

APPLICATION OF POLYSCALE METHODS FOR SPEAKER VERIFICATION

by

Sina Sedigh

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Canada

Copyright © 2018 by Sina Sedigh

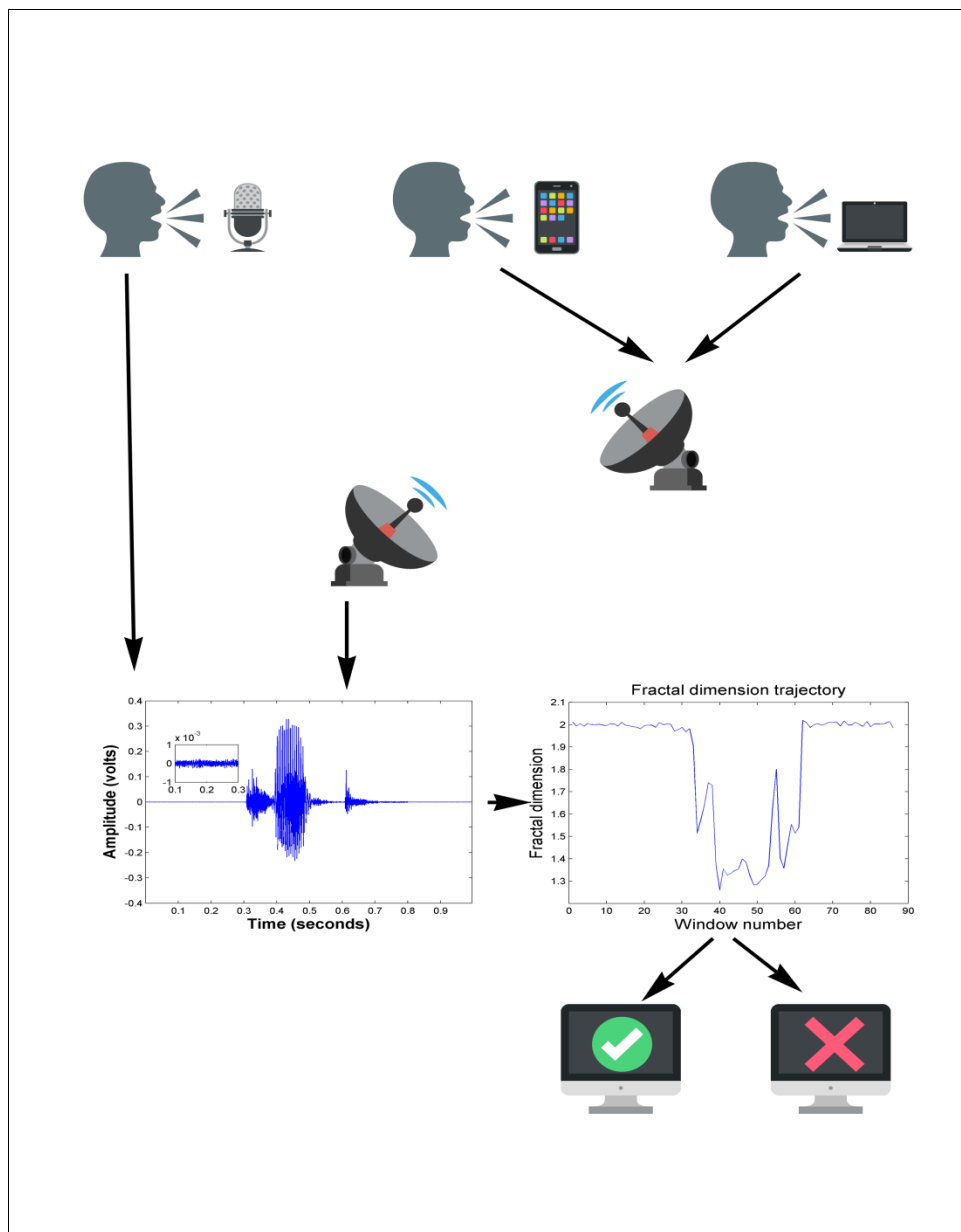
Abstract

Voice is a characteristic of the human body which is unique to an individual. Voice can be used for remote access applications, in order to verify the individual's identity. However, robust feature extraction is required and the aim of this research is the establishment of security via the speaker's voice.

All the experiments in this thesis are based on a dataset recorded in an anechoic chamber, available at the Applied Electromagnetic Laboratory at the University of Manitoba. The following dataset consists of utterances, recorded using 24 volunteers raised in the Province of Manitoba, Canada. To provide a repeatable set of test words that would cover all of the phonemes, the Edinburg Machine Readable Phonetic Alphabet [KiGr08], consisting of 44 words was used. The utterances were recorded using a sampling frequency of 44.1 kilo-samples per second (kSps). The recording sessions took place between 10 AM to 3 PM, from March 27, 2017, until September 27, 2017.

This thesis presents a study of text-independent speaker verification with the aim of experimental evaluation of features and embedding fractal algorithms to the front-end processing of the speaker verification system. A voice activity detection based on the variance fractal dimension was used to separate the non-speech segments of the signal. A fusion of multiple features, namely the linear prediction cepstral coefficients, Mel-frequency cepstral coefficients, Higuchi fractal dimension, variance fractal dimension, zero crossing rate, and turns count, was used to form the feature vectors. Meanwhile, an experimental sensitivity analysis was conducted to test the effects of each feature on the accuracy of classification using a support vector machine. The features were extracted using multiple voice activity detection algorithms. The best across-the-divide recognition accuracy of 91.60% was obtained by fusion of all the features that were extracted using the voice activity detection algorithm based on the variance fractal dimension. This shows that fusion of features and embedding of fractal methods to the front-end processing of text-independent speaker verification will increase the accuracy of the classifications.

Visual Abstract



Acknowledgements

This thesis would not have been possible without the help and support of a network of people. I would like to take this opportunity to express my humble gratitude towards them who helped me.

At first, I would like to thank Professor Witold Kinsner, my thesis advisor, for being a mentor, a teacher, and an educator throughout this degree. His knowledge, vision, and work ethics enriched my life in so many ways.

I would like to thank Professor Joe LoVetri for granting us access to the anechoic chamber available at the Applied Electromagnetic Laboratory at the University of Manitoba, where the recordings of the dataset were conducted. Moreover, I would like to thank Ms. Maryam Ghanbari and Ms.Elnaz Afsharipour who were present during the recording sessions. I would also like to thank all the volunteers who put their time and participated in this study.

Thirdly, I would like to express my gratitude to my friends and colleagues, Mohamed Nasri, Ramin Soltanzadeh, and Eden Katz, for their help, advice, and suggestions whenever I needed.

Finally, but certainly not the least, I would like to thank my parents and sister for their love, encouragement, and patience throughout my life which helped me most in my life abroad.

Table of Contents

Abstract	i
Visual Abstract.....	ii
Acknowledgements.....	iii
Table of Contents	iv
List of Figures	viii
List of Tables	xi
List of Acronyms	xiii
List of Symbols	xv
1 Introduction.....	1
1.1 Problem Statement	1
1.1.1 Motivation.....	2
1.1.2 Definition	2
1.1.3 Thesis Aim	3
1.1.4 Approaches to Achieving the Aim.....	3
1.1.5 Proposed Solution	5
1.2 Thesis Formulation	7
1.2.1 Thesis Statement	7
1.2.2 Thesis Objective.....	7
1.2.3 Research Question	8

1.3 Thesis Organization	10
2 Literature Review on Speaker Verification.....	11
2.1 The Physiology of Speech	11
2.1.1 The Process of Speech Production.....	12
2.1.2 The Mechanics of Speech Perception (the Ear).....	13
2.1.3 The Auditory Pathway and the Brain.....	15
2.2 The Phonetics of Speech.....	15
2.2.1 Initiation.....	16
2.2.2 Phonation	17
2.2.3 Articulation of English Phonemes	17
2.3 Fundamental Speech Analysis Techniques.....	21
2.3.1 Discrete Fourier Transform.....	21
2.3.2 Root Mean Square.....	22
2.3.3 Fractal Dimension	23
2.3.4 Voice Activity Detection	24
2.4 Summary	24
3 Theoretical Background.....	26
3.1 Colored Noise	26
3.2 Framing of Speech	28
3.3 Zero Crossing Rate	28
3.4 Turns Count	29
3.5 Mel-Frequency Cepstral Coefficients.....	31
3.6 Linear Prediction Cepstral Coefficients.....	32
3.7 Higuchi Fractal Dimension.....	33
3.8 Variance Fractal Dimension	34
3.9 Test Data	36
3.9.1 Weierstrass Function.....	36
3.9.2 Fractional Brownian Motion.....	37
3.10 Support Vector Machine	39

3.11 Summary	42
4 A Manitoban Speech Dataset	44
4.1 Available Databases	45
4.2 Microphone	46
4.2.1 Directionality	47
4.2.2 Frequency Response	48
4.2.3 Sensitivity	48
4.2.4 Blue Yeti	48
4.3 Software	50
4.3.1 Camtasia	50
4.3.2 Audacity	51
4.3.3 Decibel X	51
4.4 English Phonemes	52
4.5 Demographics	53
4.6 Environment	55
4.7 Recording Protocols	56
4.8 Repository	58
4.9 Summary	60
5 Design of Experiment and Result Analysis	62
5.1 Comparison of Fractal Dimension Estimation Algorithm	62
5.1.1 Weierstrass function	63
5.1.2 Fractional Brownian motion	68
5.2 Effects of Noise on Fractal Dimension Estimation	76
5.2.1 Weierstrass function	76
5.2.2 Fractional Brownian motion	82
5.3 Voice Activity Detection	86
5.3.1 Characteristics of the Background Noise	87
5.3.2 Fractal Dimension of Colored Noise	89
5.3.3 The Voice Activity Detection Algorithm	91

5.4 Feature Extraction	95
5.5 Classification of Features.....	98
5.5.1 Building of the training model and the classification of the test data.....	98
5.5.2 Comparison of the results with different voice activity detection methods.....	100
5.6 Summary	103
6 Conclusions.....	105
6.1 Thesis Overview	105
6.2 Thesis Conclusions	107
6.3 Thesis Contribution.....	109
6.4 Limitations and Future Work.....	111
Reference	113
Appendix A Voice Activity Detection.....	A1
Appendix B Software Flowcharts.....	B1
Appendix C Experimental Codes and Data.....	C1
Appendix D Colophon	D1

List of Figures

Fig. 2.1. A speech production model. (After [Lang92])	13
Fig. 2.2. The three sections of the ear. (From [Micr18])	14
Fig. 2.3. The auditory pathway. (From [FiGM07])	16
Fig. 2.4. The power spectral density of a frame containing 512 samples of speech.....	22
Fig. 2.5. The normalized root mean square of a speech signal	23
Fig. 3.1. The normalized zero crossing rate of a speech signal.	29
Fig. 3.2. The normalized turns count of a speech signal.....	30
Fig. 3.3. The Higuchi fractal dimension trajectory.....	34
Fig. 3.4. Weierstrass function generated using $H = 0.5$ and $\lambda = 2$	37
Fig. 3.5. The fractional Brownian motion generated using $H = 0.5$	38
Fig. 3.6. The separating hyperplane obtained by setting $C=0.1$	41
Fig. 3.7. The separating hyperplane obtained by setting $C=100$	42
Fig. 4.1 The blue Yeti.	49
Fig. 4.2 The frequency response of the Yeti in cardioid directionality mode. (After [Blue16]) ..	50
Fig. 4.3 The utterance test recorded using Camtasia.	51

Fig. 4.4. The utterance “church” trimmed to a 2 sec epoch using Audacity.	52
Fig. 4.5. A screenshot of Decibel X.....	53
Fig. 4.6. The Histogram of the participants.	54
Fig. 4.7. The interior of the anechoic chamber used.....	55
Fig. 4.8. The Blue Yeti in a side-address position on the left. (After [Blue17])	56
Fig. 4.9. The Blue Yeti in cardioid mode and with gain level of -9 dB.....	57
Fig. 4.10. The input level in Camatsia is set to 80%.	58
Fig. 4.11. The setup of the Blue Yeti with a pop-shield filter placed in front of it.	59
Fig. 5.1. Graph of fractal dimension values of 512 samples of the Weierstrass function.	64
Fig. 5.2. The Weierstrass function generated using Hurst value of 0.4.....	66
Fig. 5.3. The variance fractal dimension trajectory of the Weierstrass function.	67
Fig. 5.4. The Higuchi fractal dimension trajectory of the Weierstrass function.....	67
Fig. 5.5. The variance fractal dimension of the fractional Brownian motion.....	70
Fig. 5.6. The Higuchi fractal dimension of the fractional Brownian motion.....	71
Fig. 5.7. The fractional Brownian motion generated using $H = 0.5$ and the seed set to 10.....	74
Fig. 5.8. The variance fractal dimension trajectory of the fractional Brownian motion.....	75
Fig. 5.9. The Higuchi fractal dimension trajectory of the fractional Brownian motion.	75
Fig. 5.10. 172 cycles of the Weierstrass function generated using $H = 0.7$	79
Fig. 5.11. The fractional Brownian motion generated by assigning $H = 0.5$	84
Fig. 5.12. The background noise seen in the pre-silence.	88
Fig. 5.13. The power spectrum density of the noise.	88
Fig. 5.14. The variance fractal dimension trajectory of white, pink, and brown noise.	89
Fig. 5.15. The variance fractal dimension trajectory of the pre-silence.	90

Fig 5.16. The Waveform of the utterance “church”.....	92
Fig. 5.17. The variance fractal dimension of the utterance “church”.	93
Fig. 5.18. The variance fractal dimension after addition of white noise.	93
Fig. 5.19. The extracted frames containing speech.....	94
Fig. 5.20. The waveform of the utterance detected by the voice activity detection algorithm.	95
Fig. 5.21. The graphical representation of feature vector “A”.....	97

List of Tables

Table 2.1: The English Phonemes and the corresponding keyword (From [Grie96]).	18
Table 2.2: The English Consonants. (From (Grie96)).	20
Table 5.1: Fractal dimension estimation of 512 samples of Weierstrass function.	63
Table 5.2: Fractal dimension estimation of 88200 samples of the Weierstrass function	65
Table 5.3: The variance fractal dimension estimation of the fractional Brownian motion.	68
Table 5.4: The Higuchi fractal dimension estimation of the fractional Brownian motion.	69
Table 5.5: The variance fractal dimension trajectory of the fractional Brownian motion.	72
Table 5.6: The Higuchi fractal dimension trajectory of the fractional Brownian motion.	73
Table 5.7: The estimated Fractal Dimension of the Weierstrass functions after addition of -40 dB of colored noise.	77
Fig. 5.8: The estimated Fractal Dimension of the Weierstrass functions after addition of -30 dB of colored noise.	78
Table 5.9: The mean of the trajectories of the fractal dimension of the Weierstrass function by addition of -40 dB of colored noise.	80
Table 5.10: The mean of the trajectories of the fractal dimension of the Weierstrass function by addition of -30 dB of colored noise.	81
Table 5.11: The fractal dimension of the fractional Brownian motion by addition of -40 dB of colored noise.	82

Table 5.12: The fractal dimension of the fractional Brownian motion by addition of -30 dB of colored noise.	83
Table 5.13: The mean of the trajectory obtained by addition of -40 dB of colored noise to the fractional Brownian motion.	85
Table 5.14: The mean of the trajectory obtained by addition of -30 dB of colored noise to the fractional Brownian motion.	86
Table 5.15: The keywords used for training and testing the classifier.	99
Table 5.16: The accuracy of the classifications using variance fractal dimension.	100
Table 5.17: The accuracy of the classifications using Higuchi fractal dimension.	101
Table 5.18: The accuracy of classifications using amplitude threshold.	102
Table 5.19: The accuracy of the classifications using the energy.	102

List of Acronyms

VAD	Voice activity detection
FD	Fractal dimension
LPCC	Linear prediction cepstral coefficients
MFCC	Mel-frequency cepstral coefficients
HFD	Higuchi fractal dimension
VFD	Variance fractal dimension
ZCR	Zero crossing rate
TC	Turns count
SVM	Support vector machine
MRPA	Machine readable phonetic alphabet
DFT	Discrete Fourier transform
FFT	Fast Fourier transform
PSD	Power spectrum density
RMS	Root mean square
SNR	Signal to noise ratio
IPA	International phonetic alphabet

DOI	Digital object identifier
WAV	Waveform audio file format
DCT	Discrete cosine transform
LPC	Linear predictive coding
IFFT	Inverse fast Fourier transform
H	Hurst exponent
VFDT	Variance fractal dimension trajectory
fBm	Fractional Brownian motion
C	Cost function

List of Symbols

$X(f)$	Spectrum
x	Signal
N	Number of samples
V_{rms}	Root mean square
f	Frequency of signal
β	Power spectrum exponent
y	Generated colored noise
L_i	Level of additive colored noise
dB	Decibels
$\text{sgn}(\bullet)$	Sign function
$w[\bullet]$	Window function
tr_i	Turn at a time interval
S_k	Log-spectral coefficients
K	Length of log-spectral coefficients

L	Number of coefficients required
a_k	Linear predictive coding coefficients
s^*	Predicted signal
m	Initial time
k	Interval time
Lm_k	Length of each sample of the time series
$L(k)$	Length of time series
n	Time index
δt	Time displacement
Δt	Time increment
n_k	Vel size
$n_k \max$	Largest vel size
D_σ	Variance fractal dimension
E	Number of independent variables
α	Roughness parameter
γ	Circulant matrix
Z	Complex value vector
$K(\cdot, \cdot)$	Sums of a kernel function
t_i	Ideal outputs
x_i	Support vectors
ξ	Slack function

Chapter 1

Introduction

Current advancements in technology have resulted in our dependency on machines while accomplishing daily tasks. However, breaches of highly sensitive data have raised awareness that the information that can be accessed online is not safe. Some of these breaches are by human interaction and hence have resulted in the scientific community searching for alternative methods to minimize the risk of unauthorized access to personal data. One of the fields that have gained considerable attention is biometrics. Biometrics involves the use of physical characteristics of the human body that are unique to that person, such as fingerprints, iris, and voice, in order to verify that person's identity.

1.1 Problem Statement

To the listener, the speech signal carries many levels of information. While the speech transfers a message using words, it also contains information about gender, emotion, language,

and generally the identity of the speaker [Reyn02]. This section discusses the motivation behind the use of *speaker recognition* and the general categories and tasks associated with it, followed by the discussion of the aim of this work and the proposed solution.

1.1.1 Motivation

With the growing number of services accessed via telephone, web or mobile apps, maintaining and remembering multiple passwords, PIN's, and authentication details required to gain access to accounts remotely has become more challenging. Especially since security experts encourage the use of different authentications for different accounts.

Meanwhile, with the currently existing infrastructure speaker identity is a biometric that can be easily tested for remote access applications [Beig11]. This makes speaker recognition valuable for many real-world applications.

1.1.2 Definition

Speaker recognition involves the identification of the speaker based on the words they speak and it can be divided into two categories, *text-dependent*, and *text-independent*. Text-dependent requires the speaker to say the same word that was used for feature extraction, whereas text-independent can identify the speaker regardless of the words mentioned [ChLu09]. Text-dependent speaker recognition has prior knowledge of the text to be spoken [Reyn02]. Text-independent speaker recognition relies on the physiological characteristics of the speaker and does not make any assumption about the context of the speech [Beig11].

Speaker recognition can be divided into two general tasks, namely, *speaker identification* and *speaker verification* [Reyn02]. Speaker identification involves with the determination of who the

speaker is from a group of known voices or speakers. Speaker verification involves with the determination of whether a person is who he/she claims to be.

1.1.3 Thesis Aim

The aim of this work is to present an experimental evaluation of *feature extraction* techniques that could be used towards text-independent speaker verification. The Feature extraction is the process of extracting speaker-specific properties from the raw signal and storing it into a feature vector [KiLi10]. The speech signal consists of many features, all of which might not be important for speaker verification. A good feature should include the following characteristics [Rose02]:

- Should discriminate between speakers while having small within-speaker variability.
- Be robust against noise.
- Occur frequently and naturally in speech.
- Be easy to extract from the speech signal.
- Should not be susceptible to mimicry.
- Should be stable over time and not affected by speakers health.

Meanwhile, the number of features should also be considered since the number of required training samples for reliable density estimation grows exponentially with the number of features [JaDM00]. Moreover, the computational savings are also achieved with lower dimension features.

1.1.4 Approaches to Achieving the Aim

Signals can be stationary or non-stationary, and can originate from a linear or non-linear system. A signal can be recorded above the Nyquist sampling frequency, and each sample can be

quantized to a required number of bits in order to satisfy the dynamic range of the signal. The total number of samples recorded is called a record and the recorded samples constitute a time series. If the recorded signal is stationary in the entire record, analysis can be carried out on the entire record. However, if the signal is not stationary, the analysis must be conducted within short-time stationary frames of the signal. The time series can be analyzed using several approaches, including:

- i. Time domain analysis [AlMi04]
- ii. Frequency domain and spectral analysis [AlMi04][Beig11]
- iii. Time-frequency analysis [AlMi04][Groch01]
- iv. Multiscale analysis [CoZa11]
- v. Polyscale analysis [Kins11][Kins05]

In the time domain analysis, the signal is analyzed with respect to time and examples of such analysis are statistical methods, power, and zero crossing rate. In the frequency domain and spectral analysis, the signal is analyzed with respect to frequency and examples of such analysis are the Fourier transform, the Haar transform, and Mel-frequency cepstrum coefficients. In the time-frequency analysis, the signal is analyzed with respect to time and frequency simultaneously and examples of such analysis are the short-time Fourier transform and the Wigner distribution function. In the multiscale analysis, the signal is analyzed at multiple scales and examples of such analysis are wavelets and time-scale analysis. In the polyscale analysis, the signal is analyzed based on the power law relationship of the measures extracted from multiple scales and examples of such analysis is the variance fractal dimension and the Higuchi fractal dimension.

Many natural static objects and dynamic phenomena (speech signal) are independent of scale over many orders of magnitude [Kins11]. Such objects are either self-similar or self-affine. Self-similar objects have an isotropic (the same scale along different coordinates) invariance against changes in scale and self-affine objects have a non-isotropic (different scales along different coordinates) invariance against changes in scale. To quantify such object or processes there is a need for polyscale analysis [Kins11]. Therefore, in this work application of polyscale analysis for speaker verification is introduced. The polyscale analysis provides a measure of complexity using *fractal dimension* (FD) with respect to self-similarity or self-affinity of the signal using measures taken from different scales simultaneously. Complexity refers to the difficulty of describing the pattern of an object [Kins08] and FD provides the degree of roughness of an object.

1.1.5 Proposed Solution

One of the structures of any speaker verification system is the front-end processing. Front-end processing generally consists of some form of *voice activity detection* (VAD) to remove non-speech sections of the signal, followed by the extraction of features that contain the speaker's identity from the speech signal [Reyn02]. The features vectors extracted are then used to build a model of the speaker or test against the model and decide if the person is he/she claims to be.

But, before proceeding to the front-end processing, a speech signal is required. A dataset consisting of 44 words is recorded using 12 male and 12 female volunteers raised in the province of Manitoba. The choice of these 44 words is due to having enough speech data to build a model and at the same time, quick to record making it practical to use for a real-world application. Moreover, the volunteers being from a specific geographical location can limit the variety of

accents and forms of speaking and thus, the analysis will be based on the physiological factors of the speaker. The motivation behind the development of this dataset arises by the need to know details under which the recordings were conducted.

The traditional approach towards solving of speaker recognition problem involved the use of linear methods. However, the process of speech production is nonlinear [NeMM06]. Speech has nonlinear characteristics and its multifractal nature has been proven [LaSK97]. A VAD based on the FD is used to separate the non-speech segments of the signal. The choice of FD is due to the estimation of the FD based on signal complexity and not relying on the amplitude. This method would be compared to VAD detection methods that use time domain analysis which is commonly used [Beig11].

Fusion is the combination of information from multiple sources [KiLi10], which is used to combine nonlinear method to the traditional methods and form the feature vectors. The features used to form the feature vector are the *linear prediction cepstral coefficients* (LPCC), *Mel-frequency cepstral coefficients* (MFCC), *Higuchi fractal dimension* (HFD), *variance fractal dimension* (VFD), *zero crossing rate* (ZCR), and *Turns count* (TC). The theory and programming of these algorithms are fully discussed in chapter 3 and the motivation behind using them is discussed in section 5.4.

Upon extraction of the feature vectors, the *support vector machine* (SVM) is used to build a model of the speaker and test it against unseen data. The choice of SVM is due to the availability of different kernel functions suitable for different type of features and the availability of highly optimized libraries that could be used.

1.2 Thesis Formulation

This thesis comprises of three portions which include, recording of a dataset, front-end processing, and classification. The next section discusses the thesis statement followed by the thesis objective and the research questions.

1.2.1 Thesis Statement

The core of this thesis is to assess the suitability of embedding fractal methods, due to the nature of speech, to the front-end process of a speaker verification system and to investigate the effectiveness of these methods. But before proceeding to the front-end processes of any speaker verification system, a speech signal is required. Therefore, volunteers are recorded and the acquired signals are stored in a repository. Moreover, a detailed description of the recording procedures is provided to serve as a guide and ensure the repeatability of the recordings.

1.2.2 Thesis Objective

There are three main objectives in this thesis:

1. Recording of participating volunteers and establishment of a dataset that could be used for the study of text-independent speaker verification by:
 - a) Designing a set of protocols to ensure the quality and the repeatability of the recordings;
 - b) Storage of the dataset in a repository accessible by researchers to allow further studying in the field of speaker recognition.

2. Study the suitability of using FD in the front-end processing of a speaker verification system by:
 - a) Comparison of the HFD and the VFD using test data;
 - b) Studying the effects of addition of colored noise on estimation of FD using the HFD and the VFD algorithms;
 - c) Implementing a VAD based on FD of the speech signal;
3. Assess the effectiveness of the VAD algorithm and each algorithm in the fusion of features by:
 - a) Dividing the data into training and testing data;
 - b) Extracting the features from the speech part of the signal and forming multiple feature vectors based on fusion of different features to assess the effects of each algorithm;
 - c) Building a training model from the training data for each combination of feature vectors and using the testing data to measure the accuracy of the classification;
 - d) Extraction of the same features vectors using different VAD and comparing the accuracy results.

1.2.3 Research Question

The goal of this thesis is the robust feature extraction from speech, which could be used in a text-independent speaker verification system. However to achieve this goal a number of research questions arises which are addressed below:

1. What set of test words to use which would be practical for a real-world application and at the same time contains enough data for speaker verification?
2. How to record these test words to ensure quality, repeatability, and similarity of all the recordings?
3. How and where to store the recorded dataset to allow further research using the dataset?
4. Fractal dimension estimation algorithms are numerous and a question that arises is which one is more suitable for text-independent speaker verification?
5. What are the effects of noise on FD estimation?
6. Can FD be used for VAD?
7. Will using FD for VAD improve the performance of the speaker verification system in comparison to other algorithms?
8. Does fusion of multiple algorithms increase the accuracy of speaker verification?
9. How can the accuracy be compared with the literature if the dataset is different?
10. What is the effect of addition of each feature to the feature vector, on the accuracy of speaker verification?
11. Which fusion of features is more appropriate for speaker verification?
12. How to divide the training and the testing data to avoid overtraining the SVM?
13. Which kernel and cost function should be chosen for the extracted feature vectors?

1.3 Thesis Organization

This thesis presents a study of text-independent speaker verification with the aim of embedding fractal algorithms to the front-end processing of the speaker verification system. This thesis consists of 6 chapters. Chapter 2 presents a fundamental background for this study on speaker verification. This chapter discusses the physiology of the speech production and perception, the phonetics of speech, and some of the fundamental methods needed for speech processing. Chapter 3 presents the algorithms used in this thesis for this study on speaker verification. This chapter discussed the algorithms used to generate colored noise, test data (the Weierstrass function and fractional Brownian motion), feature extraction (LPCC, MFCC, HFD, VFD, ZCR, and TC), and the classifier (SVM). Chapter 4 presents the procedures for the recording of the dataset used for this study on speaker verification. This chapter discusses the hardware and software used to ensure the quality and similarity of all the recordings. Moreover, a list of English phonemes chosen for this study and the demographics of the speakers are presented, followed by the environment and a set of protocols that would be followed to ensure the repeatability of all the recordings. Chapter 5 presents the design of experiments and the analysis of the results obtained for this study on speaker verification. This chapter discusses the results of the HFD and VFD on the test data, tests the effects of colored noise on FD estimation, introduces a VAD algorithm based on FD, introduces the feature vectors used for experimental sensitivity analysis, and classifies the feature vectors to measure the accuracy. Chapter 6 presents the conclusion of this study on speaker verification. This chapter discusses a summary of the results and findings, answers the research questions and reasoning behind it, and provides suggestions and recommendation for future work.

Chapter 2

Literature Review on Speaker Verification

Speech is a type of signal that is mainly affected by the background noise or noise from the transmitting channel [Grie96]. This Chapter focuses on providing a background on some of the fundamental aspects in the area of speaker recognition and discusses the physiology of speech production and perception, the phonology of speech, and some of the fundamental speech analysis techniques which will be used in the algorithms discussed in chapter 3.

2.1 The Physiology of Speech

Understanding the mechanism of speech production and the knowledge of the auditory system may allow us to do a better job at extraction of characteristics of our voice and systems that recognize the speakers [Beig11]. In this section, a description of the process of speech production, the mechanics of speech perception, and the auditory pathway and how the brain deciphers speech is provided.

2.1.1 The Process of Speech Production

The simplest principle of speech production is that all sounds are produced by moving air. Human speech is produced by the interaction of speech organs, as shown in fig 2.1, to shape this air into specific sound. The speech organs can be divided into three sections, namely, the *pulmonary tract*, the *larynx*, and the *vocal tract* [Lang92].

The Pulmonary Tract

The pulmonary tract provides the air flow required for speech production. It consists of *lungs* and *trachea*. The lungs provide the energy source, by the respiration process [Mann17]. The lungs expand and contract, causing a decrease and an increase in air pressure in the lungs, which allows air to be drawn in and out. The trachea allows the air to pass from the lungs to the larynx.

Larynx

The larynx converts the airflow into pulses. It consists of the *vocal chords* and their muscles [Lang92]. The opening of the vocal chords is in the shape of a triangle, the front sloping up in the form of the *epiglottal wall*, surrounded from the back by the *Corniculate cartilage*, and towards the side by the *cuneiform cartilage* [Beig11]. The space in between the vocal cords is called the *glottis*.

Vocal Tract

The vocal tract controls all the articulations. It consists of a *pharynx*, an *oral cavity*, and a *nasal cavity*. The pharynx has an irregular shape and depending on the amount of air going through the vocal cord, the air goes to a different section of the pharynx to produce different sounds

[Beig11]. The determination of whether the nasal cavity is included in the production of speech is up to the *velum*. The output of the vocal tract is passed through the teeth and lips to become audible acoustic waveforms.

