{err(oct)}} = \mathrm{Log}_2\left(\frac{F_0}{F_{0_a}}\right), \tag{2.14}$$

produces a balanced plot. A 20 percent error corresponds roughly to .25 oct. Many pitch doubling errors can be seen at +1.0 oct while tripling, quadrupling, and halving errors occur at +1.6, +2.0, and −1.0 oct respectively. This definition of error is not used for the pitch deviation triangle because the pitch deviation within .25 oct is skewed which would result in a more complex definition of pitch confidence. That is, the fine errors would not be symmetric. Also, any further processing using the pitch confidence based on the octave would require the calculation of a logarithm which results in more computation time than the calculation of a percent.

All of the pitch confidence errors that occur within the confidence range of [0.30,0.65) and most of the errors below $c_0 = 0.3$ are also gross errors. Pitch smoothing will correct many of these errors.

2.2.7 Pitch Smoothing

Many of the gross errors with high pitch confidence are harmonic errors, which is apparent in Figure 2.13. Since many of the gross errors that occur

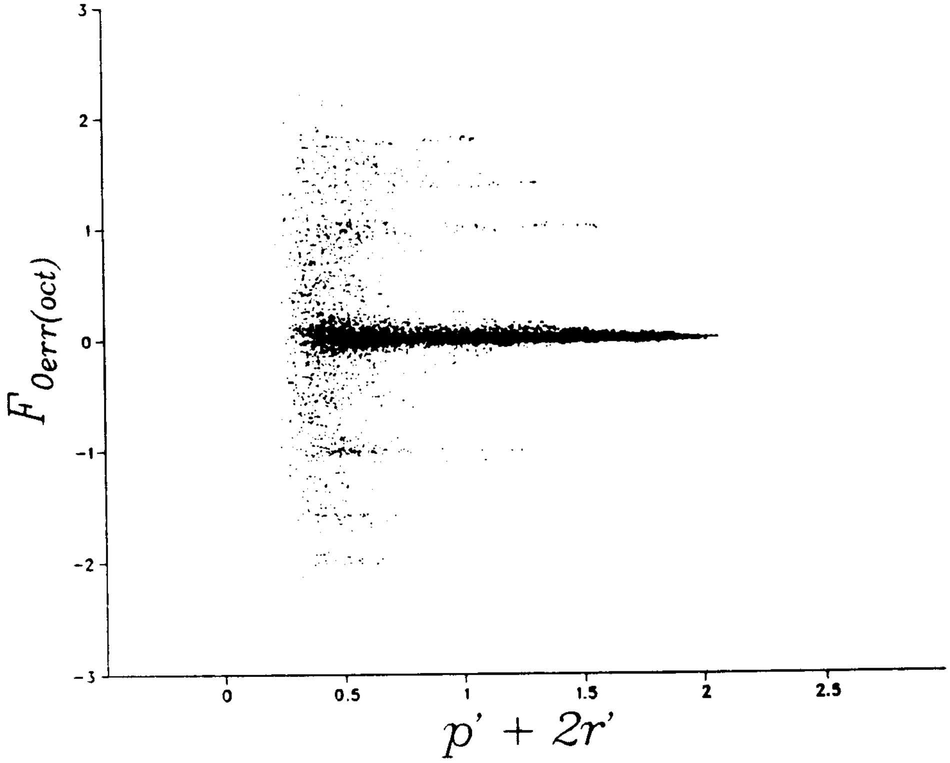

Figure 2.13: $F_{0_{err(oct)}}$ versus $p' + 2r'$ for the entire development database at seven SNRs.

are isolated, a median smoother is useful. As with the voicing smoothing, a modified median 9 smoother [36] is used to smooth the pitch estimate, F_0, resulting in $\hat{F}_0$. There are two conditions under which the original estimate is chosen instead of the median value. First, since there are no gross errors for $c_0 \geq .65$, no smoothing needs to be done. Second, if the original estimate is within 20 percent of the median value, the original estimate is chosen. This 20 percent threshold was found to be better in reducing the standard deviation of the fine errors than the originally proposed 5 percent threshold was [22].

Table 2.7 contains the results of the pitch smoothing algorithm. ($\hat{F}_{0_{err}}$ is used to signify that $F_{0_{err}}$ in Equation 2.3 is calculated using $\hat{F}_0$ instead of F_0.) Comparing Table 2.7 to Table 2.1, the percentage of gross errors has been reduced while the standard deviation of the fine errors has increased only slightly. Many of the gross errors are doubling errors. This information may be used for some pitch synchronous algorithms. If the analysis window is chosen as an even integer number of estimate periods, a pitch doubling error will still result in an integer number of periods within the window. Under this condition, a window size of four times the pitch period may be well suited for formant analysis [6]. A pitch doubling error will result in a window size of twice the pitch period.

The analysis of the pitch confidence after pitch smoothing is found

SNR (dB)	$\hat{F}_{0_{err}}$		Gross Error (%-age of Segments)	Fine Error	
	Average (%)	Standard Deviation (%)		Average (%)	Standard Deviation (%)
∞	0.761	4.259	0.533	0.510	1.974
12	0.783	4.270	0.533	0.530	1.986
6	0.701	2.971	0.388	0.578	2.126
0	0.783	3.163	0.582	0.637	2.170
−6	1.699	7.228	2.181	0.747	2.606
−12	2.358	13.449	6.108	0.803	3.940
−18	−1.257	35.764	29.811	0.884	6.429

Table 2.7: Results of the Smoothed Pitch Detection Algorithm

in Table 2.8. Compared with Table 2.6, many of the gross errors have been eliminated. Since there is no increase in confidence errors, all of the smoothed pitch estimates fall within the pitch deviation triangle. The pitch confidence has been improved by smoothing the pitch. It can be improved further by smoothing the pitch confidence itself.

2.2.8 Pitch Confidence Smoothing

There are two goals set forth in smoothing the pitch confidence, c_0. First, any pitch error that is greater than the maximum deviation indicated by the pitch confidence should have its confidence smoothed downward (toward 0.0). Second, any pitch error that is less than the maximum deviation indicated by the pitch confidence should have its confidence smoothed upward (toward +1.0).

c_0	Gross Error (%)	Confidence Error (%)	Gross Error (#)	Confidence Error (#)	Total Segments (#)
[0.00,0.05)	31.30	31.39	713	715	2278
[0.05,0.10)	9.07	10.16	50	56	551
[0.10,0.15)	5.90	6.80	26	30	441
[0.15,0.20)	3.35	3.35	16	16	477
[0.20,0.25)	1.90	2.33	9	11	473
[0.25,0.30)	0.60	1.39	3	7	503
[0.30,0.35)	0.86	0.86	5	5	582
[0.35,0.40)	0.15	0.15	1	1	657
[0.40,0.45)	0.36	0.48	3	4	842
[0.45,0.50)	0.20	0.20	2	2	1011
[0.50,0.55)	0.00	0.00	0	0	1133
[0.55,0.60)	0.00	0.00	0	0	1387
[0.60,0.65)	0.00	0.00	0	0	1333
[0.65,0.70)	0.00	0.00	0	0	1290
[0.70,0.75)	0.00	0.00	0	0	718
[0.75,0.80)	0.00	0.00	0	0	403
[0.80,0.85)	0.00	0.00	0	0	264
[0.85,0.90)	0.00	0.00	0	0	94
[0.80,0.95)	0.00	0.00	0	0	4
[0.95,1.00]	0.00	0.00	0	0	0
Totals	5.73	5.87	828	847	14441

Table 2.8: Results of the Pitch Confidence Algorithm after Pitch Smoothing

A median 9 smoother [36] is applied to c_0 (resulting in $\hat{c}_0$) with the following condition. Consulting Table 2.8, it can be seen that there are very few pitch errors (gross and confidence errors) for $c_0 \geq .45$. Except for these few errors, the first goal set forth does not apply. To still be in accordance with the second goal, the maximum of the median value and the original confidence is chosen as the smoothed confidence. For $c_0 < .45$, the standard median 9 smoother is applied.

The results of the smoothed confidence algorithm are given in Table 2.9. Although a gross error cannot be corrected by confidence smoothing, the confidence has been lowered for many of these segments (compared to Table 2.8). Smoothing has increased the total number of segments in the range of [0.45,0.85) which is in accordance with the second goal. Unfortunately a few confidence errors have appeared. Precision in determining the error deviation has caused these few errors.

Since the voicing confidence, c_v, and the pitch confidence, c_0, have been smoothed in different ways, a smoothed pitch confidence of $\hat{c}_0 = 0.04$ does not necessarily correspond to a smoothed voicing confidence of $\hat{c}_v = 0.0$.

2.3 Independent Analysis

The method presented in this chapter is an algorithm that does not require training before its use. The behavior of the three features (derived from

$\hat{c}_0$	Gross Error (%)	Confidence Error (%)	Gross Error (#)	Confidence Error (#)	Total Segments (#)
[0.00,0.05)	31.17	31.21	729	730	2339
[0.05,0.10)	5.70	6.11	28	30	491
[0.10,0.15)	4.68	4.68	21	21	449
[0.15,0.20)	2.83	4.14	13	19	459
[0.20,0.25)	2.88	3.29	14	16	486
[0.25,0.30)	2.32	3.29	12	17	517
[0.30,0.35)	0.83	1.16	5	7	604
[0.35,0.40)	0.00	0.00	0	0	570
[0.40,0.45)	0.67	0.83	4	5	600
[0.45,0.50)	0.20	0.20	2	2	1018
[0.50,0.55)	0.00	0.35	0	4	1134
[0.55,0.60)	0.00	0.14	0	2	1421
[0.60,0.65)	0.00	0.00	0	0	1374
[0.65,0.70)	0.00	0.14	0	2	1399
[0.70,0.75)	0.00	0.26	0	2	777
[0.75,0.80)	0.00	0.00	0	0	434
[0.80,0.85)	0.00	0.00	0	0	271
[0.85,0.90)	0.00	0.00	0	0	94
[0.90,0.95)	0.00	0.00	0	0	4
[0.95,1.00]	0.00	0.00	0	0	0
Totals	5.73	5.93	828	857	14441

Table 2.9: Results of the Smoothed Pitch Confidence Algorithm after Pitch Smoothing

the autocorrelation function) in noise was observed so that thresholds and curves could be fit to the data. The results presented thus far have been for the utterances used to develop the algorithm. It is therefore conceivable that the method is biased toward these utterances. To determine if this is the case, the method is applied to an entirely different database (the test database) to assure that the chosen thresholds and curves are representative of the behavior of the features.

This test database is comprised of 6 speakers, 3 male and 3 female. All six speakers are different than those in the first database. The following sentences were spoken.

1) We were away a year ago.

2) I know when my lawyer is due.

3) Every salt breeze comes from the sea.

4) I was stunned by the beauty of the view.

5) Never learn a yellow lion roar.

Sentences 1 and 3 are also contained in the development database.

The results of the smoothed pitch detection algorithm are shown in Table 2.10. The results are similar to Table 2.7. The results of the smoothed voicing decision algorithm are shown in Table 2.11. Compared with Table 2.4, the percentage of V/U errors is slightly higher for low noise and slightly lower for high noise. The results of the smoothed voicing confi-

	$\hat{F}_{0_{err}}$		Gross Error (%-age of Segments)	Fine Error	
SNR (dB)	Average (%)	Standard Deviation (%)		Average (%)	Standard Deviation (%)
∞	0.301	1.451	0.000	0.301	1.451
12	0.292	1.453	0.000	0.292	1.453
6	0.286	1.509	0.026	0.280	1.473
0	0.279	1.797	0.053	0.280	1.720
−6	0.278	2.708	0.211	0.271	2.429
−12	0.681	7.945	2.427	0.449	3.667
−18	−0.212	22.077	15.594	0.401	6.170

Table 2.10: Results of the Smoothed Pitch Detection Algorithm (Test Database)

dence algorithm are shown in Table 2.12. The percentage of V/U errors for low unvoiced confidence (negative confidence near zero) has increased (compared to Table 2.5) and the percentage of U/V errors for low voiced confidence has decreased. This is due to the fact that the ratio of voiced to unvoiced segments is larger in the test database. Comparing the normalized error ratio of the two databases, it can be seen that the expected error is more similar. The results of the smoothed pitch confidence algorithm after pitch smoothing are shown in Table 2.13. Compared with Table 2.9, there are fewer errors in the low confidence range. There are only a few confidence errors that are not gross errors. This indicates that the pitch deviation triangle is still accurate for the test database.

SNR (dB)	V/U Error (#)	V/U Error (%)	U/V Error (#)	U/V Error (%)	Total Error (#)
∞	24	0.63	1	0.07	25
12	24	0.63	2	0.15	26
6	22	0.58	2	0.15	24
0	7	0.18	42	3.10	49
−6	77	2.03	80	5.90	157
−12	424	11.19	79	5.83	503
−18	2156	56.89	81	5.97	2237

Number of Voiced Segments = 3790
Number of Unvoiced Segments = 1356
Number of Transitional Segments = 1104

Table 2.11: Results of the Smoothed Voicing Decision Algorithm (Test Database)

2.4 Conclusions

The method presented has been designed for situations where the speech signal below 600 Hz is not significantly distorted and the added noise is broadband. Consequently, only stationary white noise cases are discussed so that a basis of results is obtained on which to extrapolate to non-stationary and non-white noise results. With the assumption of broadband noise, the results of non-white noise can be implied as long as the SNR below 600 Hz is the same as the white noise case. For non-stationary noise, the results will again depend on the local SNR. Since the method is non-adaptive, the results will not depend on the direction (increasing or decreasing) of the

	Voicing	Voicing	Total	Normalized
$\hat{c}_v$	Error	Error	Segments	Error
	(%)	(#)	(#)	Ratio
$[-1.0,-0.9)$	1.61	74	4600	0.0058
$[-0.9,-0.8)$	0.00	0	0	–
$[-0.8,-0.7)$	0.00	0	0	–
$[-0.7,-0.6)$	0.00	0	1	0.0000
$[-0.6,-0.5)$	5.45	3	55	0.0206
$[-0.5,-0.4)$	18.73	47	251	0.0824
$[-0.4,-0.3)$	19.78	272	1375	0.0882
$[-0.3,-0.2)$	26.60	598	2248	0.1297
$[-0.2,-0.1)$	41.48	847	2042	0.2536
$[-0.1,+0.0)$	65.33	893	1367	0.6741
V/U Errors	22.90	2734	11939	0.1063
$[+0.0,+0.1)$	22.62	133	588	0.8170
$[+0.1,+0.2)$	14.50	76	524	0.4741
$[+0.2,+0.3)$	9.73	43	442	0.3012
$[+0.3,+0.4)$	5.10	21	412	0.1501
$[+0.4,+0.5)$	1.16	4	345	0.0328
$[+0.5,+0.6)$	2.02	7	347	0.0575
$[+0.6,+0.7)$	0.00	0	296	0.0000
$[+0.7,+0.8)$	0.00	0	306	0.0000
$[+0.8,+0.9)$	0.00	0	361	0.0000
$[+0.9,+1.0]$	0.01	3	20462	0.0004
U/V Errors	1.19	287	24083	0.0337
Number of Voiced Segments = 26530 Number of Unvoiced Segments = 9492				

Table 2.12: Results of the Smoothed Voicing Confidence Algorithm (Test Database)

$\hat{c}_0$	Gross Error (%)	Confidence Error (%)	Gross Error (#)	Confidence Error (#)	Total Segments (#)
[0.00,0.05)	19.49	19.58	670	673	3437
[0.05,0.10)	2.02	2.12	19	20	942
[0.10,0.15)	0.50	0.75	4	6	796
[0.15,0.20)	0.00	0.27	0	2	730
[0.20,0.25)	0.14	0.14	1	1	691
[0.25,0.30)	0.00	0.29	0	2	684
[0.30,0.35)	0.00	0.00	0	0	770
[0.35,0.40)	0.00	0.13	0	1	741
[0.40,0.45)	0.00	0.70	0	6	861
[0.45,0.50)	0.00	0.26	0	4	1559
[0.50,0.55)	0.00	0.00	0	0	2286
[0.55,0.60)	0.00	0.00	0	0	2094
[0.60,0.65)	0.00	0.31	0	8	2558
[0.65,0.70)	0.00	0.03	0	1	3555
[0.70,0.75)	0.00	0.11	0	3	2783
[0.75,0.80)	0.00	0.24	0	3	1272
[0.80,0.85)	0.00	0.00	0	0	632
[0.85,0.90)	0.00	0.00	0	0	139
[0.90,0.95)	0.00	0.00	0	0	0
[0.95,1.00]	0.00	0.00	0	0	0
Totals	2.62	2.75	694	730	26530

Table 2.13: Results of the Smoothed Pitch Confidence Algorithm after Pitch Smoothing (Test Database)

changing noise. The portion of the speech spectrum below 600 Hz has the greatest energy (for voiced speech) and is the most likely to survive the addition of high levels of noise. Therefore, the method is only influenced by low frequency noise.

The calculations required for this integrated method are quite reasonable. The only exception is the calculation of the autocorrelation. With fast autocorrelation techniques (Section 3.3, [37],[38]), even this calculation can be performed quickly.

The pitch detection and pitch confidence results are better for female speakers than for male speakers. The higher female pitch causes fewer harmonics in the first formant range. Most of the voiced speech energy is concentrated at these few harmonics which are not easily corrupted by the addition of white noise. For male speakers, the speech energy is spread over many harmonics which are more easily corrupted since they are lower in amplitude for a given total speech energy. Therefore the autocorrelation function is not as highly correlated at the correct period. It has been observed by Sondhi [14] and confirmed by the present author that there are cases where female speech becomes nearly sinusoidal, that is, most of the speech energy is concentrated at only one harmonic.

The method is non-adaptive and requires no a priori information. It is felt that adaptive techniques [30] could be used to improve the results for both the pitch detector and the voicing decision. Additional features could be added to the voicing decision, but observing their relative behavior in noise may be difficult.

Clipping algorithms have been used for spectral flattening [14], [18]. The clp[x] algorithm [18] was implemented with the autocorrelation pitch detector in this chapter and applied to the development database. For the smoothed pitch detection results, the number of gross errors were reduced up to 50 percent for low levels of noise. For higher levels of noise (less than 0 dB), the results degraded. This is probably a result of the clipping levels being influenced by the noise so that mostly noise components are left after clipping. In terms of spectral flattening, high levels of white noise imply a flat spectrum. A spectral flattener will probably reduce the components that are not flat, the speech components. Since a reduction of errors for speech in such noise was a goal of this chapter, clipping algorithms were not explored further.

Chapter 3

Short-Time Analysis of Noise

This chapter provides an intuitive and mathematical basis necessary for the understanding and implementation of the noise estimation algorithm described in Chapter 4 [39]. There are many techniques for estimating noise [40]. The most appropriate method depends on the application and only those techniques necessary for Chapter 4 are discussed.

This chapter is divided into a discussion of these main topics. First, a popular method of spectral averaging and spectral smoothing is detailed. Second, the spectral averaging method is extended. Although the two spectral averaging methods provide the same expected value of the power spectrum, the extended method will be shown to be superior in certain spectral separation cases (Chapter 4). Finally, cepstral windowing will be discussed as a way to improve the sidelobe performance of spectral smoothing.