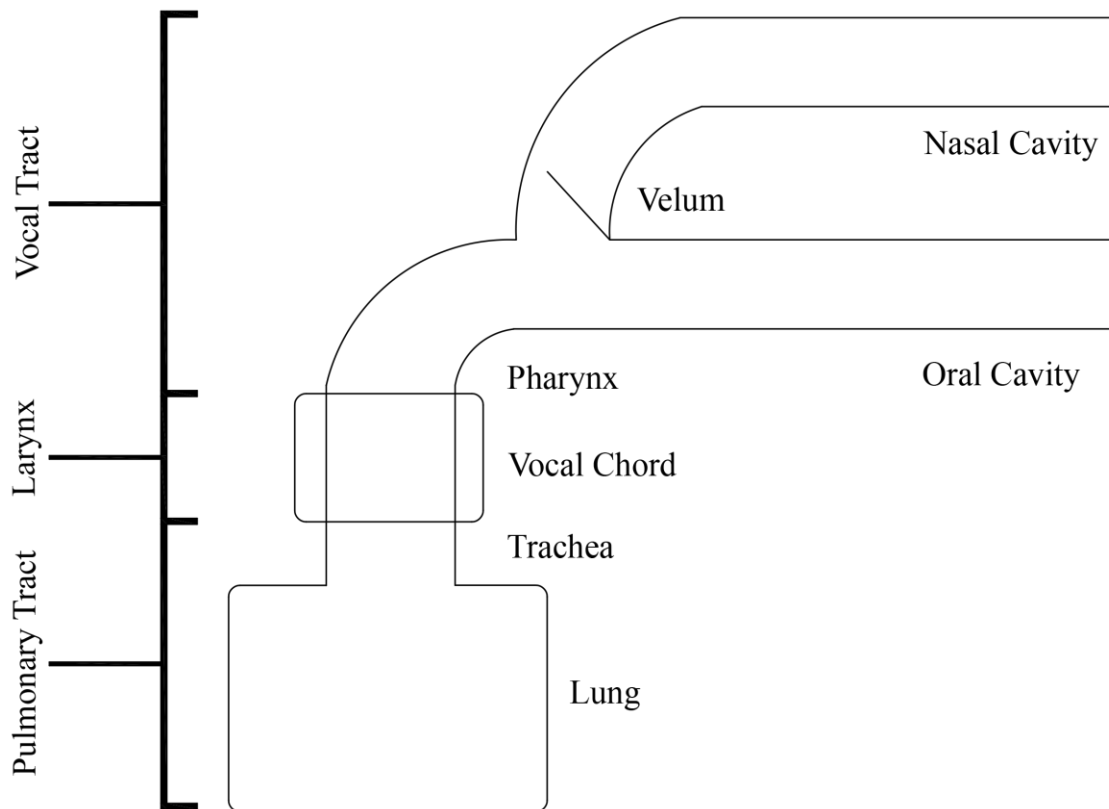


Fig. 2.1. A speech production model. (After [Lang92])

2.1.2 The Mechanics of Speech Perception (the Ear)

The Ear is the mechanical part of hearing which, consist of three sections, namely, the *external ear*, *middle ear*, and the *inner ear* which is displayed in fig 2.2. The external ear consists of folds of cartilage called the *pinna*. The pinna is the visible part of the ear and is responsible for reflecting and attenuating the sound waves, which help the brain to determine the direction of the sound.

The middle ear includes the *tympanic membrane* (ear drum) and three bones called the *malleus*, *incus*, and *stapes*. The vibrations from the ear drum that are induced by the sound waves being reflected and attenuated by the external ear are transferred through the malleus to the incus and from the incus to the stapes. From the stapes, these vibrations are then transferred to the inner ear through the oval window of the cochlea. The middle ear protects the inner ear from damage by loud sounds.

The inner ear is filled with fluid and is made up of the *cochlea* and three canals called the *superior ampulla*, the *anterior ampulla*, and the *posterior ampulla* [Beig11]. Motion from the stapes produces fluid waves in these canals which excites the hair (*cilia*) in the spiral of cochlea. The fluid waves are transformed to electrical impulses using the *cilia*. A semi-logarithmic cognitive ability of sounds is produced due to the spiral shape of the cochlea which is important in the development of speaker models and features.

Once the cilia are excited, the signal they generate is carried through the auditory nerve bundle to the brain. The ear only functions as a “transducer,” and the sound is heard with the brain [Beig11].

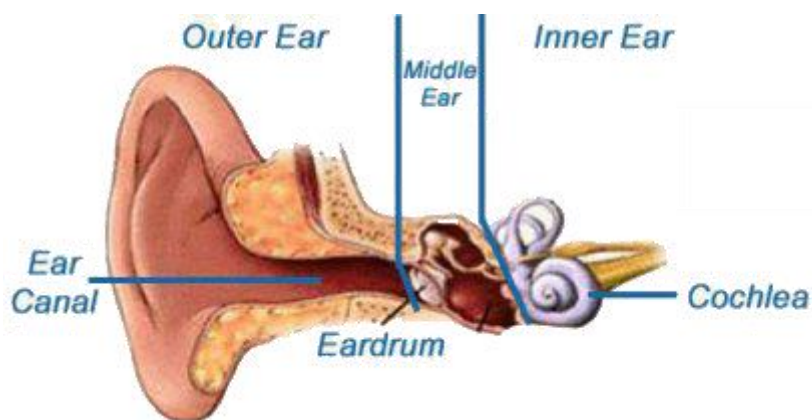


Fig. 2.2. The three sections of the ear. (From [Micr18])

2.1.3 The Auditory Pathway and the Brain

The auditory nerve carries the signals from the organ of *Corti* to the brainstem, where the auditory information is processed by the *cochlear nuclei* and superior *olivary complex*. It then travels up into the midbrain, which consists of three nuclei that are involved in localization of the sound, and decoding of basic signals such as duration, intensity, and frequency. These nuclei are called *medial superior olive*, the *lateral superior olive*, and the nucleus of the trapezoid body [Ekma12].

Further, in the midbrain, the *inferior colliculus* does higher level processing and integration of auditory information from the previous structures. From the midbrain, the electrical signals travel to the *thalamus* which integrates the sensory systems in the body and hence functions as an essential factor in the preparation of a motor response [Ekma12].

The thalamus then relays the signals to the *auditory cortex* of the brain, located in the temporal lobe. The primary auditory cortex is found in the superior temporal *gyrus*, which is above the ear on either side of the brain. At this stage, the message has already been largely decoded, however, the signal is moreover recognized, memorized and may eventually result in a response [Ekma12]. Figure 2.3, displays the auditory pathway of the human's auditory system.

2.2 The Phonetics of Speech

Phonetics is the study of sounds produced by the human vocal system regardless of the associated language [Beig11]. In this section, three elements of speech production, *initiation*, *Phonation*, and *articulation* are discussed.

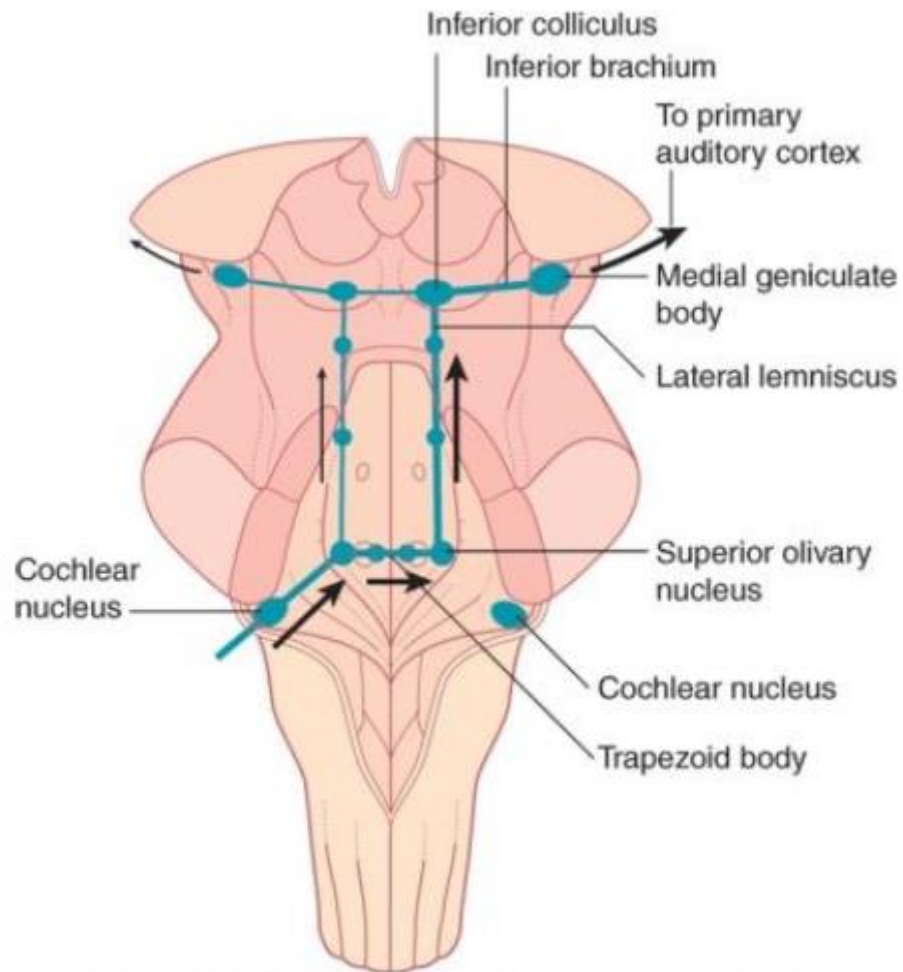


Fig. 2.3. The auditory pathway. (From [FIGM07])

2.2.1 Initiation

Initiation is a function of the airstream mechanism and the direction of airflow [Beig11]. There are three types of airstreams namely, *pulmonic*, *glottalic*, and *velaric*. The pulmonic airstream is initiated from the lungs. The glottalic airstream is initiated in the larynx when the glottis is closed. The velaric airstream is initiated when the airstream is produced by the movement of the tongue.

The air may move outwards or inwards for any of the three initiation airstreams. If the air moves outwards, it is called *egressive*, and if it moves inwards, it is called *ingressive*.

2.2.2 Phonation

Phonation is the process by which certain sounds are produced by the vibration of the vocal cords. The different kinds of phonations are *unvoiced*, *voiced*, and *whisper*. Unvoiced phonation happens when the vocal folds do not vibrate and are relaxed. Voiced phonation happens when the vocal folds are vibrating. Whisper phonation happens like generating a voiced phonation, however, the vocal folds are more relaxed causing a greater air flow to go through them.

2.2.3 Articulation of English Phonemes

Phonemes are members of the smallest unit of speech that distinguish different utterances from each other [Grie96]. Consonants and vowels are two categories of phonemes. Consonants are produced when the airflow from the lungs is obstructed in the middle of the vocal tract and when this obstruction does not occur vowels are produced. Table 2.1 shows the Edinburg *Machine Readable Phonetic Alphabet* (MRPA) representation of English phonemes and its corresponding keywords [Gri96].

Consonants

The consonants can be divided into several subcategories based on the voicing state, the manner of articulation, and the place of articulation [Carr93]. Speech *articulation* is the way a speech sound is being generated [Lang92].

The manner of articulation is characterized by the amount of airflow being obstructed by the

articulator. Stop and nasal sounds are produced when there is a complete closure at some point in the vocal tract, with the nasal also includes the lowering of the velum. This closure causes the-

Table 2.1: The English Phonemes and the corresponding keyword (From [Grie96]).

No	MRPA	Keyword	No	MRPA	Keyword	No	MRPA	Keyword
1	/p/	Pip	16	/zh/	Measure	31	/oo/	For
2	/b/	Barb	17	/h/	Hand	32	/u/	Book
3	/t/	Test	18	/r/	Rear	33	/uu/	Boot
4	/d/	Deed	19	/l/	Loyal	34	/uh/	Bud
5	/k/	Kick	20	/m/	Mime	35	/@@/	Bird
6	/g/	Gag	21	/n/	None	36	/@/	Banana
7	/ch/	Church	22	/ng/	Ringing	37	/ei/	Bay
8	/jh/	Judge	23	/y/	Year	38	/ou/	Boat
9	/f/	Fife	24	/w/	Weal	39	/ai/	Buy
10	/v/	Verve	25	/ii/	Bead	40	/au/	Bough
11	/th/	Thirtieth	26	/i/	Bid	41	/oi/	Boy
12	/dh/	Other	27	/e/	Bed	42	/i@/	Beer
13	/s/	Cease	28	/a/	Bad	43	/e@/	Bear
14	/z/	Zoos	29	/aa/	Baard	44	/u@/	Poor
15	/sh/	Sheepish	30	/o/	Body			

-airflow to build up a pressure and a sudden release that result in sounds with a noise like spectrum. Fricative sounds are produced when there is a stricture in the vocal tract causing the air flow to be forced through the vocal tract and a turbulent flow of air to be created resulting in a sound that looks like noise. Approximants are produced when the opening between articulators is wide enough to avoid turbulent airflow [Carr93]. Affricatives are sounds that are made of stops followed by a fricative [Beig11].

The place of articulation defines the part of the vocal tract that is acting as the articulator [Grie96]. Bilabial sounds are produced by closing the lips to build up a pressure in the mouth and then releasing it to produce an impulsive sound. Labiodental sounds are produced using the lips and teeth to produce a turbulent air flow. Dental sounds are produced by the tip of the tongue and the upper teeth, producing a turbulent high-frequency vibration. Alveolar sounds are produced with the upper tip of the tongue placed on the alveolar ridge of the roof of the mouth. Post-alveolar sounds are produced behind the alveolar ridge. Palate-alveolar sounds are produced using the blade of the tongue and the back of the alveolar ridge. Palatal sounds are produced using the front of the tongue and the hard palate. Velar sounds are produced by the back of the tongue touching the soft palate. Uvular sounds are produced by the back of the tongue and the uvula. Pharyngeal sounds are produced by the walls of the pharynx. Glottal sounds are produced by using the closure and opening of the vocal cords.

The parameters discussed above can be used to distinguish consonants as shown in table 2.2.

Vowels

The vowels can be divided into two types namely, the monophthongal vowels and the diphthongal vowels. The monophthongal vowels can be categorized by a set of articulatory

positions, namely, the height of the tongue body, the front/back position of the tongue body, and the presence or absence of lip rounding [Grie96].

Table 2.2: The English Consonants. (From (Grie96)).

		bilabial	Labiodental	Dental	Alveolar	Post-alveolar	Palatal	Velar	Uvular	Pharyngeal	Glottal
STOPS	Voiceless	/p/			/t/			/k/			
	Voiced	/b/			/d/			/g/			
NASALS		/m/			/n/			/ŋg/			
AFFRICATIVE S	Voiceless					/tʃ/					
	Voiced					/dʒ/					
FRICATIVES	Voiceless		/f/	/θ/	/s/	/ʃ/					
	Voiced		/v/	/ð/	/z/	/ʒ/					
APPROXIMAN TS	Central	/w/				/r/	/j/				/h/
	Lateral				/l/						

The height of the tongue body is categorized by, when the tongue body is near the hard or soft palate as close vowels, and when the tongue body is as far as possible from the roof of the mouth as open vowels. There are two intermediate heights which are half open and half closed

for the remaining vowels [Grie96]. The front-back dimension consists of three categories, namely, front, central, and back, where the position of the tongue is as far forward, as far back, and at an intermediate position respectively. Round vowels are produced by the spread of the lips (lip rounding) and unrounded vowels are produced when this rounding does not occur.

The diphthongal vowels are articulated starting in the positions of one monophthongal vowel and glide through a transition to the position of another monophthongal vowel. For example, the vowel \ai\ starts its articulation as close, back, and unrounded and undergoes a transition to close, front, and unrounded.

2.3 Fundamental Speech Analysis Techniques

Numerous techniques have been used in the analysis of speech. This section aims at introducing some of the fundamental techniques and concepts that are used in the algorithms discussed in chapter 3.

2.3.1 Discrete Fourier Transform

In order to compute the spectrum, $X(f)$, of a signal, $x[n]$, the *discrete Fourier transform* (DFT) is calculated on a window of N samples [Grie96] which is defined by

$$X(f) = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)fn} \quad (2.1)$$

The DFT computed using the above equation requires $O(N^2)$ operations to be done [Beig11]. Hence, by using the *fast Fourier transform* (FFT) the complexity of this problem is

reduced to $O(N \log(N))$ and the spectrum is calculated more efficiently [Grie96].

The *power spectral density* (PSD) of a signal can be obtained by multiplying the spectrum by their complex conjugates. Figure 2.4, displays the PSD of a frame containing 512 samples of speech. The FFT is used in the algorithms to obtain LPCC and MFCC described in the next chapter, and the PSD is used to study the characteristics of noise which is discussed in section 5.3.1.

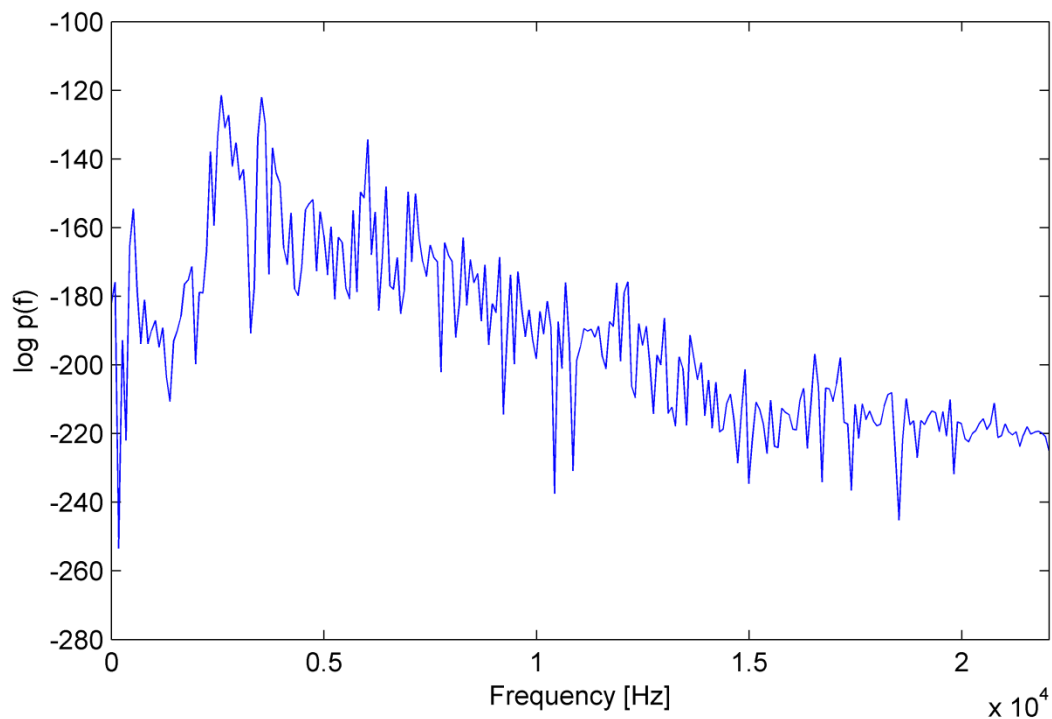


Fig. 2.4. The power spectral density of a frame containing 512 samples of speech.

2.3.2 Root Mean Square

The *root mean square* (RMS) is concerned with the magnitude rather than just the values of samples. It is a second-order statistics that provides a good indication of the deviation of the samples from its origin [Beig11]. The RMS is defined as

$$V_{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N |x(n)|^2} \quad (2.2)$$

where N represents the total number of samples in a given window. Figure 2.5, displays the RMS of a speech signal. Please note that RMS amplitude is normalized. The RMS is used for the addition of colored noise which is described in section 3.1.

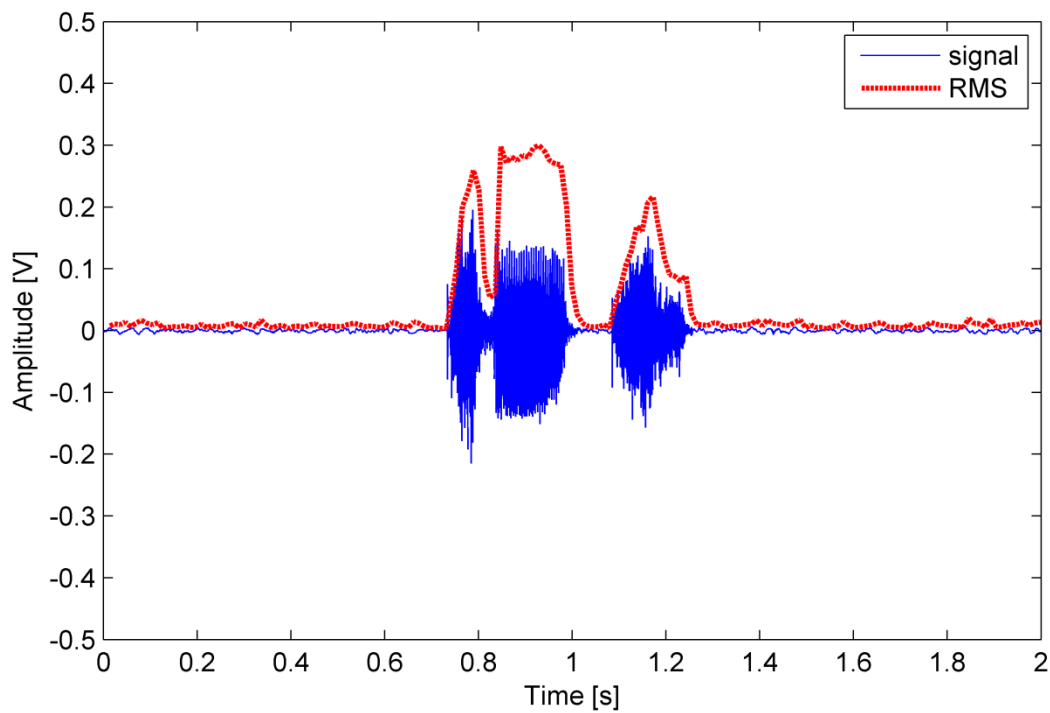


Fig. 2.5. The normalized root mean square of a speech signal

2.3.3 Fractal Dimension

Fractal dimensions are used to estimate the complexity of stationary and self-affine monofractals. However, if the time series is multifractal, the FD has to be calculated over a repeated series of stationary windows to represent the complexity of the signal.

The calculation of nonlinear methods (such as FD) for real-world signals is challenging due

to the presence of noise. Hence, a reliable estimation of the fractal dimension in the time series requires the satisfaction of the conditions of stationarity, a sufficient number of data points and a good *signal to noise ratio* (SNR) [SOAM02].

Moreover, care has to be taken in the selection of the FD estimation algorithm depending on the task. For instance, morphological FD estimates the dimension of a shape using its geometry, while the entropy-based FD can deal with non-uniform distribution in the fractal [Kins05]. The FD estimation algorithms used in this work are the HFD and the VFD. Both these algorithms estimate the FD from the temporal features of speech. The speed in the computation of the FD using both these algorithms makes them suitable for purpose of speech analysis. A description of these algorithms is provided in chapter 3.

2.3.4 Voice Activity Detection

Voice Activity Detection aims at detecting samples of speech in an audio file [KiLi10]. It was reported by [Beig11], that approximately 30% of any normal audio recording is comprised of silence frames, and thus, by removing the silence any computation on the audio files should become faster by the same rate, due to the reduced number of frames. A simple and efficient method commonly used for VAD is setting a threshold on signal energy to detect speech samples ([Beig11], [KiLi10]).

2.4 Summary

This chapter provided a description of the physiology of the speech production and perception, followed by, the phonetics of speech and three elements of speech production is

discussed. Furthermore, some of the speech analysis techniques and concepts that would be used in the upcoming chapters are discussed.

In the next chapter, the algorithms used in this thesis for this study on speaker verification are discussed.

Chapter 3

Theoretical Background

This chapter presents the theoretical background and the programming of the algorithms which will be used in chapter 5. At first, colored noise is discussed briefly, followed by the framing of the speech signal into stationary segments. Moreover, the algorithms that will be used for feature extraction (ZCR, TC, LPCC, MFCC, HFD, and VFD) are discussed, followed by, the test signals (Weierstrass function and fBm) which will be used to study the accuracy of the FD estimation algorithms used. In the same time, the SVM which will be used to build a model and test the unseen data is discussed.

3.1 Colored Noise

A broadband signal can be characterized by its PSD. If we assume the PSD has the following power law [Kins05]

$$P(f) \sim \frac{1}{f^\beta} \quad (3.1)$$

Where f is the frequency of the signal and β is the exponent, then this signal can be characterized by colored noise depending on the value of β ; for $\beta = 0, 1, 2$, the noise is white, pink, and brown, respectively.

White noise has equal intensity over all frequencies, thus, producing a flat power spectrum. Pink noise has a spectral decay of $\beta = 1$ and is mildly non-stationary [Pott08]. Pink noise is dominant in nature [Kins15] and it is common in biological systems and relaxation processes [Pott08]. Brown noise has a spectral decay of $\beta = 2$ and is non-stationary. Brown noise is a good model for natural shapes and physical processes [Kins15].

The characteristics and generation of colored noise are of interest because of the use in the study of complex time signals [Kins15]. Colored noise is generated using the built-in code “*dsp.ColoredNoise*” in Matlab [Math16].

Different levels of colored noise in *decibels* (dB) is added to the signal using the equation defined as

$$noise = y \times \left(\frac{V_{RMS}(x)}{V_{RMS}(y)} \right) \times 10^{\left(\frac{L_i}{20} \right)} \quad (3.2)$$

where y is the generated colored noise, x is the signal, and L_i is the desired level of additive colored noise in dB. Addition of colored noise will be used in section 5.2 to test the effects of noise on the estimation of FD and in section 5.3 it will be used in the VAD used for this work. The matlab code used for the addition of colored noise to the signal is provided in the attached CD.

3.2 Framing of Speech

The speech signal is a non-stationary signal [Beig11] and in order to perform any analysis, it needs to be divided into frames that are stationary. At the first glance, it might seem appropriate to use a frame of 80 ms which is the average length of the phonemes. However, some of the stops might be in the order of 5 ms and thus their effects might be missed.

Therefore, all the algorithms used in this thesis are on frames of 512 samples of speech which is equivalent to 11.6 ms of speech which is used by [KiGr09]. The flowchart for the programming of the framing of speech algorithm is provided in appendix B, fig. B.1 and the Matlab code is provided on the attached CD.

3.3 Zero Crossing Rate

The ZCR is the measure of the number of times the amplitude of a signal crosses the value of zero in a given frame. The ZCR is defined as [TeKi16]

$$ZCR = \sum_{m=-\infty}^{\infty} [|\text{sgn}[x(m)] - \text{sgn}[x(m-1)]|] w[n-m] \quad (3.3)$$

where $\text{sgn}(\bullet)$ represents the sign function and is defined as

$$\text{sgn}(x[n]) = \begin{cases} 1, & x[n] \geq 0 \\ -1, & x[n] < 0 \end{cases} \quad (3.4)$$

and $w[\bullet]$ represents a window containing a stationary segment of a signal and is defined as

$$w[n] = \begin{cases} \frac{1}{2N}, & 0 \leq n \leq N-1 \\ 0, & \textit{otherwise} \end{cases} \quad (3.5)$$

where N represents the total number of samples in a given window. The ZCR is a monoscale measure that provides an estimate of the frequency of the signal in a given frame. The ZCR of an unvoiced speech is greater than that of voiced speech and it is an important parameter for endpoint detection of speech [Beig11]. The ZCR is used for feature extraction which is discussed in detail in section 5.4. Figure 3.1, displays the normalized ZCR of a speech signal. The flowchart for the programming of the ZCR algorithm is provided in appendix B, fig. B.2 and the Matlab code is provided on the attached CD.

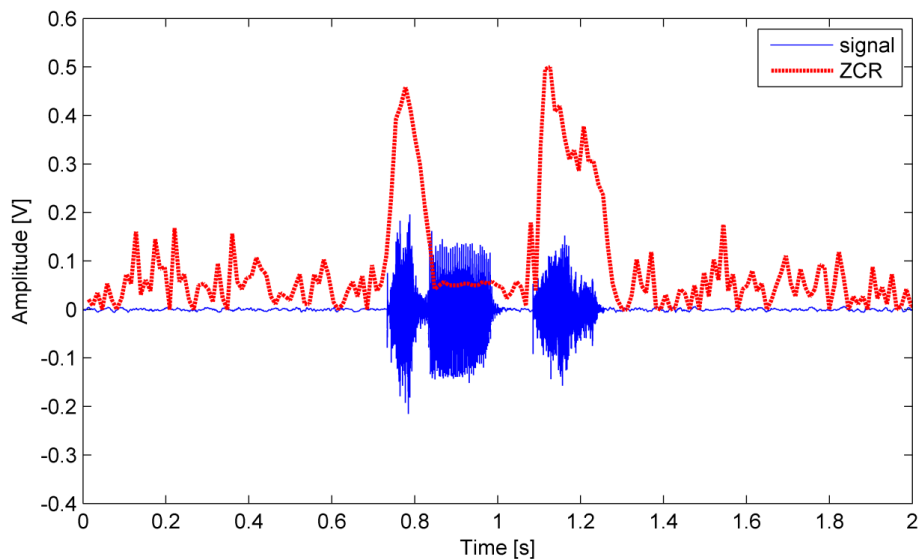


Fig. 3.1. The normalized zero crossing rate of a speech signal.

3.4 Turns Count

Turns count is a method of counting every change in the phase (direction of slope) of the speech signal [Rang01]. Let's assume a signal $x(n)$ containing N number of samples. A turn

occurs if

$$tr_i = x(n) > x(n+1) \& x(n+1) < x(n+2) \quad (3.6)$$

or

$$tr_i = x(n) < x(n+1) \& x(n+1) > x(n+2) \quad (3.7)$$

where tr_i is the occurred turn at a specific time interval. The TC is then defined by

$$TC = \sum_{i=1}^{\infty} tr_i \quad (3.8)$$

Figure 3.2, displays the normalized TC of a speech signal. The TC is used for feature extraction which is discussed in detail in section 5.4. The flowchart for the programming of the TC algorithm is provided in appendix B, fig. B.3 and the Matlab code is provided on the attached CD.

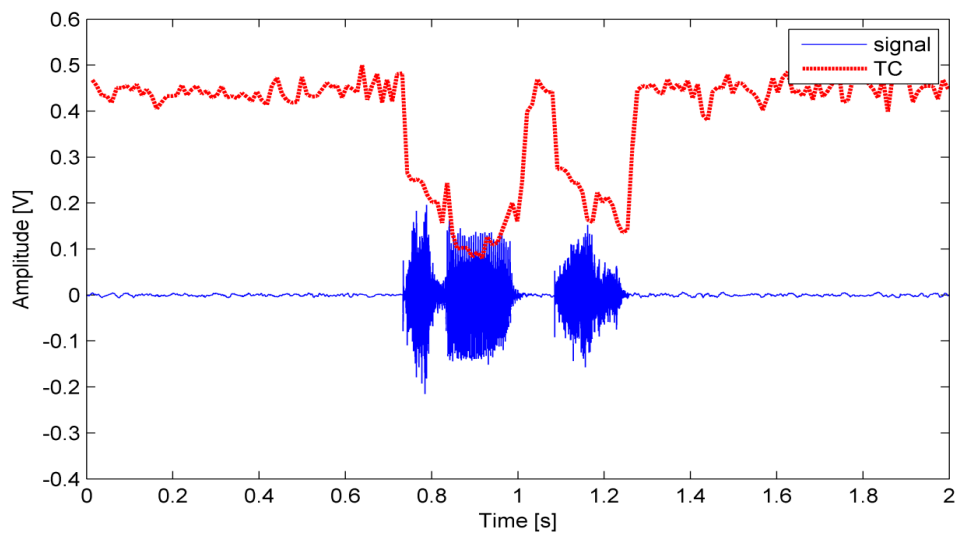


Fig. 3.2. The normalized turns count of a speech signal.

3.5 Mel-Frequency Cepstral Coefficients

Mel-frequency cepstral coefficients are used to extract short-term spectral features based on human perception of speech [KiLi10]. At first, the signal is divided into frames containing a stationary segment of the speech, and its Fourier transform is calculated using the FFT algorithm [Math17]. Upon calculation of the FFT, the power spectrum is obtained by extracting the absolute value of the FFT.

The envelope of the spectrum is of interest because the spectrum presents many fluctuations and smoothing the spectrum reduces the spectral vectors size [BBFG04]. The spectrum is then mapped into the Mel scale which is an auditory scale similar to the frequency scale of the human ear [BBFG04]. The Mel scale is given by

$$f_{mel} = 2595 \times \log_{10} \left(1 + \frac{f}{700} \right) \quad (3.9)$$

where f is the frequency of the signal. We then calculate the log of the spectral envelope to obtain the spectral vectors. Finally, to obtain the MFCC the *discrete cosine transform* (DCT) is applied to the spectral vectors which is given by [BBFG04]

$$c_n = \sum_{k=1}^K S_k \cdot \cos \left[n \left(k - \frac{1}{2} \right) \frac{\pi}{K} \right], \quad n = 1, 2, \dots, L, \quad (3.10)$$

where S_k are the log-spectral coefficients, K is the length of log-spectral coefficients, and L is the number of MFCC required to be calculated ($L \leq K$). We then obtain the MFCC for each frame of the signal. The MFCC is used for feature extraction which is discussed in detail in section 5.4. The flowchart for the programming of the MFCC algorithm is provided in appendix

B, fig. B.4 and the Matlab code is provided on the attached CD.

3.6 Linear Prediction Cepstral Coefficients

Linear prediction cepstral coefficients are coefficients that are transformed from *linear predictive coding* (LPC) coefficients due to being more robust and less correlated [KiLi10]. The LPC analysis is based on a linear model of speech productions [BBFG04]. The LPC in the time domain is defined as [KiLi10]

$$s^*[n] = \sum_{k=1}^P a_k x[n-k] \quad (3.11)$$

where $x[n]$ is the signal, a_k is the LPC coefficient, and $s^*[n]$ is the predicted signal.

At first, the signal is divided into frames containing a stationary segment of the speech. Then we find the autocorrelation vector by finding the *inverse fast Fourier transform* (IFFT) of the square of the spectrum for each frame of speech. The LPC coefficients a_k are then found by the Levinson-Durbin algorithm. The spectral model is defined as

$$H(z) = \frac{1}{1 - \sum_{k=1}^P a_k z^{-k}} \quad (3.12)$$

We then calculate the log of the spectrum of the LPC coefficients. Finally, the LPCC coefficients are found by calculating the IFFT of the log-spectrum. The LPCC is used for feature extraction which is discussed in detail in section 5.4. The flowchart for the programming of the LPCC algorithm is provided in appendix B, fig. B.5 and the Matlab code is provided on the attached CD.

3.7 Higuchi Fractal Dimension

Consider a given finite set of time series $x(1), x(2), \dots, x(N)$. From the given time series, a new time series is constructed which is defined as follows

$$x_k^m = x(m), x(m+k), x(m+2k), \dots, x(m + (\frac{N-m}{k})k) \quad (3.13)$$

for $m = 1, 2, \dots, k$ where m indicates the initial time, k indicates the interval time, N is the total length of the time series and only the integer part of $(\frac{N-m}{k})$ is taken. The length of each of the time series x_k^m is defined as follows

$$L_m(k) = \left[\left(\sum_{i=1}^{\lfloor \frac{N-m}{k} \rfloor} |x(m+ik) - x(m+(i-1)k)| \right) \frac{N-1}{\lfloor \frac{N-m}{k} \rfloor \cdot k} \right] / k \quad (3.14)$$

where $(N-1)/\lfloor \frac{N-m}{k} \rfloor \cdot k$ is the normalizing factor. The length of the time series $L(k)$ is computed as the mean of the k values, for $m = 1, 2, \dots, k$, as following

$$L(k) = \sum_{m=1}^k L_m(k) \quad (3.15)$$

The HFD estimate is the slope of $\log(L(k))$ over $\log(1/k)$ where $k = 1, 2, \dots, k_{max}$ [Higu88]. Figure 3.3, displays the HFD trajectory of the utterance church. The HFD is tested in section 5.1 and 5.2 and used for feature extraction in section 5.4. The flowchart for the programming of the HFD algorithm is provided in appendix B, fig. B.6 and the Matlab code is provided on the attached CD.

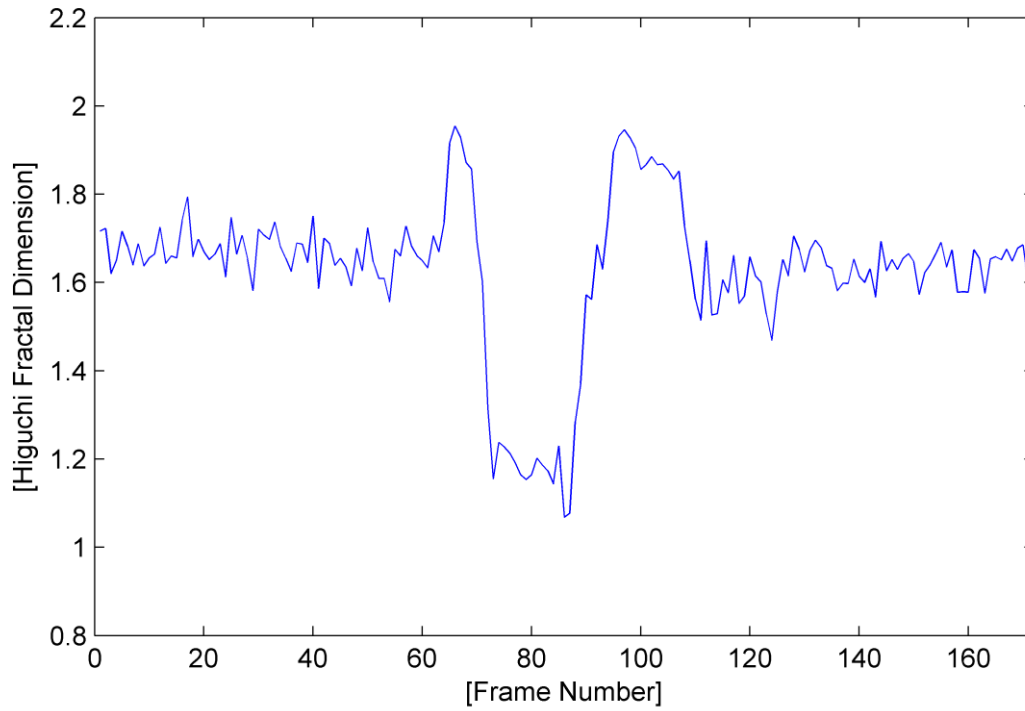


Fig. 3.3. The Higuchi fractal dimension trajectory.

3.8 Variance Fractal Dimension

The VFD can be used to estimate the FD of a signal [Grie96]. Let's consider a sequence containing N samples at times, t_n , given by

$$t_n = n\delta t, n = 0, 1, 2, \dots, N - 1 \quad (3.16)$$

where n is the time index, and δt is the time displacement between individual samples. A dyadic time displacement is chosen for the measurement of the increments, Δt , as shown below

$$\Delta t = n_k \delta t, n_k = 2^0, 2^1, 2^2, \dots, n_k \max \quad (3.17)$$

where n_k is the vel size and $n_k \max$ is the largest vel size. The increment, $b(n\delta t, n_k \delta t)$, between

the signal samples is determined by

$$b(n\delta t, n_k \delta t) = x(n\delta t + n_k \delta t) - x(n\delta t) \quad (3.18)$$

where $x(t)$ is the signal.

The variance of the increments, $V\{b(n\delta t, n_k \delta t)\}$, can then be calculated by

$$V\{b(n\delta t, n_k \delta t)\} = \frac{\sum_{n=1}^{N-n_k} b(n\delta t, n_k \delta t)^2}{N - n_k} \quad (3.19)$$

The slope of the log-log plot can be calculated by

$$s = \frac{k_{hi} \sum_{i=1}^{k_{hi}} X_i Y_i - \sum_{i=1}^{k_{hi}} X_i \sum_{i=1}^{k_{hi}} Y_i}{k_{hi} \sum_{i=1}^{k_{hi}} X_i^2 - \left(\sum_{i=1}^{k_{hi}} X_i \right)^2} \quad (3.20)$$

where $X_i = \log(n_k)$ and $Y_i = \log(V\{b(n\delta t, n_k \delta t)\})$. The *Hurst exponent* (H) can be computed

from the slope by

$$H = \frac{1}{2} s \quad (3.21)$$

and the VFD ($D\sigma$) for a given number of independent variables E can be computed from

$$D\sigma = E + 1 - H \quad (3.22)$$

Where $E = 1$ for a single-variable time series, thus the VFD must be in the range of 1 (for a line) and 2 (for white noise) [Kins15]. This method of calculating the VFD leads to the analysis of

data in real time due to its simplicity [Kins94a] [Kins94b]. The VFD is calculated continuously for each stationary window which forms a trajectory. This is called *variance fractal dimension trajectory* (VFDT). The VFD is tested in section 5.1 and 5.2 and used for VAD and feature extraction discussed in sections 5.3 and 5.4. Figures of the VFDT of the utterances can be found in section 5.3 and appendix A of this document. The flowchart for the programming of the VFD algorithm is provided in appendix B, fig. B.7 and the Matlab code is provided on the attached CD.

3.9 Test Data

This section provides the description of the test data that can be generated with known FD. These test data are used in section 5.1 and 5.2 to test the VFD and the HFD and study the effects of noise on the estimation of the FD. The two sets of the waveforms used are the Weierstrass function and the *fractional Brownian motion* (fBm).

3.9.1 Weierstrass Function

The Weierstrass function is a function that is continuous but nowhere differentiable. It is defined as [RaDu09]

$$W(t) = \sum_{k=0}^{\infty} \lambda^{-kH} \cos(2\pi\lambda^k t) \quad (3.23)$$

where $\lambda > 1$ and $0 < H < 1$. The theoretical FD of the Weierstrass function can be calculated from H [KiGr09], and thus the fractal dimension could be expressed as

$$\text{FD} = 2 - H \quad (3.24)$$

Thus, due to the possibility of generating the Weierstrass function with different FD, it can be used to test the performance of FD estimation algorithms for signals with different complexity. Figure 4, displays the Weierstrass function generated using $H = 0.5$ and $\lambda = 2$. The Matlab code for the generation of the Weierstrass function is provided on the attached CD.

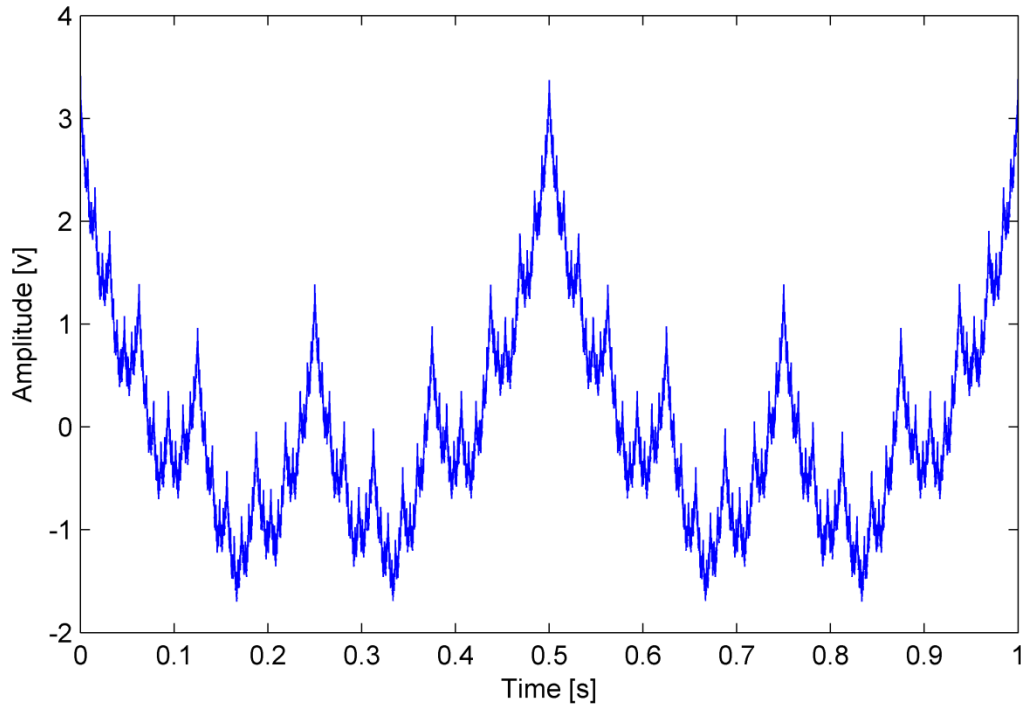


Fig. 3.4. Weierstrass function generated using $H = 0.5$ and $\lambda = 2$.

3.9.2 Fractional Brownian Motion

Fractional Brownian motion is a continuous zero-mean Gaussian process $\{W_t, t \geq 0\}$ with covariance function [KrBo13]

$$\text{Cov}(W_t, W_s) = \frac{1}{2} \left(|t|^\alpha + |s|^\alpha - |t-s|^\alpha \right), \quad t, s \geq 0 \quad (3.25)$$

where α is the roughness parameter and equivalent to $\alpha = 2H$. Similar to the Weierstrass

function, the theoretical FD of the fBm can be calculated from H and is equivalent to equation 3.24 [RaDu09]. Figure 3.5, displays the fBm generated using $H = 0.5$ and the seed for the random number generator set to 1.

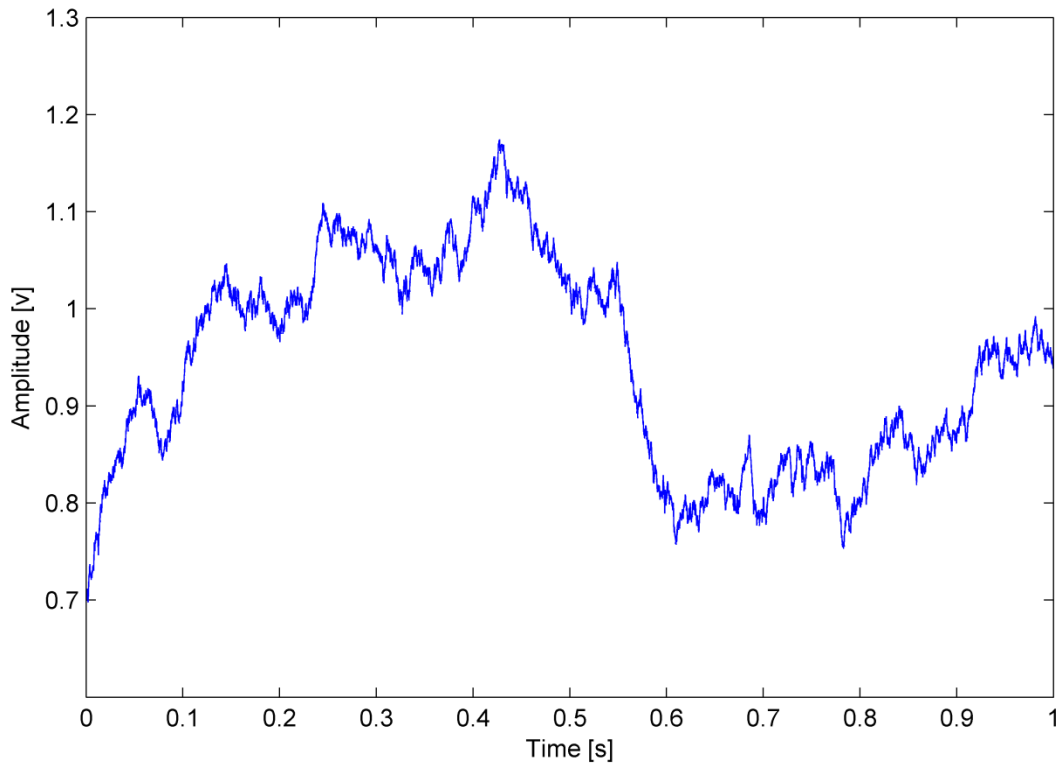


Fig. 3.5. The fractional Brownian motion generated using $H = 0.5$.

The fBm is generated using the circulant embedding method described and implemented in [KrBo13]. The fBm is generated by generating an increment process known as the fractional Gaussian noise that can be characterized as a discrete zero-mean stationary Gaussian process with covariance

$$\text{Cov}(X_i, X_{i+k}) = \frac{1}{2} \left(|k+1|^\alpha - 2|k|^\alpha + |k-1|^\alpha \right), \quad k = 0, 1, 2, \dots \quad (3.26)$$

where $X_i = W_i - W_{i-1}$, and then delivering the cumulative sum

$$W_i = N^{-H} \sum_{k=1}^i X_k \quad (3.27)$$

where N is the number of fBm samples generated and $i = 1, \dots, N$. The fractional Gaussian noise is generated and stored in the first row (r_1, \dots, r_{n+1}) . Then the first row of the circulant matrix is built and is given by

$$\mathbf{r} = (r_1, \dots, r_{n+1}, r_n, r_{n-1}, \dots, r_2) \quad (3.28)$$

From the circulant matrix γ is calculated where

$$\gamma = F \mathbf{r} \quad (3.29)$$

and

$$F_{j,k} = \exp\left(\frac{-2\pi ijk}{2N}\right) / \sqrt{2N}, \quad (3.30)$$

where $j, k = 0, 1, \dots, 2N - 1$. The fractional Brownian motion is then achieved by the first $N + 1$ components of the real and imaginary part of $F * \text{diag}(\sqrt{\gamma})Z$, where Z is a $2N \times 1$ complex value vector. The Matlab code for the generation of the fBm is provided on the attached CD.

3.10 Support Vector Machine

Support vector machine is one of the most robust and popular classifiers in speaker verification due to its good performance in classifying unseen data [KiLi10]. Support vector

machine is a binary classifier that separates two classes of data using a separating hyperplane. It optimizes the decision boundary in order to have a separating hyperplane that maximizes the geometric margin. The higher the geometric margin is the more confident we are that the system separates the two classes. In speaker verification, one class of training data from the authorized person is labeled as +1 and another class of training data from an imposter is labeled as -1.

Meanwhile, some of the problems are highly nonlinear which the kernel function allows us to solve. Kernel function allows the increase of dimensions of the input features [Fere15], and in a higher dimension, the two classes of data could be separated with a hyperplane. This allows solving of nonlinearly separable problems.

The SVM is constructed from the sums of a kernel function $K(\cdot, \cdot)$ [CCRS06]

$$f(x) = \sum_{i=1}^N \alpha_i t_i K(x, x_i) + d \quad (3.31)$$

where t_i are the ideal outputs, $\sum_{i=1}^N \alpha_i t_i = 0$, $\alpha_i > 0$, and x_i are the support vectors obtained by an optimization process of the training data.

Moreover, to make the algorithm work for non-linearly separable datasets and less sensitive to outliers, the *slack function* (ξ) and the *cost function* (C) are introduced to the optimization process [Wang16]. The ξ determine the error and C determine how strict the hyperplane should separate the test data. when $C = 0$, the SVM ignores the data and tries to find a hyperplane with the greatest margin, and when $C = \infty$, the SVM tries to find a hyperplane which will separate all the data [Wang16]. Figure 3.6, displays the separating hyperplane obtained by the SVM by

setting $C=0.1$ and Fig. 3.7, displays the separating hyperplane obtained by the SVM by setting $C=100$ for the same set of data. In both the images, the support vectors are circled, the separating hyperplane is a solid line, and the margin is the perpendicular distance from the solid line to the dotted line. Please note the outlier on the far left at about $(0.1,4.1)$. By setting $C = 0.1$ the outlier is ignored and the SVM tries to find a separating hyperplane with the greatest margin, whereas when setting $C=100$ the separate all the data.

There are different types of kernel functions used, such as linear kernel, polynomial kernel, sigmoidal kernel and the radial basis function kernel. Depending on the non-linearity of the kernel the can be used for different situations [Beig11] and in this thesis, the linear kernel is used. A Linear kernel may be used in situations where the data is linearly separable. It is the-

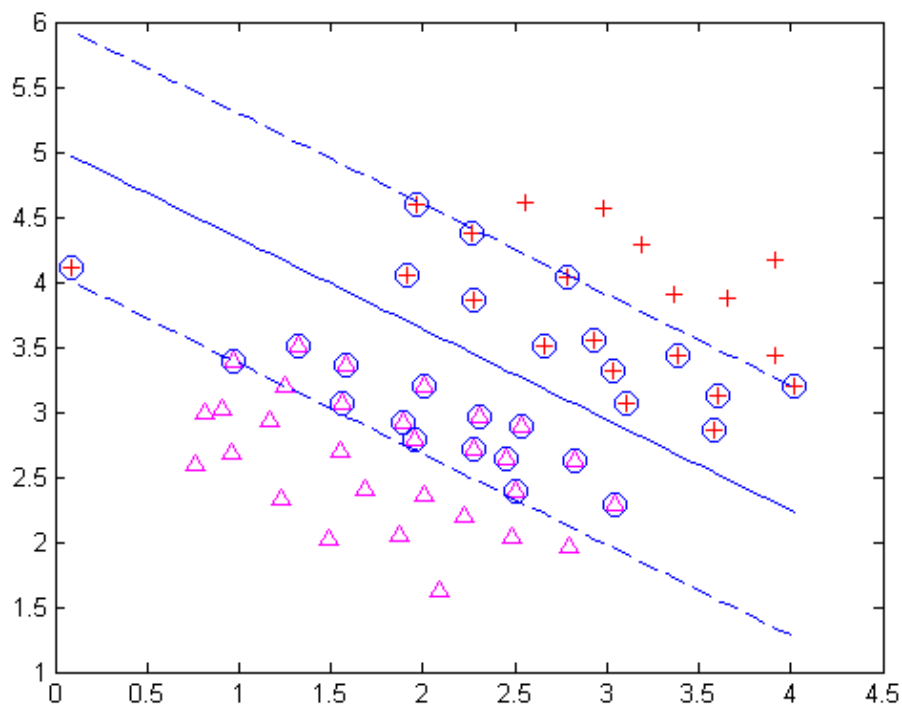


Fig. 3.6. The separating hyperplane obtained by setting $C=0.1$.

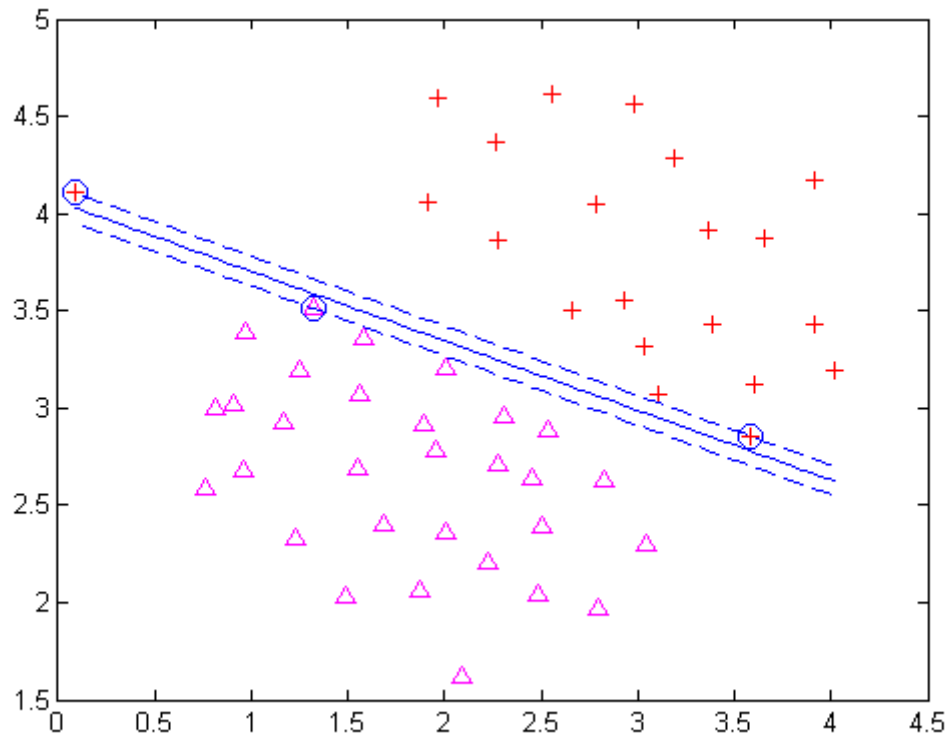


Fig. 3.7. The separating hyperplane obtained by setting $C=100$.

-inner product of the vectors in observation and is defined by [Wang16]

$$\mathbf{K}(x, x') = (x^T x') \quad (3.32)$$

There are highly optimized versions of SVM available, where LIBSVM [ChLi15] is used for this thesis. The SVM is used in section 5.5 to build a model from the training feature vectors and test the model using the testing feature vectors.

3.11 Summary

In this chapter, the theory and the programming aspects of the algorithms that will be used in chapter 5 for testing, VAD, feature extraction, and classification, was discussed. But before

proceeding to the experimental analysis, a set of utterances is required for any study on speaker verification. The next chapter provides the detailed description of the recording procedures to record the dataset that would be used for analysis in this thesis.

Chapter 4

A Manitoban Speech Dataset

Throughout the day, we are exposed to different levels of background noise depending on our surrounding environment. Hence, for the purpose of this thesis which is the robust feature extraction from the speech, all the recordings must be done with similar conditions in order to ensure the features extracted are not biased due to the surrounding conditions and are based on the physiological characteristics of the speaker. A dataset consisting of 24 participants was recorded. The recordings were conducted after obtaining approval from research ethics board at the University of Manitoba. The process of obtaining this approval is time-consuming and as a result, many researchers do not record a dataset. However, this process is very educational and is designed to teach and assure that there will be no harm to the participants during or after the session. The motivation behind the development of this dataset is driven by the need to obtain

the lowest level of background noise in the recordings and to know the details not available in other recordings. These details include the list of conditions under which the development of the dataset was conducted, the time at which the recordings were conducted, and the type of background noise in the record

This section starts with a short introduction of a few of the databases available followed by, describes the hardware and software requirements needed to ensure the quality and similarity of all the recordings. Moreover, a list of English phonemes chosen for this study and the demographics of the speakers are presented, followed by the environment and a set of protocols that would be followed to ensure the repeatability of all the recordings. Furthermore, the repository at which the recorded speech is stored is discussed.

4.1 Available Databases

There have been a considerable number of datasets produced for different fields of speech analysis. In this section, the TIMIT, KING, and YOHO databases are introduced since they are commonly used for speaker identification and speaker verification.

The TIMIT database consists of 630 speakers with 8 different English dialects from across the United States. There are 438 male and 192 female speakers that have been asked to read out 10 sentences. The recordings were conducted in a controlled environment using a sampling rate of 16 kSps. This database is not used because, it is not gender neutral, consists of 8 dialects so the features might be based on the way of speaking, and was recorded using a low sampling rate.

The KING database consists of 51 male speakers who were recorded using a combination of wideband microphone and telephone handsets. The KING dataset was recorded in a clean

environment in 10 sessions that were spread out over several weeks. This database is not used because it does not contain speech samples of female speakers, and some of the recordings were conducted using telephone handsets. The frequency bandwidth of telephone handsets is limited to 300 – 3300 Hz, which limits high frequency information which is important for speech intelligibility [KeTS92].

The YOHO database consists of 106 male speakers and 32 female speakers who were recorded in a clean office environment using a sampling rate of 3.8 kSps. The speakers have been recorded using prompted digits in 4 enrollment sessions and 10 verification sessions per speaker. This database is not used because of its low sampling rate, meaning due to the low bandwidth key information would be missed. Moreover, because the dataset contains only digits, this dataset is not very useful for real conditions especially for text-independent speaker verification [Beig11].

Although, these databases are commonly used for speaker identification and speaker verification they were not designed for this task. The dataset recorded in this work is designed for speaker verification using a sampling rate of 44.1 kSps to capture the independence of scale over many orders of magnitude.

4.2 Microphone

There are several characteristics of microphones that make them different from each other and as a result, there is no single microphone that is suitable for all conditions. This section provides a brief discussion of different types of microphones along with specifications that should be considered for a specific application.

Microphones generally utilize two major technologies, the *carbon button* type microphone, and the *electrets* type microphone. Carbon button microphones were used in telephones until the mid-1980 and are becoming harder to find these days [Beig11]. The carbon button microphone's technology is usually made of two metal plates with carbon granules in between them. The audio wave excitation causes the carbon granules to be compressed resulting in varying resistance. Direct current is passed through the metal plates and the varying resistance changes the flow, causing the production of the audio wave's electrical signal.

The electrets type microphone is a type of condenser microphone that uses a stable dielectric material. As a result, this type of microphone does not require any polarizing power. The electrets microphone comes in different forms such as the diaphragm, back electrets, front electrets and the latter.

4.2.1 Directionality

Microphones have different directionalities, which describe the microphone's sensitivity to audio from different directions. The manufacturers usually provide different directionalities, which can be suitable for different applications. They are often categorized into *omnidirectional* and *unidirectional* microphones. Omnidirectional microphones can capture audio from different directions, whereas unidirectional microphones can generally capture audio in the form of a cardioid around a microphone.

Omnidirectional microphones increase the possibility of intercepting noise and other speakers. Therefore, a cardioid microphone is more suitable for the purpose of recording audio in this thesis.

4.2.2 Frequency Response

The frequency response of a microphone provides the sensitivity of the microphone over a range of frequencies. Frequency response can be generalized into two types, *flat* frequency response, and *tailored* frequency response. Flat frequency response has the same output level over all audio frequencies. This is suitable for applications where the audio is recorded without any changing.

A tailored frequency response is designed to enhance audio for a particular application. For example, a microphone may have a peak in the 2 – 8 kHz range to increase intelligibility for live vocals.

4.2.3 Sensitivity

The sensitivity of a microphone states what voltage it would produce at a certain audio pressure. A microphone produces a high voltage output if it has a high sensitivity and therefore, it does not require as much gain as a microphone with a lower sensitivity.

4.2.4 Blue Yeti

The Yeti is an advanced and versatile USB microphone offered by Blue [Blue17], which features a triple capsule array. This microphone allows the recording of audio in four different directionality modes, including the cardioid mode, which is used in this thesis. Figure 4.1, shows the Yeti which is used for the recording of the participants.

The Yeti has a 16-bit analog to digital converter, which allows it to be connected directly to a computer via a USB port. Moreover, it has a number of built-in features, such as, a headphone

amplifier, simple controls for headphone volume, pattern selection and most importantly microphone gain that makes recording easier to control.

The frequency response of the Blue Yeti is in the range of 20 Hz to 20 kHz along with the sensitivity of 4.5mV/Pa. The cardioid directionality setting which is used in this thesis offers an almost flat frequency response, thus, making it suitable for the purpose of controlled recordings.

Figure 4.2, shows the frequency response of the Yeti in the cardioid directionality mode.

The Blue Yeti has received THX certification, which involves factors such as the frequency response, SNR and performance consistency. Therefore, this microphone is used along with an addition of a pop shield for the recording of audio signals for this thesis. A pop shield helps to block the burst of air caused by plosive words, which can cause a massive pressure change.



Fig. 4.1 The blue Yeti.

4.3 Software

The task of recording the participants for this thesis is not limited to just a microphone. A set of software is required in order to record the audio with high quality, edit the audio if required and store the audio without any compression. Moreover, software is used to measure the surrounding environment noise so that all the recordings are done in an environment with similar background noise.

4.3.1 Camtasia

The software *Camtasia* is software used for creating video tutorials and presentations via the chosen parts of the screen along with recorded audio at the same time. After recording, Camtasia will import the recordings and provides the option to edit them separately. Camtasia can be used to record high-quality audio as well. All the utterances are recorded using the software Camtasia V.8.6 [Tech17], using a sampling frequency of 44.1 kSps and stored in WAV format. Figure 4.3, displays the utterance test recorded using the software Camtasia.