3.1 Introduction

It is often necessary to estimate the power spectrum of a random sequence [40]. Broadband noise (not impulse or sinusiodal noise) can be modeled as a white noise sequence with zero mean and unit variance driving a linear filter which may be time varying. The noise estimation then becomes a filter estimation problem. If the filter has only poles, the noise estimation can be made from the autoregressive (AR) model [41]. This is also know as LPC analysis [42]. If the filter has only zeroes, the estimation can be made from the moving average (MA) model [41]. The autocorrelation function of a MA model of a noise sequence is finite in lag [43].

Noise estimation for any linear model can effectively be accomplished with windowing [44] and short-time Fourier analysis [45] [37] [46]. Noise estimation using these techniques is applied in Chapter 4 and is therefore discussed in Chapter 3. It is important to note that a finite number of poles can be modeled using an infinite number of zeroes [43] so any noise that is modeled using poles can be approximated with a finite number of zeroes.

3.2 Definitions

The **power spectrum** of a noise sequence is defined as the Fourier transform of its **autocorrelation function** [43]. The purpose of this chapter is to describe techniques for estimating this power spectrum using a finite sequence of the noise.

A distinction is made between a **point** and a **sample**. A point is a single value obtained from, for example, an A/D converter. A sample is a number of consecutive points from a discrete time sequence [47]. A sample may also be referred to as a **segment**.

The **sample autocorrelation function** is the autocorrelation of a sample. The **sample power spectrum** is the Fourier transform (DFT) of the sample autocorrelation function. It will be shown that the sample power spectrum can be calculated using the short-time Fourier transform (DFT) of the sample. This method of calculating the sample power spectrum is more efficient. The sample power spectrum is also referred to as a **periodogram** [47].

If a windowing function (other than rectangular) is applied to the sample, the resulting DFT is called a **windowed sample power spectrum**. Often the term "windowed" is dropped and the type of window (rectangular or otherwise) is implied by the context. The windowed sample power spectrum is also referred to a **modified periodogram** [45].

It turns out that a sample power spectrum is not a good estimate of the power spectrum [47]. The next section will show the mathematical relationship between the sample, the sample autocorrelation, and the sample power spectrum. The following sections will then show a variety of ways to estimate the power spectrum using sample power spectrums.

3.3 Sample Power Spectrum

The DFT of the sample autocorrelation function of an N point sample, $x(n)$, is equal to the squared magnitude of the DFT divided by N. The sample autocorrelation function needs to be interpreted as shown in the following proof and the sample needs to be padded before performing the DFT. First, the squared magnitude of the DFT of the sample is solved in terms of $x(n)$.

$$x_1(n) = \begin{cases} x(n) & n = 0, 1, \ldots, N-1 \\ 0 & n = N, N+1, \ldots, 2N-1 \end{cases} \tag{3.1}$$

$$\begin{aligned}
X_1(k) &= \sum_{n=0}^{2N-1} x_1(n) e^{-j\frac{2\pi}{2N}nk} \quad k = 0, 1, \ldots, 2N-1 \\
X_1(k) &= \sum_{n=0}^{N-1} x(n) e^{-j\frac{\pi}{N}nk} \\
X_1(k) &= \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi}{N}nk\right) - j \sum_{n=0}^{N-1} x(n) \sin\left(\frac{\pi}{N}nk\right) \\
|X_1(k)|^2 &= \left[\sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi}{N}nk\right)\right]^2 + \left[\sum_{n=0}^{N-1} x(n) \sin\left(\frac{\pi}{N}nk\right)\right]^2
\end{aligned}$$

$$|X_1(k)|^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)\cos\left(\frac{\pi}{N}nk\right) x(m)\cos\left(\frac{\pi}{N}mk\right) + \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)\sin\left(\frac{\pi}{N}nk\right) x(m)\sin\left(\frac{\pi}{N}mk\right)$$

$$|X_1(k)|^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} \left[x(n)\cos\left(\frac{\pi}{N}nk\right) x(m)\cos\left(\frac{\pi}{N}mk\right) + x(n)\sin\left(\frac{\pi}{N}nk\right) x(m)\sin\left(\frac{\pi}{N}mk\right)\right]$$

$$|X_1(k)|^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)x(m)\left[\cos\left(\frac{\pi}{N}nk\right)\cos\left(\frac{\pi}{N}mk\right) + \sin\left(\frac{\pi}{N}nk\right)\sin\left(\frac{\pi}{N}mk\right)\right]$$

$$|X_1(k)|^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)x(m)\cos\left[\frac{\pi}{N}nk - \frac{\pi}{N}mk\right]$$

$$|X_1(k)|^2 = \sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)x(m)\cos\left[\frac{\pi}{N}(n-m)k\right] \tag{3.2}$$

Second, the sample autocorrelation is solved in terms of $x(n)$. Initially, define the sample autocorrelation as follows [47].

$$r'(m) = \begin{cases} 0 & m = -N \\ \dfrac{1}{N+m}\displaystyle\sum_{n=0}^{N-1+m} x(n)x(n-m) & m = -N+1,\ldots,-1 \\ \dfrac{1}{N-m}\displaystyle\sum_{n=0}^{N-1-m} x(n)x(n+m) & m = 0,1,\ldots,N-1 \end{cases}$$

$r'(m)$ is an unbiased estimate of the autocorrelation function and the variance at a given lag goes to zero as N goes to infinity. A second definition of the sample autocorrelation will be used in this chapter [47].

$$r(m) = \begin{cases} 0 & m = -N \\ \frac{1}{N} \sum_{n=0}^{N-1+m} x(n)x(n-m) & m = -N+1, \ldots, -1 \\ \frac{1}{N} \sum_{n=0}^{N-1-m} x(n)x(n+m) & m = 0, 1, \ldots, N-1 \end{cases}$$

$r(m)$ is biased, but both the bias and the variance of a given lag go to zero as N goes to infinity so the estimate is still consistent [47].

The motivation for the use of this second definition of the sample autocorrelation function is that it will make the first statement of this section true. Nuttall and Carter [46] use the unbiased estimate, $r'(m)$, and discuss the biases and variances associated with $r'(m)$, $r(m)$, and the resulting sample power spectrums. They calculate the mathematically efficient $r(m)$ and use lag weighting to arrive at $r'(m)$. This work is beyond the scope of this chapter since it is not used in Chapter 4.

Now, $p(l)$ is defined so that the efficient DFT can be used for the calculations.

$$p(l) = \begin{cases} r(l) & l = 0, 1, \ldots, N-1 \\ 0 & l = N \\ r(l-2N) & l = N+1, \ldots, 2N-1 \end{cases}$$

$p(l)$ is even so its DFT is its cosine transform.

$$P(k) = \frac{1}{N} \sum_{l=0}^{2N-1} p(l) \cos\left(\frac{2\pi}{2N} lk\right) \quad k = 0, 1, \ldots, 2N-1$$

$$P(k) = \frac{1}{N} \left\{ \sum_{l=0}^{N-1} r(l) \cos\left(\frac{\pi}{N} lk\right) + \sum_{l=N+1}^{2N-1} r(l-2N) \cos\left(\frac{\pi}{N} lk\right) \right\}$$

$$\begin{aligned} P(k) = \frac{1}{N} \Bigg\{ & \sum_{l=0}^{N-1} \sum_{n=0}^{N-1-l} x(n) x(n+l) \cos\left(\frac{\pi}{N} lk\right) \\ & + \sum_{l=N+1}^{2N-1} \sum_{n=0}^{l-N-1} x(n) x(n-l+2N) \cos\left(\frac{\pi}{N} lk\right) \Bigg\} \end{aligned}$$

$$\begin{aligned} P(k) = \frac{1}{N} \Bigg\{ & \sum_{l=0}^{N-1} \sum_{n=0}^{N-1-l} x(n) x(n+l) \cos\left(\frac{\pi}{N} lk\right) \\ & + \sum_{l=1}^{N-1} \sum_{n=0}^{N-1-l} x(n) x(n+l) \cos\left(\frac{\pi}{N} lk\right) \Bigg\} \end{aligned}$$

$$\begin{aligned} P(k) = \frac{1}{N} \Bigg\{ & \sum_{n=0}^{N-1} \sum_{l=0}^{N-1-n} x(n) x(n+l) \cos\left(\frac{\pi}{N} lk\right) \\ & + \sum_{n=0}^{N-2} \sum_{l=1}^{N-1-n} x(n) x(n+l) \cos\left(\frac{\pi}{N} lk\right) \Bigg\} \end{aligned}$$

$$\begin{aligned} P(k) = \frac{1}{N} \Bigg\{ & \sum_{n=0}^{N-1} \sum_{m=n}^{N-1} x(n) x(m) \cos\left[\frac{\pi}{N}(m-n)k\right] \\ & + \sum_{n=0}^{N-2} \sum_{m=n+1}^{N-1} x(n) x(m) \cos\left[\frac{\pi}{N}(m-n)k\right] \Bigg\} \end{aligned}$$

$$P(k) = \frac{1}{N}\left\{\sum_{n=0}^{N-1} x^2(n) + 2\sum_{n=0}^{N-2}\sum_{m=n+1}^{N-1} x(n)x(m)\cos\left[\frac{\pi}{N}(m-n)k\right]\right\}$$

$$P(k) = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} x(n)x(m)\cos\left[\frac{\pi}{N}(n-m)k\right]$$

From Equation 3.2,

$$P(k) = \frac{1}{N}|X_1(k)|^2 \quad k = 0,1,\ldots,2N-1. \tag{3.3}$$

This result proves the statement at the beginning of this section. A fast autocorrelation could be performed by taking the IFFT of the squared magnitude of the FFT of the padded sample and dividing by N. The resulting autocorrelation needs to be interpreted as $p(l)$. Recall that $p(l)$ is defined so that a standard FFT can be used to reduce computation time when calculating the autocorrelation. The coefficients of $p(l)$ may be rearranged or scaled to obtain the autocorrelation definition of choice. Chapter 4 will use the $p(l)$ definition of the autocorrelation function because of its computational efficiency.

3.4 Windowed Sample Power Spectrum

All of the mathematics in Section 3.3 still apply if the sample is windowed. The only difference is that $P(k)$ needs to be normalized for the window [45] so that the mean energy of $P(k)$ is not biased. Let $x(n)$ in Equation 3.1 equal $s(n)w(n)$ where $s(n)$ is the N point sample of the noise and $w(n)$ is the N point window. $P(k)$ in Equation 3.3 now becomes [45]

$$P(k) = \frac{|X_1(k)|^2}{\sum_{m=0}^{N-1} w^2(m)} \quad k = 0, 1, \ldots, 2N-1. \tag{3.4}$$

It can be seen that for a rectangular window, Equation 3.4 reduces to Equation 3.3.

3.5 Sample Power Spectrum Examples

For the following examples, white gaussian noise is passed through an FIR filter to create non-white noise. Figure 3.1 shows the power response of the FIR filter versus the normalized frequency. This power response is the power spectrum of the noise. Figure 3.2 shows the sample power spectrum for a 1024 point sample. The expected value of the sample power spectrum does not equal the power spectrum. The expected value of Figure 3.2 is the rectangular window power response convolved with the power spectrum [45].

The rectangular window has a rolloff of −6 dB/oct. With more points in the window, the number of octaves in the normalized frequency domain increases. Figure 3.3 shows the sample power spectrum for a 16384 point

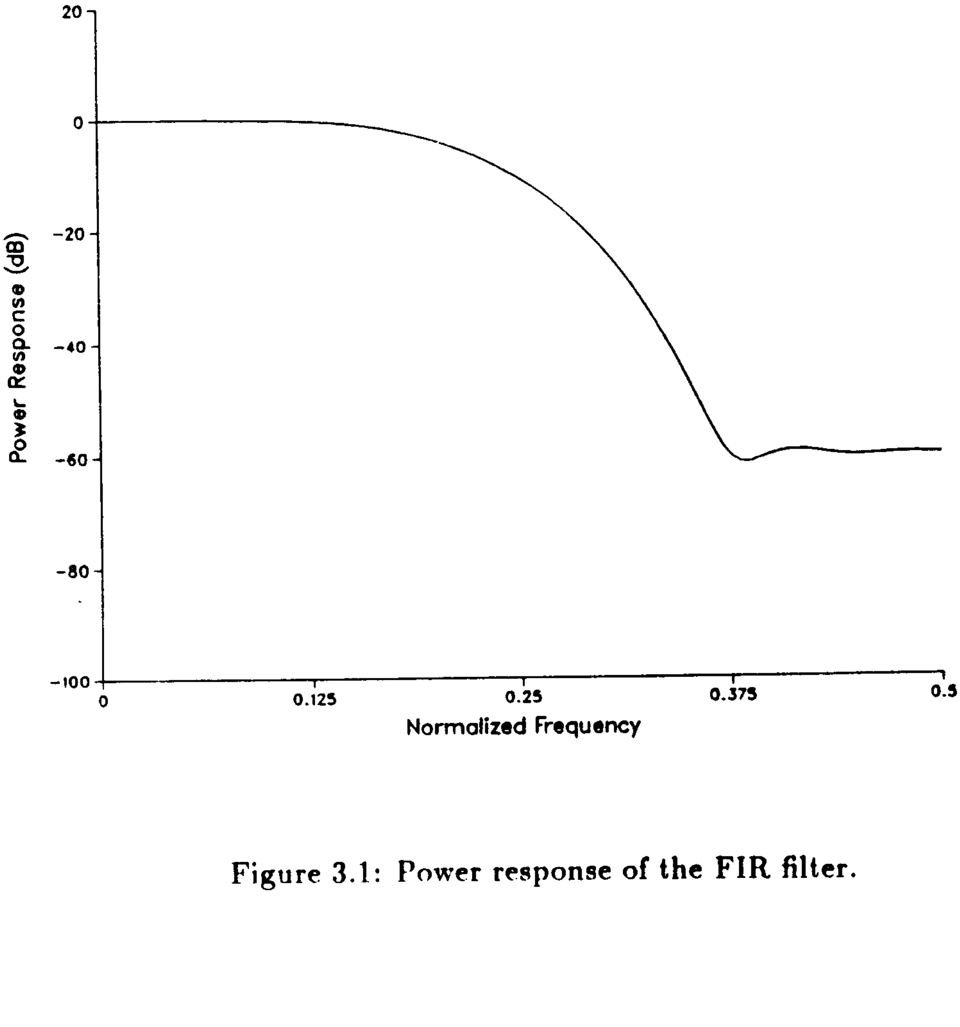

Figure 3.1: Power response of the FIR filter.

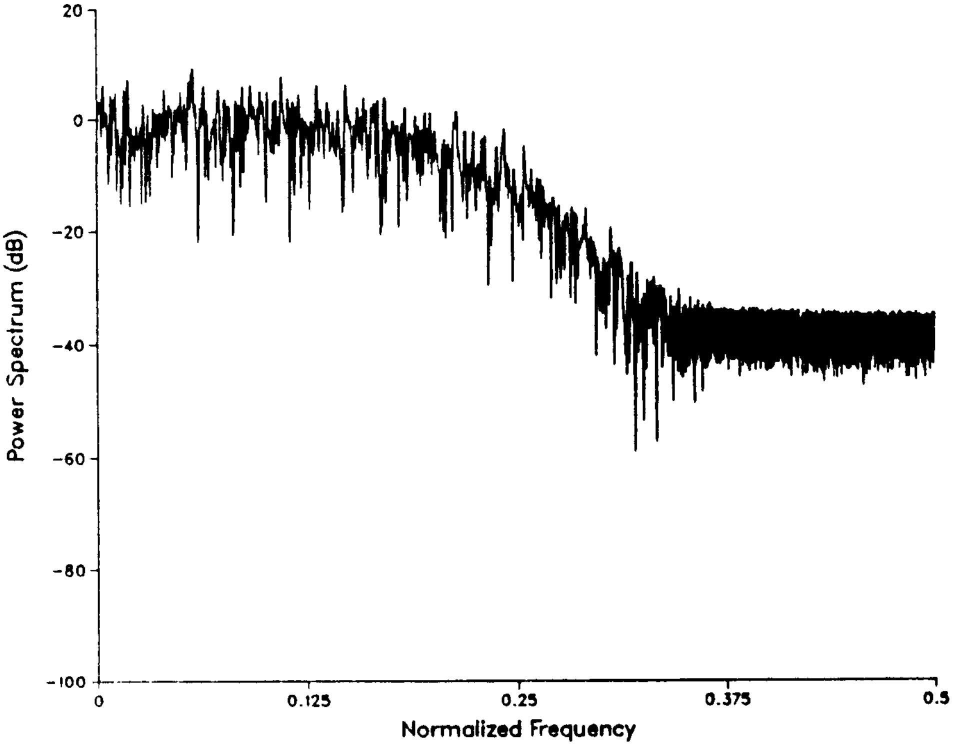

Figure 3.2: Sample power spectrum with 1024 points.

sample. As the number of points in the sample goes to infinity, the expected value of the sample power spectrum will converge to the power spectrum. (This can be seen in [47] by letting the number of points go to infinity in the equation for the expected value of the periodogram.) Unfortunately, the variance does not go to zero. Therefore, the estimate of the power spectrum is not consistent [47].

3.5.1 Averaging Periodograms

The variance can be reduced if the sample size is fixed and consecutive samples are averaged together. This method is called "Averaging Periodograms" (the Bartlett method) [47] [48]. Figure 3.4 shows the 16384 points from Figure 3.3 combined as the average of 16 consecutive 1024 point samples. If the number of samples being averaged goes to infinity, the variance goes to zero, but the expected value does not converge to the power spectrum. Instead, the expected value is the same as in Figure 3.2. For a finite number of points (in this case 16384), selection of the sample size results in a tradeoff between variance reduction (and potentially poor sidelobe performance), Figure 3.4, and frequency resolution, Figure 3.3.

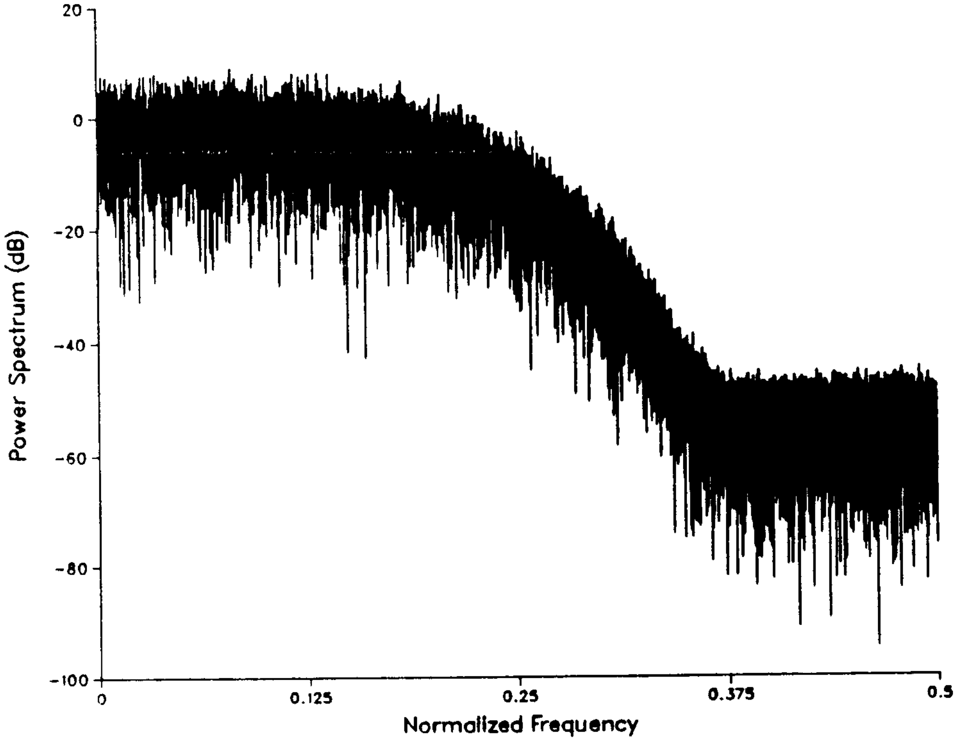

Figure 3.3: Sample power spectrum with 16384 points.