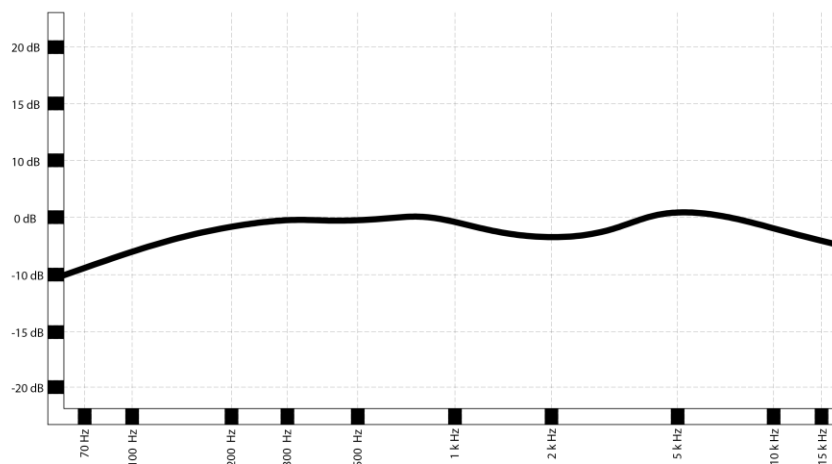


Fig. 4.2 The frequency response of the Yeti in cardioid directionality mode. (After [Blue16])

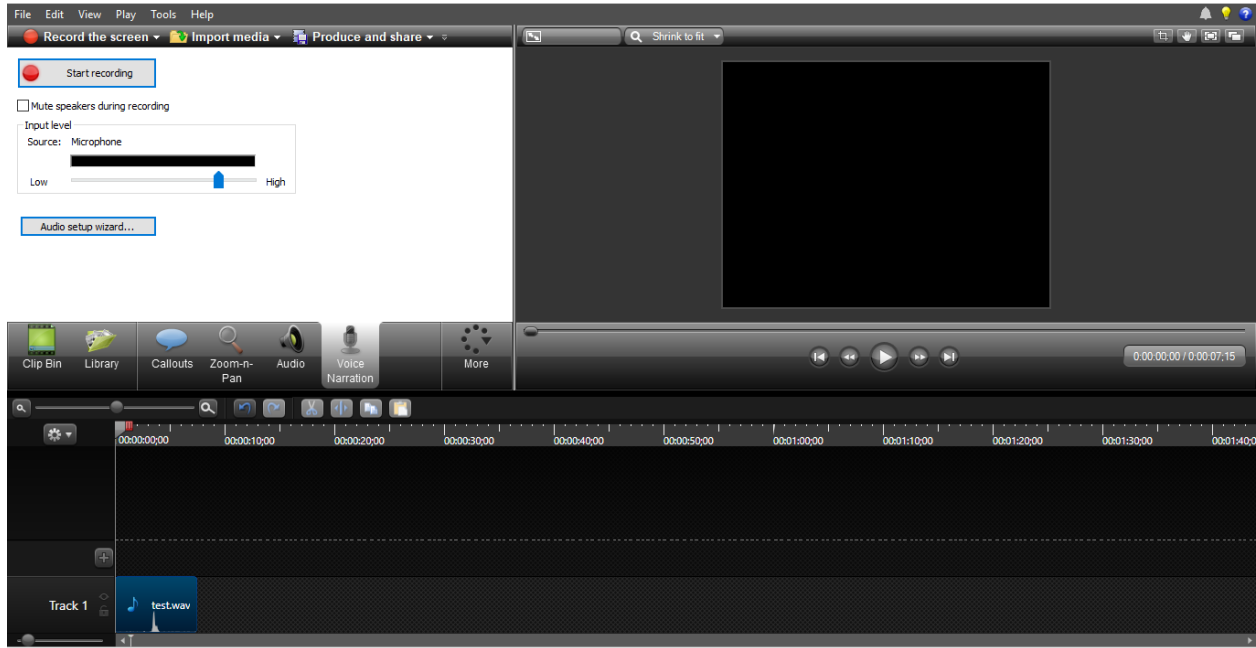


Fig. 4.3 The utterance test recorded using Camtasia.

4.3.2 Audacity

The software *Audacity* is a popular open source audio editor and recorder that is both incredibly powerful and versatile. It can be used to record audio, edit, and mix if required, and store the audio in a number of different formats including *waveform audio file format* (WAV), which is uncompressed for data collection. The software Audacity V.2.1.2 [Auda17] is used to trim the recorded utterances into epoch's of 2 sec with the utterance in the center of the epoch and silence before and after the utterance. Figure 4.4, displays the utterance church trimmed to a 2 sec epoch using the software Audacity.

4.3.3 Decibel X

The software *Decibel X* V.6.0.1 is a Smartphone app that measures the sound pressure level using the Smartphone's microphone and displays it in decibels [Skyp17]. It is used to measure

the background noise in the recording environment in order to ensure the similarity of recording conditions for all recordings. The background noise recorded in the chamber during the recordings was between 35 dB to 45 dB. Figure 4.5, displays a screenshot of the app while measuring the background noise in the recording environment.

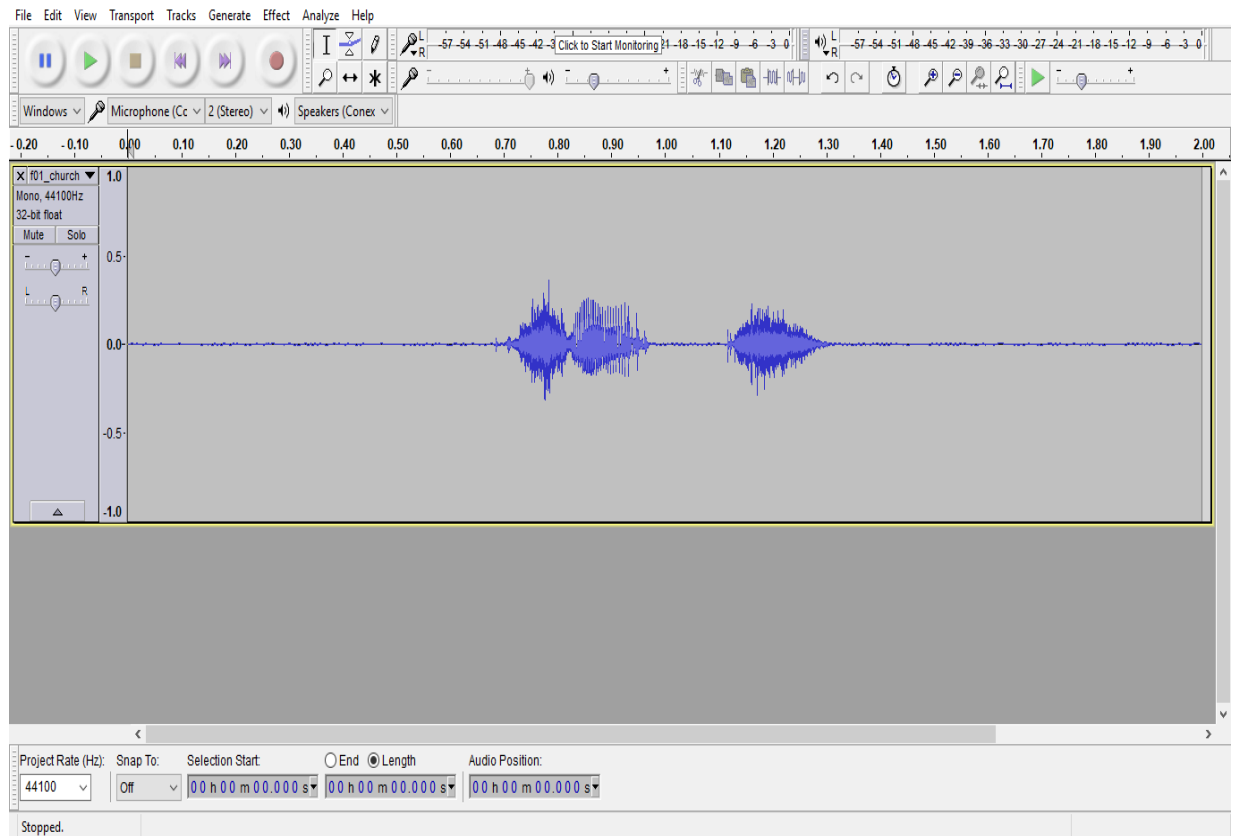


Fig. 4.4. The utterance “church” trimmed to a 2 sec epoch using Audacity.

4.4 English Phonemes

Phonemes are members of the smallest unit of speech that distinguish different words from each other. Consonants and vowels are two categories of phonemes. Consonants are produced when the airflow from the lungs is obstructed in the middle of the vocal and when this obstruction does not occur vowels are produced.

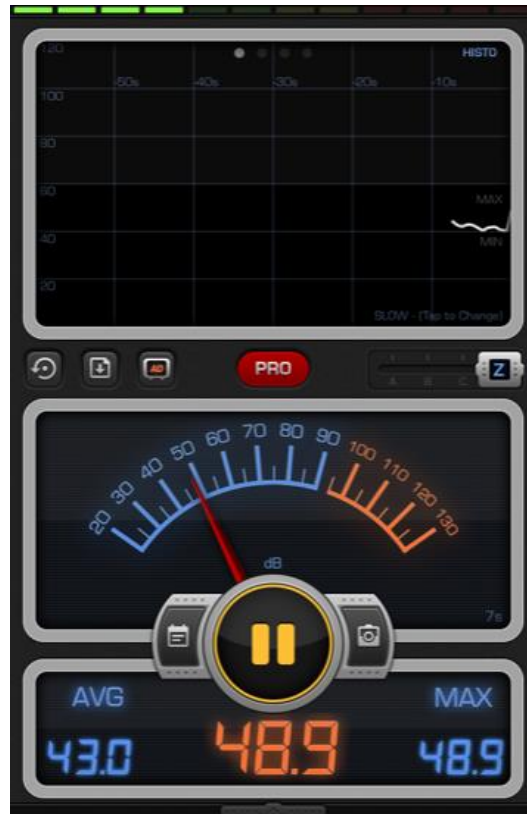


Fig. 4.5. A screenshot of Decibel X.

In order to provide a repeatable set of test words that would cover all of the phonemes, the 44 keywords from the Edinburg MRPA shown in table 2.1 is used and recorded for each participant in a recording session. The choice of MRPA over the *International Phonetic Alphabet* (IPA) is due to the IPA not being machine readable [KiGr08]. The MRPA covers all the English phonemes, has enough speech data to build a model and at the same time, quick to record making it practical to use for a real-world application.

4.5 Demographics

The volunteers participated in this study consists of 12 male and 12 female participants, all raised in the Province of Manitoba. The volunteers being from a specific geographical location

can limit the variety of accents and thus, the features extracted will not be biased by the forms of speaking. The male volunteers aged between 19 to 60 years and the female volunteers between 18 to 44 years. Figure 4.6, displays a histogram of the number of participant against their age. Please note that the distribution of the participants is due to the majority of the participants in this study being students and researchers at the University of Manitoba. All the recordings were conducted in one continuous session, each approximately 15 to 20 minutes in duration. The recording sessions took place between 10 AM to 3 PM, from March 27, 2017, until September 27, 2017. The 44 utterances recorded for each participant are stored in a file named with the alphanumeric number of each participant, along with a recording of the silence before and after the recordings. Moreover, a file named “Readme” is included, which contains the age range of the participants, as well as the date, time, weather, humidity, and pressure during the recording session.

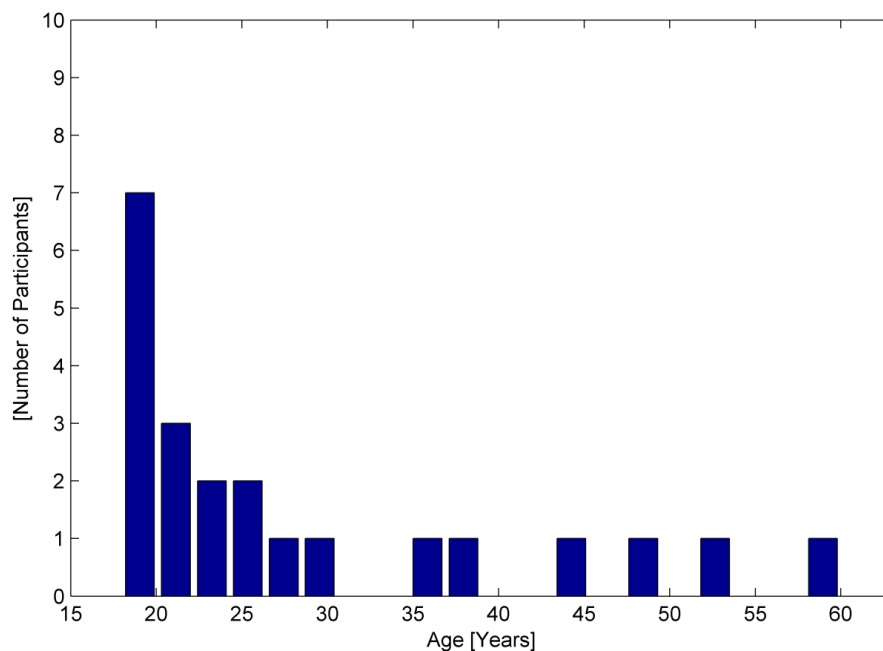


Fig. 4.6. The Histogram of the participants.

4.6 Environment

The environment in which recordings are done plays an important role in the overall quality of the recordings. In general, our environment is surrounded by background noise. A quiet bedroom at night can have 40-50dB of noise. Another factor is the reverberation of noise. A reverberation occurs when a sound wave hits obstacles or a wall and is reflected.

In order to minimize the background noise and the effects of reverberation, all the participants are recorded at the University of Manitoba's applied electromagnetic laboratory's anechoic chamber shown in Fig. 4.7. An anechoic chamber is a room that completely absorbs reflections of sound. The anechoic chamber used has a length of 6.4m, a width of 2.4m and a height of 2.3m. The interior walls of the anechoic chamber are composed of wedges that are made of radiation absorbent material. Radiation absorbent material is designed and shaped to absorb radio frequency radiation from as many directions in the most effective way possible. The wedges on the walls are 305mm long and the wedges on the ceiling are 153mm long.



Fig. 4.7. The interior of the anechoic chamber used.

4.7 Recording Protocols

The setup of the equipment plays an important role in the overall quality and the repeatability of all the recordings. The following describes the proposed setup for all the recordings:

- The microphone is placed approximately 6 inches from the speaker's mouth.
- The Blue Yeti is a side-address microphone, therefore, it is placed in an upright position, perpendicular to the speaker. Figure 4.8, displays the Blue Yeti in a side-address position on the left and in front-address position on the right.

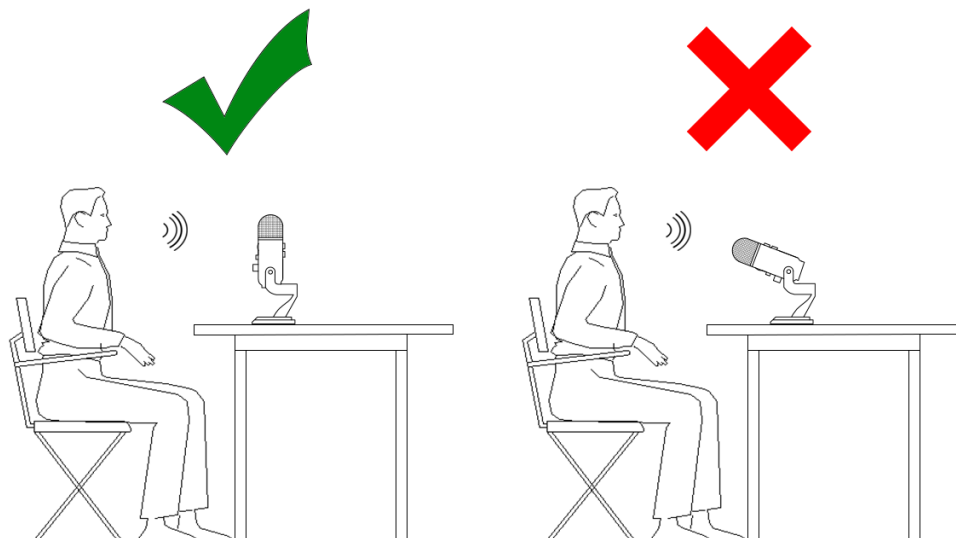


Fig. 4.8. The Blue Yeti in a side-address position on the left. (After [Blue17])

- The microphone is set to the cardioid mode and the gain level is set to -9 dB, as displayed in fig. 4.9. The input level in Cantasia is set to 80%, as displayed in fig. 4.10.

- The Apex 6 inch dual screen nylon pop-shield filter [Apex17] is placed in front of the microphone to block the burst of air when the speakers utter plosive words. Figure 4.11, displays the setup of the Blue Yeti and the placement of the pop-shield filter in front of it.
- Few seconds of silence is recorded before and after the recordings. The average duration of the recordings is 6 s and 260 ms. The minimum duration of a recording is 4 s and 250 ms and the maximum duration of a recording is 10 s and 750 ms. Please, note that all recordings are trimmed to 2 s.



Fig. 4.9. The Blue Yeti in cardioid mode and with gain level of -9 dB.

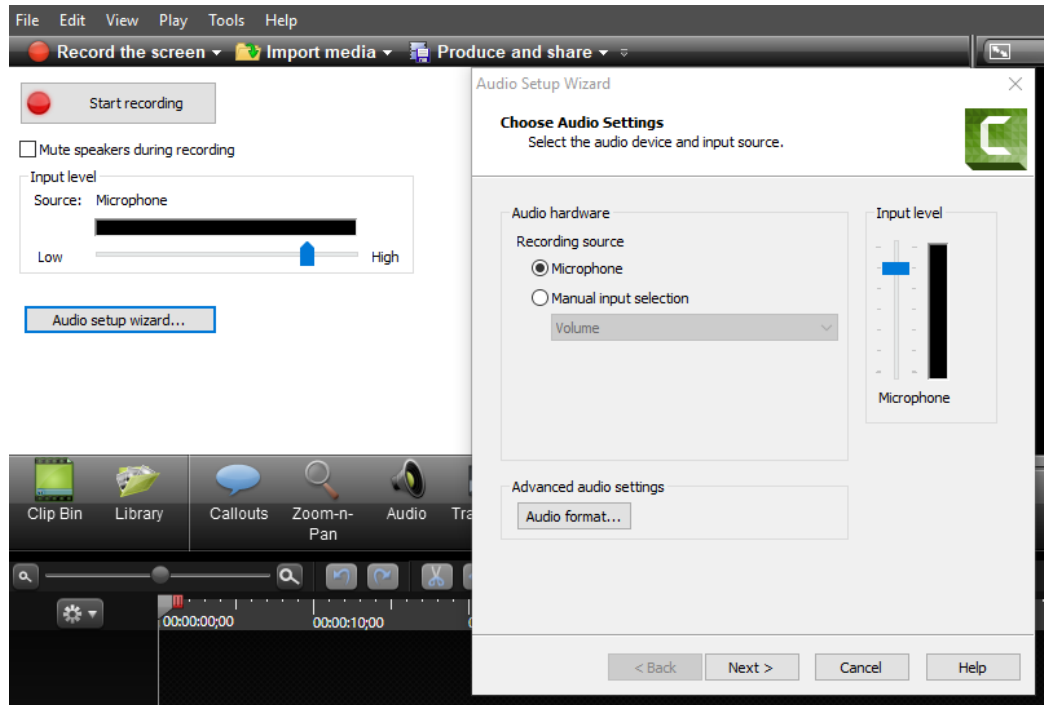


Fig. 4.10. The input level in Camtasia is set to 80%.

4.8 Repository

The repository chosen to store all the recorded utterances is the IEEE dataport [IEEE17]. IEEE dataport allows the access to datasets and data analysis tools. It has the capability to accept different varieties of datasets up to 2 Tb, provides downloading capabilities and access to cloud services to enable data analysis in the cloud.

IEEE dataport offers three options for uploading a dataset. The first option is called the ‘standard dataset’. This option allows the storage of the dataset and related files at no cost and assigns a *digital object identifier* (DOI) to the dataset. Along with the uploading of the dataset files, the author is required to provide the metadata for citation of the authors, abstract and image of the dataset, instructions on how to use the dataset and an optional analysis of the dataset. The datasets uploaded using the standard dataset option will be available for downloading or access

in the cloud by IEEE paid subscribers only.



Fig. 4.11. The setup of the Blue Yeti with a pop-shield filter placed in front of it.

The second option is called the ‘open access dataset’. This option allows the storage of the dataset with a onetime fee of 1,950\$ and assigns a DOI to the dataset. Just like the previous option, the author is required to provide the metadata for citation of the authors, abstract and image of the dataset, and optional instructions on how to use the dataset, along with uploading of the dataset files. The datasets uploaded using the open access dataset option, as its name suggests, will be accessible to all logged in dataport users.

The third option is called the ‘data competition’. This option allows the uploading of the dataset and instructions at no cost and assigns a *digital object identifier* (DOI) to the data competition. The administrator of the competition is required to provide the metadata for citation of the authors, abstract and image of the dataset, and optional instructions on how to use the dataset, along with uploading of the dataset files. The administrator is enabled to establish the competition duration, manage participation and update the competition as needed.

Although authors uploading dataset through any of the options described above to IEEE dataport have the option to provide an analysis of the dataset, they are not required to provide a detailed description of the recording/collecting procedures of the data. Availability of detailed description of the recording/collecting procedures of the data can provide the user the knowledge to repeat the recording/collecting of data with similar quality. Moreover, such description allows the user to know if the data is suitable for their analysis before downloading the data. Therefore, this chapter provided a detailed description of the recording procedures and protocols used to record the dataset. The recorded dataset is uploaded to IEEE dataport using the ‘standard dataset’ option and can be found at [SeKi18].

4.9 Summary

This chapter provided a detailed description of the procedures used to record a dataset. The following dataset consists of utterances, recorded using 24 volunteers raised in the Province of Manitoba, Canada. To provide a repeatable set of test words that would cover all of the phonemes, the Edinburg MRPA, consisting of 44 words is used. Each recording consists of one word uttered by the volunteer and recorded in one continuous session. All the recordings are conducted in an anechoic chamber, available at the Applied Electromagnetic Laboratory at the

University of Manitoba, using a Blue Yeti microphone, with a sampling frequency of 44.1 kSps. All the recordings are stored in WAV, which is uncompressed and could be easily loaded into software programs like Matlab or Audacity for analysis. Each participant is numbered alphanumerically, male participants starting with M and female participants starting with F. The volunteers participated in this study consists of 12 male and 12 female participants, all raised in the Province of Manitoba. The male volunteers aged between 19 to 60 years and the female volunteers between 18 to 44 years. All the recordings were conducted in one continuous session, each approximately 15 to 20 minutes in duration. The recording sessions took place between 10 AM to 3 PM, from March 27, 2017, until September 27, 2017. The 44 utterances recorded for each participant are stored in a file named with the alphanumeric number of each participant, along with a recording of the silence before and after the recordings. Moreover, a file named “Readme” is included, which contains the age range of the participants, as well as the date, time, weather, humidity, and pressure during the recording session. The next chapter presents the experimental results and analysis.

Chapter 5

Design of Experiment and Result Analysis

This chapter presents the experiments conducted and the analysis of the results. At first, the test data is used to test the suitability of using the VFD and the HFD for speech analysis. Then the effect of noise on the estimation of FD is studied by addition of colored noise to the test data. Furthermore, a VAD detection algorithm that utilizes the FD characteristics of the background noise to detect speech segments is introduced. Moreover, the feature vectors that will be used to conduct an experimental sensitivity analysis are discussed. Subsequently, the feature vectors are extracted using different VAD and used to train and test the SVM.

5.1 Comparison of Fractal Dimension Estimation Algorithm

ˆ In order to test the performance of the VFD and HFD, test data is generated by methods that

produce known fractal dimensions. Two of these methods that were discussed in chapter 3 are the Weierstrass function and the fBm.

5.1.1 Weierstrass function

The Weierstrass function is generated using 512 samples to resemble the same number of samples used in 1 frame of speech in this work. Assigning $\lambda = 2$ and using nine H values spaced equally from 0.1 to 0.9, nine Weierstrass functions with a FD of 1.1 to 1.9 are produced. The VFD and the HFD are used to estimate the FD of these waveforms and the results are shown in table 5.1. Figure 5.1 displays the graphical representation of these results.

Table 5.1: Fractal dimension estimation of 512 samples of Weierstrass function.

Hurst Exponent	Theoretical Fractal Dimension	Variance Fractal Dimension	Higuchi Fractal Dimension
0.9	1.10	1.17	1.16
0.8	1.20	1.24	1.23
0.7	1.30	1.32	1.32
0.6	1.40	1.41	1.41
0.5	1.50	1.50	1.50
0.4	1.60	1.60	1.59
0.3	1.70	1.70	1.68
0.2	1.80	1.80	1.76
0.1	1.90	1.90	1.83

The results indicate that both the VFD and HFD are overestimating the FD of the Weierstrass function when H is set to 0.8 and 0.9. However, this estimation error is reduced when H is set to 0.4, 0.5, 0.6, and 0.7, as the estimated FD's are equal to the theoretical FD or have a slight error. The main difference in performance between the VFD and the HFD for this test occurs when H is set to 0.1, 0.2, and 0.3. The FD estimated using the VFD is equal to the theoretical FD, while the HFD underestimates the FD, with a greater error towards the FD of 1.9. Thus, even though the VFD, compared to the HFD, is slightly overestimating the FD of the Weierstrass function when H is set to 0.8 and 0.9, it is providing a more accurate result for Weierstrass functions with FD of 1.5 to 1.9.

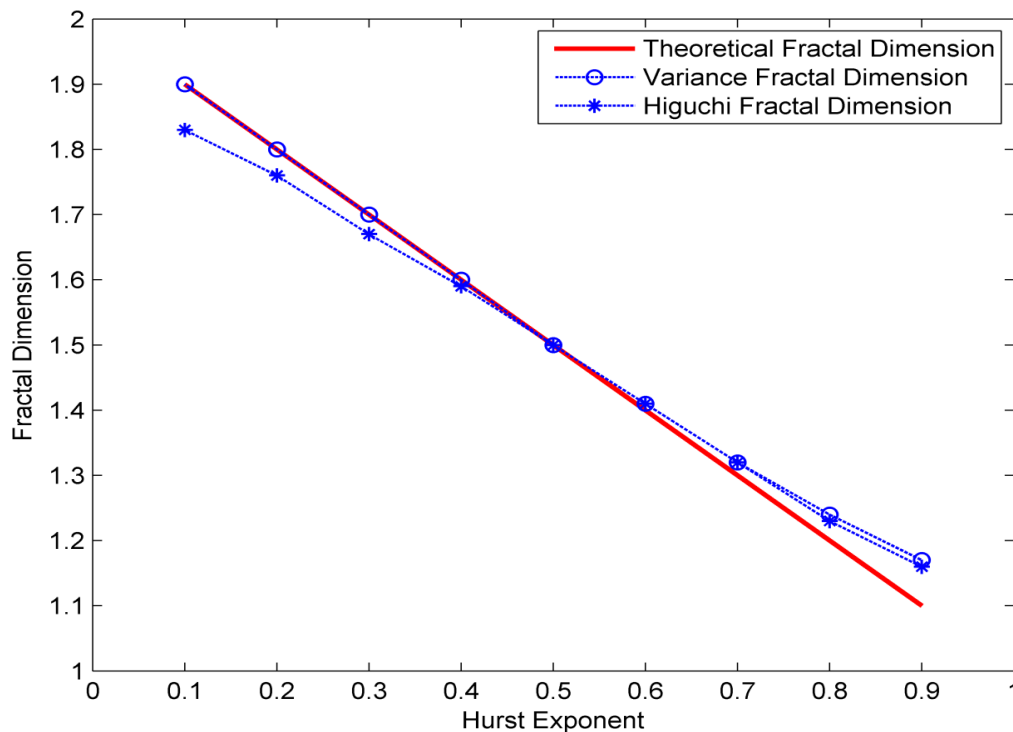


Fig. 5.1. Graph of fractal dimension values of 512 samples of the Weierstrass function.

Moreover, the Weierstrass function is generated using 88200 samples to resemble the same number of samples in one recording. As in the previous test, $\lambda = 2$ and nine equally spaced H

values from 0.1 to 0.9 are used to generate nine Weierstrass functions with a FD of 1.1 to 1.9. The Weierstrass function is then framed into frames of 512 samples to replicate the framing method used for the analysis of speech in this work. The VFD and HFD are used to estimate FD of the frames which leads to a trajectory of FD. The mean and variance of this trajectory are calculated and the results are displayed in table 5.2. The objective of this test is to measure the performance of the VFD and HFD on signals that are continuous. Please note that the Weierstrass function is self-similar and thus, in theory, all the frames must have the same FD or a trajectory, which is a straight line against the respective FD.

Table 5.2: Fractal dimension estimation of 88200 samples of the Weierstrass function

Hurst Exponent	Theoretical Fractal Dimension	Variance Fractal Dimension Trajectory		Higuchi Fractal Dimension Trajectory	
		Mean	Variance	Mean	Variance
0.9	1.10	1.126	0.0011	1.121	0.0014
0.8	1.20	1.208	0.0007	1.205	0.0008
0.7	1.30	1.302	0.0003	1.300	0.0004
0.6	1.40	1.400	0.0001	1.399	0.0002
0.5	1.50	1.500	0.0001	1.497	0.0001
0.4	1.60	1.600	0.0001	1.594	0.0001
0.3	1.70	1.700	0.0002	1.687	0.0002
0.2	1.80	1.798	0.0004	1.777	0.0003
0.1	1.90	1.891	0.0008	1.862	0.0004

The results indicate that when H is set to 0.9 the mean of the trajectories of the VFD and the HFD are slightly higher than the theoretical FD. However, when H is set to 0.4, 0.5, 0.6, 0.7, and 0.8, the mean of the trajectory of both the algorithms is close to the theoretical values. Moreover, as the results in the estimation of the FD for 512 samples of the Weierstrass function, the mean of the trajectory of the HFD tends to underestimate the FD, with a greater error towards the FD of 1.9, whereas, the mean of the trajectory of the VFD is close to the theoretical FD. Meanwhile, the variance of the trajectories of both the algorithm is the lowest when H is set to 0.5 and increasing as H is going away from 0.5. The variance of the VFD is lower than the HFD with the exception of H being set to 0.1 and 0.2. Thus, the results indicate that for the majority of the tests using the Weierstrass function, although slightly, the VFD provides a closer estimate of the FD to the theoretical FD with less variability. Figure 5.2 displays 88200 samples of the Weierstrass function generated by setting H to 0.4, while, figure 5.3 displays the VFDT and figure 5.4 displays the HFDT of this waveform.

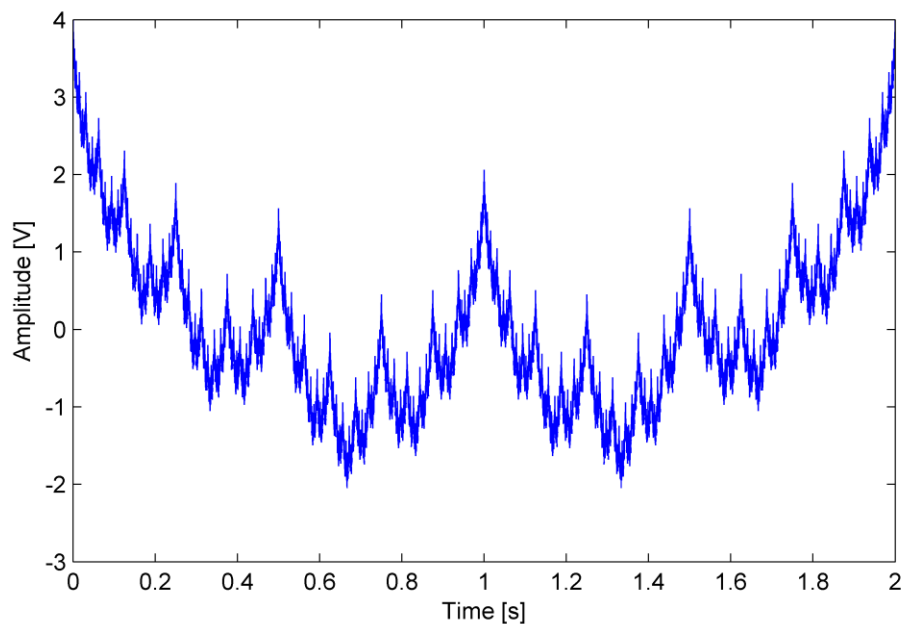


Fig. 5.2. The Weierstrass function generated using Hurst value of 0.4.

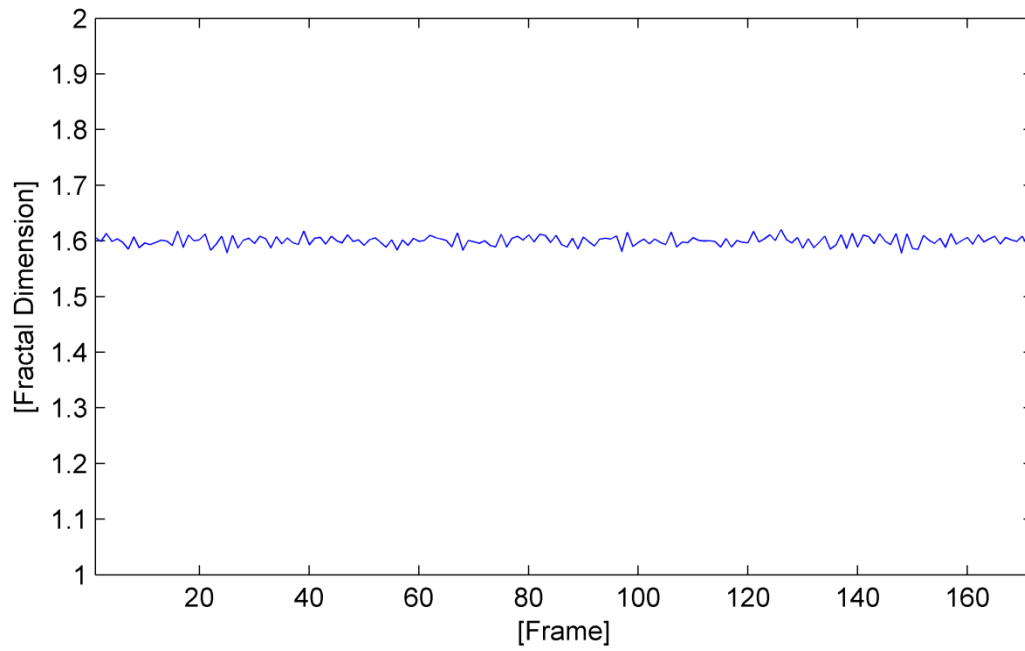


Fig. 5.3. The variance fractal dimension trajectory of the Weierstrass function.

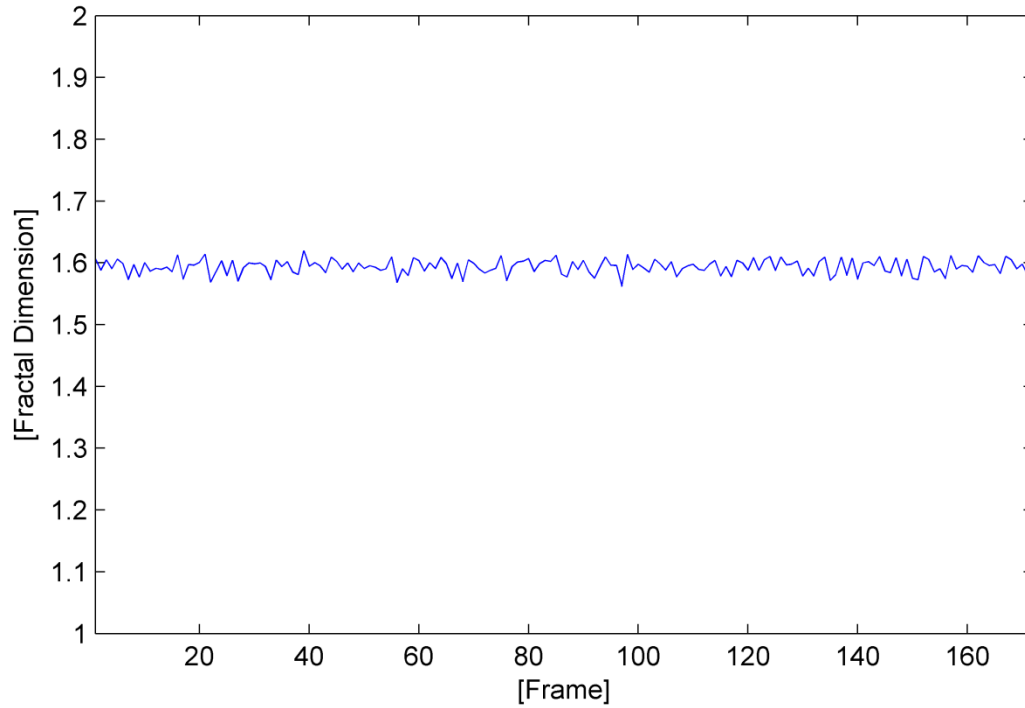


Fig. 5.4. The Higuchi fractal dimension trajectory of the Weierstrass function.

5.1.2 Fractional Brownian motion

Like the Weierstrass function, 512 samples of the fBm is generated using nine H values spaced equally from 0.1 to 0.9 to produce nine fBm with a FD of 1.1 to 1.9. Moreover, since the generation of the fBm requires random numbers which will affect the estimation of the FD, the fBm for each H value is generated by setting the seed of the random number generator to 1, 10, 30, 50, and 100. This will allow for the reproduction of the same fBm waveform. The VFD and the HFD are used to estimate the FD of these waveforms. Table 5.3, displays the results obtained using the VFD and table 5.4, displays the results obtained using HFD.

Table 5.3: The variance fractal dimension estimation of the fractional Brownian motion.

Hurst Exponent	Theoretical Fractal Dimension	Variance Fractal Dimension				
		Seed 1	Seed 10	Seed 30	Seed 50	Seed 100
0.9	1.10	1.05	1.12	1.06	1.22	1.12
0.8	1.20	1.11	1.20	1.13	1.28	1.22
0.7	1.30	1.21	1.29	1.23	1.35	1.32
0.6	1.40	1.34	1.39	1.35	1.43	1.43
0.5	1.50	1.47	1.49	1.47	1.52	1.53
0.4	1.60	1.58	1.58	1.58	1.60	1.63
0.3	1.70	1.68	1.67	1.68	1.69	1.71
0.2	1.80	1.77	1.76	1.77	1.78	1.80
0.1	1.90	1.87	1.86	1.87	1.87	1.89

Table 5.4: The Higuchi fractal dimension estimation of the fractional Brownian motion.

Hurst Exponent	Theoretical Fractal Dimension	Higuchi Fractal Dimension				
		Seed 1	Seed 10	Seed 30	Seed 50	Seed 100
0.9	1.10	1.03	1.11	1.02	1.21	1.12
0.8	1.20	1.10	1.19	1.11	1.28	1.24
0.7	1.30	1.20	1.29	1.24	1.36	1.34
0.6	1.40	1.33	1.40	1.36	1.45	1.42
0.5	1.50	1.45	1.49	1.47	1.53	1.53
0.4	1.60	1.57	1.59	1.58	1.60	1.62
0.3	1.70	1.68	1.68	1.68	1.69	1.71
0.2	1.80	1.78	1.77	1.77	1.78	1.80
0.1	1.90	1.89	1.87	1.87	1.88	1.90

The results obtained when the seed of the random number generator is set to 1, shows that both the VFD and the HFD are underestimating the FD of the fBm waveforms, with this underestimation being greater for the FD of 1.1, 1.2, 1.3, and 1.4. As the FD increase, this underestimation is reduced and the estimated FD has fewer errors. The VFD is providing a slightly more accurate FD estimation for the FD of 1.1 to 1.6, the HFD is providing a slightly more accurate FD estimation for the FD of 1.8 and 1.9, and both the VFD and the HFD having the same estimation for the FD of 1.7. When the seed of the random number generator is set to 10, both the VFD and HFD provide a close estimation for the FD of 1.1 to 1.6. The VFD and the HFD tend to start underestimating the FD of 1.7 to 1.9 with this underestimation being slightly

more in the VFD. When the seed of the random number generator is set to 30, both the VFD and HFD are underestimating the FD with this underestimation being reduced for the FD of 1.5 to 1.9. When the seed of the random number generator is set to 50, both the VFD and HFD are estimating the FD of 1.6 to 1.9 correctly or with a slight underestimation and the FD of 1.1 to 1.5 with overestimation, with the overestimation being greater towards the FD of 1.1. When the seed of the random number generator is set to 100, both the VFD and HFD are slightly overestimating the FD of the fBm. Figure 5.5, displays the graphical representation of the results obtained using the VFD and Fig. 5.6 displays the graphical representation of the results obtained using the HFD for the mentioned above H and random number generator seed values. Please note the seed is displayed in logarithmic scale in the graph.

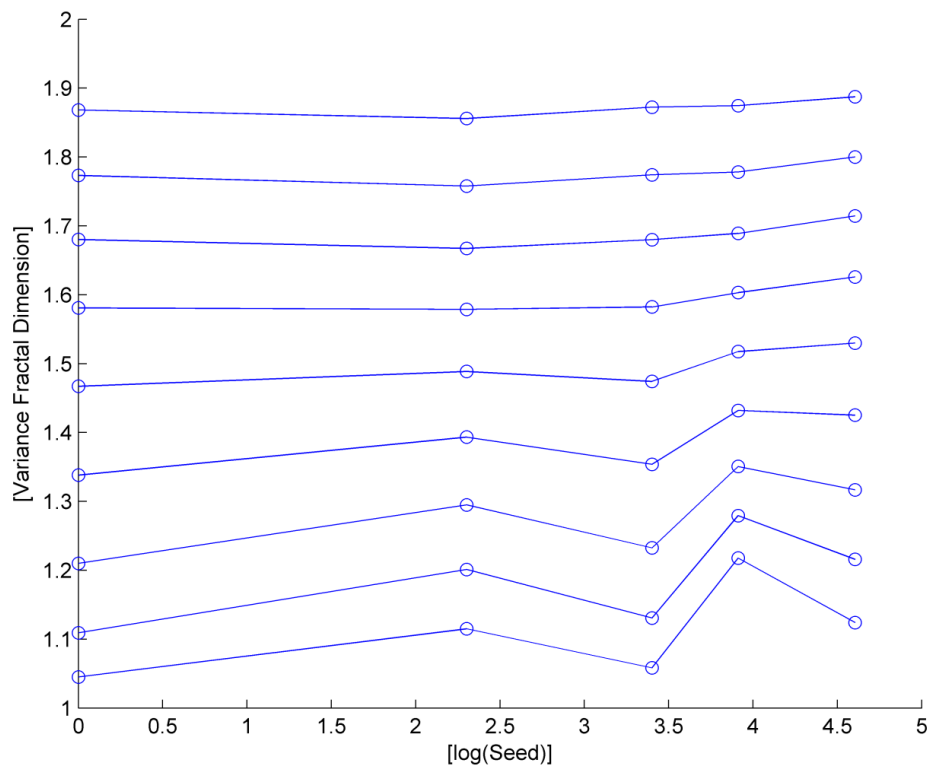


Fig. 5.5. The variance fractal dimension of the fractional Brownian motion

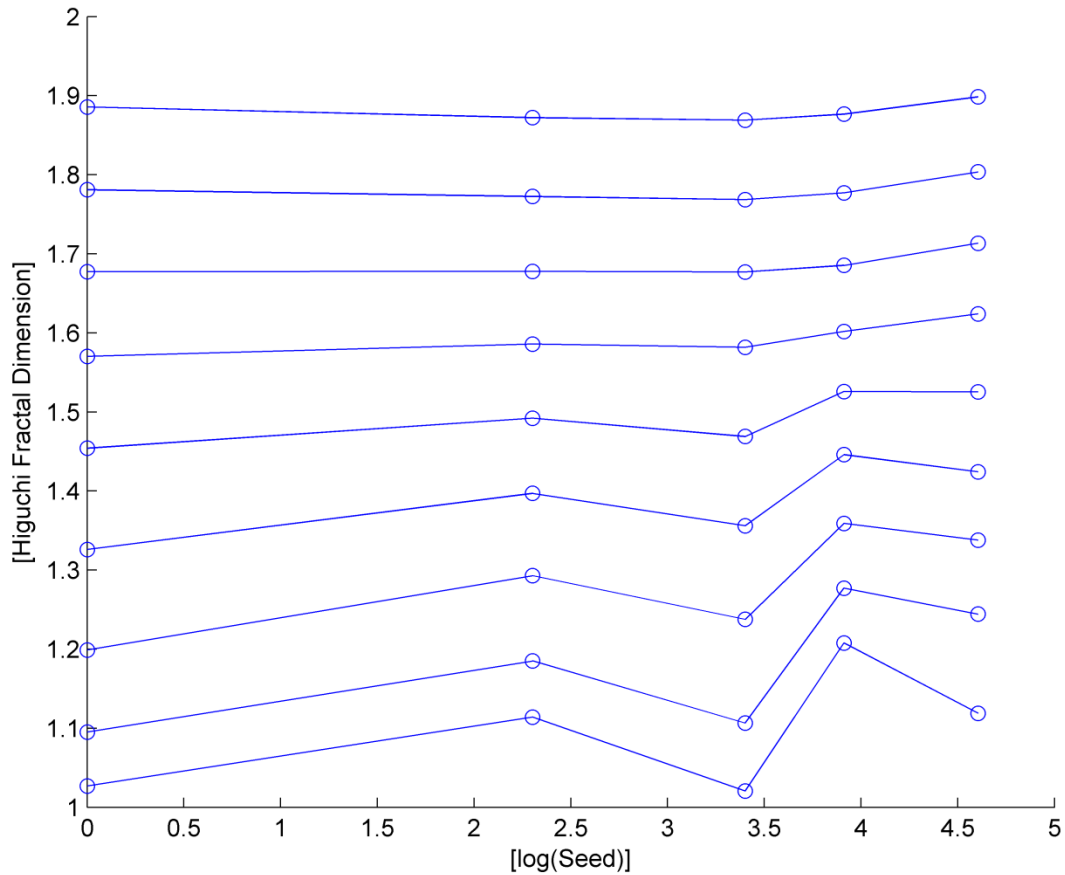


Fig. 5.6. The Higuchi fractal dimension of the fractional Brownian motion.

Moreover, like the Weierstrass function, 88200 samples of the fBm is generated for the mentioned above H and seed values. The VFD and the HFD of frames of 512 samples are estimated and the mean and variance of the trajectory are calculated and displayed in table 5.5 and 5.6. Please note that just like the Weierstrass function, in theory, all the frames of the fBm must have the same FD or a trajectory, which is a straight line against the respective FD. Figure 5.7, displays 88200 samples of fBm generated by setting the H to 0.5 and the seed to 10 and fig. 5.8 and fig. 5.9 display the variance fractal dimension trajectory and the Higuchi fractal dimension trajectory of the fBm waveform respectively.

Table 5.5: The variance fractal dimension trajectory of the fractional Brownian motion.

Hurst Exponent	Theoretical Fractal Dimension		Variance Fractal Dimension				
			Seed 1	Seed 10	Seed 30	Seed 50	Seed 100
0.9	1.1	Mean	1.06	1.11	1.09	1.14	1.11
		Variance	0.0009	0.0025	0.0025	0.0025	0.0025
0.8	1.2	Mean	1.19	1.21	1.20	1.22	1.21
		Variance	0.0036	0.0025	0.0025	0.0025	0.0025
0.7	1.3	Mean	1.30	1.31	1.30	1.31	1.30
		Variance	0.0025	0.0016	0.0025	0.0025	0.0016
0.6	1.4	Mean	1.40	1.41	1.40	1.41	1.40
		Variance	0.0016	0.0016	0.0016	0.0016	0.0016
0.5	1.5	Mean	1.50	1.51	1.50	1.51	1.50
		Variance	0.0016	0.0009	0.0016	0.0016	0.0016
0.4	1.6	Mean	1.60	1.60	1.60	1.60	1.60
		Variance	0.0016	0.0016	0.0016	0.0016	0.0009
0.3	1.7	Mean	1.70	1.70	1.70	1.70	1.70
		Variance	0.0016	0.0016	0.0016	0.0016	0.0016
0.2	1.8	Mean	1.80	1.80	1.80	1.80	1.80
		Variance	0.0016	0.0016	0.0016	0.0016	0.0009
0.1	1.9	Mean	1.90	1.90	1.90	1.90	1.90
		Variance	0.0009	0.0009	0.0009	0.0009	0.0009

Table 5.6: The Higuchi fractal dimension trajectory of the fractional Brownian motion.

Hurst Exponent	Theoretical Fractal Dimension		Higuchi Fractal Dimension				
			Seed 1	Seed 10	Seed 30	Seed 50	Seed 100
0.9	1.1	Mean	1.04	1.10	1.07	1.13	1.11
		Variance	0.0016	0.0036	0.0036	0.0036	0.0036
0.8	1.2	Mean	1.18	1.20	1.19	1.22	1.20
		Variance	0.0049	0.0036	0.0036	0.0036	0.0036
0.7	1.3	Mean	1.30	1.30	1.30	1.31	1.30
		Variance	0.0036	0.0025	0.0036	0.0025	0.0025
0.6	1.4	Mean	1.40	1.40	1.40	1.41	1.40
		Variance	0.0025	0.0016	0.0025	0.0025	0.0025
0.5	1.5	Mean	1.50	1.50	1.50	1.51	1.50
		Variance	0.0016	0.0016	0.0016	0.0016	0.0016
0.4	1.6	Mean	1.60	1.60	1.60	1.60	1.60
		Variance	0.0016	0.0016	0.0016	0.0016	0.0016
0.3	1.7	Mean	1.70	1.70	1.70	1.70	1.70
		Variance	0.0016	0.0016	0.0016	0.0016	0.0016
0.2	1.8	Mean	1.80	1.80	1.80	1.80	1.80
		Variance	0.0016	0.0016	0.0016	0.0016	0.0009
0.1	1.9	Mean	1.90	1.90	1.90	1.90	1.90
		Variance	0.0009	0.0009	0.0009	0.0009	0.0009

The results show that when H is set to 0.1 to 0.4, the mean of the trajectories of both the VFD and the HFD is the same as the theoretical values. Meanwhile, when H is set to 0.5 to 0.7 the mean of the trajectories of both the VFD and the HFD is the same as the theoretical values, with a slight overestimation when the seed of the random number generator is set to 50 and in case of the VFD when the seed is set to 10. Moreover, when H is set to 0.8 there is slight underestimation for the mean of trajectories of both the VFD and HFD for the seed value of 1 and a slight overestimation for the seed value of 50. However, when H is set to 0.9 the errors start to increase. When the seed is set to 1 and 30, the mean of the trajectory of both the VFD and the HFD underestimate the FD with this underestimation being greater with HFD, while, when the seed is set to 50 both the algorithms are overestimating with this overestimation being greater for the VFD. On the other hand, the variance of the trajectory of both the VFD and the HFD tends to be the highest when H is set to 0.9 and decrease gradually towards the H of 0.1 with the exception of the fBm generated using a seed value of 1 and H set to 0.9. Moreover, the results in this section show that the variance of the trajectory of the VFD is lower or equal to the variance of the trajectory of the HFD in all cases.

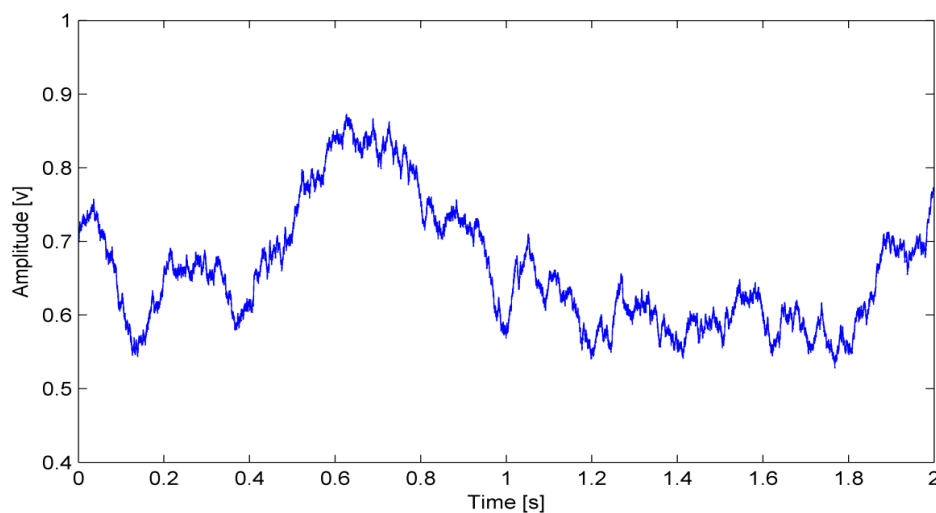


Fig. 5.7. The fractional Brownian motion generated using $H = 0.5$ and the seed set to 10.

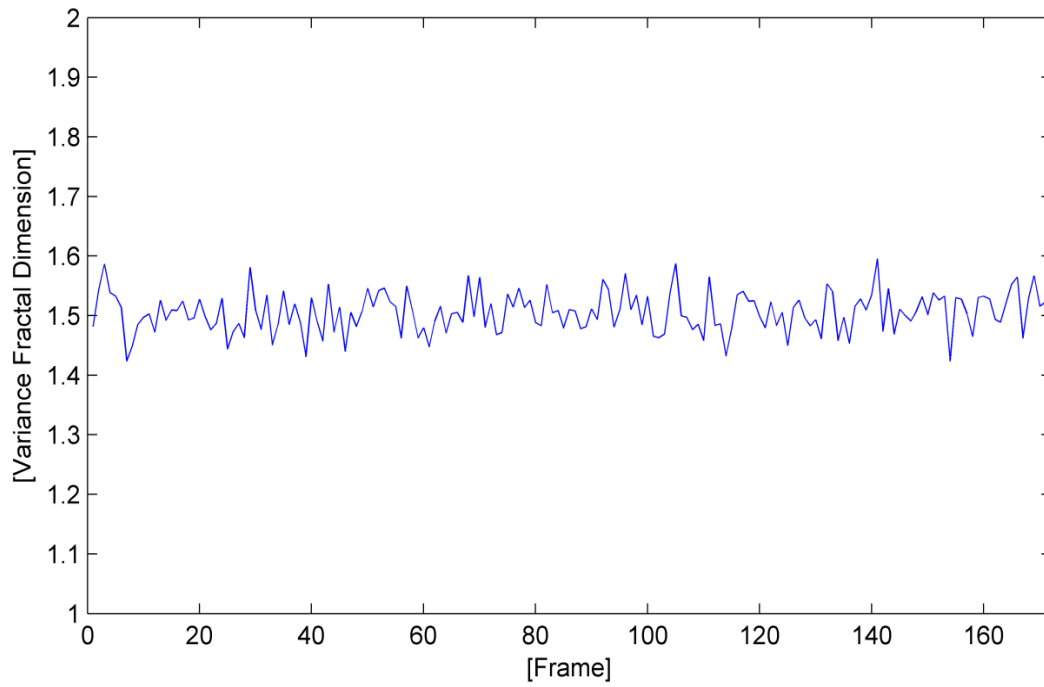


Fig. 5.8. The variance fractal dimension trajectory of the fractional Brownian motion.

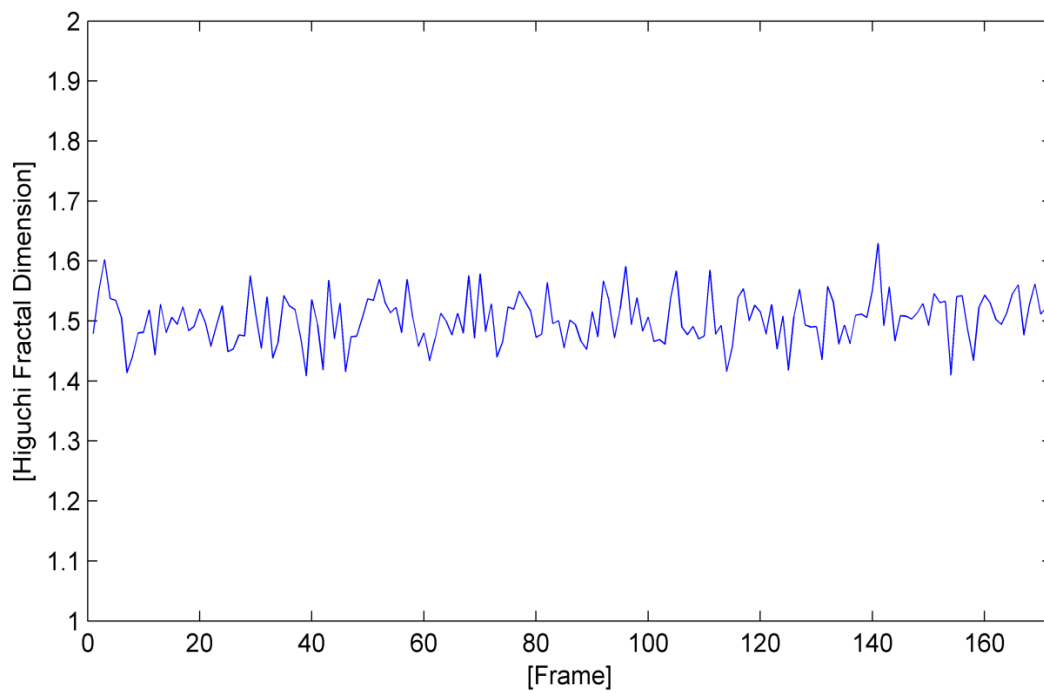


Fig. 5.9. The Higuchi fractal dimension trajectory of the fractional Brownian motion.

5.2 Effects of Noise on Fractal Dimension Estimation

As speakers, regardless of the background conditions, we are always exposed to different levels and types of noise. Recording of different types of real noise is beyond the scope of this thesis and to study the effects of noise, colored noise is used due to being compatible with the different conditions and the natural phenomena [Kins11]. In this section, the VFD and the HFD are tested by addition of white, pink, and brown noise to the Weierstrass function and the fBm. The level of addition of noise is -40 dB and -30 dB.

5.2.1 Weierstrass function

As per the previous section, the Weierstrass function is generated using 512 samples at first, by assigning $\lambda = 2$ and using nine H values spaced equally from 0.1 to 0.9. Then for each of the waveforms, -40 dB and -30 dB of white, pink, and brown noise is added and the VFD and the HFD are used to estimate the fractal dimension of these waveforms. Table 5.7, displays the results obtained by addition of -40 dB of colored noise and table 5.8, displays the results obtained by addition of -30 dB of colored noise.

The results obtained by addition of -40 dB of colored noise shows the overestimation of the FD for the waveforms with a lower FD in comparison to FD estimation of the Weierstrass function without any additional noise. Addition of white noise causes the greatest amount of overestimation, followed by pink noise. Addition of -40 dB of brown noise has no impact on the FD estimation. Although the amount of overestimation due to the addition of -40 dB of colored noise is very low, this overestimation tends to be higher with the VFD in comparison to the HFD. Moreover, a similar behavior is seen with the addition of -30 dB of colored noise. Greater

overestimation of the FD is seen for the Weierstrass function with a lower FD and this overestimation tends to get lower and fade as the FD of the Weierstrass function increases. White noise has the greatest effect on the FD estimation, causing the greatest amount of overestimation, followed by the pink noise. Brown noise does not cause any overestimation to the FD estimation. The overestimation tends to be more for the VFD in comparison to HFD.

Table 5.7: The estimated Fractal Dimension of the Weierstrass functions after addition of -40 dB of colored noise.

-40 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.17	1.19	1.18	1.17	1.16	1.17	1.16	1.16
0.8	1.20	1.24	1.25	1.24	1.24	1.23	1.24	1.24	1.23
0.7	1.30	1.32	1.33	1.32	1.32	1.32	1.32	1.32	1.32
0.6	1.40	1.41	1.41	1.41	1.41	1.41	1.41	1.41	1.41
0.5	1.50	1.50	1.51	1.50	1.50	1.50	1.50	1.50	1.50
0.4	1.60	1.60	1.60	1.60	1.60	1.59	1.59	1.59	1.59
0.3	1.70	1.70	1.70	1.70	1.70	1.68	1.68	1.68	1.68
0.2	1.80	1.80	1.80	1.80	1.80	1.76	1.76	1.76	1.76
0.1	1.90	1.90	1.90	1.90	1.90	1.83	1.83	1.83	1.83

Fig. 5.8: The estimated Fractal Dimension of the Weierstrass functions after addition of -30 dB of colored noise.

-30 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.17	1.31	1.22	1.17	1.16	1.26	1.20	1.16
0.8	1.20	1.24	1.33	1.27	1.23	1.23	1.30	1.25	1.24
0.7	1.30	1.32	1.38	1.34	1.32	1.32	1.35	1.33	1.32
0.6	1.40	1.41	1.44	1.42	1.41	1.41	1.42	1.41	1.41
0.5	1.50	1.50	1.52	1.51	1.50	1.50	1.51	1.50	1.50
0.4	1.60	1.60	1.61	1.60	1.60	1.59	1.59	1.59	1.59
0.3	1.70	1.70	1.70	1.70	1.70	1.68	1.68	1.68	1.68
0.2	1.80	1.80	1.80	1.80	1.80	1.76	1.76	1.76	1.76
0.1	1.90	1.90	1.90	1.90	1.90	1.83	1.83	1.83	1.83

Please note that the addition of colored noise is tested using -40 dB and -30 dB only, since the addition of colored noise lower than -40 dB has no impact on the FD estimation and addition of colored noise greater than -30 dB causes greater overestimation.

In the same time, 88200 samples of the Weierstrass function are generated, by assigning $\lambda = 2$ and using nine H values of 0.1 to 0.9 to resemble the same number of samples in 1

recording. However, unlike the previous section, the 88200 samples consist of 172 cycles of the Weierstrass function which is displayed in Fig 5.10. Then for each of the waveforms, -40 dB and -30 dB of white, pink, and brown noise is added and the VFD and the HFD are used to estimate the fractal dimension of these waveforms. Table 5.9, displays the mean of the trajectory obtained by addition of -40 dB of colored noise and table 5.10, displays the mean of the trajectory obtained by addition of -30 dB of colored noise. The results obtained by addition of colored noise to the signal that consist of 172 cycles of the Weierstrass function shows a similar behavior to the Weierstrass function generated using 512 samples.

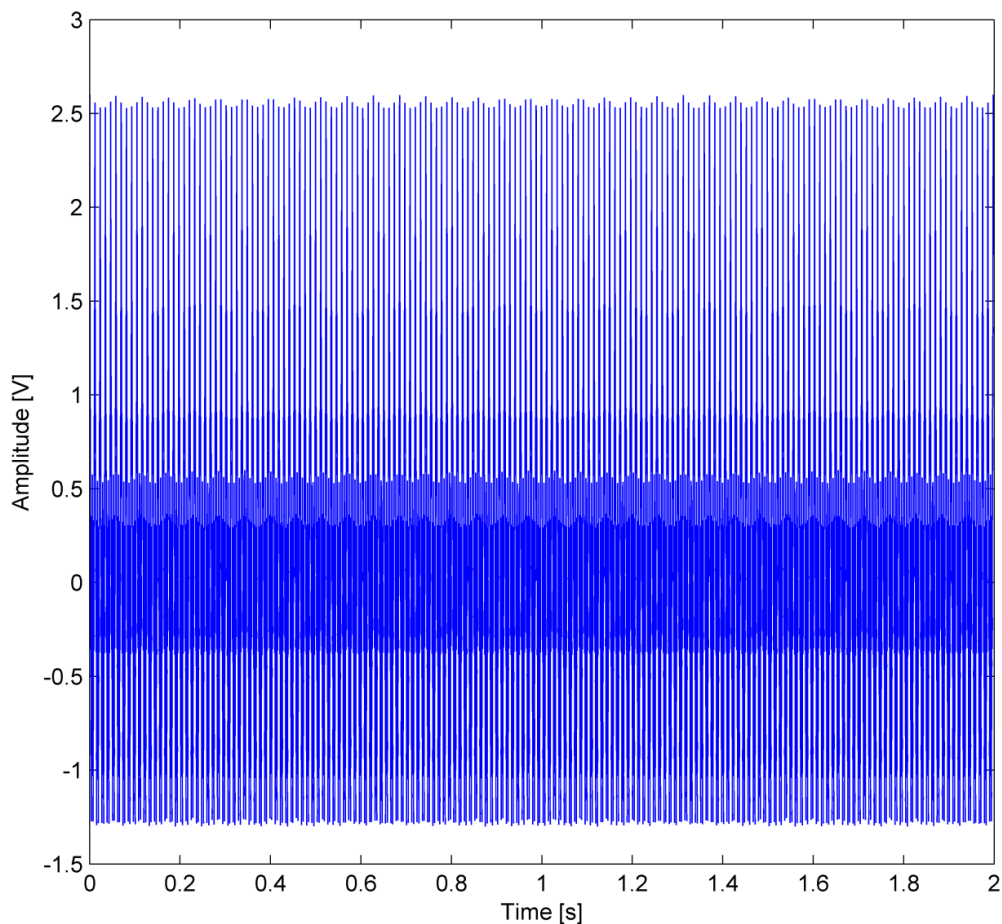


Fig. 5.10. 172 cycles of the Weierstrass function generated using 88200 $H = 0.7$.

As per the previous test, the mean of the trajectory obtained by addition of -40 dB of colored noise shows the overestimation of the FD for the waveforms with a lower FD in comparison to FD estimation of the Weierstrass function without any additional noise. Addition of white noise causes the greatest amount of overestimation, followed by pink noise. Addition of -40 dB of brown noise has no impact on the FD estimation. Although the amount of overestimation due to the addition of -40 dB of colored noise is very low, this overestimation tends to be higher with the VFD in comparison to the HFD.

Table 5.9: The mean of the trajectories of the fractal dimension of the Weierstrass function by addition of -40 dB of colored noise.