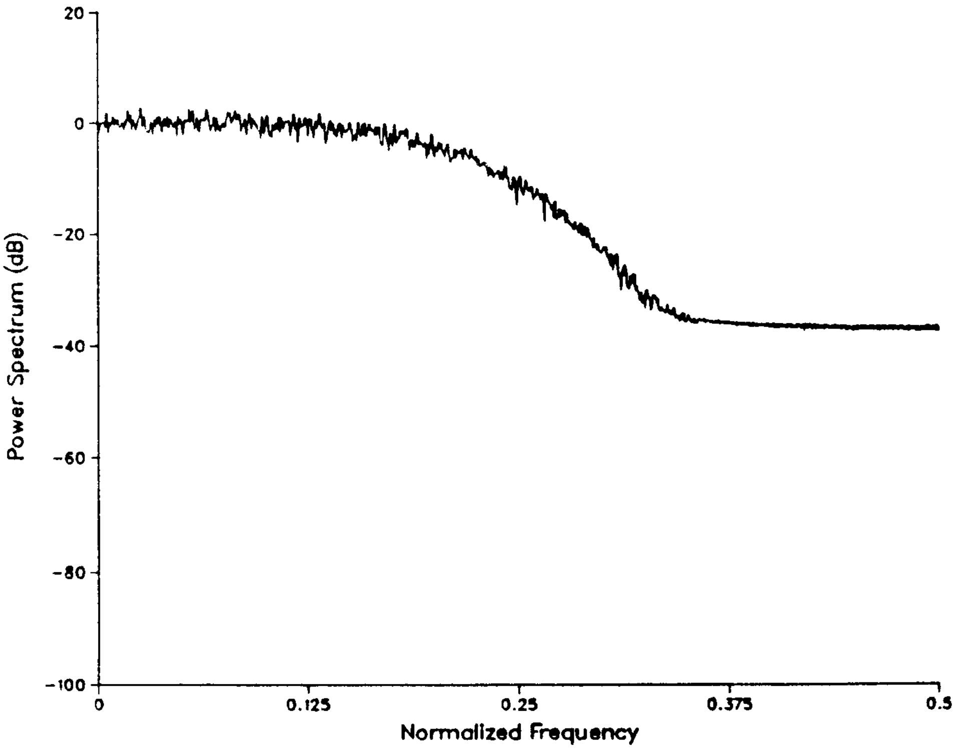

Figure 3.4: Average power spectrum of 16 consecutive 1024 point samples.

3.5.2 Averaging of Modified Periodograms

Since a finite window must always be used for short-time analysis, a window with sidelobes lower than a rectangular window can be chosen. This method is called "Averaging of Modified Periodograms" (the Welch method) [47] [45]. Figure 3.5 shows the same 16384 points as in Figure 3.4, but a minimum 3-term window is used [44]. Now, the expected value of the average (windowed) sample power spectrum (Section 3.4) is the convolution of the power spectrum with the new window [45].

3.6 Spectral Smoothing via Lag Windowing

For a finite number of samples being averaged, the variance is not zero. Further variance reduction can be achieved by spectral smoothing. This also results in a tradeoff between variance reduction and frequency resolution.

Convolution with a filter in the frequency domain can be efficiently computed in the lag domain [46]. The original samples were padded with zeroes so the autocorrelation is not aliased. Spectral smoothing is performed using a lag window.

The window needs to have a non-negative power response [47]. Additionally, the window needs to be equal to one at zero lag so that the mean energy of the spectrum is not biased. A triangle window satisfies these

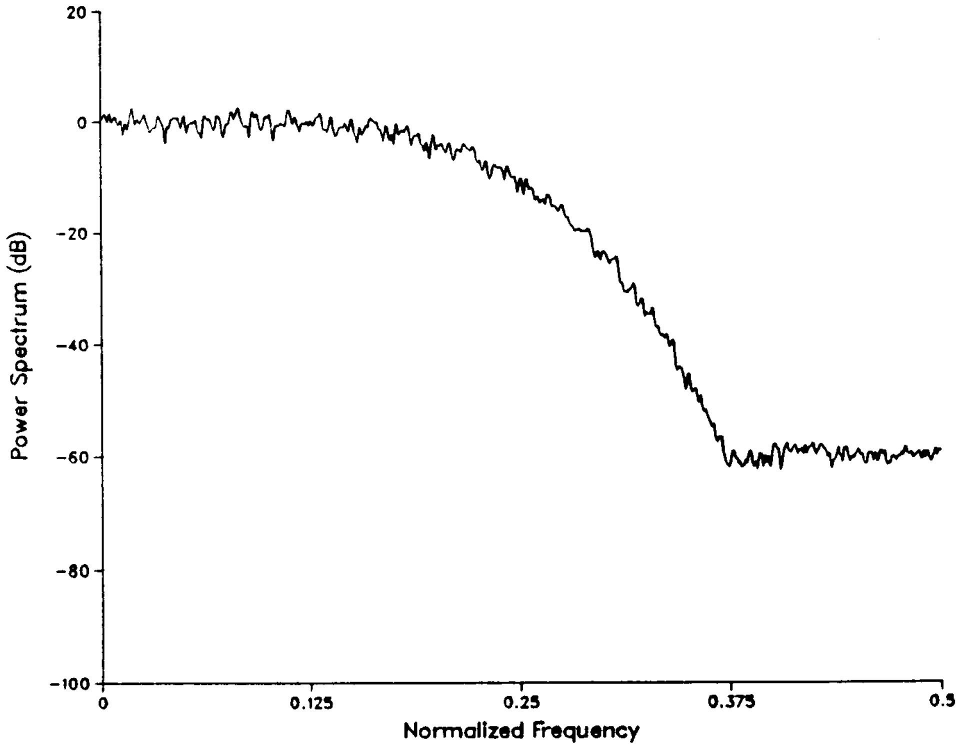

Figure 3.5: Average power spectrum of 16 consecutive 1024 point windowed samples.

criteria [47]. After the IDFT is performed, the following window is applied to $p(l)$.

$$w_c(l) = \begin{cases} \dfrac{M-l}{M} & l = 0, 1, \ldots, M-1 \\ 0 & l = M, \ldots, 2N-M \\ \dfrac{M+l-2N}{M} & l = 2N-M+1, \ldots, 2N-1; M \neq 1 \end{cases}$$

where M determines the amount of smoothing and ranges from 1 to N. The smoothed spectrum is then obtained from the DFT of the windowed lag function.

Figure 3.6 shows a smoothed version of Figure 3.5. The variable M is set to 32. The variance is clearly reduced, yet the sidelobes of the triangular lag window distort the intended spectrum. Other lag windows could be used [41], but another, more effective way of dealing with this problem is discussed in Section 3.10.

3.7 Non-Stationary Noise

Until now, stationary noise has been assumed. If the power spectrum is slowly time varying, the assumption can be made that it is stationary for a short amount of time. This short-time power spectrum can then be estimated from a finite number of samples covering this time period. Choosing the length of time over which the noise is assumed to be stationary results in

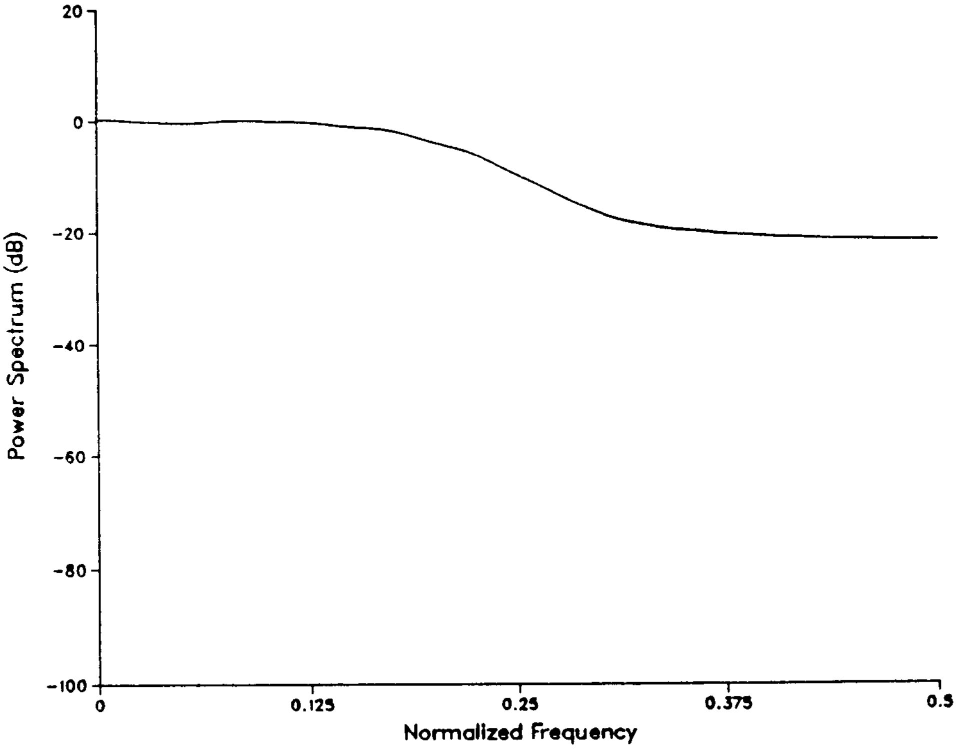

Figure 3.6: Smoothed average power spectrum of 16 consecutive 1024 point windowed samples.

a tradeoff between the convergence rate of the non-stationary power spectrum estimate and the variance of the estimate. Recall that the variance can also be traded for frequency resolution by selection of the sample size (Section 3.5) and the length of the lag window (Section 3.6).

3.8 Practical Implementation

In spectral separation, a single error can cause much damage. For example, in the algorithm described in Chapter 4, 256 samples are averaged at a given frequency. If 255 of the frequency samples are 20 dB (100) and only 1 is 60 dB (1000000), the average value is 36 dB. The averaging of the logarithm of the samples may be a useful method, but the expected value of the logarithm is not equal to the logarithm of the expected value since the logarithmic function is not linear. The next section will extend the method of averaging of modified periodograms to averaging of logarithmic modified periodograms.

3.9 Averaging of Logarithmic Modified Periodograms

The following discussion will show that the logarithm of the expected value of the sample power spectrum, $P(k)$, (Equation 3.4, Section 3.4) is the expected value of the logarithm of the sample power spectrum plus a constant. The proof is based on the fact that for time domain gaussian noise, the power spectrum has a scaled chi-squared density function with two degrees of freedom [41]. Although the proof is only for gaussian noise, a discussion is given in Section 4.3.2 that will generalize this proof for other types of noise.

3.9.1 Sample Power Spectrum Density Function

Any linear transformation of gaussian random variables results in gaussian random variables [49]. A sample may be windowed from an infinite sequence of zero mean gaussian noise. Windowing is a linear transformation. The windowed sample may be padded with zeroes and transformed into the frequency domain using a DFT. Whether or not padding is done, both the real and the imaginary parts of the DFT are a weighted sum of the windowed sample by definition of the DFT. The squared real part is added to the squared imaginary part to result in the squared magnitude of the DFT which therefore has a scaled chi-squared density function [50].

Let $\mathbf{x}$ be defined as a random variable having a chi-squared density function with two degrees of freedom. The random variable, $\mathbf{y}$, representing the sample power spectrum at a given frequency is $\mathbf{y} = \frac{c}{2}\mathbf{x}$, where $\frac{c}{2}$ is chosen as the scaling factor so that $E[\mathbf{y}] = c$. The random variable, $\mathbf{y}$, is described by the following density function [41].

$$f_{\mathbf{y}}(y) = \begin{cases} \frac{1}{c}e^{-y/c} & y \geq 0 \\ \\ 0 & y < 0 \end{cases}$$

3.9.2 Logarithmic Averaging Proof

Log $E[\mathbf{y}]$ is solved in terms of $E[\text{Log } \mathbf{y}]$, but first $E[\text{Log } \mathbf{y}]$ will be solved in terms of Log $E[\mathbf{y}]$. All logarithms are base e unless otherwise specified. "log" is the standard multi-valued complex function. "Log" is the principle value of the "log" function.

$$\begin{aligned} E[\text{Log } \mathbf{y}] &= \int_{-\infty}^{\infty} (\text{Log } y) f_{\mathbf{y}}(y)\, dy \\ &= \frac{1}{c}\int_{0}^{\infty} (\text{Log } y) e^{-y/c}\, dy \end{aligned}$$

Let $r = e^{-y/c}$, so $y = -c \log r$ and $dy = -\frac{c}{r} dr$.

$$\begin{aligned} E[\text{Log } \mathbf{y}] &= \frac{1}{c}\int_{1}^{0} [\, \text{Log } (-c \, \log r) \,]\, r \left(-\frac{c}{r}\right)\, dr \\ &= \int_{0}^{1} [\text{Log } (c) + \text{Log } (-\log r)]\, dr \\ &= \int_{0}^{1} \text{Log } (c)\, dr + \int_{0}^{1} \text{Log } (-\log r)\, dr \end{aligned}$$

Let $s = \log r$, so $e^s = r$ and $e^s ds = dr$.

$$E[\text{Log } \mathbf{y}] = \text{Log } c + \int_{-\infty}^{0} [\text{Log } (-s)]\, e^s\, ds$$

Let $t = -s$, so $-t = s$ and $-dt = ds$.

$$E[\text{Log } \mathbf{y}] = \text{Log } c + \int_0^\infty (\text{Log } t)\, e^{-t}\, dt$$

The solution to the definite integral can be found in a standard table of integrals [51] and is equal to the negative of Euler's constant, γ.

$$E[\text{Log } \mathbf{y}] = \text{Log } E[\mathbf{y}] - \gamma$$

Solving for Log $E[\mathbf{y}]$,

$$\text{Log } E[\mathbf{y}] = E[\text{Log } \mathbf{y}] + \gamma.$$

γ is equal to .577215664901532860606512...[52]. Since a power spectrum is usually presented in decibels, the constant may be adjusted accordingly.

$$\begin{aligned}\frac{\text{Log } E[\mathbf{y}]}{\text{Log } 10} &= \frac{E[\text{Log } \mathbf{y}]}{\text{Log } 10} + \frac{\gamma}{\text{Log } 10} \\ \text{Log}_{10} E[\mathbf{y}] &= E[\text{Log}_{10} \mathbf{y}] + \frac{\gamma}{\text{Log } 10} \\ 10 \, \text{Log}_{10} E[\mathbf{y}] &= E[10 \, \text{Log}_{10} \mathbf{y}] + \frac{10\gamma}{\text{Log } 10}\end{aligned}$$

Let $\zeta = 10\gamma/\text{Log } 10 = 2.506815781\ldots\text{dB}$.

$$10 \, \text{Log}_{10} E[\mathbf{y}] = E[10 \, \text{Log}_{10} \mathbf{y}] + \zeta$$

3.9.3 Logarithmic Averaging Example

Figure 3.7 shows the same 16384 points as used in Figure 3.5 of Section 3.5.2. The only difference is that the logarithmic average is used. Although the results are similar, any error (as discussed in Section 3.8) will not cause as much damage. Comparisons of Welch and logarithmic averaging are made throughout Chapter 4.

3.10 Spectral Smoothing via Cepstral Windowing

This power spectrum estimate can now be smoothed using an IDFT and a lag window as described in Section 3.6. The result will be similar to Figure 3.6 and the poor sidelobe performance still exists. Instead, the logarithm of the power spectrum estimate is inverse transformed. Because of the logarithmic function in the spectral domain, the lag window is now more appropriately called a cepstral [53] window. Figure 3.8 uses the triangular cepstral (lag) window as discussed in Section 3.6.

Since the logarithm is used on the spectrum before smoothing, the cepstral window's power response is essentially convolved with the logarithm of the spectrum instead of the spectrum resulting in excellent sidelobe reduction. Even if the method of "Averaging of Modified Periodograms"

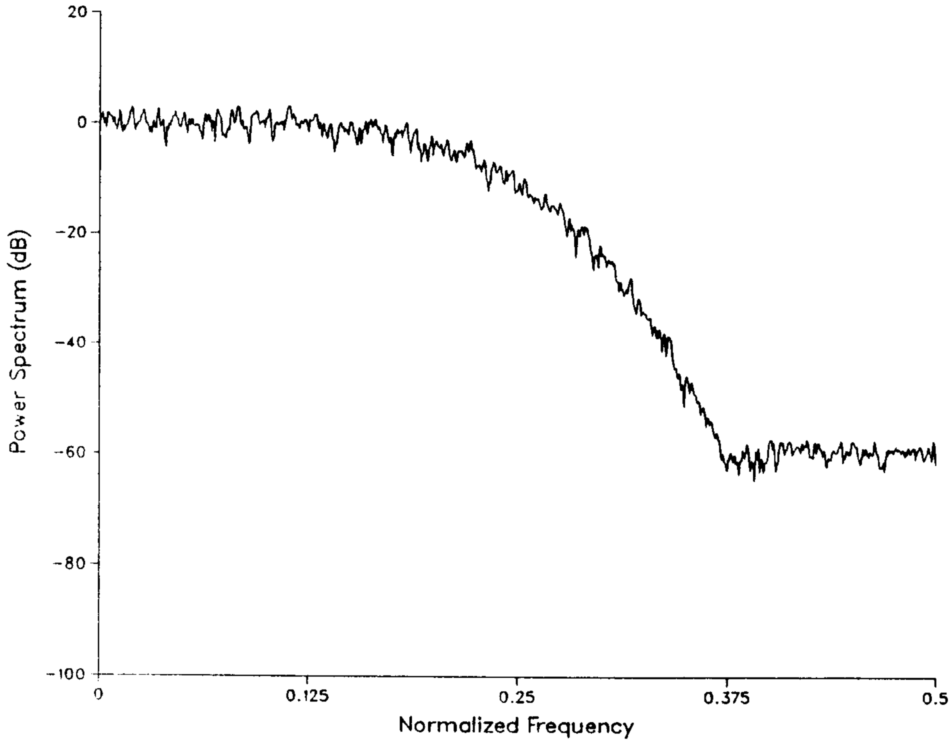

Figure 3.7: Logarithmic average power spectrum of 16 consecutive 1024 point windowed samples.

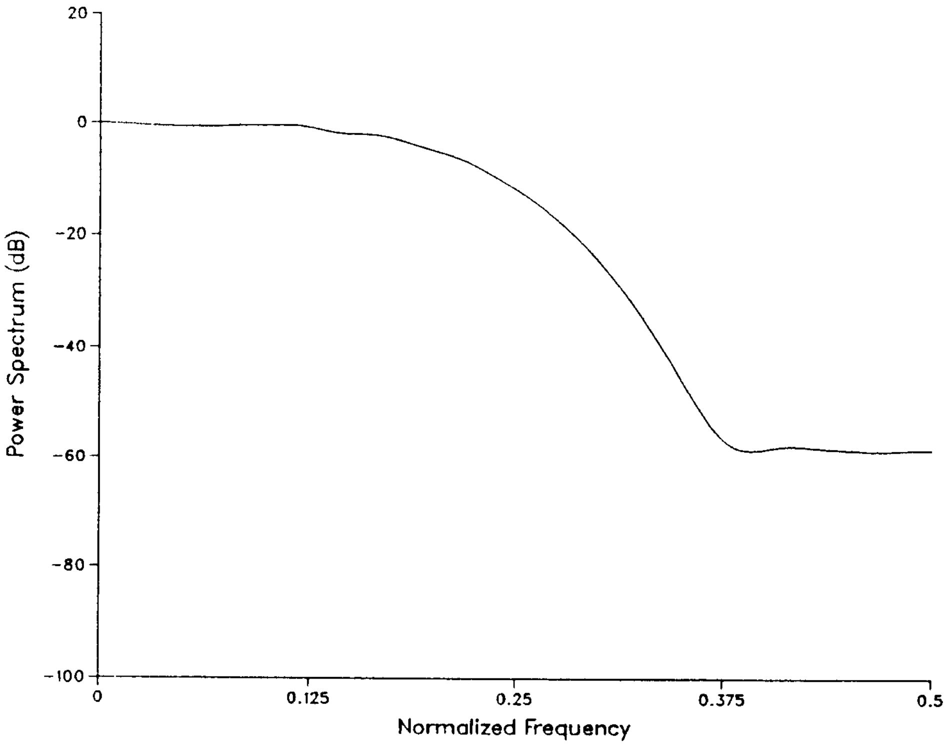

Figure 3.8: Smoothed logarithmic average power spectrum of 16 consecutive 1024 point windowed samples.

(Section 3.5.2) is used prior to smoothing, processing the spectrum with a logarithm before smoothing would still result in excellent sidelobe reduction.

Chapter 4

Estimation of Noise Corrupting Speech Using Extracted Speech Parameters and Averaging of Logarithmic Modified Periodograms

An algorithm is described which provides an estimate of the power spectrum of broadband noise corrupting a wideband speech signal [54]. Estimated parameters of the speech signal (voicing confidence and pitch) are provided by the algorithm in Chapter 2. The noise estimate is derived using both unvoiced and voiced speech and is shown to provide a better estimate than using unvoiced speech alone. The method of averaging of logarithmic modified periodograms (Section 3.9) is used to reduce the effect of speech components corrupting the noise estimate. Results are provided for both Welch (Section 3.5.2) and logarithmic averaging so that comparisons can be made. The results include stationary white noise estimation for only unvoiced speech, for only voiced speech, and for a combination of both. Examples are given for stationary non-white noise and non-stationary white noise using the logarithmic averaging method.

4.1 Introduction

For many speech processing applications [55], [56], [57], it is necessary to have an estimate of the noise level and spectrum corrupting the speech signal. Such an estimate may be obtained from a speech plus noise signal when it is known a priori where non-speech occurs. The assumption is then made that the noise spectrum does not change during the speech.

To eliminate the need for a priori knowledge of the signal, automated speech activity detectors have been developed [58], [3], [59]. Preuss [58] uses a segment based voicing decision. In this method, voiced regions are first located and eliminated. Then, 250 ms before and 500 ms after a voiced region have the potential to be low energy or unvoiced speech and are therefore not used for noise estimation. The remaining signal is assumed to be noise and is used to update the noise estimate. In conversational speech, this method may have difficulty tracking changing noise unless considerable periods of non-speech exist.

Harrison, Lim, and Singer [3] describe the difficulties of detecting speech over a range of background noise levels. They employ an adaptive energy threshold as a speech/no speech detector in a two microphone, noise cancelling system. Those portions of the signal that are classified as no speech are used for noise estimation. It was found that errors in the speech/no speech detector during low energy speech did not cause significant degradation of the noise cancelled speech. In this method, the sampling rate is 10 kHz. As a result, the noise spectrum above 5 kHz is not estimated. The initial estimate of the noise (used for setting thresholds) is determined by assuming the first second of the signal is only noise.

A method to estimate the noise spectrum by making use of a frequency domain voicing decision based on the energy below 1 kHz was investigated by Kang and Fransen [59]. The noise estimate is updated during unvoiced regions. Although they state that unvoiced speech can corrupt the noise estimate, they reason that the total duration of the unvoiced speech relative to silence is such that the corruption is not too adverse. In this method, the sampling rate is 8 kHz. As a result, noise in the frequency range above 4 kHz is not estimated.

Paliwal [60] estimates the level of additive noise during speech using a method that is based on the assumption that the noise is white and that the speech can be modeled as a tenth order autoregressive signal. The method allows tracking of the noise, even during speech, by mathematically separating the two signals based on the properties of the assumed models of noise and speech.

In a related method, Kasuya et al. [61] estimate inter-harmonic energy during voiced speech to differentiate between normal and pathological voices. The speech is modeled as a periodic component and an additive noise component. The method was applied to sustained vowels, after the removal of transitional segments, and was found successful for detecting laryngeal pathologies. While not a method for additive noise estimation, this previous work is an interesting application of the use of inter-harmonic voiced energy.

The method presented in this chapter produces a spectral estimate of additive broadband noise corrupting a wideband speech signal. The method is automatic and requires no a priori knowledge of the noise spectrum. The method uses information regarding the location of speech in the signal. This information is provided by a separate algorithm (Chapter 2). The only time constraint imposed on the signal is that it be no shorter than 51.2 ms (one segment). The noise need not be white, stationary, or gaussian. For smoothing of the noise spectrum, the assumption is made that the noise can be modeled with at most 32 zeroes. This is a parameter that may be adjusted. The speech need not obey a statistical model. The speech is assumed, however, to follow basic phoneme spectral and occurrence statistics reported in the literature.

Section 4.2 describes the parameters extracted from the speech signal (Chapter 2) used to locate speech components in the spectrum. Section 4.3 examines estimating the noise spectrum from unvoiced speech. The sampling rate is 20 kHz and high frequency fricative energy is included in the analysis. A method is included to minimize the corrupting effects of unvoiced speech. Section 4.4 explores the noise information that can be extracted from the voiced spectrum. In Section 4.5, both unvoiced and voiced speech are used to estimate the noise spectrum. White noise results are provided in Section 4.6. In Section 4.7, examples are provided for stationary non-white noise and non-stationary white noise. Concluding remarks are made in Section 4.8.

4.2 Extracted Speech Parameters and Power Spectrum

The entire algorithm for the extraction of the speech parameters is detailed in Chapter 2. A more detailed analysis of the power spectrum is described in Chapter 3. This section is a summary of those results for readers who are interested in only the noise estimation algorithm.

The parameter extraction (Chapter 2) and the noise estimation (this chapter) have been designed as part of a speech enhancement system [8].

The added noise is assumed to be broadband. The method is not intended for impulse or sinusoidal noise.

Two estimated speech parameters are used to identify those frequency values in the power spectrum of the segment where speech energy exists. These parameters are the voicing confidence, $\hat{c}_v$, and the pitch estimate, $\hat{F}_0$. Some factors regarding these parameters are described in the next paragraphs.

The speech to be analyzed is lowpass filtered with a 6 pole Butterworth filter having a cutoff frequency of 8 kHz and sampled with a 10 bit analog-to-digital converter at a rate of 20 kHz. Gaussian noise, computer generated to be white to 10 kHz [33], is added to the sampled speech at the appropriate signal-to-noise ratio (SNR). The SNR is determined as ten times the common logarithm of the mean square level of speech to the mean square level of noise.

The speech plus noise is digitally bandpass filtered. The filter is comprised of a 6 pole Butterworth highpass filter at 20 Hz and a 6 pole Butterworth lowpass filter at 600 Hz. The normalized autocorrelation function (NACF) is calculated for each 51.2 ms, 75 percent overlapped segment. The voicing confidence, $\hat{c}_v$, is derived from the largest value and the rms energy of the NACF in the pitch range (3–20 ms). Therefore, $\hat{c}_v$ reflects the periodicity of the signal and is independent of signal amplitude. $\hat{c}_v$ ranges from

-1 to $+1$ where negative values specify unvoiced segments and positive values specify voiced segments. The magnitude indicates the certainty with which the voicing decision is made. As higher levels of noise are added, the periodicity of the voiced segments diminishes and $\hat{c}_v$ approaches 0. When the periodicity is absent ($\hat{c}_v < 0$), it is difficult to determine if the segment is unvoiced or if it is voiced with much noise. The magnitude does give some indication of the probability of voicing, but the precise probability is dependent on the SNR and the voiced-to-unvoiced ratio of the speech. Whether the segment is actually voiced or not is insignificant since a highly corrupted voiced segment is very useful in estimating the noise.

The pitch estimate, $\hat{F}_0$, specifies where the speech harmonics are located in the power spectrum. A pitch confidence, $\hat{c}_0$, has been defined (Section 2.2.8) to indicate the probabilistic maximum deviation of the actual pitch, F_{0_a}, from the estimated pitch. It has been found that this measure is not particularly useful for the research presented in this chapter. Reasons for this observation are discussed in Section 4.4.

A 51.2 ms analysis window is chosen based upon $\hat{c}_v$ and $\hat{F}_0$ (Section 4.3 and Section 4.4). The window is applied to the speech plus noise signal (before the bandpass filter used in the extraction of $\hat{c}_v$ and $\hat{F}_0$). The resulting 1024 points are padded with 1024 zeroes and the squared magnitude of the DFT is calculated. The power spectrum of the windowed segment (the modified periodogram, $P(k)$) is determined by dividing the squared magnitude of the DFT by the number of points in the window and by the mean square value of the window [45]. Let $s(n)$ be the 1024 point segment of the signal padded with 1024 zeroes and let $w(n)$ be the 1024 point window padded with 1024 zeroes. Therefore,

$$P(k) = \frac{\frac{1}{1024}\left|\sum_{n=0}^{2047} s(n)w(n)e^{-j2\pi nk/2048}\right|^2}{\frac{1}{1024}\sum_{m=0}^{1023} w^2(m)} \qquad k = 0, 1, \ldots, 2047. \quad (4.1)$$

See also Section 3.4.

4.3 Noise Estimation using Unvoiced Speech

Techniques for estimating noise during an unvoiced signal (silence and voiceless speech) have been previously considered [58], [59]. The fricative energy is generally assumed low relative to the corrupting noise [3]. In those cases when it is not low, the relative occurrence is assumed low enough to cause minimal distortion of the overall noise estimate [59]. These methods use band limited speech. The presentation here explores the effects of unvoiced, wideband speech on the noise estimate and suggests a way to minimize its effect.

4.3.1 Averaging of Modified Periodograms

Figure 4.1 is a time domain plot of the utterance "Every salt breeze comes from the sea" spoken by a female. This utterance was chosen as an example because of the fricative /s/ and the vowel /i/. Of all the fricatives, /s/ occurs most often [62], [63] and is one of the most intense for English [64]. /ʃ/ is also very intense, but does not occur as often. /f/ and /θ/ occur less often than /s/ and are also less intense. The vowel /i/ has a high frequency third formant [65] which nearly overlaps with /s/ [66] and therefore must be carefully separated from it. Figure 4.2 shows the same utterance with white noise added to produce a SNR of 6 dB. White noise is used during the discussion of the algorithm but, as will be seen later, the noise need not be white. Figure 4.3 is the voicing confidence, $\hat{c}_v$, resulting from the corrupted speech (Section 2.2.5).

For unvoiced segments ($\hat{c}_v < 0$), a minimum 3-term window [44] is applied to the segment. The modified periodograms of the present and all past unvoiced segments are averaged using the Welch method [45], [67].

$$N_U(k) = \frac{1}{M_U} \sum_i P_i(k) \tag{4.2}$$

where $N_U(k)$ is the power spectrum estimate of the noise at the kth frequency, M_U is the number of unvoiced segments, and the summation is over the periodograms of only unvoiced segments. Finally, the noise estimate,

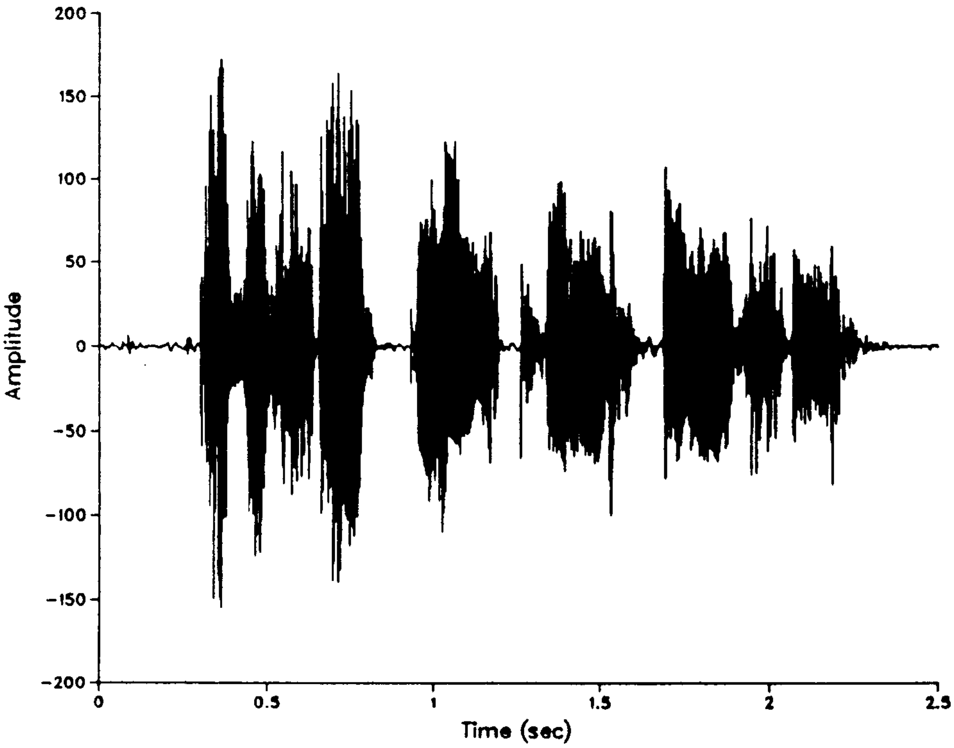

Figure 4.1: The female time domain utterance "Every salt breeze comes from the sea."

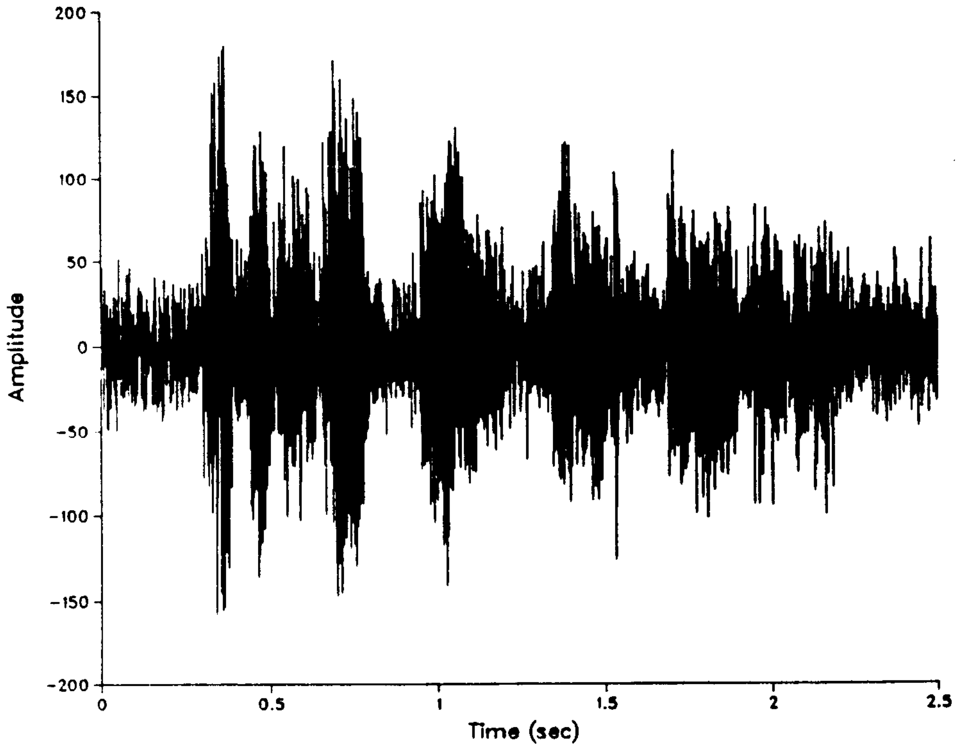

Figure 4.2: The female time domain utterance "Every salt breeze comes from the sea" with white noise added (6 dB SNR).

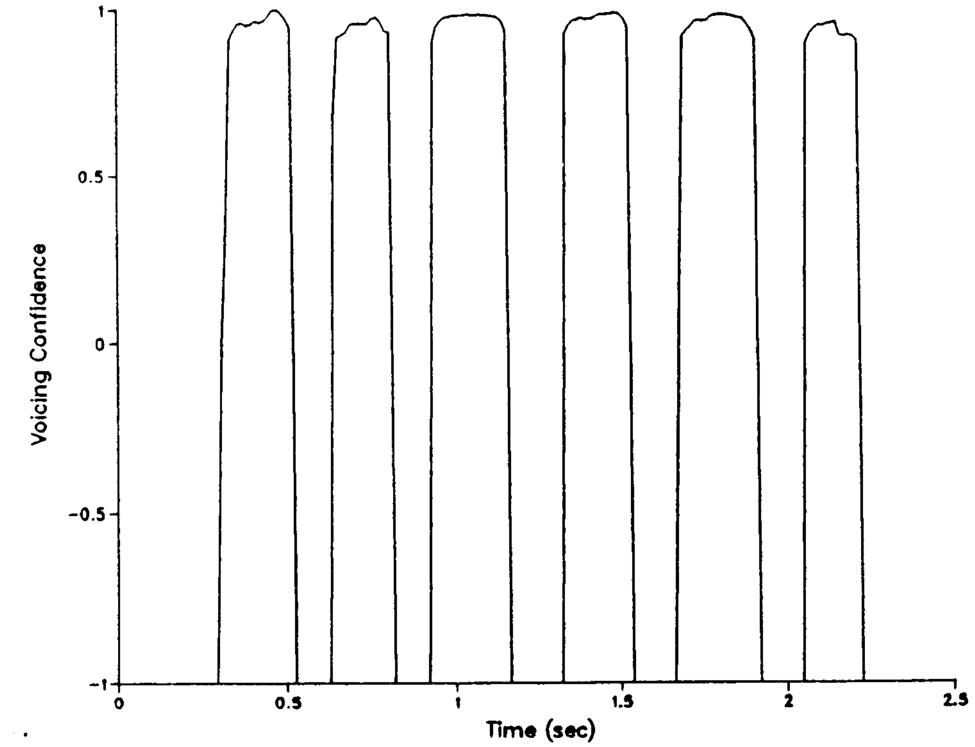

Figure 4.3: The voicing confidence of the female utterance "Every salt breeze comes from the sea" with white noise added (6 dB SNR).