Addition of -40 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.18	1.20	1.19	1.18	1.17	1.18	1.17	1.17
0.8	1.20	1.25	1.26	1.25	1.25	1.24	1.25	1.24	1.24
0.7	1.30	1.33	1.34	1.33	1.33	1.32	1.33	1.33	1.32
0.6	1.40	1.42	1.42	1.42	1.42	1.41	1.41	1.41	1.41
0.5	1.50	1.51	1.51	1.51	1.51	1.50	1.50	1.50	1.50
0.4	1.60	1.60	1.60	1.60	1.60	1.59	1.59	1.59	1.59
0.3	1.70	1.70	1.70	1.70	1.70	1.67	1.67	1.67	1.67
0.2	1.80	1.80	1.80	1.80	1.80	1.76	1.76	1.76	1.76
0.1	1.90	1.89	1.89	1.89	1.89	1.83	1.83	1.83	1.83

Similarly, with the addition of -30 dB of colored noise, the mean of the trajectories shows a greater overestimation of the FD for the Weierstrass function with a lower FD and this overestimation tends to get lower and fade as the FD of the Weierstrass function increases. White noise has the greatest effect on the FD estimation, causing the greatest amount of overestimation, followed by the pink noise. Brown noise does not cause any overestimation to the FD estimation. The overestimation tends to more for the VFD in comparison to HFD.

Table 5.10: The mean of the trajectories of the fractal dimension of the Weierstrass function by addition of -30 dB of colored noise.

Addition of -30 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.18	1.32	1.23	1.18	1.17	1.28	1.21	1.17
0.8	1.20	1.25	1.34	1.28	1.25	1.24	1.31	1.26	1.24
0.7	1.30	1.33	1.38	1.34	1.33	1.32	1.36	1.33	1.32
0.6	1.40	1.42	1.44	1.42	1.42	1.41	1.43	1.42	1.41
0.5	1.50	1.51	1.52	1.51	1.51	1.50	1.51	1.50	1.50
0.4	1.60	1.60	1.61	1.60	1.60	1.59	1.59	1.59	1.59
0.3	1.70	1.70	1.70	1.70	1.70	1.67	1.68	1.68	1.67
0.2	1.80	1.80	1.80	1.80	1.80	1.76	1.76	1.76	1.76
0.1	1.90	1.89	1.89	1.89	1.89	1.83	1.83	1.83	1.83

5.2.2 Fractional Brownian motion

Like the Weierstrass function, the fBm is generated using 512 samples at first, using nine H values spaced equally from 0.1 to 0.9. However, unlike the previous section, the fBm is generated using a seed number of 100. This is due to the noise having the same effect on the estimation of the FD for the fBm generated using different seed numbers. Then for each of the waveforms, -40 dB and -30 dB of white, pink, and brown noise are added and the VFD and the HFD are used to estimate the fractal dimension of these waveforms. Table 5.11, displays the results obtained by addition of -40 dB of colored noise and table 5.12, displays the results obtained by addition of -30 dB of colored noise.

Table 5.11: The fractal dimension of the fractional Brownian motion by addition of -40 dB of colored noise.

Addition of -40 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.12	1.53	1.34	1.13	1.12	1.52	1.34	1.13
0.8	1.20	1.22	1.50	1.35	1.21	1.24	1.51	1.37	1.25
0.7	1.30	1.32	1.50	1.39	1.31	1.34	1.49	1.40	1.34
0.6	1.40	1.43	1.53	1.46	1.42	1.42	1.51	1.46	1.42
0.5	1.50	1.53	1.59	1.55	1.53	1.53	1.58	1.54	1.52
0.4	1.60	1.63	1.66	1.64	1.63	1.62	1.66	1.64	1.62
0.3	1.70	1.71	1.74	1.73	1.72	1.71	1.73	1.73	1.72
0.2	1.80	1.80	1.82	1.81	1.80	1.80	1.82	1.81	1.81
0.1	1.90	1.89	1.90	1.90	1.90	1.90	1.91	1.91	1.90

Table 5.12: The fractal dimension of the fractional Brownian motion by addition of -30 dB of colored noise.

Addition of -30 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.12	1.87	1.67	1.17	1.12	1.87	1.69	1.19
0.8	1.20	1.22	1.84	1.64	1.23	1.24	1.84	1.67	1.27
0.7	1.30	1.32	1.81	1.62	1.32	1.34	1.80	1.63	1.35
0.6	1.40	1.43	1.80	1.63	1.42	1.42	1.78	1.62	1.42
0.5	1.50	1.53	1.80	1.66	1.53	1.53	1.78	1.66	1.52
0.4	1.60	1.63	1.82	1.72	1.63	1.62	1.81	1.71	1.63
0.3	1.70	1.71	1.86	1.78	1.72	1.71	1.85	1.78	1.72
0.2	1.80	1.80	1.90	1.84	1.81	1.80	1.90	1.85	1.81
0.1	1.90	1.89	1.95	1.91	1.90	1.90	1.96	1.91	1.90

The results obtained shows that white and pink noises dominate the fBm, unlike the Weierstrass function. This is due to the structure of the fBm which has characteristics of noise itself [KrBo13] thus, with the addition of -40 dB of white and pink noise there is a significant overestimation for the waveforms with a lower FD and with the addition of -30 dB this overestimation is higher and closer to the FD of the respective additive noise. Meanwhile, the addition of Brown noise causes overestimation of the FD for the lower FD waveform which gets reduced and eventually fades as the FD of the waveform increases.

In the same time, 88200 samples of the fBm are generated, by assigning the seed of the

random number generator to 100 and using nine H values of 0.1 to 0.9 to resemble the same number of samples in 1 recording. Then for each of the waveforms, -40 dB and -30 dB of white, pink, and brown noise is added and the VFD and the HFD are used to estimate the FD of these waveforms. Figure 5.11 displays 88200 samples of the fBm generated by assigning $H = 0.5$ and setting the seed to 100. Table 5.13, displays the mean of the trajectory obtained by addition of -40 dB of colored noise and table 5.14, displays the mean of the trajectory obtained by addition of -30 dB of colored noise. The results obtained by addition of colored noise to the fBm generated using 88200 samples shows a similar behavior to the fBm generated using 512 samples.

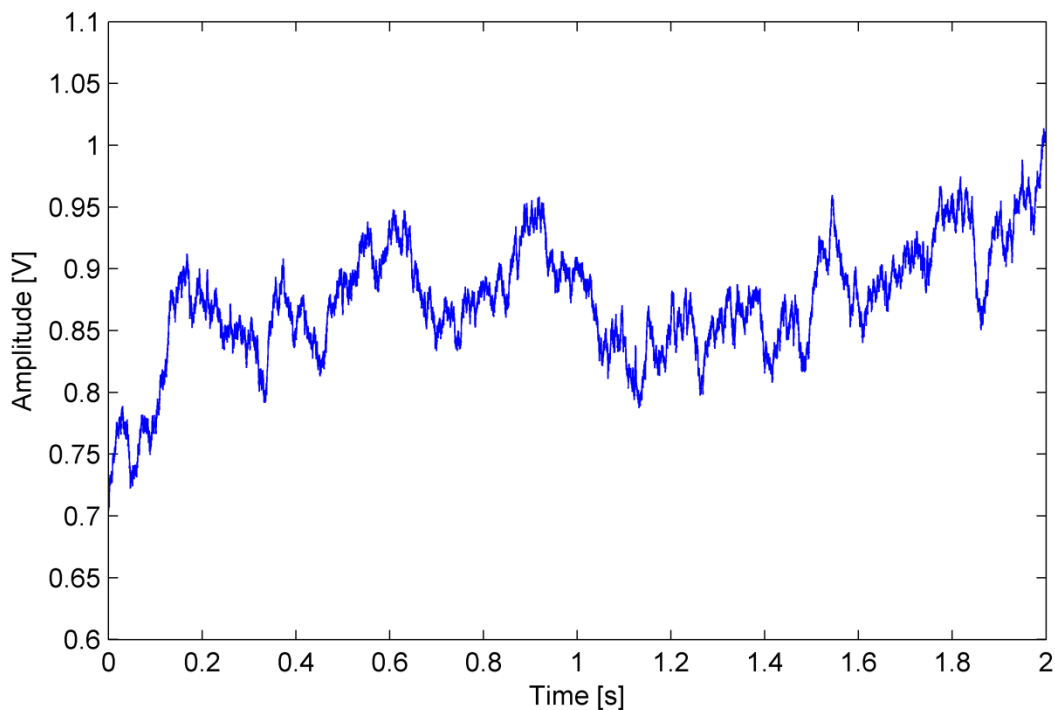


Fig. 5.11. The fractional Brownian motion generated by assigning $H = 0.5$.

As per the previous test, the mean of the trajectory obtained by addition of -40 dB and -30

dB of white and pink noise shows that the FD changes to the FD of the additive noise. Addition of brown noise causes an overestimation of the FD for the fBm waveform with a lower FD with this overestimation being higher for the colored noise added at -30 dB level.

Table 5.13: The mean of the trajectory obtained by addition of -40 dB of colored noise to the fractional Brownian motion.

Addition of -40 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.11	2.00	1.85	1.32	1.11	2.00	1.85	1.31
0.8	1.20	1.21	2.00	1.85	1.26	1.20	2.00	1.85	1.26
0.7	1.30	1.30	2.00	1.84	1.31	1.30	2.00	1.85	1.31
0.6	1.40	1.40	1.99	1.83	1.40	1.40	1.99	1.84	1.40
0.5	1.50	1.50	1.98	1.82	1.50	1.50	1.98	1.83	1.50
0.4	1.60	1.60	1.97	1.82	1.60	1.60	1.97	1.82	1.60
0.3	1.70	1.70	1.98	1.83	1.70	1.70	1.98	1.83	1.70
0.2	1.80	1.80	1.99	1.84	1.80	1.80	1.99	1.85	1.80
0.1	1.90	1.90	1.99	1.85	1.90	1.90	1.99	1.86	1.90

Table 5.14: The mean of the trajectory obtained by addition of -30 dB of colored noise to the fractional Brownian motion.

Addition of -30 dB Noise		Variance Fractal Dimension				Higuchi Fractal Dimension			
Hurst	Theoretical Fractal Dimension		White Noise	Pink Noise	Brown Noise		White Noise	Pink Noise	Brown Noise
0.9	1.10	1.11	2.00	1.85	1.47	1.11	2.00	1.85	1.46
0.8	1.20	1.21	2.00	1.85	1.41	1.20	2.00	1.85	1.40
0.7	1.30	1.30	2.00	1.85	1.36	1.30	2.00	1.85	1.36
0.6	1.40	1.40	2.00	1.85	1.41	1.40	2.00	1.85	1.41
0.5	1.50	1.50	2.00	1.84	1.50	1.50	2.00	1.85	1.50
0.4	1.60	1.60	2.00	1.84	1.60	1.60	2.00	1.85	1.60
0.3	1.70	1.70	2.00	1.84	1.70	1.70	2.00	1.85	1.70
0.2	1.80	1.80	2.00	1.85	1.79	1.80	2.00	1.85	1.79
0.1	1.90	1.90	2.00	1.85	1.89	1.90	2.00	1.85	1.89

5.3 Voice Activity Detection

Chapter 4 described the procedures taken in order to have recordings with similar conditions and a reduced amount of noise. However, under even, the most controlled conditions the pre-

silence, the silence, and the post-silence part of the utterance will contain noise. When we zoom on the silence we can see that these silence parts look like noise with specific characteristics and as a result when we apply the algorithms to extract features, the silence part will be seen in the trajectory as well. This makes the task of feature extraction difficult since it would be difficult to distinguish the features from the silence. Therefore, the characteristics of the background noise are studied, followed by the estimation of the fractal dimension of colored noise and the background noise. Upon establishment of the characteristics of the background noise, a VAD which is based on the VFD is introduced.

5.3.1 Characteristics of the Background Noise

It was reported in chapter 4 that during the recordings the background noise measured was between 35 dB to 45 dB, during the silence moments using the “DecibelX” smartphone app. Although the amplitude of the silence in the recording is related to other factors, such as the gain of the microphone, the amplitude of the background noise in the silence is in the range of -45 to -40 dB. Figure 5.12, displays 500 ms of pre-silence in one of the recordings. The background noise can be characterized by its PSD. The PSD of the background noise is displayed in fig. 5.13, along with the PSD of generated pink noise and the PSD of theoretical pink noise. The spectral decay shows a higher drop in the lower frequencies. However, the spectral decay of the higher frequency is similar to the spectral decay of pink noise or in other words $\beta = 1$. The higher drop in the spectral decay of the lower frequencies is due to the attenuation of the reverberation of the background noise in the anechoic chamber and the background noise can be characterized as having pink noise characteristics.

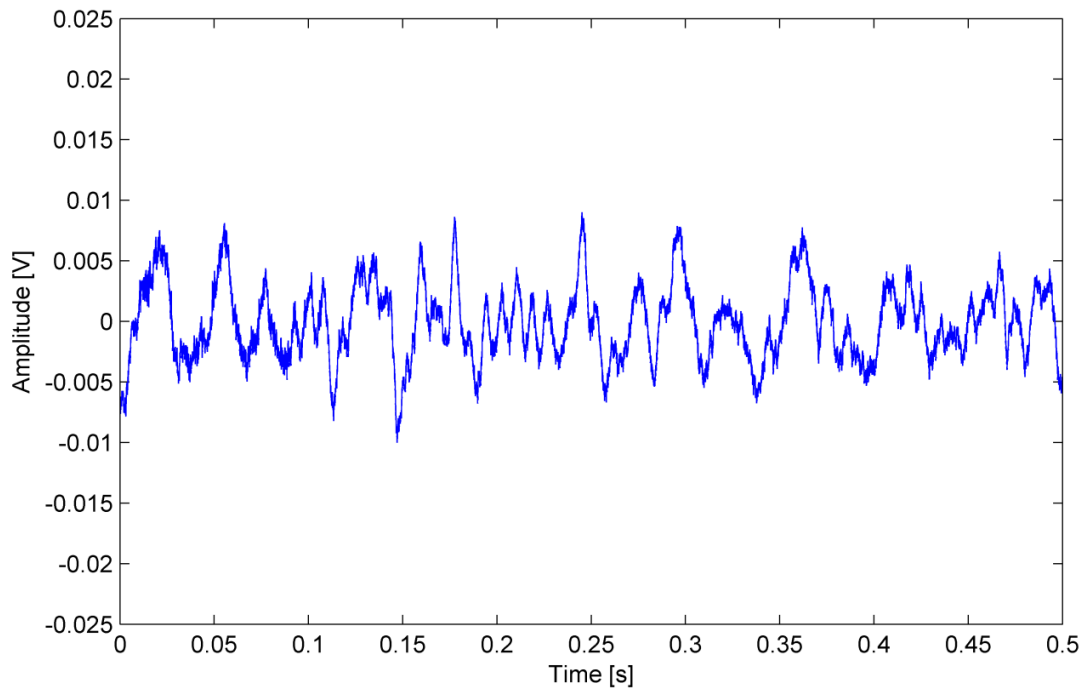


Fig. 5.12. The background noise seen in the pre-silence.

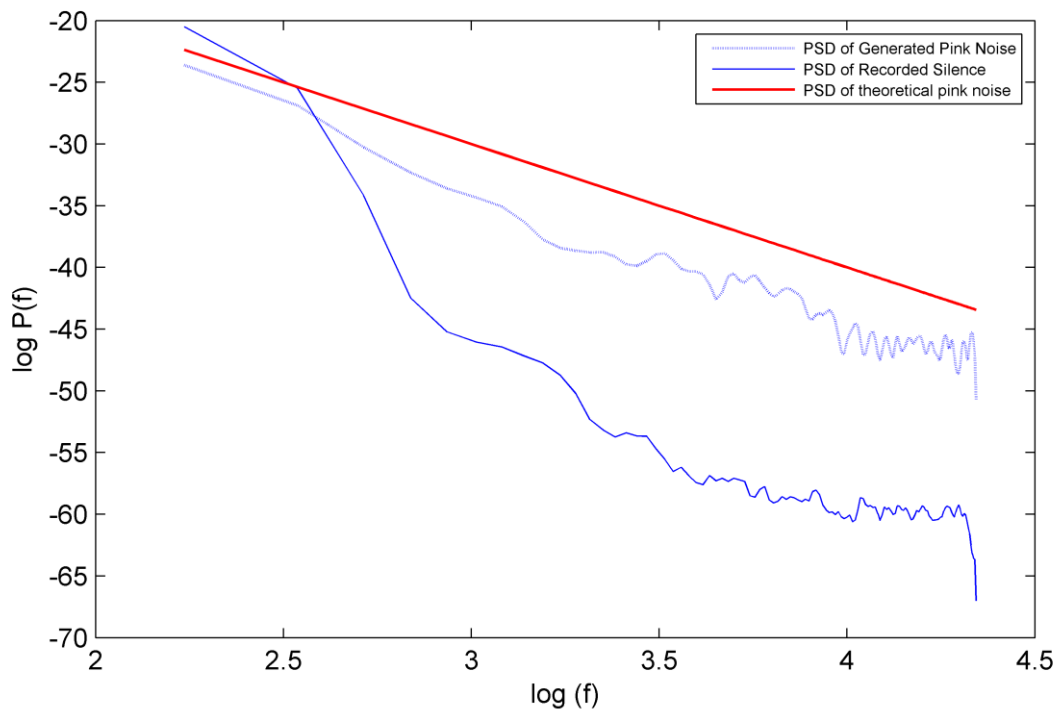


Fig. 5.13. The power spectrum density of the noise.

5.3.2 Fractal Dimension of Colored Noise

It is reported by [KiGr08] that the stop consonants and the fricatives have trajectories with dimension level of background noise, making the detection of features difficult since they could be easily mistaken by the background noise.

Colored noise has specific FD for each color, the FD of white noise is in the range of 2, pink noise is in the range of 1.8 and brown noise is in the range of 1.5. To test the FD of colored noise, 88200 samples of white, pink, and brown noise is generated and the VFD is used to estimate the FD. The trajectory of each color of the noise obtained is displayed in fig. 5.14.

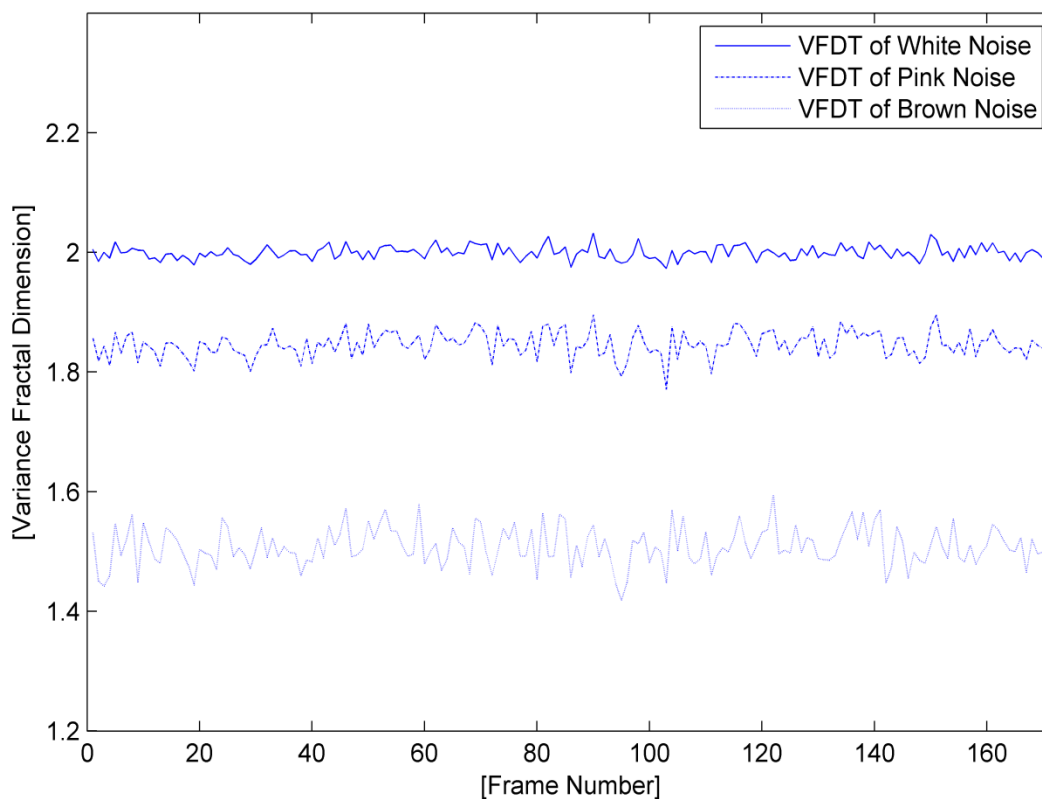


Fig. 5.14. The variance fractal dimension trajectory of white, pink, and brown noise.

Similarly, the VFD is used to estimate the FD of the recorded noise. The trajectory obtained is displayed in fig. 5.15, which shows that the FD of the background noise is in the range of 1.6 to 1.8. Please note that in some instances the FD does go below or above this range, however in most cases the FD of the background noise is within the 1.6 to 1.8 range.

The estimated FD of the background noise in the silence is in the range, which is in between the FD of pink and brown noise. As it was established in the previous section that background noise has the characteristics of pink noise, however, the lower frequencies have a greater spectral decay, due to attenuation. This attenuation affects the complexity of the background noise and as a result, causes the FD to be lower than pink noise. Moreover, there might be minor complexity reduction in comparison to synthetically generated colored noise due to numerical noise introduced to the signal upon recording.

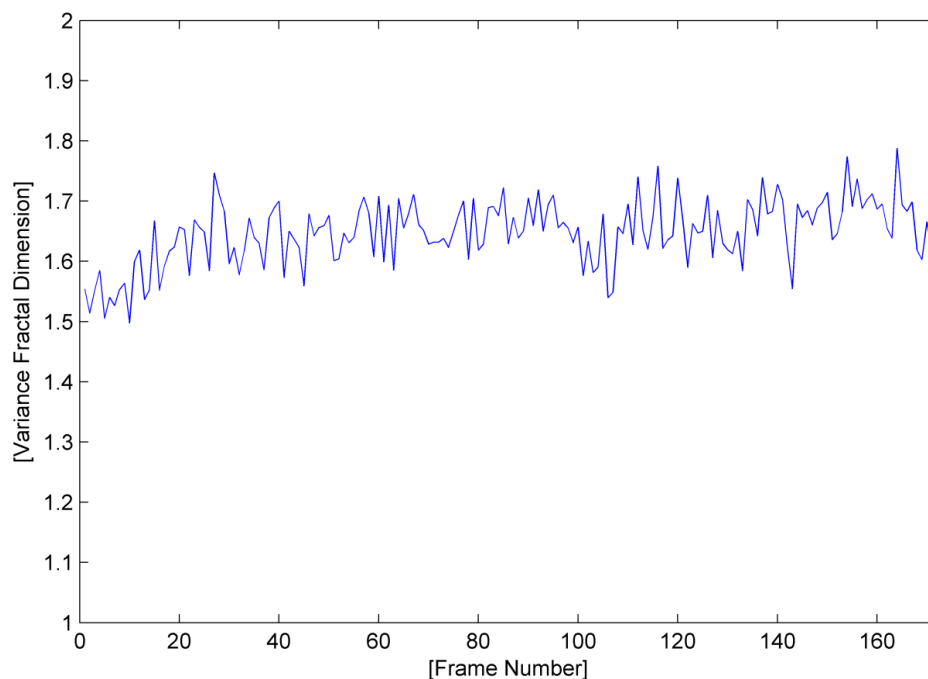


Fig. 5.15. The variance fractal dimension trajectory of the pre-silence.

5.3.3 The Voice Activity Detection Algorithm

Fractal dimension estimation algorithms can estimate the complexity of a signal regardless of its amplitude. It was established that the background noise is in the range of -45 to -40 dB and has the characteristics of pink noise with the FD being in the range of 1.6 to 1.8 in the majority of instances. Moreover, in section 5.2 it was established that addition of -30 dB and -40 dB of colored noise to the Weierstrass function does not significantly affect the FD estimation of the majority of the waveforms.

Addition of colored noise to speech can help control the FD level of background noise by increasing or decreasing it, causing it to be different from the features without affecting the features itself. This will make the task of feature extraction simple since the dimension of the background noise would change but the dimension of the features will be the same due to the speech having a higher amplitude and thus, the unchanged data could be clearly extracted.

It is trivial to note that acquiring a low SNR while recording the utterances is of crucial importance because not only the features would contain less amount of noise, but also the amplitude of the noise available in the silences would be lower. This would allow the addition of colored noise at lower SNR in order to minimize the effects of noise on the features.

Hence, for the purpose of VAD in this work, the VFD is used to estimate the FD trajectory of the utterances. The choice of VFD is due to providing a more accurate estimation of the FD of the test waveforms in section 5.1 with FD of 1.6 to 1.9 and having a lower variability in its trajectory. Addition of colored noise is done using -30 dB of white noise to the signal, which can increase the FD level of the background noise to 2 without affecting the speech significantly, due to the speech having a higher amplitude. A threshold value of 0.05 is set and any frame of the

signal that has a change in FD greater or less than the threshold is considered as noise. Figure 5.16, displays the waveform of the utterance “church”. Figure 5.17, shows the VFD of the waveform of the utterance “church”. This image shows that the background noise has a FD similar to the phoneme /ch/ and it is very difficult to distinguish the Phoneme.

White noise is added to the waveform of the utterance “church” at -30 dB SNR. Figure 5.18, displays the VFD of the waveform of the utterance “church” after addition of -30 dB of white noise. This image shows that the FD trajectory of the pre-silence, silence, and the post-silence increase to around 2, due to addition of white noise, an overestimation which is in the range of 0.2 to 0.3 in the FD estimation of the frames that contain silence, while, there is no change or an overestimation of below 0.05 in the FD estimation of the frames that contain speech. Please note that if the overestimation in the FD is greater than 0.05 for the frames that contain speech it is due to the framing having background noise in a greater SNR.

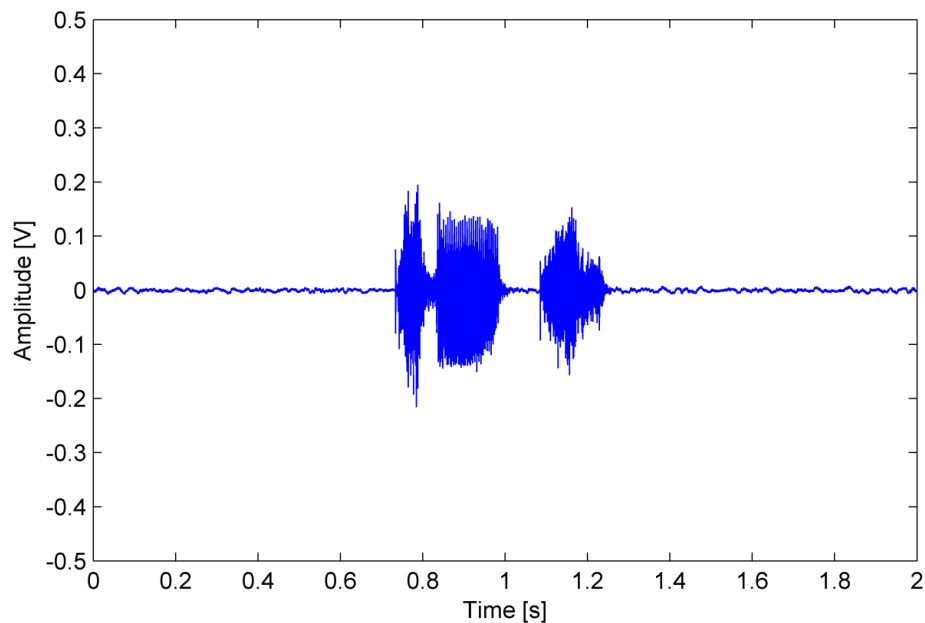


Fig 5.16. The Waveform of the utterance “church”.

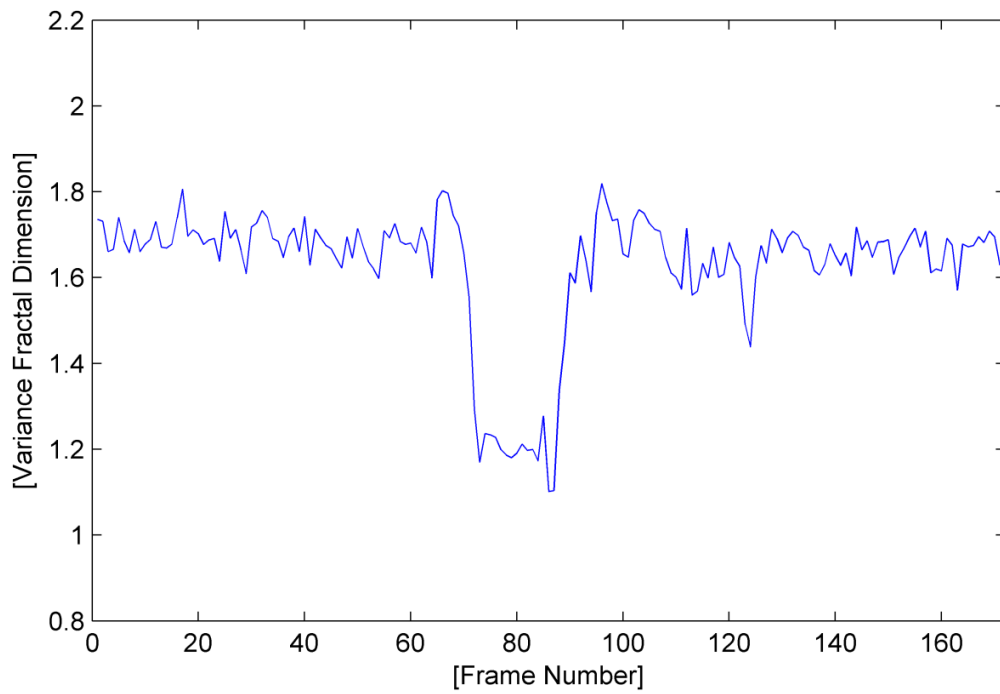


Fig. 5.17. The variance fractal dimension of the utterance "church".

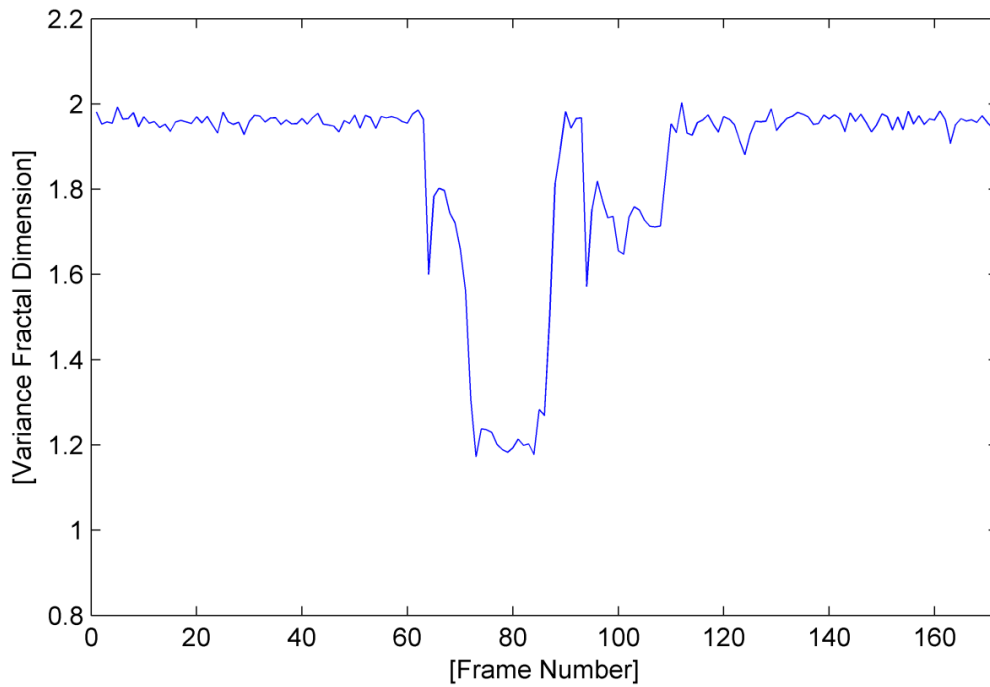


Fig. 5.18. The variance fractal dimension after addition of white noise.

Therefore, the frames of the signal that have a FD increase or decrease of 0.05 are considered as the frames that contain speech signal, which are displayed in fig. 5.19 for the utterance church. Figure 5.20, displays the waveform of the utterance “church” after extraction of the frames containing speech using the VAD algorithm described. This image shows that the pre-silence, silence, and the post-silence have been removed. In section 5.5 this method is tested and compared with different VAD to see the difference in speaker verification accuracy.

Appendix A displays the process of VAD for a set of 44 utterances recorded by one of the participants. As in this section for each utterance, the image of the waveform, the VFDT of the utterance, the VFDT after addition of noise, the extracted VFDT, and the extracted waveform are displayed. Furthermore, the flowchart of this algorithm is provided in appendix B, fig. B.8 of this document.

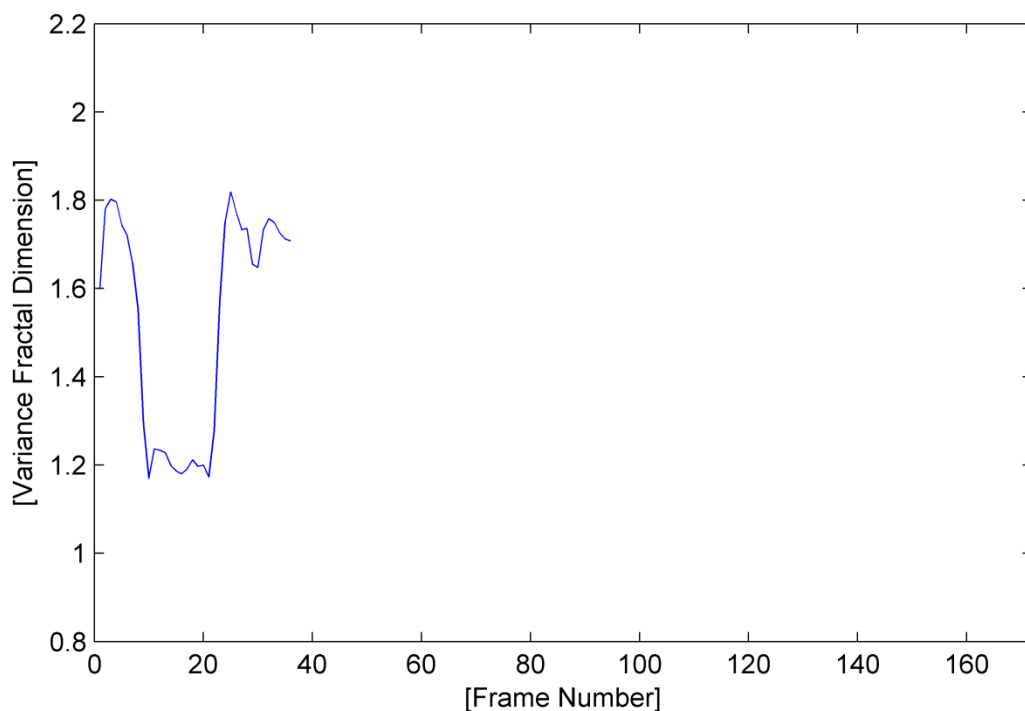


Fig. 5.19. The extracted frames containing speech.

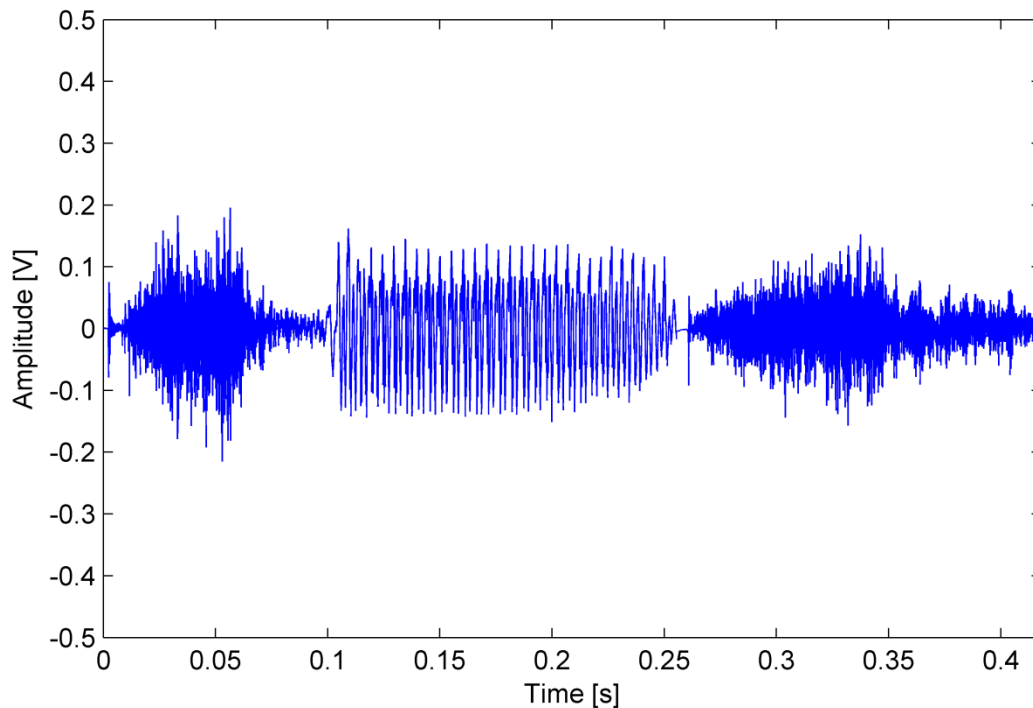


Fig. 5.20. The waveform of the utterance detected by the voice activity detection algorithm.

5.4 Feature Extraction

This section presents the features extracted from the frames that contain speech. We would like to set the baseline to the LPCC algorithm obtained using a 23rd order LPC and 23 LPCC vector which will be referred to as LPCC23. The choice of the 23rd order LPCC is due to the use of this algorithm by [Reyn94], which yielded an accuracy of 92%. Moreover, the results obtained by [Reyn94], shows that changing the training and testing data yields different results. Hence, due to the use of a different speech dataset in this work and in order to compare the performance of the features, LPCC23 is set as the baseline and a feature vector based on the fusion of multiple features described in chapter 3 is created and used for training and testing the data. Fusion is widely used in the area of speaker recognition ([CaRD03]; [ChWC97]; [HaDC04]; [KiA109]).

The next feature that is added to the feature vector is 16 MFCC vector obtained using a 16th order MFCC which will be referred to as the MFCC16. The choice of 16th order MFCC is due to the intention of not significantly exceeding the 39 feature vector commonly used to solve the text-independent speaker recognition problem ([KiLi10]; [ToPu11]; [ChZF16]). In the same time, it was reported by [Reyn94] that when MFCC and LPCC used are lower than the model order, the accuracy is reduced, most significantly for noisy signals.

Asides from the LPCC23 and MFCC16, the HFD and the VFD is added to the feature vector. The HFD is obtained by setting $kmax = 16$ and the VFD is obtained using a dyadic time displacement by assigning $n_k = 16, 8, 4, 2, 1$. It is specified by [KiGr09] that for the VFD algorithm $n_k = 1$ should be omitted to reduce the effect of correlation between adjacent signal samples. It is noticed in this work that omission of $n_k = 1$ causes the VFDT to go significantly over 2 for phonemes that have a turbulent structure such as the fricatives a matter which is noticed in [Grie96] as well. It is noticed that with the addition of $n_k = 1$, the VFDT is reduced to a FD close to 2, in some cases slightly crossing 2 which is compatible with HFD as well. The highest VFD values are notice for phoneme /s/, where for 2 of the participants, a peak value of 2.10 is noticed. Speech has nonlinear characteristics and its multifractal nature has been proven [LaSK97]. The choice of the HFD and VFD is due to both the algorithms estimating the FD quickly due to their simplicity and the possibility of implementing them for real-time applications. Moreover, the 42nd and 43rd features are allocated to the ZCR and the TC. The ZCR provides an estimate of the frequency and the TC provides an estimate of the bandwidth of the signal in a given frame.

An experimental sensitivity analysis is conducted in the next section to test the effects of each feature on the accuracy of classification of the test data using the SVM. The feature vectors

used in these tests are as follows:

A: Consists of all 6 features mentioned and each feature vector is 43 fields.

B: Consists of only LPCC23 and each feature vector are 23 fields.

C: Consists of only MFCC16 and each feature vector are 16 fields.

D: Consists of LPCC23 and MFCC16 and each feature vector are 39 fields.

E: Consists of LPCC23, MFCC16, and HFD and each feature vector are 40 fields.

F: Consists of LPCC23, MFCC16, and VFD and each feature vector are 40 fields.

G: Consists of LPCC23, MFCC16, and ZCR and each feature vector are 40 fields.

H: Consists of LPCC23, MFCC16, and TC and each feature vector are 40 fields.

For simplicity, each combination of the feature vector will be referred to by the alphabet assigned to them above. The graphical representation of feature vector A which consists of all the features is displayed in fig. 5.21. In the next section, the mentioned above features are extracted by different VAD methods and the SVM is used to train the model and classify the test data.

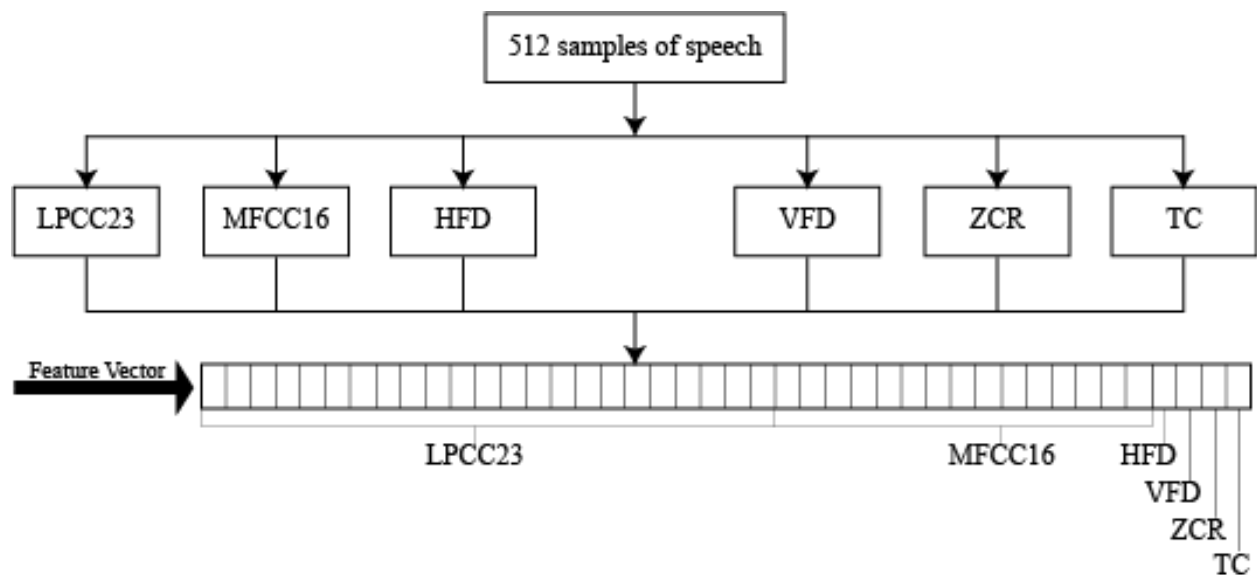


Fig. 5.21. The graphical representation of feature vector "A".

5.5 Classification of Features

Upon extraction of features, it is the turn to build a training model and classify the feature vectors extracted from the test data. This section presents the results obtained for the different combination of feature vectors discussed above which were extracted using the VAD algorithm discussed in section 5.3. These results are then compared with results obtained by using the same the feature vectors that were extracted using the HFD, amplitude threshold, and energy.

5.5.1 Building of the training model and the classification of the test data.

The set of test words used are the 44 keywords from the MRPA displayed in table 2.1. The keywords are arranged in alphabetical order with the odd keywords used for training and the even keywords used for testing the classifier. The 22 keywords used for training and the 22 keywords used for testing are displayed in table 5.15. The motivation behind the division of the keywords in this manner is to have training and testing words that contain similar phonemes but yet are completely different. Meanwhile, dividing the data into half will ensure that the classifier will not be over-trained and thus upon exposure to new data its performance will not be reduced. Please note that in this work the recorded utterances are not warped, thus the amount of training and testing data is variable from one speaker to another. Using the VAD algorithm discussed in section 5.3.3 to obtain the frames containing speech from the recorded utterances of the 24 participants and by addition of -30 dB of white noise, the average training speech extracted is 7.43 sec with a maximum of 9.76 sec and a minimum of 5.21 sec. Similarly, for the testing data, the average testing speech extracted is 7.59 sec with a maximum of 9.88 sec and a minimum of 5.43 sec.

Table 5.15: The keywords used for training and testing the classifier.

Training						Testing					
No	MRPA	Keyword	No	MRPA	Keyword	No	MRPA	Keyword	No	MRPA	Keyword
1	/a/	Bad	12	/f/	Fife	1	/@/	Banana	12	/oo/	For
2	/b/	Barb	13	/g/	Gag	2	/aa/	Bard	13	/h/	Hand
3	/ei/	Bay	14	/jh/	Judge	3	/ii/	Bead	14	/k/	Kick
4	/e@/	Bear	15	/l/	Loyal	4	/e/	Bed	15	/zh/	Measure
5	/i@/	Beer	16	/m/	Mime	5	/i/	Bid	16	/n/	None
6	/@@/	Bird	17	/dh/	Other	6	/ou/	Boat	17	/p/	Pip
7	/o/	Body	18	/u@/	Poor	7	/u/	Book	18	/r/	Rear
8	/uu/	Boot	19	/ng/	Ringin	8	/au/	Bough	19	/sh/	Sheepish
9	/oi/	Boy	20	/t/	Test	9	/uh/	Bud	20	/th/	Thirtieth
10	/ai/	Buy	21	/v/	Verve	10	/s/	Cease	21	/w/	Weal
11	/ch/	Church	22	/y/	Year	11	/d/	Deed	22	/z/	Zoos

The SVM is used for training a model and classification of the testing data. All the experiments are conducted by setting the kernel function to the linear kernel and assigning the $C = 10$. The choice of the kernel function and the cost function is motivated by experimental results. All the classifications were done using a one-vs-one approach resulting in 276

classifications. The mean of the accuracy of the classifications of the test data obtained for each of the mentioned above tests is displayed in table 5.16.

Table 5.16: The accuracy of the classifications using variance fractal dimension.

A	B	C	D	E	F	G	H
91.60 %	86.79 %	76.86 %	91.27 %	91.34 %	91.40 %	91.38 %	91.41 %

The results obtained shows that the feature vector “B” yields an accuracy of 86.79%. This is a reduction of almost 5% in comparison to the results obtained by [Reyn94]. The drop in accuracy percentage can be related to having fewer training data. Meanwhile, feature vector “C” yields 76.86 % due to having a low filter order. Feature vector “D” which is a combination of feature vector “B” and “C” shows a significant improvement resulting in an accuracy of 91.27%. The classification accuracies increase for feature vectors “E” to “H”, due to the addition of speech features not measured by the LPCC23 and MFCC16. Feature vector “A” yields the highest result which is 91.60% showing that addition of nonlinear features improves the accuracy of the speaker verification.

5.5.2 Comparison of the results with different voice activity detection methods.

The focus of this work is the comparison of different features and hence, all the processes of building the model and the classification steps are kept constant. This way, the changes in performance can be linked mostly to features and the VAD. In this section, the features are extracted using different VAD methods.

At first, the VFD is replaced with HFD in the algorithm described in section 5.3.3. The average training speech extracted using this method is 7.41 sec with a maximum of 10.52 sec and

a minimum of 5.07 sec. Similarly, for the testing data, the average testing speech extracted is 7.57 sec with a maximum of 9.88 sec and a minimum of 5.43 sec. This shows a slight decrease in the average training and testing times, however an increase in the maximum and a decrease in the minimum training and testing times. The results of the classification of the test data by extraction of features using this method are displayed in table 5.17.

Table 5.17: The accuracy of the classifications using Higuchi fractal dimension.

A	B	C	D	E	F	G	H
91.43 %	86.49 %	76.91 %	91.09 %	91.16 %	91.16 %	91.18 %	91.22 %

The results display a slight decrease in the classification accuracy for all the feature vectors besides from feature vector “C”. This decrease in accuracy can be linked to the reduced training and test time. Thus, this reduction is not due to noise being omitted, but rather, they are frames containing speech being omitted due to FD estimation error.

In the next approach, the VAD is established by placing a threshold on the amplitude of the signal. Any sample greater than 0.01 V or any sample less than -0.01 V is considered a speech signal. The threshold is set to 0.01 V due to the background noise being in the range of -45 to -40 dB. The average training speech extracted using this method is 6.76 sec with a maximum of 9.49 sec and a minimum of 4.45 sec. Similarly, for the testing data, the average testing speech extracted is 6.73 sec with a maximum of 9.24 sec and a minimum of 4.62 sec. The results of the classification of the test data by extraction of features using this method are displayed in table 5.18.

Table 5.18: The accuracy of classifications using amplitude threshold.

A	B	C	D	E	F	G	H
86.10 %	77.84 %	76.50 %	85.51 %	85.67 %	85.73 %	85.63 %	85.78 %

The results display a 5% to 6% decrease in the accuracy for feature vectors “A”, “D”, “E”, “F”, “G”, and “H”. However, the decrease is greater for feature vector “B” and very small for feature vector ‘C’.

Moreover, the VAD is established by setting a threshold on signal energy. The threshold is set to 0.0005 V with the motivation of having a threshold that is slightly higher than -40 dB level. The extracted speech extracted using this method is reduced as expected. The average training speech extracted using this method is 4.99 sec with a maximum of 7.78 sec and a minimum of 2.87 sec. Similarly, for the testing data, the average testing speech extracted is 4.90 sec with a maximum of 7.16 sec and a minimum of 2.98 sec. The results of the classification of the test data by extraction of features using this method are displayed in table 5.19.

Table 5.19: The accuracy of the classifications using the energy.

A	B	C	D	E	F	G	H
86.06 %	77.05 %	75.55 %	85.20 %	85.27 %	85.36 %	85.44 %	85.62 %

The results show a slight decrease in accuracy in comparison to setting a threshold on amplitude, however, show the same pattern. The slight decrease in accuracy can be related to the decrease in average training and testing times.

Meanwhile, it was reported by [ReRo95], that LPC spectral representations, such as LPCC,

can be severely affected by noise and directly computed filterbank features are more robust to noisy speech. The significant difference in overall accuracy of feature vectors extracted using the method discussed in section 5.3.3 using VFD and HFD, in comparison to setting a threshold on signal amplitude and energy can be related to capturing of phonemes that have similar amplitude to the noise level and omission of frames that contain speech, but yet the speech is contaminated with noise. The VFD and the HFD can estimate the FD of a signal regardless of the amplitude of the signal and addition of noise can cause overestimation of the FD which allows frames contaminated with noise to be detected and omitted. The significant decrease in the accuracy of the feature vector “B” that consist of LPCC23 and a slight decrease in the accuracy of the feature vector “C” which consist of MFCC16, extracted using VAD based on setting a threshold on the amplitude and energy of the signal can be related to speech samples that were contaminated by noise. The LPCC23 are severely affected by noise and the MFCC16 are more robust to noisy speech.

5.6 Summary

This chapter presented the results and the analysis. Firstly, the VFD and the HFD were tested using the Weierstrass function and the fBm. The results show that the VFD has less variability for most of the waveforms and is slightly more accurate for the Weierstrass function with FD 1.7 to 1.9. Secondly, colored noise was added to the test data to study the effects of noise on the estimation of FD. It was found that the VFD is more sensitive to the addition of noise. Thirdly, a VAD algorithm that utilizes the characteristics of the background noise and is based on FD was introduced. Finally, an experimental sensitivity analysis of the extracted features was conducted using the SVM. The features were extracted using the VAD detection algorithm based on the

VFD, the HFD, the amplitude threshold, and the energy. The feature vector that consisted of all the features and was extracted using the VAD that utilized the VFD provided the highest accuracy, which was 91.60%.

In the next chapter, the thesis overview is presented followed by the thesis conclusions. Furthermore, the thesis contributions are presented, followed by the limitation of this work and possible future extensions.

Chapter 6

Conclusions

6.1 Thesis Overview

This thesis presented a study of fusion of multiple features with the focus of embedding fractal methods to the front end-processing of a text-independent speaker verification system. The fusion of the features which were extracted using a VAD detection based on FD showed an improved accuracy rate on the dataset that was recorded. Chapter 2 began with the discussion of the anatomy of human speech production and perception, followed by the discussion of the organization of sound in the human language known as phonology. Moreover, this chapter introduces the fundamental techniques and concepts that are used in the algorithms discussed in chapter 3.

Meanwhile, the methodology and programming of the algorithms used are discussed in chapter 3. Chapter 3 starts with the discussion of colored noise which was used in the VAD detection algorithm, followed by the framing of the speech signal into stationary segments for analysis. Moreover, the algorithms used to form the feature vectors (ZCR, TC, LPCC, MFCC, HFD, and VFD) are discussed, followed by, the test signals (Weierstrass function and fBm) which was used to study the accuracy of the FD estimation algorithms used. In the same time, the SVM which was used to build a model and test the unseen data was discussed In this chapter.

Any study on speaker verification requires a dataset. In this thesis, a dataset is recorded under controlled conditions using 24 volunteers raised in the Province of Manitoba, Canada. Chapter 4 provides a detailed description of the recording hardware and software used followed by the discussion of the test words used, the demographics of the participants, and the protocols used. Moreover, the recorded dataset is stored in a repository, to facilitate other researchers gaining access to the data for further research. The aim of this chapter is to provide a detailed guideline to allow the repeatability of the recordings.

The experimental results and analysis are presented in chapter 5. At first, since different phonemes have a different FD, the accuracy of the HFD and VFD is tested for FD rang of 1.1 to 1.9 using the Weierstrass function and the fBm, followed by, the study of effects of colored noise on the estimation of the FD. In the same time, a VAD algorithm based on changing the characteristics of the background noise and estimation of FD is introduced and used to extract features. An experimental sensitivity analysis of the features is conducted by using the SVM to model and test the feature vectors extracted using VAD based on VFD, HFD, amplitude threshold, and signal energy. The fusion of the 6 features discussed, which were extracted using a VAD based on the VFD provided the highest accuracy of 91.60%.

6.2 Thesis Conclusions

This thesis addressed a number of research questions in section 1.2.3, related to recording of a dataset, feature extraction, and training a classifier to test the features extracted. This section links the results to the research questions to provide insight into the answers and potential future research.

At first, 24 volunteers consisting of 12 male and 12 female participants raised in the Province of Manitoba, Canada, are recorded. In order to provide a repeatable set of test words that would cover all of the phonemes, the 44 keywords from the Edinburg MRPA shown in table 2.1 is used and recorded for each participant in a recording session. The choice of MRPA over the IPA is due to the IPA not being machine readable [KiGr08]. The MRPA covers all the English phonemes, has enough speech data to build a model and at the same time, quick to record making it practical to use for a real-world application. All the recordings were conducted in one continuous session, each approximately 15 to 20 minutes in duration. In this work, to eliminate any background noise from outside of the chamber each word was recorded separately. The recording session time could be reduced if the keywords are recorded continuously. In the same time, chapter 4 provides a detailed description of the hardware, the software, and the set of protocols used in order to ensure quality, repeatability, and similarity of all the recordings. Furthermore, to allow further research using the dataset by researchers, the recorded dataset is uploaded to IEEE dataport and can be found at [SeKi18].

Secondly, the HFD and the VFD are selected. Both these algorithms use the temporal features of the signal to estimate the FD and are fast, making them suitable for speech processing. Both these algorithms are tested using the Weierstrass function and the fBm

generated with different complexity. The results which are displayed in section 5.1, and show that the VFD has less variability for most of the waveforms and is slightly more accurate for the Weierstrass function with FD 1.7 to 1.9. This is due to the VFD using variance to estimate the FD. Moreover, both the algorithms are tested with colored noise in section 5.2. The results show that white noise causes the greatest overestimation in the FD, followed by pink noise. This is due to structure and complexity of the noises. Brown noise does not cause significant overestimation of the FD. It is noticed that the VFD is more sensitive to the addition of colored noise and has more overestimation. This is due to the VFD estimating the FD by finding the variance of the increments, which can be found in equation 3.19, causing an error to increase by the power of 2.

Thirdly, the characteristics of the noise in the background of the recordings are studied and it is established that the noise is pink noise with a FD of 1.6 to 1.8. The tests in section 5.2, showed that addition of white noise will increase the FD of noise to 2 (As per the tests with the fBm) and does not affect the FD of the segments containing speech to that extent. Therefore, using this property a VAD algorithm is introduced in section 5.3 that uses the VFD to detect the segments containing speech by addition of white noise to the signal. This method of VAD is more robust to noise as it does not rely on the amplitude but the complexity of the speech. Therefore, this property of the VAD allows it to detect segments that contain speech and are above the energy threshold level but yet are corrupted with noise. The results obtained in section 5.5 support this claim. The results show that when FD is not used for the VAD, the accuracy of classification of the feature vectors containing LPCC is degraded significantly in comparison to other feature vectors, since LPC spectral representations, such as LPCC, are severely affected by noise [ReRo95]. In these tests the VAD that utilized the VFD provided the highest accuracy rate, higher than the HFD. This can be linked to the VFD provided a better estimation of the FD of the

Weierstrass function and thus and be suggested that it is slightly estimating the FD of speech more accurately.

Finally, in section 5.4, multiple combinations of feature vectors are discussed and an experimental sensitivity analysis is conducted in section 5.5. Meanwhile, knowing that training data affects the accuracy of the correctly identified frames and in order to compare the accuracy of correctly identified frames to the literature the LPCC23 is used and yielded an accuracy 86.79% which is a reduction of almost 5% in comparison to the results obtained by [Reyn94]. This is due to having fewer training data in this work. The results showed that fusion of features increases the percentage of the correctly identified frames, due to the addition of features not captured by a single algorithm. Fusion of MFCC16 with LPCC23 provided the highest increase in accuracy, followed by the TC and VFD. The combination of the features from the 6 algorithms extracted using the VAD that utilized the VFD provided the highest accuracy of 91.60%. As discussed in section 5.5.1, for all the tests 22 keywords are used for training and 22 keywords are used for testing. All the tests were conducted using the SVM, using a linear kernel and setting $C = 10$ due to obtaining the highest accuracy rates in experimental results.

6.3 Thesis Contribution

This thesis contributes to the current knowledge of text-independent speaker verification by the recording of a dataset under controlled conditions, embedding of fractal algorithms to the front-end processing of a speaker verification system, and conduction an experimental sensitivity analysis to determine the effects of fusion of each feature in the feature vector. The main contributions are:

1. In order to provide a set of test signals a group 24 volunteers raised in the province of Manitoba, Canada, were recorded under controlled conditions. The set of test words used were from the Edinburg MRPA which consists of all the phonemes [KiGr08] and is easy and quick to record. This dataset is posted on IEEE dataport [SeKi18] and can be used for research in the field of speech.
2. A detailed description of the recording procedures taken to serve as a guideline to any future recordings is provided. This guideline ensures the repeatability of the recordings with equivalent quality to allow extension of the number of participants to the current dataset or recording of a different dataset with similar quality.
3. The HFD and the VFD are used to add the multifractal features of speech to the front-end processing due to the speed of their computation. It is specified by [KiGr09] that for the VFD algorithm $n_k = 1$ should be omitted to reduce the effect of correlation between adjacent signal samples. However, it was noticed that that omission of $n_k = 1$ causes the VFDT to go significantly over 2 for phonemes that have a turbulent structure such as the fricatives, a matter which is noticed in- [Grie96] as well. It is noticed that with the addition of $n_k = 1$, the VFDT is reduced to a FD close to 2, in some cases slightly crossing 2 which is compatible with HFD.
4. The Weierstrass function and the fBm are used to study the performance of the HFD and the VFD and the effects of addition of colored noise on the estimation of the FD. The experiments show that the VFD has less variability for most of the waveforms and is slightly more accurate for the Weierstrass function with FD 1.7 to 1.9. However, the VFD is more sensitive to the addition of colored noise and has greater error.

5. A VAD algorithm is introduced in section 5.3 which is based on changing the FD of the background noise available in the recorded speech. This algorithm proved to be more effective in comparison to setting a threshold on the energy of the signal. The results show that this algorithm is capable of detecting segments that contain speech and omitting the segments that contain speech but are noisy. This is due to the FD relying on the complexity of the signal rather than the amplitude.
6. An experimental sensitivity analysis was conducted showing that the fusion of features improves the accuracy of the classifications. The fusion of features yielded a 43 feature vector. It was intended not to significantly exceed the 39 feature vector commonly used in the literature since with the increased number of features the number of training samples for reliable density estimation grows exponentially [JaDM00]. The SVM was used to train the models and test them.

6.4 Limitations and Future Work

This thesis provided some key contributions to the area of text-independent speaker verification. However, there are some limitations in this study which can be improved. These limitations and areas for possible extensions to this research are:

1. The recorded dataset consisted of 44 keywords which were used for the experiments in this work. This dataset can be expanded to include sentences. Sentences can consist of more training data and thus the experiments can be repeated with more training data to see if they yield the same accuracy. Moreover, availability of sentences in a dataset makes the data suitable for research on speaker recognition.

2. The SVM was used to train and test a model. A possible extension to this work can be the investigation of the performance of the feature vectors extracted using other machine learning algorithms. The current machine learning algorithm popular in the literature is the deep learning algorithms. Meanwhile, the traditional machine learning algorithms such as the Gaussian mixture models and the hidden Markov model can be used for the comparison of the results.

3. A VAD detection based of FD was introduced in this work. A possible extension to this work is the investigation of this algorithm for other application. A possible field where this algorithm could improve the results is speaker recognition due to the elimination of segments that contain noisy speech. This objective can be achieved by identifying the segments that contain speech and normalizing the segments in time. For this purpose, normalization is required to remove the variability in the speed of speaking between one speaker to another.