$\hat{N}_U(k)$, is obtained by smoothing $N_U(k)$.

$$\hat{N}_U(k)_{dB} = \mathcal{F}[w_c \mathcal{F}^{-1}[N_U(k)_{dB}]] \tag{4.3}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are the standard 2048 point DFT and IDFT. $N_U(k)$ in dB is inverse transformed to the cepstral domain [6], [53] and a triangular window is applied (Section 3.10). Using the index l for the independent axis of the cepstral domain, the 63 point triangular window is equal to one at $l = 0$ and equal to zero at $l = 32$. The window remains equal to zero until $l = 2048 - 32$. The triangular window then approaches one as l approaches, but does not reach, 2048. This smoothing technique assumes the noise can be modeled with at most 32 zeroes. (The length of the autocorrelation of the impulse response of a 32 zero FIR filter is 63 points.) If it is known that the noise can be modeled with fewer zeroes, a shorter window may be used to reduce the variance of $\hat{N}_U(k)$.

The triangular window was chosen so that the smoothed spectral estimate would be non-negative [47]. The smoothing window is applied in the cepstral domain rather than the lag domain [46] because in the lag domain, its poor sidelobe level (relative to other windows) causes spectral leakage which is unacceptable for estimating a noise spectrum with a large difference in amplitude for various frequencies. It has been shown (Section 3.10) that applying the window in the cepstral domain gives better effective sidelobe performance since the spectrum of the window is essentially convolved with the logarithm of the power spectrum instead of the power spectrum.

The upper curve in Figure 4.4 is the error of $\hat{N}_U(k)$ versus time. The square root of the mean square error in dB was shown to be an informative unit of measure [7].

$$E = \left[\frac{1}{2048}\sum_{k=0}^{2047}\left(\hat{N}_U(k)_{dB} - N_{a\,dB}\right)^2\right]^{\frac{1}{2}} \qquad (4.4)$$

where $N_{a\,dB}$ is the actual power spectrum of the noise in dB. Since white noise is used for this example, N_a is independent of k. The error for each segment is plotted at the time corresponding to the center of the window. (Computation time and storage space may be reduced by transforming only the 63 non-zero points (and one zero) from the cepstral domain in Equation 4.3. The summation in Equation 4.4 would then be over only 64 points.)

The actual noise is defined to be the rms energy requested of the noise generator. For statistical reasons, the exact energy of the noise generated during the unvoiced segments is rarely equal to the requested rms energy. This small difference is ignored for simplicity in calculating the error and does not detract from the discussion which follows.

Since only unvoiced segments are being used, the estimate and therefore the error (using stationary noise) is held constant over the voiced regions.

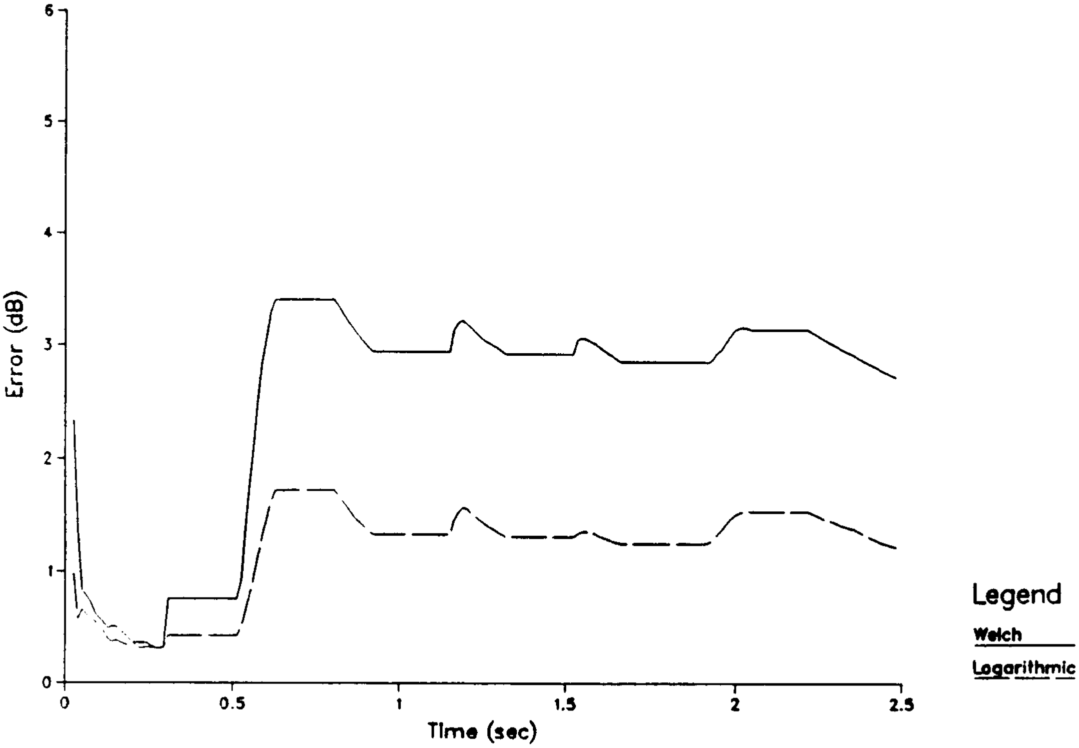

Figure 4.4: The error of the noise estimate of the female utterance "Every salt breeze comes from the sea" with white noise added (6 dB SNR).

The noise estimate converges quickly during the non-speech signal at the beginning of the utterance. At about .5 seconds, the /s/ in "salt" begins to corrupt the noise estimate. Some correction occurs during the stop gap at .8 seconds. The /s/ at the end of "breeze" after the voicing has stopped (about 1.2 sec) and the /s/ in "sea" (about 2 sec) again corrupt the estimate. Finally, the estimate improves during the .25 seconds of non-speech at the end of the utterance.

4.3.2 Averaging of Logarithmic Modified Periodograms

The corrupting effect of the fricatives can be reduced. The method of averaging of modified periodograms has been extended to the method of averaging of logarithmic modified periodograms in Section 3.9. This section shows that those theoretical results have practical advantages over the Welch method in a number of noise cases.

The error plotted as the upper curve in Figure 4.4 is calculated using the present and all past unvoiced segments. As an example, assume that for a given segment, 256 unvoiced present and past segments are used. With these estimates, the Welch method of averaging of modified periodograms (Section 3.5.2, [45], [67]) could be employed, but there is a disadvantage of that method for this application. If 255 of the samples are 20 dB (100) relative to the quantization level and only 1 is 60 dB (1000000), the average value is 36 dB. Similarly, high energy fricatives and doubling errors (Section 4.4) corrupt the noise estimate.

The averaging of the logarithm of the modified periodograms (windowed sample power spectrums) may be a useful method, but the expected value of the logarithm is not equal to the logarithm of the expected value since the logarithmic function is not linear. However, for gaussian noise, the power spectrum has a scaled chi-squared density function with two degrees of freedom [41]. ($P(k)$ at $k = 0$ and at $k = 1024$ has a scaled chi-squared density function with one degree of freedom. Using the approximation of two degrees of freedom is not significant to this algorithm.) It can be shown (Section 3.9.2) that the natural logarithm of the expected value is the expected value of the natural logarithm plus Euler's constant ($\gamma = 0.57721566...$). When adjusted for ten times the common logarithm (decibel) the constant becomes $\zeta = 2.506815781...$ Therefore, the average of the individual power spectrums in dB plus ζ results in an accurate estimate of the power spectrum.

$$N_U(k)_{dB} = \left(\frac{1}{M_U} \sum_i P_i(k)_{dB} \right) + \zeta \qquad (4.5)$$

Compare Equation 4.5 to Equation 4.2. Using Equation 4.3, $\hat{N}_U(k)_{dB}$ is obtained by smoothing $N_U(k)_{dB}$.

If the noise is not time domain gaussian, this method may still prove useful. The central limit theorem states that for independent samples (white noise), a weighted average (properly normalized) of random variables from any density function will approach a normal distribution as more samples are used [68]. Since windowing and Fourier transforming causes a weighted averaging of 1024 time domain samples, the real and imaginary parts of the transform can be approximated by a gaussian distribution (see Section 3.9.1) and the method will work for white noise. Logarithmic averaging has been tested with white, uniform noise with results similar to white, gaussian noise.

If the noise is not white but can be modeled as white noise that has passed through a linear filter, this method may again be used. This is due to the fact that a linear filter can be described by an all zero model using an infinite number of zeroes to model any pole. The output at any sample is a weighted average of the independent input samples and the central limit theorem shows that the output samples approach a gaussian distribution [49]. Therefore, the less white the noise becomes because of more zeroes in the noise model, the more gaussian the output becomes. Whether or not logarithmic averaging will prove to be an improvement over the Welch method will depend on the application.

The lower curve in Figure 4.4 represents the error of $\hat{N}_U(k)$ versus time for the logarithmic averaging method. The noise estimate converges more quickly at the beginning than the Welch averaging method. At .5 seconds, the /s/ still corrupts the estimate, but not nearly as much. The relative improvement with regard to the second /s/ (at about 2 sec) and the convergence rate during the last .25 seconds is not as noticeable, yet the absolute error at the end of the utterance is nearly one-half.

As the SNR increases, the reduction of the fricative's adverse effect on the noise estimate becomes more pronounced. As the SNR is decreased to about −6 dB, the difference between the two averaging methods becomes less significant and the increase in computation time for the new method may not be warranted. The difference is also less pronounced as more past samples are used and there is little difference for speakers that produce quiet fricatives.

At the end of the utterance, all 97 unvoiced segments are used to estimate the noise spectrum (Figure 4.5). The white noise added to the signal has an rms level of 24.88 dB. It can be seen that most of the error occurs at the higher frequencies. This is because of the high frequency fricative /s/. For band limited speech, the noise spectrum can effectively be estimated using only unvoiced speech [59]. For full bandwidth speech, voiced speech should also be considered.

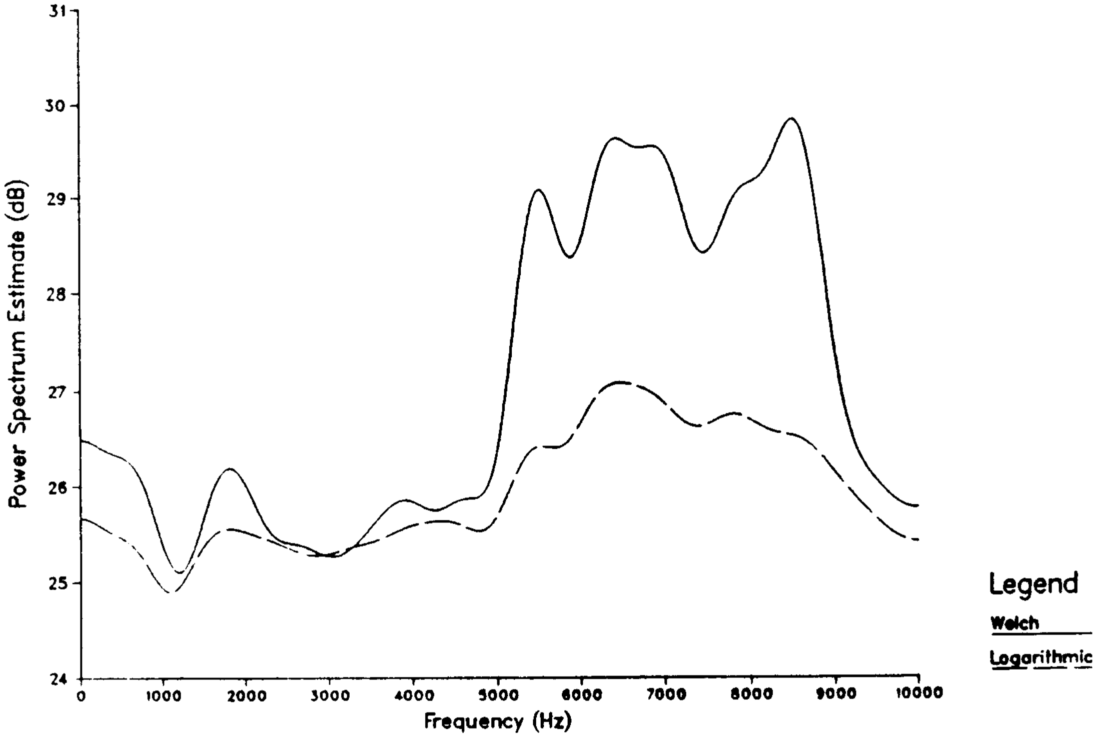

Figure 4.5: Noise spectrum estimate at the end of the utterance using only unvoiced speech (6 dB SNR).

4.4 Noise Estimation using Voiced Speech

In this section, only voiced segments will be used to estimate the noise. The author is aware of only one previous method that has been used to estimate additive non-white noise corrupting running speech using only voiced speech [7]. Because of the relatively slow convergence, this method requires an a priori estimate of the noise or at least some time at the beginning of the signal on which to converge. Similar techniques of using inter-harmonic energy [61] and comb filtering [4] have been used for different applications.

For voiced segments ($\hat{c}_v \geq 0$), the chosen window is based on the pitch estimate, $\hat{F}_0$. For high frequency pitch, the harmonics are separated and the energy is concentrated primarily at the few harmonics in the first formant frequency region. For low frequency pitch, the harmonics are more closely spaced and most of the energy is distributed among many harmonics in the first formant region. Therefore, a minimum 3-term window [44] is chosen when $\hat{F}_0 \geq 175$ Hz and a Hamming window is chosen when $\hat{F}_0 < 175$ Hz. In this way the high energy harmonics of high frequency pitch have extra sidelobe reduction and the low frequency pitch harmonics have a narrower bandwidth so that inter-harmonic spectral space is assured to exist.

Since the segment is voiced, $0 \leq \hat{c}_v < 1$. As described in Section 4.2, $\hat{c}_v$

indicates a level of periodicity. Through experimentation with white noise, it was found that for low levels of periodicity (due to high noise levels), all but the first formant region were corrupted by noise and could therefore be used for noise estimation. As the periodicity (measured by $\hat{c}_v$) increased, the second and then third formant emerged out of the noise and the associated regions could not be used for direct noise estimation. Even for high levels of periodicity, no significant amount of speech energy was found above 6000 Hz. (Voiced fricatives have high frequency energy during voiced speech. This is not a significant factor because voiced fricatives occur relatively infrequently [62], the particular speaker often does not produce a strong fricative component, and the voicing component significantly reduces the air flow as compared to the corresponding fricative [69].)

Based on the results of experimentation, the voiced spectrum is divided into two regions. Recalling that $P(k)$ is an even function, only the first 1025 points (0 to 1024) are described while the remaining 1023 points are found by reflecting the spectrum about the point 1024. The two regions are defined as

$$\begin{array}{llcl} \text{Region 1:} & 0 & - & \text{nint}[614.4\,\hat{c}_v] \\ \text{Region 2:} & \text{nint}[614.4\,\hat{c}_v] & - & 1024, \end{array}$$

where "nint" means nearest integer. In terms of real frequency, the regions are

Region 1: 0 kHz – $6\,\hat{c}_v$ kHz

Region 2: $6\,\hat{c}_v$ kHz – 10 kHz.

In the first region, valid noise estimates are found between the pitch harmonics [61]. The pitch specifies the center frequency for each harmonic. The bandwidth of the harmonic is determined by the chosen window and remains constant for each harmonic. Because of the pitch range (50 to 333 Hz) and the chosen window, the pitch harmonics are never too closely spaced and inter-harmonic spectral energy is assured to exist. All of the points in the second region are considered to be valid noise estimates.

Some initial work [7] motivated research to find a probabilistic maximum deviation of the actual pitch, F_{0_a}, from the estimated pitch, $\hat{F}_0$, defined as the pitch confidence, $\hat{c}_0$ (Section 2.2.8). The bandwidths of the harmonics would then be widened to accommodate the possible errors. This measure was not found to be useful in the present research for the following reasons. First, if the predicted deviation was the true deviation for an estimate, it is not known whether the error is positive or negative so that one-half of the discarded inter-harmonic noise samples are actually valid. Second, the typical deviation is about one third of the predicted maximum deviation. Again, many valid noise samples are discarded. Third, intra-segment pitch variations cause the higher harmonics to become skewed rendering the pitch confidence less than useful for the higher frequencies. (The wide 51.2 ms window is required for spectral resolution.) Fourth, if the pitch deviation is caused by high levels of noise in the first formant region, under white noise conditions, the level is high enough in the second and third formant regions so that the speech has little effect on the noise estimate because the second and third formants are lower in amplitude than the first formant. Without the pitch confidence measure, pitch errors will occur that will not be directly compensated for, but when averaged with the many valid samples (especially using logarithmic averaging), the resulting noise estimate was found to be consistently better.

Figure 4.6 shows the number of estimates available at each frequency for all of the voiced segments at the end of the utterance. The utterance and SNR are the same as in Figure 4.2 and Figure 4.5.

The noise estimate is found by averaging all available estimates for each frequency, k. Since some frequencies have few or no estimates to average, interpolation must be performed. For any frequency that has less than 16 (or the number of segments available, whichever is less) estimates, nearby frequency estimates are added to the averaging. If the number of estimates available at the next highest frequency, $k+1$, plus the number of estimates at the present frequency, k, is greater than or equal to 16, all of these estimates are used. If 16 has not been reached, the next lowest frequency, $k-1$, is added to the averaging. If 16 has not yet been reached, the next

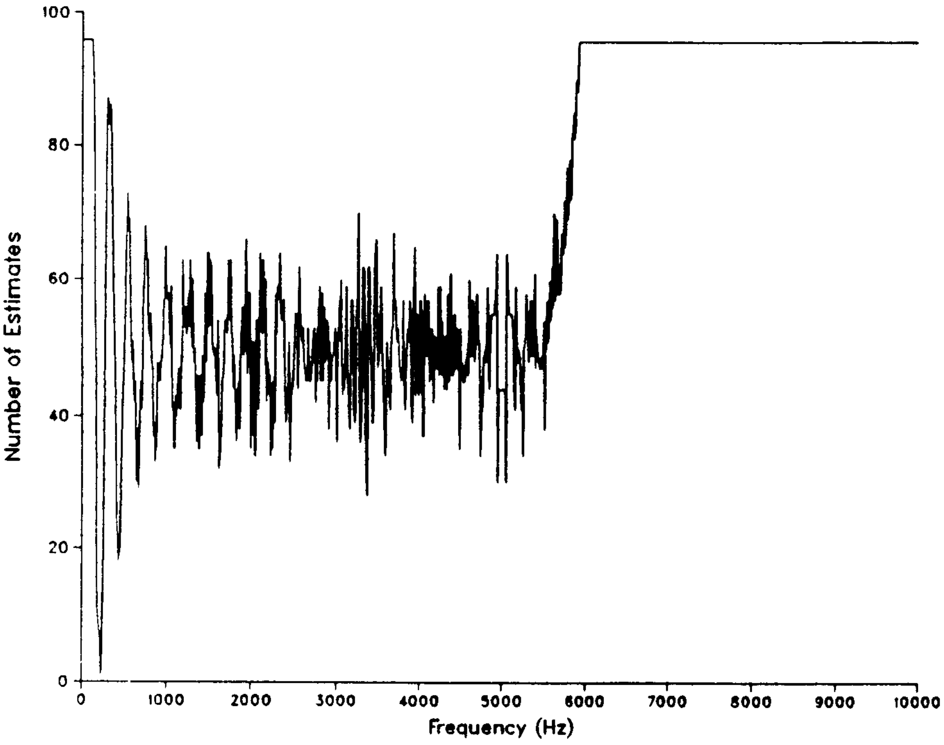

Figure 4.6: Number of estimates available from voiced speech at the end of the utterance (6 dB SNR).

highest frequency, $k+2$, is added to the averaging. This process continues, right then left, until at least 16 estimates are averaged.

After interpolation, the entire noise spectrum, $N_V(k)$, is smoothed using the method described in Section 4.3.1 and $\hat{N}_V(k)$ results. Figure 4.7 shows the error of $\hat{N}_V(k)$ versus time. The error remains constant throughout the unvoiced segments. Since voiced segments are not available at time zero, the first noise estimate is made at about .3 seconds.

If the pitch extraction algorithm makes a gross error ($\frac{|\hat{F}_0 - F_{0_a}|}{\hat{F}_0} > .2$), it is likely to be a doubling error (Section 2.2.6). Such an error will cause every other harmonic to be treated as noise and will degrade the noise estimate in Region 1. Doubling errors are more likely to occur when the level of low frequency noise increases and corrupts the low harmonics. This corruption might not be a serious problem for the low frequency noise estimate, but if the noise level for higher frequencies is lower than the low frequency noise, the noise estimate can be degraded. This problem occurs more often for low pitch speakers.

The voiced estimate can be degraded by doubling errors, by interharmonic speech energy, by high frequency voiced energy, and by sidelobe energy. To reduce these effects, logarithmic averaging can be used. The result is shown in Figure 4.7.

Figure 4.8 shows the noise estimate at the end of the utterance. Most

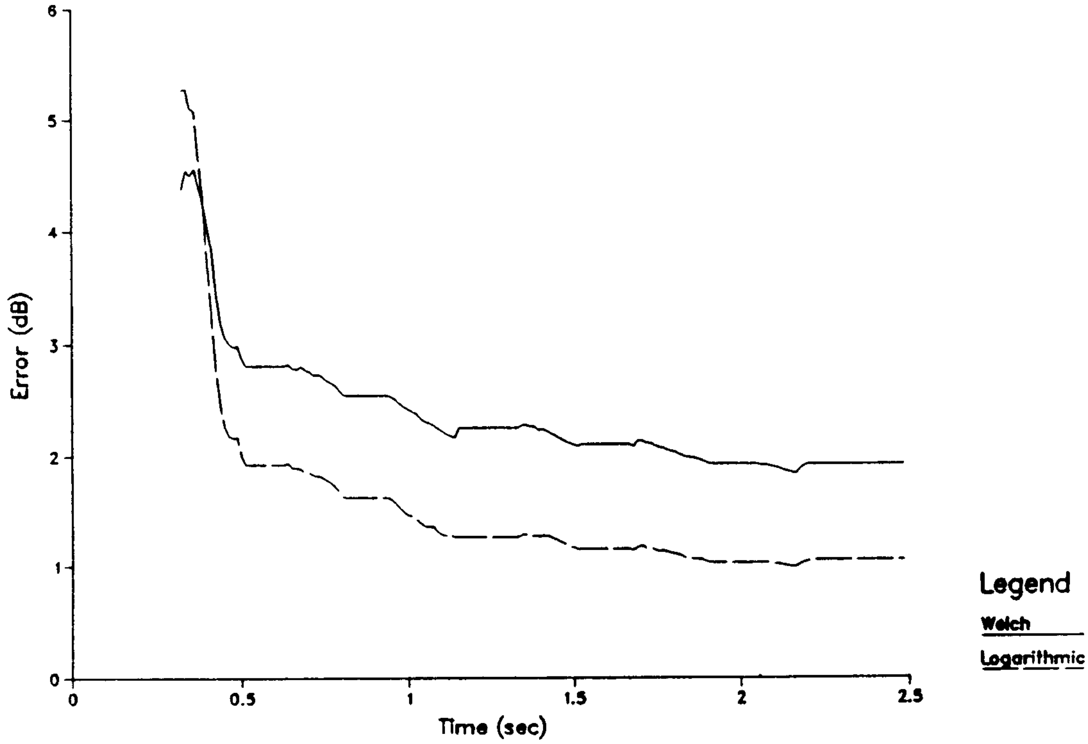

Figure 4.7: The error of the noise estimate versus time using only voiced speech (6 dB SNR).

of the error is caused by the inter-harmonic speech energy that exists in the first three formant frequency regions. This low frequency error is in contrast to the high frequency error that occurs when only unvoiced speech is used (Figure 4.5). As a result, these two methods can be combined to produce a better overall noise spectral estimate.

4.5 Noise Estimation using both Voiced and Unvoiced Speech

The noise estimate derived from only unvoiced segments can exhibit high frequency error due to fricatives. If derived from voiced segments, the estimate can be corrupted by low frequency speech. An improved estimate can be obtained by using the low frequency spectrum of the unvoiced estimate and the high frequency spectrum of the voiced estimate.

Determination of the frequency at which to convert from the unvoiced estimate of the noise to the voiced estimate is based on typical frequency regions of voiced and unvoiced speech. The vowel /i/ for female speech has the highest average third formant (3310 Hz) of all voiced speech (excluding children) [65] and the bandwidth is about 171 Hz [70]. Using this data, the highest vowel frequency is 3396 Hz and is 27 dB below the first formant amplitude [65]. A strong /s/ can be as intense as the first formant and is 30 dB below its maximum at about 3500 Hz [66]. This may suggest an

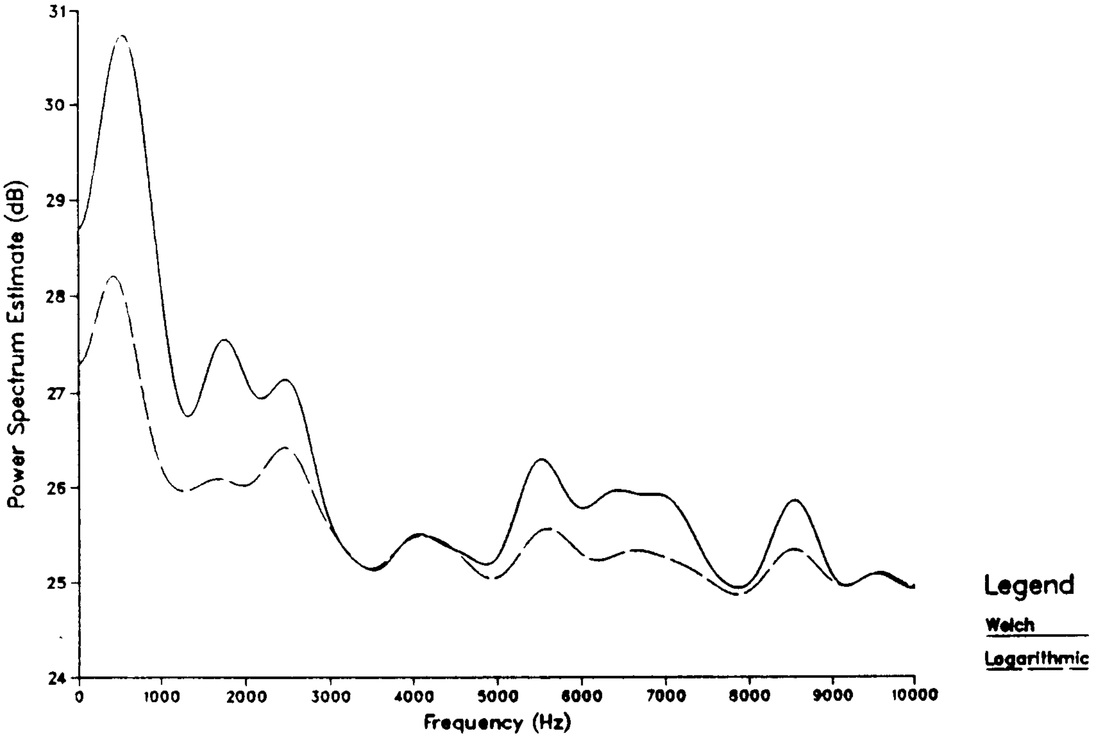

Figure 4.8: Noise spectrum estimate at the end of the utterance using only voiced speech (6 dB SNR).

appropriate transitional frequency of 3400 to 3500 Hz, but quite often /s/ is not as intense as voiced sounds nor does it occur as often. Therefore, 3000 to 6000 Hz is chosen as a transitional region.

$$N(k) = \begin{cases} N_U(k) & k = 0, 1, \ldots, 307 \\ \frac{k-308}{306} N_V(k) + \frac{614-k}{306} N_U(k) & k = 308, \ldots, 614 \\ N_V(k) & k = 615, \ldots, 1024 \end{cases} \tag{4.6}$$

where $N(k)$ is the spectral estimate before smoothing.

Although this produces a good estimate at the end of the utterance, there are some side effects that are not immediately apparent. For example, if only one segment is available, as it would be at the beginning of an utterance, the entire spectrum must be estimated whether the segment is unvoiced or voiced. As another example, assume the first few segments are unvoiced and one voiced segment is received. The high frequencies of the voiced estimate might not be a better estimate than the many unvoiced segments already available because the estimate based on one segment has a large variance. This can cause the error at the transition between unvoiced and voiced speech to suddenly increase.

To reduce these side effects, the final noise estimate is calculated using the following algorithm instead of Equation 4.6.

```
$m = 16$
$d = \min[m, M_U + M_V]$
$w = \min[m, M_U]/d$
for $k = 0$ to $307$
    $N(k) = wN_V(k) + (1 - w)N_U(k)$
next $k$
for $k = 308$ to $614$
    $z = (k - 308)/306$
    $w = (z \min[m, M_V] + (1 - z) \min[m - \min[m, M_U], M_V])/d$
    $N(k) = wN_V(k) + (1 - w)N_U(k)$
next $k$
$w = \min[m, M_V]/d$
for $k = 615$ to $1024$
    $N(k) = wN_V(k) + (1 - w)N_U(k)$
next $k$
```

M_U is the number of unvoiced segments available and M_V is the number of voiced segments available. The algorithm can be considered as favoring the unvoiced estimate in the low frequency range and favoring the voiced estimate in the high frequency range. If at least 16 unvoiced and at least 16 voiced segments are available, the algorithm reduces to Equation 4.6.

The final noise estimate, $\hat{N}(k)$, is obtained by smoothing $N(k)$ using the method described in Section 4.3.1.

Figure 4.9 shows the error of $\hat{N}(k)$ versus time for the estimate based on the present and all past segments. The upper curve is produced using the Welch method and the lower curve is produced using logarithmic averaging. Both methods of averaging are superior to the corresponding method based on unvoiced or voiced segments alone (Figure 4.4 and Figure 4.7).

Figure 4.10 shows the noise estimate at the end of the utterance. It can be seen that the high frequency error in Figure 4.5 and the low frequency error in Figure 4.8 has been reduced for both averaging methods.

4.6 Results for Stationary White Noise

Six variations of the algorithm were run on each of 30 utterances for seven SNRs. The variations include using only unvoiced speech, only voiced speech, and both unvoiced and voiced speech for the Welch and logarithmic averaging methods. A noise estimate is obtained at the end of each utterance and a table of errors is given for each variation.

The database is comprised of 6 speakers, 3 male and 3 female, each speaking 5 sentences. Each utterance consists of the sentence and approximately .25 seconds of silence before and after the sentence. This amount of silence (about 25 percent of the total utterance time) was chosen to be

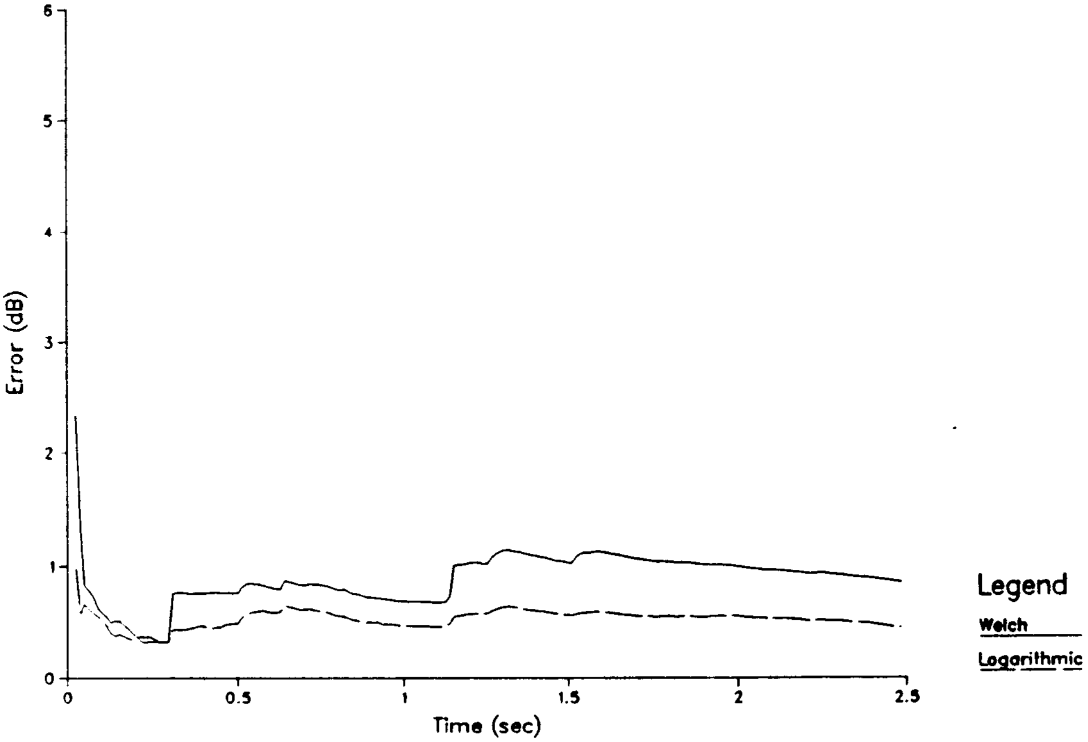

Figure 4.9: The error of the noise estimate versus time using both unvoiced and voiced speech (6 dB SNR).

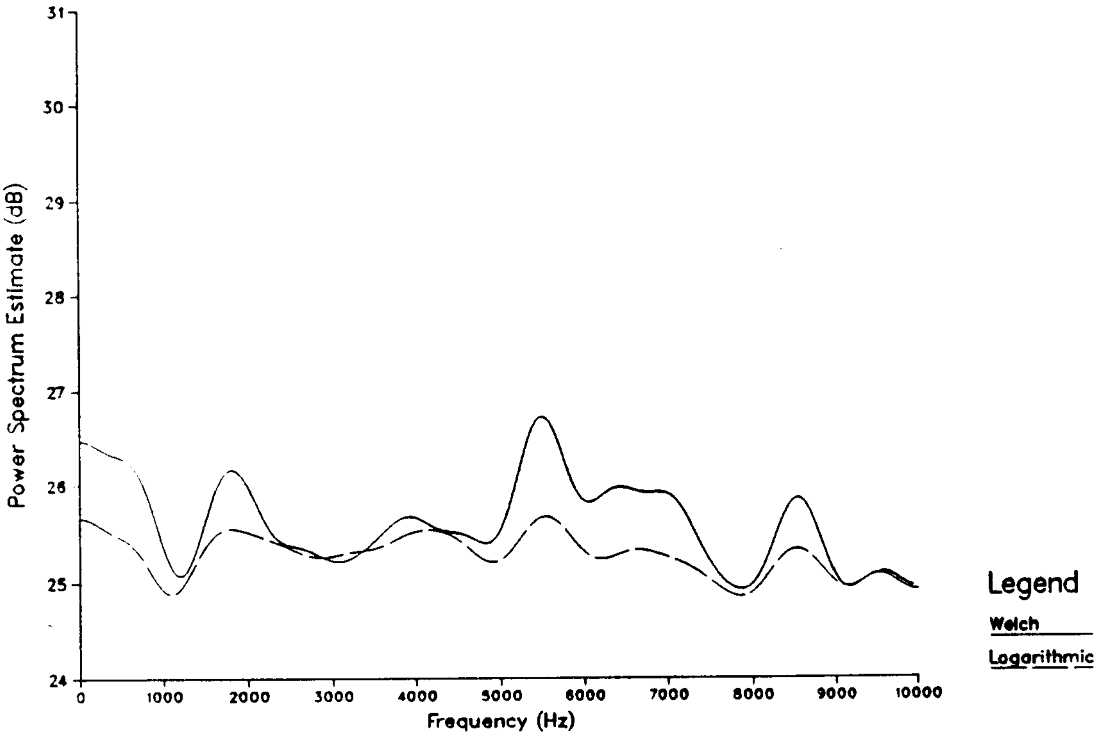

Figure 4.10: Noise spectrum estimate at the end of the utterance using both unvoiced and voiced speech (6 dB SNR).

typical of conversational speech. (The mean square level of speech, used in the SNR calculation, does not include the surrounding silence.) A more complete description of the database is available [22].

The error is calculated over four frequency ranges at each SNR. $\hat{N}(j,k)$ is the power spectrum estimate of the noise of the jth utterance at the kth frequency. Here, a 2048 point DFT is used after the 63 point triangular cepstral window. $N_a(j)$ is the actual power spectrum of the noise of the jth utterance. For all of the tables, white noise is added to the utterances so the actual noise is independent of frequency. E_1, E_2, E_3, and E_4 are the four error measures used in each table.

$$0 - 1\,\mathrm{kHz}: \quad E_1 = \left[\frac{1}{30}\sum_{j=1}^{30}\frac{1}{308}\sum_{k=0}^{307}\left(\hat{N}(j,k)_{dB} - N_a(j)_{dB}\right)^2\right]^{\frac{1}{2}}$$

$$1 - 6\,\mathrm{kHz}: \quad E_2 = \left[\frac{1}{30}\sum_{j=1}^{30}\frac{1}{307}\sum_{k=308}^{614}\left(\hat{N}(j,k)_{dB} - N_a(j)_{dB}\right)^2\right]^{\frac{1}{2}}$$

$$6 - 10\,\mathrm{kHz}: \quad E_3 = \left[\frac{1}{30}\sum_{j=1}^{30}\frac{1}{410}\sum_{k=615}^{1024}\left(\hat{N}(j,k)_{dB} - N_a(j)_{dB}\right)^2\right]^{\frac{1}{2}}$$

$$0 - 10\,\mathrm{kHz}: \quad E_4 = \left[\frac{1}{30}\sum_{j=1}^{30}\frac{1}{2048}\sum_{k=0}^{2047}\left(\hat{N}(j,k)_{dB} - N_a(j)_{dB}\right)^2\right]^{\frac{1}{2}}$$

E_4 is defined so that it is equivalent to Equation 4.4 in Section 4.3.1.

Table 4.1 shows the results of the algorithm using only unvoiced speech

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	2.460	3.978	3.941	3.574
12	1.028	2.059	2.180	1.868
6	0.404	0.913	1.012	0.842
0	0.257	0.359	0.408	0.354
-6	0.234	0.189	0.200	0.208
-12	0.210	0.142	0.150	0.168
$-\infty$	0.114	0.114	0.111	0.113

Table 4.1: Noise Estimation Error Using only Unvoiced Speech and the Welch Averaging Method

and the Welch method of averaging. Although this method of estimating noise is not original, these results for wideband speech are useful for comparison.

The first observation is that higher frequencies have a greater error than lower frequencies for SNRs at 0 dB and above. This is consistent with the discussion in Section 4.3. High frequency fricatives corrupt the noise estimate. For SNRs below 0 dB, the error is similar for all frequency ranges because the noise sufficiently corrupts the fricative sound. For some speakers with stronger than average fricative sounds, the friction still corrupts the noise estimate at SNRs below 0 dB.

The second observation is that the results improve as the SNR is decreased. As the noise is increased relative to the speech, the noise spectrum becomes less corrupted by the speech and is easier to estimate.

The error at $-\infty$ dB is not zero. Although there is no speech, the voicing decision occasionally misclassifies a segment so that the noise estimation algorithm proceeds on false data. For the error to be zero, the voicing decision would need to be correct (always unvoiced). The actual noise would need to be defined as the average power spectrum of the output by the noise generator over the corresponding overlapped segments smoothed by a 63 point triangular cepstral window and the averaging method would depend on the method used in the algorithm (Welch or logarithmic). These definitions of the noise seem cumbersome and artificial. Consequently, defining the actual noise as the rms level requested of the white noise generator causes non-zero errors at $-\infty$ dB even if the voicing decision is always correct.

Table 4.2 shows the results of the algorithm using only unvoiced speech, but now logarithmic averaging is used. The noise is gaussian. Compared with Table 4.1, the results have consistently improved for SNRs above -12 dB. For -12 and $-\infty$ dB, the results are generally the same for the two methods.

Table 4.3 shows the results of the algorithm using only those segments which are classified as voiced. The Welch method of averaging is employed. Lower frequencies have a greater error than higher frequencies for all SNRs. This is due to voiced speech which is more intense at lower frequencies.

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	1.176	1.651	1.426	1.430
12	0.565	0.963	0.874	0.824
6	0.302	0.521	0.502	0.458
0	0.242	0.274	0.295	0.274
-6	0.228	0.182	0.198	0.203
-12	0.212	0.151	0.160	0.175
$-\infty$	0.127	0.121	0.128	0.125

Table 4.2: Noise Estimation Error Using only Unvoiced Speech and the Logarithmic Averaging Method

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	9.025	4.023	0.910	5.442
12	5.071	1.961	0.336	2.985
6	2.348	0.772	0.198	1.359
0	0.885	0.301	0.147	0.520
-6	0.315	0.182	0.149	0.220
-12	0.266	0.206	0.179	0.216
$-\infty$	1.416	1.270	1.058	1.238

Table 4.3: Noise Estimation Error Using only Voiced Speech and the Welch Averaging Method

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	5.915	2.595	0.396	3.545
12	3.327	1.381	0.224	1.977
6	1.617	0.629	0.178	0.956
0	0.703	0.315	0.166	0.435
−6	0.333	0.200	0.166	0.237
−12	0.281	0.196	0.197	0.225
−∞	0.711	0.603	0.614	0.641

Table 4.4: Noise Estimation Error Using only Voiced Speech and the Logarithmic Averaging Method

The results improve as the SNR is decreased which was the case for unvoiced speech. The one exception is at $-\infty$ dB. Although there are no voiced segments, the voicing decision algorithm does make occasional mistakes. The increase in error is due to a large variance because of few segments and due to incorrectly treating unvoiced segments as voiced.

The results of the logarithmic averaging method applied to voiced speech are shown in Table 4.4. Compared with Table 4.3, high SNR errors have been reduced considerably while lower SNR errors are similar. The increase in error at $-\infty$ dB is also exhibited here.

For either method of averaging, noise estimation is generally corrupted at high frequencies for unvoiced speech and at low frequencies for voiced speech. Table 4.5 shows the results for the combined method with Welch averaging. As described earlier, for the combined method the $0 - 3$ kHz

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	2.438	3.013	0.888	2.194
12	1.015	1.435	0.328	0.984
6	0.398	0.598	0.198	0.413
0	0.256	0.242	0.148	0.214
−6	0.234	0.148	0.149	0.178
−12	0.209	0.147	0.179	0.180
$-\infty$	0.115	0.180	0.234	0.189

Table 4.5: Noise Estimation Error Using both Unvoiced and Voiced Speech and the Welch Averaging Method

range is derived from the unvoiced speech, the $6-10$ kHz range is derived from the voiced speech, and the $3-6$ kHz range is derived from a combination of both unvoiced and voiced speech (Section 4.5).

Comparing Table 4.5 to Table 4.1 and Table 4.3, the results have generally improved for all frequency ranges and all SNRs. Even though the $0-3$ kHz range is derived from the unvoiced speech, the error has been reduced because smoothing has increased the error for the unvoiced estimate (Table 4.1) due to the large error in the $3-6$ kHz range. Since the error in the $3-6$ kHz range has been reduced (Table 4.5), smoothing does not degrade the low frequency estimate as much as in Table 4.1. A similar argument is true for the $6-10$ kHz range and the voiced estimate of the noise (Table 4.3). The increase in error due to the voiced estimate at $-\infty$ dB for $3-10$ kHz is apparent.

SNR (dB)	Error 0-3 kHz (dB)	Error 3-6 kHz (dB)	Error 6-10 kHz (dB)	Error 0-10 kHz (dB)
18	1.170	1.449	0.372	1.372
12	0.561	0.778	0.216	0.542
6	0.300	0.398	0.176	0.295
0	0.242	0.219	0.166	0.207
-6	0.228	0.152	0.167	0.183
-12	0.212	0.146	0.197	0.188
$-\infty$	0.127	0.149	0.194	0.163

Table 4.6: Noise Estimation Error Using both Unvoiced and Voiced Speech and the Logarithmic Averaging Method

The results of logarithmic averaging applied to the combined method are shown in Table 4.6. Compared with Table 4.5, the results have clearly improved for high SNRs. For low SNRs, the results are generally the same for the two averaging methods.

In the data used to create Table 4.5 and Table 4.6, an average of about 25 percent of each utterance is silence. The amount of silence will only affect the low frequency results. As the number of non-speech segments are increased, the results will improve for the low frequencies. The high frequency results are dependent only on the voiced segments. If more voiced segments are added, the high frequency results will improve.

4.7 Stationary Non-White and Non-Stationary White Noise Examples

For all of the examples in this section, the noise is gaussian and logarithmic averaging is used with the combined method (unvoiced and voiced).

Several tests of the noise estimation algorithm have been conducted for non-white noise. As an example, consider Figure 4.11 which shows the power spectrum for a typical test case. This noise is generated by passing white gaussian noise with zero mean and unit variance through an FIR filter of length 16 [71] which is less than the maximum of 32 assumed by the smoothing algorithm (Section 4.3). To simplify calculations, the actual noise spectrum is defined to be the filter response. The low frequency noise level is equivalent to -6 dB of white noise and the high frequency noise level has been reduced to 12.88 dB which is equivalent to 18 dB of white noise. This noise was added to the utterance used to generate the results shown in Figure 4.1 through Figure 4.10.

The power spectrum estimate at the end of the utterance is also shown in Figure 4.11. The estimate is about 1 dB greater than the actual power spectrum in the high frequency range. This is due to the fact that for voiced segments the algorithm considers all the energy above 6 kHz to be noise. The low level high frequency noise in this example is easily corrupted by

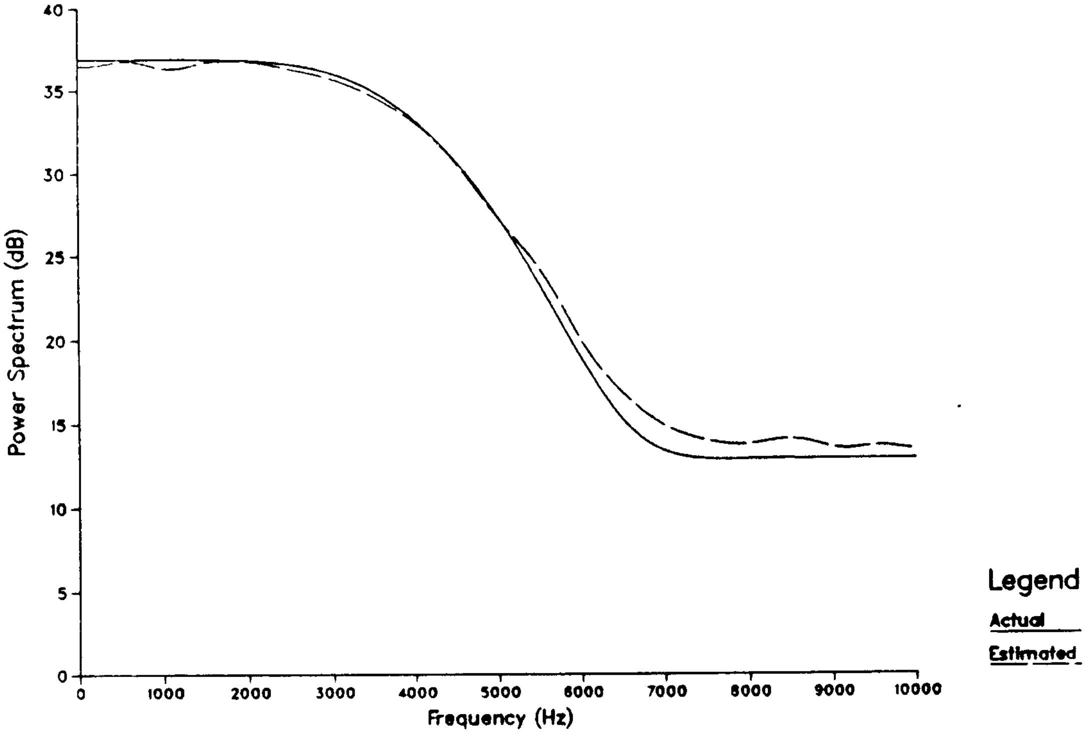

Figure 4.11: Actual and estimated power spectrum of non-white noise.

the speech energy in that range.

In previous sections of this chapter, results and examples are provided for single sentence utterances. In practice, the algorithm is intended to be used for continuous speech. To illustrate how the algorithm will perform for continuous speech, 5 utterances for a male speaker were concatenated to form one long utterance. The male speaker, selected for illustrative purposes, had typical results. The 5 sentences are "We were away a year ago," "Every salt breeze comes from the sea," "Never kill a snake," "Veils of aversion," and "Bread and butter." The time domain plot is shown in Figure 4.12.

In all previous examples, the present and all past segments are used. This method produces the best results for stationary noise. To address non-stationary noise, the assumption is made that the noise is stationary over a finite number of finite number of segments (finite unit of time). The algorithm is then modified to estimate the noise based on the present and past segments up to 16, 64, or 256 segments total. Figure 4.13 shows the error versus time for these three algorithm variations at an SNR of 0 dB (white noise). The lowest curve results from averaging the past 256 segments. In this case, the algorithm begins to discard past segments after 3.2 seconds. The middle curve results from averaging the past 64 segments and begins to discard segments after .8 seconds. The upper curve results

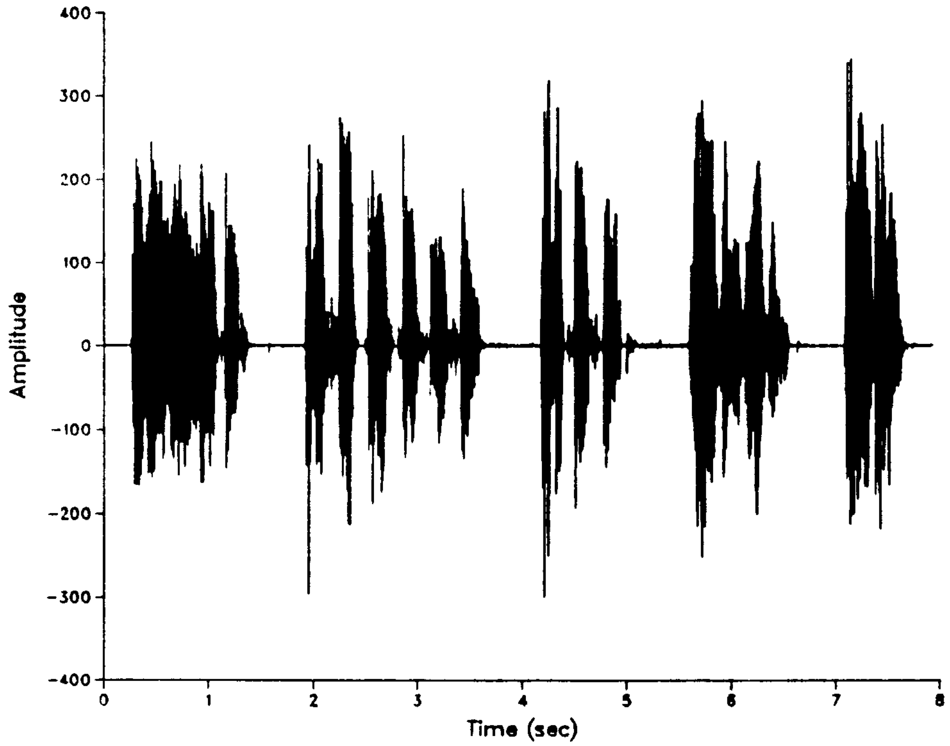

Figure 4.12: The time domain plot of a long utterance formed by concatenating 5 utterances.

from averaging the past 16 segments and begins to discard segments after only .2 seconds.

An algorithm making use of an increased number of past segments produces a smaller error for stationary noise because the variance at each frequency is reduced. Such an algorithm will be unable to track a changing noise spectrum as rapidly as one that uses fewer past segments. Figure 4.14 shows the error versus time starting at an SNR of 0 dB. At 4.2 seconds into the utterance, the noise is increased 12 dB resulting in a SNR of −12 dB. The algorithm using 16 past segments quickly tracks the sudden change while the algorithm using 256 past segments requires more time to adjust. The time 4.2 seconds was chosen so that the algorithm would have to adjust during speech activity.

The error is greater for a sudden change in noise than it is at time zero where the three algorithms use all available segments up to the predetermined maximum for that algorithm. However, for a sudden change in noise level, each algorithm uses past segments that are now invalid.

4.8 Conclusions

A power spectrum estimation algorithm has been described. The algorithm estimates the spectrum of noise corrupting a speech signal. The estimation is performed in the presence of the speech signal and makes use of the voic-

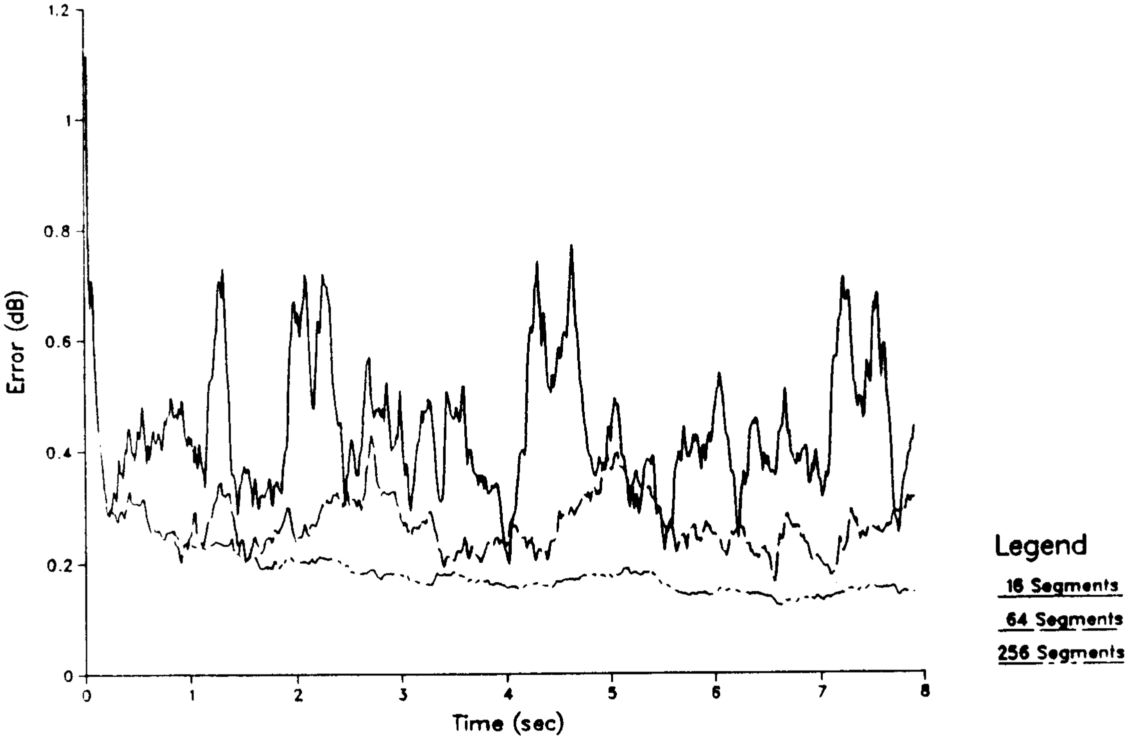

Figure 4.13: The error of the noise estimate for 3 variations of the logarithmic averaging method applied to the long utterance at an SNR of 0 dB.

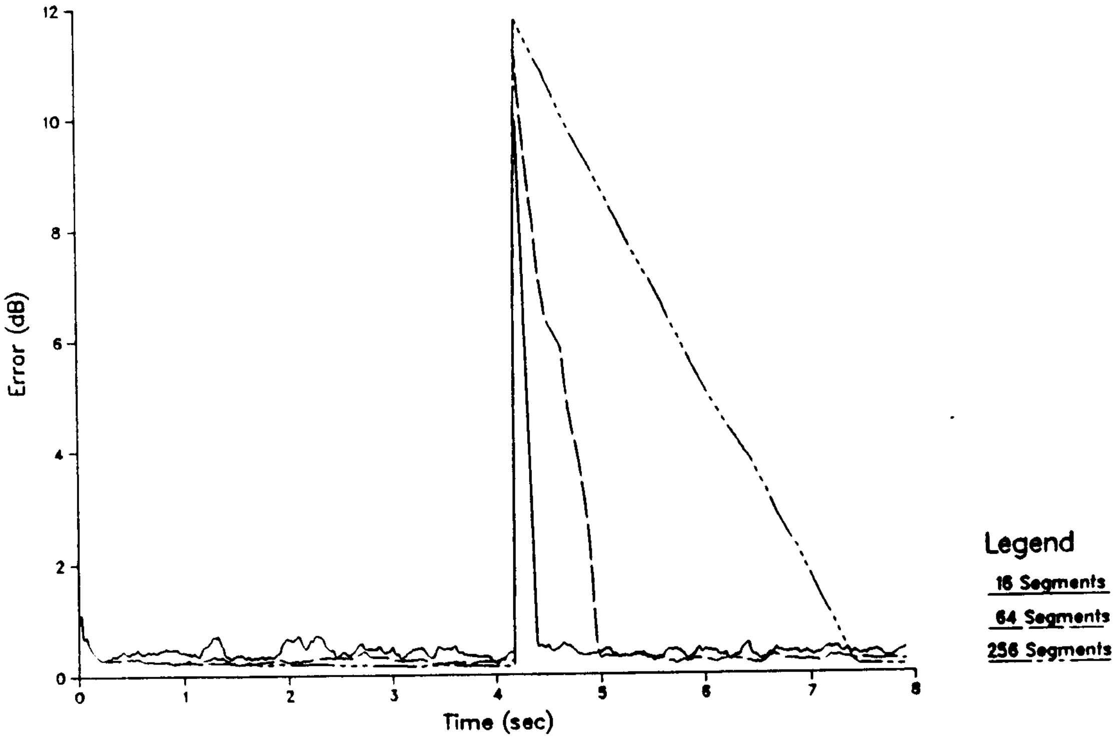

Figure 4.14: The error of the noise estimate for an SNR starting at 0 dB and decreased to −12 dB at t = 4.2 seconds.

ing confidence and pitch which are extracted from the speech plus noise signal. Both unvoiced and voiced speech are used to estimate the noise. This has been found to produce a better broadband estimate of noise corrupting wideband speech than using unvoiced speech alone. Although both types of speech are used, the algorithm is still capable of estimating the noise if only voiced or only unvoiced speech exists.

A logarithmic spectral averaging method has been introduced. This method reduces the effect of speech components corrupting the noise estimate. For noise other than gaussian, this method is an approximation so results have been included for the typical Welch spectral averaging method.

Several parameters in the algorithm may be varied depending upon the particular situation for which the estimated noise is to be used. It has been illustrated that the number of past segments can be reduced to increase tracking time but at the expense of increased variance of the estimated noise spectrum. The length of the cepstral window may be increased to increase frequency resolution or it may be decreased if the noise is known to be nearly white so that the variance of the spectral estimate is decreased. Obviously, there are many tradeoffs in the choice of parameters which can be selected as appropriate to a given situation.

Chapter 5

Summary and Comments for Future Research

This dissertation has presented two new algorithms (Chapters 2 and 4) and a new spectral averaging method (Chapter 3) which have been shown to be useful for the extraction of parameters from a noise corrupted speech signal. The first algorithm makes a voicing decision and produces an estimate of the fundamental frequency (pitch) of speech. Chapter 2 introduces, defines, and develops a confidence measure for each of these parameters. The parameters and confidence measures are extracted and smoothed under the consideration of high levels of noise. The second algorithm produces a spectral estimate of the noise which is corrupting the speech signal. The spectral estimate makes use of the new spectral averaging method designed to reduce the effects of the speech components corrupting the noise estimate.

Unvoiced-to-voiced (U/V) errors remain below a few percent for SNRs down to −18 dB and are reasoned to remain low for even lower SNRs. V/U errors remain below a few percent down to −6 dB, below 20 percent at −12 dB and increase above 50 percent below −18 dB. The pitch (gross) errors are below a few percent down to −12 dB and below 30 percent at −18 dB.

Comparison to previous work is difficult since little quantitative evaluation has been made for pitch determination and voicing decision in cases where speech is corrupted by high levels of additive noise. One paper [22] gives quantitative results for a wide range of SNRs and was found to be comparable to other work for SNRs above 0 dB. Comparing the results of the algorithm in Chapter 2 to the results in [22], the pitch (gross) errors are less than one-half for all SNRs considered. The voicing decision is clearly superior for both V/U and U/V errors for all SNRs except for V/U errors at −18 dB. Because of the different methods used in the two algorithms, the V/U errors in very high levels of noise approach 100 percent for the algorithm in Chapter 2 while the errors in [22] remain around 50 percent.

The voicing confidence measure is found to be successful for indicating the probabilistic accuracy of the voicing decision. The exact probability is dependent on the SNR and the voiced-to-unvoiced ratio of the speech material. Without this information, the voicing confidence can only be used qualitatively. The pitch confidence measure was also found successful, in this case for determining a probable limit on the deviation of the actual pitch from the estimated pitch. Since both of these measures have been introduced in this dissertation, comparison to previous methods is not possible.

Previous work in spectral estimation of noise which is corrupting a speech signal has been referenced throughout Chapter 4. Nearly all of this work has concentrated on speech which is band limited to about 3500 Hz. The results of the popular technique of estimation using only unvoiced speech is included for comparison. It was found that for wideband speech, high frequency fricative energy significantly corrupts the noise estimate above 3500 Hz.

One previous method [7] estimates the noise using only voiced speech and two other methods [3], [60] can estimate noise during voiced speech under a number of restrictions. Chapter 4 has explored the corrupting effect of the energy from voiced speech on the noise estimate. This corruption was found to be significant for low frequencies.

Using a specially designed technique introduced in Chapter 4, the final noise estimate is comprised of the relatively uncorrupted low frequency estimate of the unvoiced speech and the relatively uncorrupted high frequency estimate of the voiced speech. When only voiced or only unvoiced speech is available, the regional corrupting effects still exist. Whether both parts of speech are available or not, the speech, to a greater or lesser degree, corrupts the noise estimate.

To reduce the speech components corrupting the noise estimate, a new spectral averaging method has been introduced. This method has been found successful for significantly reducing the noise estimation corruption for SNRs above 0 dB. The method works well for broadband noise which need not be white, stationary, or gaussian.

These two algorithms are fundamental concepts of the speech enhancement system outlined in Chapter 1. The two remaining blocks in Figure 1.1, "formant estimation" and "speech processing," are presently being researched by the speech and signal processing group at Marquette University.

T. V. Sreenivas [10], [11], [12] is completing a formant estimation algorithm which provides estimates of the first three formants and associated confidences. The algorithm uses the voicing decision produced by the algorithm described in Chapter 2 of this dissertation. The formant estimator presently does not use the noise estimate outlined in Chapter 4 so the assumption of white noise must be made. The pitch estimate, which is also available to the formant estimator (Figure 1.1), is not used either. Further research could be done on the formant estimator to include the noise estimate and to perform pitch synchronous analysis [6].

R. J. Conway [8], [9] is presently working on the "speech processing" block in Figure 1.1. His most resent work uses all of the information available to him. This includes the original speech plus noise signal, the pitch estimate, the pitch confidence, the voicing decision, the voicing confidence, the noise estimate, and the formant estimates with their associated confidences.

Future research in speech intelligibility enhancement could include spectral subtraction techniques improved by the use of extracted speech parameters. Let s be the speech signal, n be the additive noise signal, S and N be the Fourier transforms of s and n, and let $\hat{}$ signify an estimate of one of the variables at a point in time. s and n are functions of time while S and N are functions of time and frequency. s can be approximated by passing $s+n$ through one of the following time varying filters.

$$\hat{H}_1 = \frac{(\widehat{S+N}) - \hat{N}}{(\widehat{S+N})}$$

$$\hat{H}_2 = \frac{\hat{S}_f}{(\widehat{S+N})}$$

$$\hat{H}_3 = \frac{(\widehat{S+N}) - \hat{N}}{\hat{S}_f + \hat{N}}$$

$$\hat{H}_4 = \frac{\hat{S}_f}{\hat{S}_f + \hat{N}}$$

The estimate $(\widehat{S+N})$ may be obtained by applying the spectral estimation techniques discussed in Chapter 3 to the $s+n$ signal. $\hat{S}_f$ is an estimate of the speech spectrum constructed from estimated formant data (for example, see [7]).

Filtering $s+n$ using $\hat{H}_1$ is the standard spectral subtraction technique [57]. It is conceivable that more intelligible speech could result from a more sophisticated filtering operation using an estimate of the speech spectrum which would be based on information important to the intelligibility of speech. Such an estimate, $\hat{S}_f$, could be used in similar filtering techniques resulting in $\hat{H}_2$, $\hat{H}_3$, and $\hat{H}_4$. Since these techniques have not been researched, it is difficult to determine what advantage one might offer over another.

Bibliography

[1] G. J. Borden and K. S. Harris, *Speech Science Primer: Physiology, Acoustics, and Perception of Speech*, 2nd ed., Baltimore, MD: Williams & Wilkins, 1984.

[2] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Englewood Cliffs, NJ: Prentice-Hall, 1978.

[3] W. A. Harrison, J. S. Lim, and E. Singer, "A new application of adaptive noise cancellation," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-34, pp. 21–27, Feb. 1986.

[4] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.

[5] I. Fonagy, "A new method of investigating the perception of prosodic features," *Language and Speech*, vol. 21, pp. 34–49, 1978.

[6] R. W. Schafer and L. R. Rabiner, "System for automatic formant analysis of voiced speech," *J. Acoust. Soc. Am.*, vol. 47, pp. 634-648, Feb. 1970.

[7] C. R. Giordano, ***Obtaining a Spectral Estimate of Time Varying, Broadband Noise Corrupting a Speech Signal***, M.S. Thesis, Marquette University, Milwaukee, WI, 1987.

[8] R. J. Conway and R. J. Niederjohn "Adaptive processing with feature extraction to enhance the intelligibility of noise-corrupted speech," in *Proc. 1987 IEEE Intern. Conf. Indust. Electronics, Control and Instrumentation*, Cambridge, MA, Nov. 1987, pp. 997–1002.

[9] R. J. Conway, ***Intelligibility Enhancement of Noise-Corrupted Speech using Features Extracted from the Noise-Corrvpted Speech Signal***, Doctoral Thesis, Marquette University, Milwaukee, WI, under preparation.

[10] T. V. Sreenivas and R. J. Niederjohn, "Zero-crossing based spectral analysis versus 'SVD' spectral analysis for formant frequency estimation in noise," ***IEEE Trans. Acoust., Speech, and Signal Processing,*** accepted for publication.

[11] T. V. Sreenivas and R. J. Niederjohn, "Growing line segment (GLS) algorithm for determinating formant contours from noisy spectral data,"

IEEE Trans. Acoust., Speech, and Signal Processing, submitted for publication.

[12] T. V. Sreenivas, "Report on formant estimation of noise corrupted speech," Tech. Rep., Speech and Signal Processing Laboratory, Dept. of EECE, Marquette University, Milwaukee, WI, 1990.

[13] D. A. Krubsack and R. J. Niederjohn, "An autocorrelation pitch detector and voicing decision with confidence measures developed for noise-corrupted speech," *IEEE Trans. Acoust., Speech, and Signal Processing*, accepted for publication.

[14] M. M. Sondhi, "New methods of pitch extraction," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 262-266, June 1968.

[15] B. Gold and L. Rabiner, "Parallel processing techniques for estimating pitch periods of speech in the time domain," *J. Acoust. Soc. Am.*, vol. 46, pp. 442-448, Feb. 1969.

[16] M. J. Ross, H. L. Shaffer, A. Cohen, R. Freudberg, and H. Manley, "Average magnitude difference function pitch extractor," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 353-362, Oct. 1974.

[17] N. J. Miller, "Pitch detection by data reduction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 72-79, Feb. 1975.

[18] L. R. Rabiner, "On the use of autocorrelation analysis for pitch detection," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 24-33, Feb. 1977.

[19] A. M. Noll, "Pitch determination of human speech by the harmonic product spectrum, the harmonic sum spectrum, and a maximum likelihood estimate," in *Proc. Symposium Computer Processing in Communications*, Brooklyn, NY, Apr. 1969, pp. 779-797.

[20] S. Seneff, "Real-time harmonic pitch detector," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, pp. 358-365, Aug. 1978.

[21] T. V. Sreenivas and P. V. S. Rao, "Pitch Extraction from corrupted harmonics of the power spectrum," *J. Acoust. Soc. Am.*, vol. 65, pp. 223-228, Jan. 1979.

[22] M. Lahat, R. J. Niederjohn, and D. A. Krubsack, "A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 741-750, June 1987.

[23] A. M. Noll, "Cepstrum pitch determination," *J. Acoust. Soc. Am.*, vol. 47, pp. 293-309, Feb. 1967.

[24] J. D. Markel, "The SIFT algorithm for fundamental frequency estimation," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 367-377, Dec. 1972.

[25] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 399-417, Oct. 1976.

[26] B. S. Atal and L. R. Rabiner, "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 201-212, June 1976.

[27] L. J. Siegel, "A procedure for using pattern classification techniques to obtain a voiced/unvoiced classifier," *IEEE Trans. Acoust. Speech Signal Processing*, vol. ASSP-27, pp. 83-89, Feb. 1979.

[28] C. K. Un and H. H. Lee, "Voiced/unvoiced/silence discrimination of speech by delta modulation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 398-407, Aug. 1980.

[29] L. J. Siegel and A. C. Bessey, "Voiced/unvoiced/mixed excitation classification of speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 451-460, June 1982.

[30] H. Kobatake, "Optimization of voiced/unvoiced decisions in nonstationary noise environments," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 9-18, Jan. 1987.

[31] S. Y. Kwon and A. J. Goldberg, "An enhanced LPC vocoder with no voiced/unvoiced switch," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 851-858, Aug. 1984.

[32] D. A. Krubsack and R. J. Niederjohn, "Comparison of pitch tracking methods for speech in additive white gaussian noise," in *Proc. Midwest Symposium on Circuits and Systems*, pp. 1262-1265, Aug. 1987.

[33] M. E. Muller, "A comparison of methods for generating normal deviates on digital computers," *Association for Computing Machinery*, vol. 6, pp. 376–383, 1959.

[34] D. A. Krubsack and R. J. Niederjohn, "A logarithmic approach to fundamental frequency error measurement in speech," *J. Acoust. Soc. Am.*, pp. 1782-1784, Apr. 1989.

[35] D. L. Thomson, "A multivariate voicing decision rule adapts to noise,

distortion, and spectral shaping," in *Proc. Internat. Conf. on Acoust., Speech, Signal Processing*, pp. 197-200, Apr. 1987.

[36] L. R. Rabiner, M. R. Sambur, and C. E. Schmidt, "Applications of a nonlinear smoothing algorithm to speech processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 552-557, Dec. 1975.

[37] C. M. Rader, "An improved algorithm for high speed autocorrelation with applications to Spectral Estimation," *IEEE Trans. Audio and Electroacoust.*, vol. AU-18, pp. 439-441, Dec. 1970.

[38] R. E. Blahut, *Fast Algorithms for Digital Signal Processing.* Readings, MA: Addison-Wesley, 1985.

[39] D. A. Krubsack, "Short-time analysis of noise models," Tech. Rep., Dept. of ECBE, Marquette University, Milwaukee, WI, 1989.

[40] S. M. Kay and S. L. Marple, Jr., "Spectrum analysis - a modern perspective," *Proc. IEEE*, vol. 69, pp. 1380-1419, Nov. 1981.

[41] M. B. Priestley, *Spectral analysis and time series*, New York, NY: Academic Press, 1981.

[42] J. Makhoul, "Linear prediction: a tutorial review," *Proc. IEEE*, vol. 63, pp. 561-580, Apr. 1975.

[43] G. P. Box and G. M. Jenkins, *Time Series Analysis Forecasting and Control*, Oakland, CA: Holden-Day, 1976.

[44] A. H. Nuttall, "Some windows with very good sidelobe behavior," *IEEE Trans. Acoust. Speech and Signal Processing*, vol. ASSP-29, pp. 84-91, Feb. 1981.

[45] P. D. Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Trans. Audio and Electroacoust.*, vol. AU-15, pp. 70-73, Jun. 1967.

[46] A. H. Nuttall and G. C. Carter, "Spectral estimation using combined time and lag weighting," *Proc. IEEE*, vol. 70, pp. 1115-1125, Sep. 1982.

[47] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1975.

[48] M. S. Bartlett, *An Introduction to Stochastic Processes with Special Reference to Methods and Applications*, New York, NY: Cambridge University Press, 1960.

[49] K. S. Shanmugan and A. M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*, New York, NY: John Wiley & Sons, 1988.

[50] J. E. Freund and R. E. Walpole, ***Mathematical Statistics***, 4th Ed., Englewood Cliffs, NJ: Prentice-Hall, 1987.

[51] W. H. Beyer, ***CRC Standard Mathematical Tables***, Boca Raton, FL: CRC Press, 1981.

[52] M. Abramowitz and I. A. Stegun, ed., ***Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables***, Washington, D.C.: U.S. Government Printing Office, 1964.

[53] A. M. Noll, "Short-time spectrum and "cepstrum" Techniques for Vocal-Pitch Detection," *J. Acoust. Soc. Am.*, vol. 36, pp. 296-302, Feb. 1964.

[54] D. A. Krubsack and R. J. Niederjohn, "Estimation of Noise Corrupting Speech Using Extracted Speech Parameters and Averaging of Logarithmic Modified Periodograms," *IEEE Trans. Acoust., Speech, and Signal Processing*, submitted for publication.

[55] M. S. Ahmed, "Comparison of noisy speech enhancement algorithms in terms of LPC perturbation," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-37, pp. 121–125, Jan. 1989.

[56] R. J. Niederjohn, P. Lee, and F. Josse, "Factors related to spectral subtraction for speech in noise enhancement," in *Proc. 1987 IEEE Intern.*

Conf. Indust. Electronics, Control and Instrumentation, Cambridge, MA, Nov. 1987, pp. 985–996.

[57] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-27, pp. 113–120, Apr. 1979.

[58] R. D. Pruess, "A frequency domain noise cancelling preprocessor for narrowband speech communications systems," in *Proc. International Conference Acoust., Speech, and Signal Processing*, Washington, D.C., Apr. 1979, pp. 212–215.

[59] G. S. Kang and L. J. Fransen, "Quality improvement of LPC-processed noisy speech by using spectral subtraction,"*IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-37, pp. 939–942, June 1989.

[60] K. K. Paliwal, "Estimation of noise variance from the noisy AR signal and its application in speech enhancement," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-36, pp. 292–294, Feb. 1988.

[61] H. Kasuya, S. Ogawa, K. Mashima, and S. Ebihara, "Normalized noise energy as an acoustic measure to evaluate pathologic voice," *J. Acoust. Soc. Am.*, vol. 80, pp. 1329–1334, Nov. 1986.

[62] P. B. Denes, "On the statistics of spoken English," *J. Acoust. Soc. Am.*, vol. 35, pp. 892–904, June 1963.

[63] W. S-Y. Wang and J. Crawford, "Frequency studies of English consonants," *Language and Speech*, vol. 3, pp. 131–138, 1960.

[64] P. Strevens, "Spectra of fricative noise in human speech," *Language and Speech*, vol. 3, pp. 32–49, 1960.

[65] G. E. Peterson and H. L. Barney, "Control methods used in a study of the vowels," *J. Acoust. Soc. Am.*, vol. 24, pp. 175–184, Mar. 1952.

[66] M. F. Schwartz, "Identification of speaker sex from isolated, voiceless fricatives," *J. Acoust. Soc. Am.*, vol. 43, pp. 1178–1179, May 1968.

[67] G. C. Carter and A. H. Nuttall, "On the weighted overlapped segment averaging method for power spectral estimation," *Proc. IEEE*, vol. 68, pp. 1352–1354, Oct. 1980.

[68] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, New York, NY: McGraw-Hill Book Company, 1984.

[69] C. Scully, "A comparison of /s/ and /z/ for an English speaker," *Language and Speech*, vol. 14, Apr.-Jun. 1971.

[70] H. K. Dunn, "Methods of measuring vowel formant bandwidths," *J. Acoust. Soc. Am.*, vol. 33, pp. 1737–1746, Dec. 1961.

[71] D. A. Krubsack, "DFT-even linear-phase FIR filters," Tech. Rep., Dept. of ECBE, Marquette University, Milwaukee, WI, 1988.

This dissertation has been approved by the following committee:

______________________________ Chairperson

____

____

____

____