Reference

- [AlMi04] Ronald Allen and Duncan Mills, *Signal Analysis: Time, Frequency, Scale, and Structure*. Hoboken, NJ: Wiley-Interscience, 2004, 966 pp. {ISBN-13: 978-0-471-23441-8}
- [Apex17] Apex electronics, *MWS-206DLX*, 2017. Retrieved October 30, 2017 from apexelectronics at:
http://apexelectronics.com/mic_accessories/pop_filters/product/mws-206dlx/
- [Auda17] Audacityteam, *Audacity*, 2017. Retrieved October 24, 2017 from Audacityteam at:
<http://www.audacityteam.org/home/>
- [BBFG04] Frederic Bimbot, Jean-Francois Bonastre, Corinne Fredouille, Guillaume Gravier, Ivan Magrin-Chagnolleau, Sylvain Meignier, Teva Merlin, Javier Ortega-Garcia, Dijana Petrovska-Delacretaz, and Douglas Reynolds, "A tutorial on text-independent speaker verification," *Springer*, pp. 430-451, April 2004. Retrieved October 9, 2017 from Springer at:
<https://link.springer.com/content/pdf/10.1155/S1110865704310024.pdf>
- [Beig11] Homayoon Beigi, *Fundamentals of Speaker Recognition*. Springer, 2011 (1st Ed), 942 PP. {ISBN-13: 978-0-387-77592-0}
- [Blue16] Bluemic, *Blue Yeti Manual*. 2016. Retrieved August 27, 2016 from BlueMic at:
http://cd.bluemic.com.s3.amazonaws.com/pdf/manuals/yeti/Yeti%20Manual_English.pdf

-
- [Blue17] Bluemic, *Yeti*. 2017. Retrieved October 24, 2017 from BlueMic at: <http://www.bluemic.com/products/yeti/>
- [CaRD03] Joseph Campbell, Douglas Reynolds, and Robert Dunn, “Fusing high- and low-level features for speaker recognition,”. In *Proc. 8th European Conf. on speech communication and technology*, ISCA03 (Geneva, Switzerland; 1-4 September 2003) 4 pp. 2003. Retrieved January 1, 2018 from ISCA at: http://www.isca-speech.org/archive/archive_papers/eurospeech_2003/e03_2665.pdf
- [Carr93] Philip Carr, *Phonology: An Introduction (Palgrave Modern Linguistics)*, Palgrave Macmillan, 1993, 336 PP. {ISBN-13: 978-0-333-51908-6}
- [CCRS06] William Campbell, Joseph Campbell, Douglas Reynolds, Elliot Singer, and Pedro Torres-Carrasquillo, “Support vector machine for speaker and language recognition,” *Elsevier*, vol.20, pp. 210-229, 2004. Retrieved January 1, 2017 from science Direct at: <https://www.sciencedirect.com/science/article/pii/S0885230805000318>
- [ChLi15] Chih-Chung Chang and Chih-Jen Lin, “LIBSVM: A library for support vector machines,” *ACM Transaction on Intelligent Systems and Technology*, vol. 2, no. 27, 2011. {DOI: 10.1145/1961189.1961199}. Retrieved November 12, 2015 from ACM at: <http://dl.acm.org/citation.cfm?id=1961199>
- [ChLu09] Shi Huang Chen and Yu Ren Luo, “Speaker verification using MFCC and support vector machine,” in *Proc. Int. Engineers and Computer Scientists conf.*, IMECS09, (Hong Kong; 18-20 March 2009), vol. 1, 4 pp., 2009. Retrieved December 25, 2015 from IAENG at: http://www.iaeng.org/publication/IMECS2009/IMECS2009_pp532-535.pdf
- [ChWC97] Ke Chen, Lan Wang, and Huisheng Chi, “Methods of combining multiple classifiers with different features and their applications to textindependent speaker recognition,” *Int. J. Pattern Recognition Artificial Intelligence*. Vol. 11, no. 3, pp. 417–445, May 1997. Retrieved January 1, 2018 from the university of Manchester at: <http://www.cs.man.ac.uk/~kechen/publication/ijprai97.pdf>
- [ChZF16] Rania Chakroun, Leila Beltaifa Zouari, and Mondher Frikha, “An improved approach to text-independent speaker recognition,” *Int. journal of advanced computer science and applications*., vol. 7, no. 8, pp. 343-348, 2016. Retrieved October 27, 2017 from SAI at: https://thesai.org/Downloads/Volume7No8/Paper_46-An_Improved_Approach_for_Text_Independent_Speaker_Recognition.pdf
- [CoZa11] Jonathan Cohen and Ahmed Zayed, *Wavelets and Multiscale Analysis: Theory and Applications*. New York, NY: Springer, 2011, 338 pp. {ISBN-13: 978-0-8176-8095-4}
- [Ekma12] Laurie Lundy-Ekman, *Neuroscience: Fundamentals for Rehabilitation*. Elsevier, 2012 (4th Ed), 552 PP. {ISBN-13: 978-1-455-70643-3}
-

-
- [EsEC96] Anna Esposito, Eugene Ezin, and Michele Ceccarelli, “Preprocessing and neural classification of English stop consonants [b,d,g,p,t,k]”, in *Proc. Fourth Int. spoken language conf., ICSLP96* (Philadelphia, PA, 3-6 October), 4 pp., 1996. Retrieved October 13, 2016 from research gate at:
https://www.researchgate.net/publication/221481550_Preprocessing_and_neural_classification_of_English_stop_consonants_b_d_g_p_t_k
- [Fere15] Ken Ferens, “Applied computational intelligence course notes,” *Technical Report*, University of Manitoba, 2015.
- [FiGM07] M.J.T FitzGerald, Gregory Gruener, and Estomih Mtui, *Clinical Neuroanatomy and Neuroscience*. Elsevier, 2006 (5st Ed), 368 PP. {ISBN-13: 978-1-416-03445-2}
- [Grie96] Warren Grieder, *Variance fractal dimension for signal feature enhancement and segmentation from noise*, M.Sc. thesis, University of Manitoba, 1996.
- [Groch01] Karlheinz Grochenig, *Foundations of Time-Frequency Analysis*. New York, NY: Springer, 2001, 360 pp. {ISBN-13: 978-0-8176-4022-4}
- [HaDC04] Asmaa El Hannani, Dijana Petrovska-Delacretaz, and Gerard Cholle, “Linear and non-linear fusion of ALISP-based and GMM systems for textindependent speaker verification.” In *Proc. Speaker Odyssey: the Speaker Recognition Workshop, Odyssey04*, (Toledo, Spain; May 2004) pp. 111–116. Retrieved January 1, 2018 from ISCA at:
http://www.isca-speech.org/archive_open/archive_papers/odyssey_04/ody4_111.pdf
- [Higu88] T Higuchi, “Approach to an irregular time series on the basis of the fractal theory,” *Physics D*, vol. 31, no. 2, pp. 277-283, 1998. Retrieved January 06, 2016 from *Science direct* at:
<http://www.sciencedirect.com.uml.idm.oclc.org/science/article/pii/0167278988900814>
- [IEEE17] IEEE, *IEEE Dataport*, 2017. Retrieved December 01, 2017 from IEEE Dataport at:
<https://iee-dataport.org/>
- [JaDM00] Anil Jain, Robert Duin, and Jianchang Mao, “Statistical pattern recognition: a review,” *IEEE Trans. Pattern Analysis and Machine Intelligence.*, vol. 22, no. 1, pp. 4-37, January 2000. Retrieved January 15, 2018 from Bilkent University at:
http://www.cs.bilkent.edu.tr/~saksoy/courses/cs551-Spring2009/papers/jain00_pr_survey.pdf
- [KeTS92] Laura Jennings Kepler, Mark Terry, and Richard Sweetman, “Telephone usage in the hearing-impaired population,” *Ear and Hearing*, vol. 13, no. 5, 1992, PP. 311 – 330. Retrieved May 10, 2018 from researchgate at :
-

- https://www.researchgate.net/publication/21668812_Telephone_Usage_in_the_Hearing-Impaired_Population
- [KiAl09] Tomi Kinnunen and Paavo Alki, “On separating glottal source and vocal tract information in telephony speaker verification.” In: *Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2009)*, Taipei, Taiwan, April 2009, pp. 4545–4548.
- [KiGr08] Witold Kinsner and Warren Grieder, “Speech segmentation using multifractal measures and amplification of signal features,” in *Proc. Seventh IEEE Cognitive Informatics conf., ICCI08*, (Stanford, CA; 14-16 August 2008), 7 pp., 2008. Retrieved July 1, 2015 from IEEE xplore at:
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4639188&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D463918
- [KiGr09] Witold Kinsner and Warren Grieder, “Amplification of signal features using variance fractal dimension trajectory,” in *proc. IEEE Cognitive Informatics Conf., ICCI 09*, (15-17 June, 2009), 4 pp, 2009. Retrieved June 18, 2015 from *IEEE explore* at:
<http://ieeexplore.ieee.org/document/5250750/?reload=true>
- [KiLi10] Tomi Kinnunen and Haizhou Li, “An overview of text-independent speaker recognition: from features to supervectors,” *Elsevier*, vol. 52, no. 1, pp. 12-40, 2010. Retrieved October 10, 2015 from *sciencedirect* at:
<http://www.sciencedirect.com/science/article/pii/S0167639309001289>
- [Kins94a] Witold Kinsner, “Fractal dimension: Morphological, entropy, spectrum, and variance classes,” *Technical Report*, University of Manitoba, 1994.
- [Kins94b] Witold Kinsner, "Batch and real-time computation of a fractal dimension based on variance of a time series," *Technical Report*, University of Manitoba, 1994.
- [Kins05] Witold Kinsner, “A unified approach to fractal dimensions”, in *Proc. fourth IEEE Cognitive Informatics conf., ICCI05*, (8-10 August 2005), 14 pp., 2005. Retrieved June 13, 2015 from IEEE xplore at:
<http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1532616&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F10264%2F32698%2F01532616>
- [Kins08] Witold Kinsner, “Complexity and its measures in cognitive and other complex systems.” in *Proc. seventh IEEE Cognitive Informatics & Cognitive Computing conf., ICCI*CC 08*, (Stanford, CA; 14-16 August 2008), 17 pp.
Retrieved May 02, 2018 from IEEE xplore at:
<https://ieeexplore.ieee.org/document/4639147/>

-
- [Kins11] Witold Kinsner, “It’s time for multiscale analysis and synthesis in cognitive systems”, in *proc. IEEE tenth Int. Cognitive Informatics & Cognitive Computing Conf, ICCI*CC11*, (Banff, AB; 18-21 August 2011), 4 pp., 2011. Retrieved February 10, 2016 from University of Calgary at:
http://www.ucalgary.ca/icci_cc/files/icci_cc/icci11-Multiscale_v29-NoF.pdf
- [Kins15] Witold Kinsner, “Fractal and chaos engineering course notes,” *Technical Report*, University of Manitoba, 2010.
- [KrBo13] Dirk Kroese and Zdravko Botev, “Spatial process generation,” pp. 41, August 2013. Retrieved May 18, 2017 from Researchgate at:
https://www.researchgate.net/publication/254863177_Spatial_Process_Generation
- [Lang92] Armein Langi, *Code-excited linear predictive coding for high-quality and low bit-rate speech*, M.Sc. thesis, University of Manitoba, 1992.
- [LaSK97] Armein Langi, Kudrat Soemintapura, and Witold Kinsner, “Multifractal processing of speech signals,” in *Proc. Int. Information, Communication and Signal Processing Conf., ICICS97*, (Singapore; 9-12 September 1997), 5 pp., 1997. Retrieved December 13, 2015 from IEEE xplore at:
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=647154&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D64715
- [Mann17] Robert Mannell, Macquarie University, *An introduction to speech production*, Sydney, Australia, 2017. Retrieved January 3, 2016 from Macquarie University at:
<http://clas.mq.edu.au/speech/phonetics/phonetics/introduction/>
- [Math16] Mathworks, *colored noise generator*, Mathworks, 2016. Retrieved December 25, 2016 from Mathworks at:
<https://www.mathworks.com/help/dsp/ref/dsp.colorednoise-class.html>
- [Math17] Mathworks, *Fast Fourier transform*, 2017. Retrieved November 20, 2017 from Mathworks at: <https://www.mathworks.com/help/matlab/ref/fft.html>
- [Micr18] Microtia, *Hearing and Ear Anatomy*. San Antonio, TX, 2018. Retrieved January 28, 2018 from Microtia at: <http://microtia.net/hearing-ear-anatomy/>
- [MiZh03] Hou Li-Min and Wang Shou-Zhong, “Speaker identification based on fractal dimension,” *Shanghai University*, vol. 7, no. 1, pp. 60-63, 2003. {DOI: 10.1007/s11741-003-0053-4} Retrieved August 13, 2015 from Springer at:
<http://link.springer.com/article/10.1007/s11741-003-0053-4>
- [NeMM06] Fulufhelo Nelwamondo, Unathi Mahola, and Tshilidzi Marwala, “Multi-scale fractal dimension for speaker identification system,” in *Proc. eight int. Automatic Control, Modeling and Simulation Conf., WSEAS06*, (Prague, Czech Republic; 12-14 March 2006), 7 pp., 2006. Retrieved August 6, 2015 from WSEAS at:
-

- <http://www.wseas.us/e-library/conferences/2006prague/papers/514-164.pdf>
- [Pott08] Michael Potter, *Convergence of dynamical features under ICA with application to fetal ECG*, Phd thesis, University of Manitoba, 2008.
- [RaDu09] B.S. Raghavendra and D. Narayana Dutt, "A note on fractal dimensions of biomedical waveforms," *Computers in Biology and Medicine*, vol. 39, no. 11, pp. 1006-1012, 2009. Retrieved January 17, 2016 from *ScienceDirect* at: <http://www.sciencedirect.com/science/article/pii/S0010482509001486>
- [Rang01] Rangaraj Rangayyan, *Biomedical Signal Analysis*. Wiley-IEEE press, 2001 (1st Ed), 552 PP. {ISBN-13: 978-0-471-20811-2}
- [ReRo95] Douglas Reynolds and Richard Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE transactions on speech and audio processing*, vol. 3, no. 1, pp. 72-83, January 1995. Retrieved September 13, 2016 from IEEE xplore at: <http://ieeexplore.ieee.org/document/365379/>
- [Reyn94] Douglas Reynolds, "Experimental evaluation of features for robust speaker identification," *IEEE transaction on speech and audio processing*, vol. 2, no. 4, pp. 639-643, October 1994. Retrieved January 10, 2018 from IEEE xplore at: <http://ieeexplore.ieee.org/document/326623/>
- [Reyn02] Douglas Reynolds, "An overview of automatic speaker recognition" in *Proc. Acoustics, speech, and signal processing conf. ICASSP02*, (Orlando, FL, USA; May 13-17 2002), 4 pp., 2002. Retrieved January 12, 2018 from IEEE xplore at: <http://ieeexplore.ieee.org/document/5745552/>
- [Rose02] Philip Rose, *Forensic Speaker Identification*. CRC Press, 2002 (1st Ed), 384 PP. {ISBN-13: 978-0-415-27182-7}
- [SeKi18] Sina Sedigh and Witold Kinsner, "A Manitoban Speech Dataset", *IEEE Dataport*, 2018. [Online]. Retrieved January 25, 2018 from IEEE dataport at: <http://dx.doi.org/10.21227/H2KM16>.
- [SHGK01] Jungpa Seo, S. Hong, J Gu, M Kim, I Baek, Y Kwon, K Lee, and Sung Il Yang, "New speaker recognition feature using correlation dimension," in *Proc. IEEE Int. Symposium on industrial electronics*. ISIE2001, (Pusan, South Korea; 12-16 June 2001), 3 pp., 2001. Retrieved July 22, 2015 from IEEE xplore at: http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=931843&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D931843
- [Skyp17] Skypaw, *Decibel X*, 2017. Retrieved November 11, 2017 from skypaw at: <http://www.skypaw.com/decibel10.html>

-
- [SOAM02] Y. Shena, Eckehard Olbrich, P. Achermann, and P.F. Meier “Dimensional complexity and spectral properties of the human sleep EEG” *Elsevier*, vol. 114, pp. 199-209, 2002. Retrieved from July 1, 2015 from *NCBI* at:
http://www.ncbi.nlm.nih.gov/pubmed/1255922642d061e8311d&enrichSource=Y292ZXJQYWdlOzI2MDAwNjk3MTtBUzoxNzQ5ODExNDUxNzgxMTRAMTQxODczMDE2NTAxMg%3D%3D&el=1_x_2
- [Tech17] Techsmith, *Camtasia*, 2017. Retrieved October 24, 2017 from Techsmith at:
<https://www.techsmith.com/video-editor.html>
- [TeKi16] Jesus David Terrazas Gonzalez and Witold Kinsner, “Zero-crossing analysis of Lévy walks for real-time feature extraction,” in *Proc IEEE Int. Electro/Information Technology Conf.*, EIT, (Grand Forks, ND, USA; April 2016), 9 pp., 2016. Retrieved September 28, 2016 from IEEE explore at:
<http://ieeexplore.ieee.org/document/7535276/>
- [ToPu11] Robert Togneri and Daniel Pullella, “An Overview of Speaker Identification: Accuracy and Robustness Issues,” In: *IEEE Circuits And Systems Magazine*, Vol. 11, No. 2, pp. 23-61, ISSN: 1531-636X, 2011. Retrieved January 1, 2018 from IEEE xplore at: <http://ieeexplore.ieee.org/document/5871484/>
- [Wang16] Yang Wang, “Machine learning,” *Technical Report*, University of Manitoba, 2015.
- [ZhWZ10] Yuhuan Zhou, Jinming Wang, and Xiongwei Zhang, “Research on speaker recognition based on multifractal spectrum feature,” in *Proc. Second Int. IEEE Computer Modeling and Simulation Conf.*, ICCMS10, (Sanya, Hainan; 22-24 January 2010), 4 pp., 2010. Retrieved November 24, 2015 from IEEE Xplore at:
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=5421347&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D5421347

APPENDIX A

Voice Activity Detection

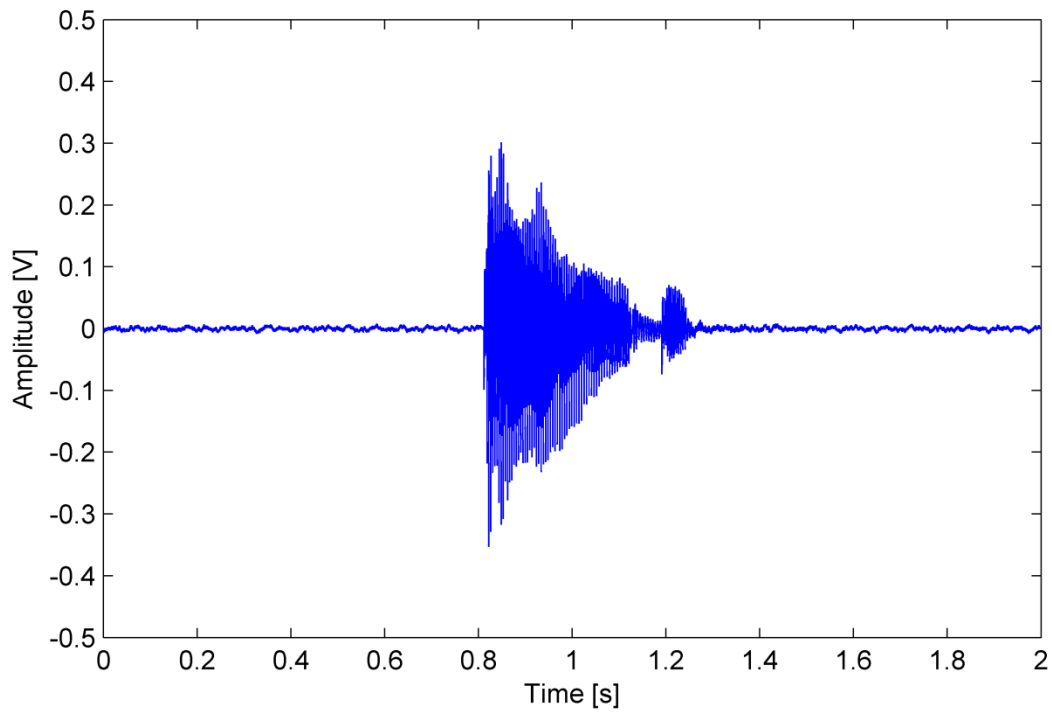


Fig. A.1. The waveform of the utterance “barb”.

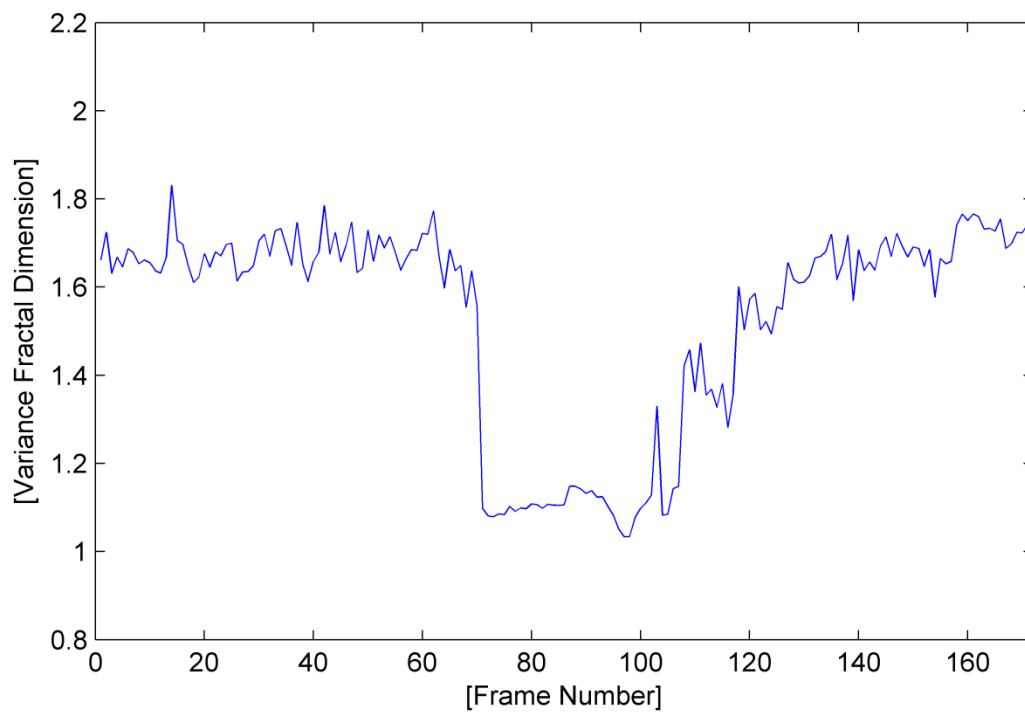


Fig. A.2. The variance fractal dimension trajectory of the utterance “barb”.

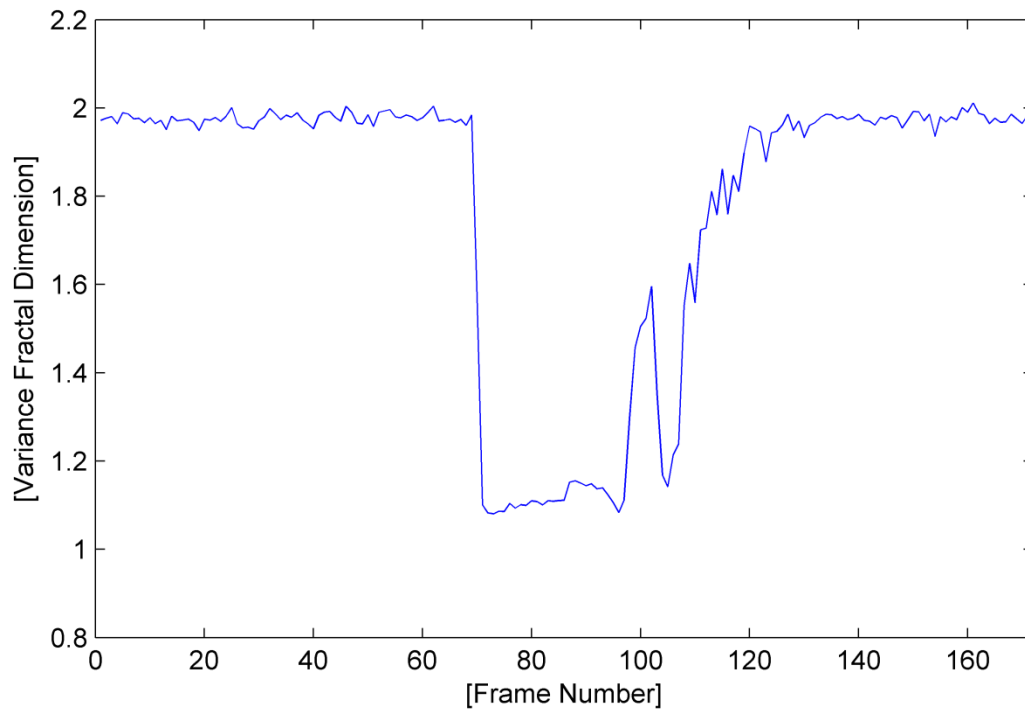


Fig. A.3. The variance fractal dimension trajectory of the utterance “barb” after addition of white noise.

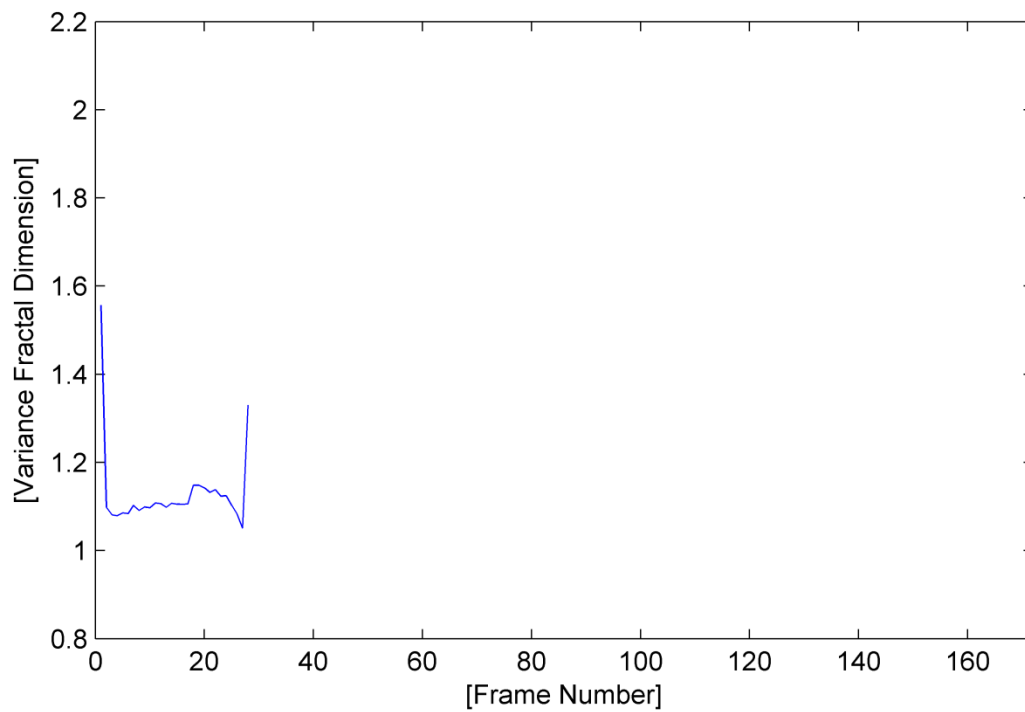


Fig. A.4. The trajectory of the utterance “barb” detected by the voice activity detection algorithm.

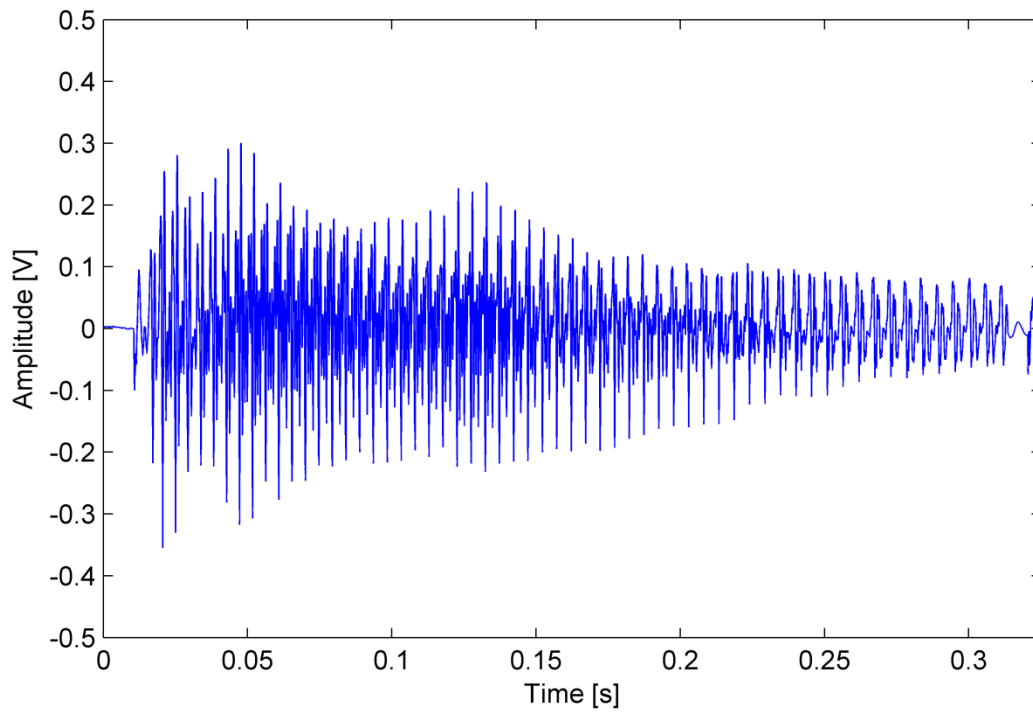


Fig. A.5. The waveform of the utterance “barb” detected by the voice activity detection algorithm.

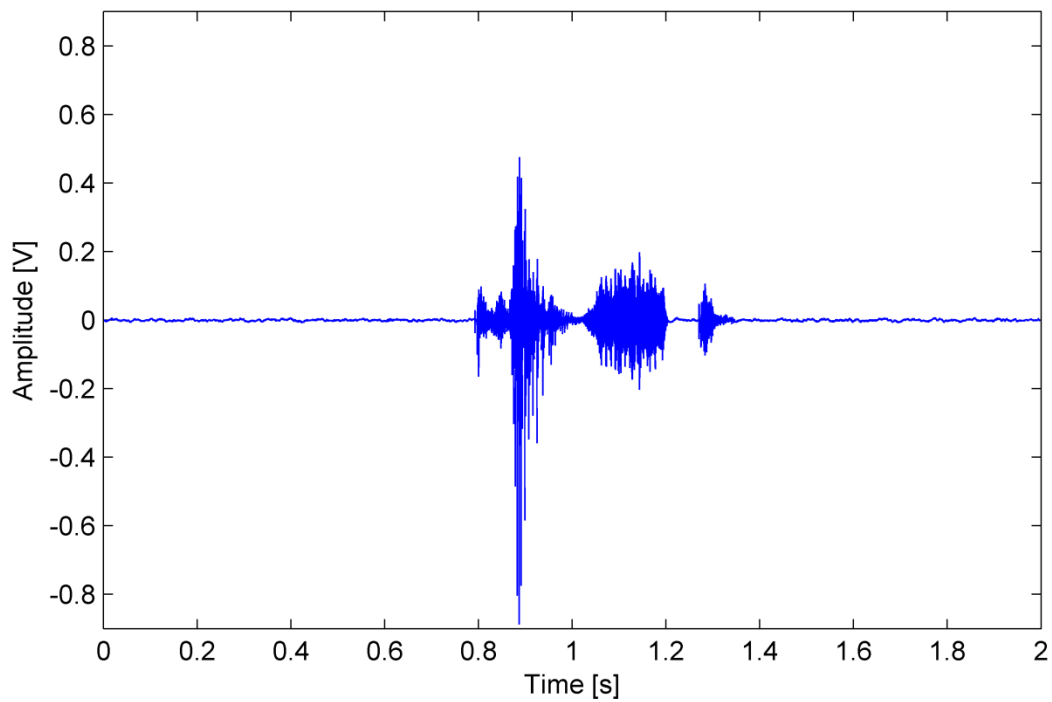


Fig. A.6. The waveform of the utterance “test”.

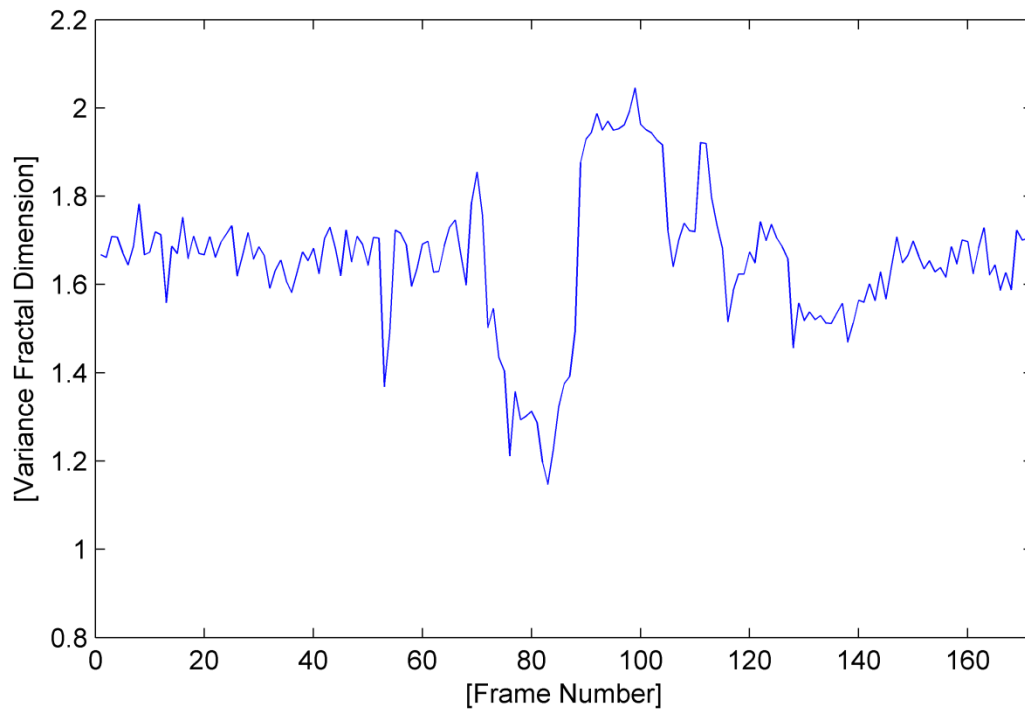


Fig. A.7. The variance fractal dimension trajectory of the utterance "test".

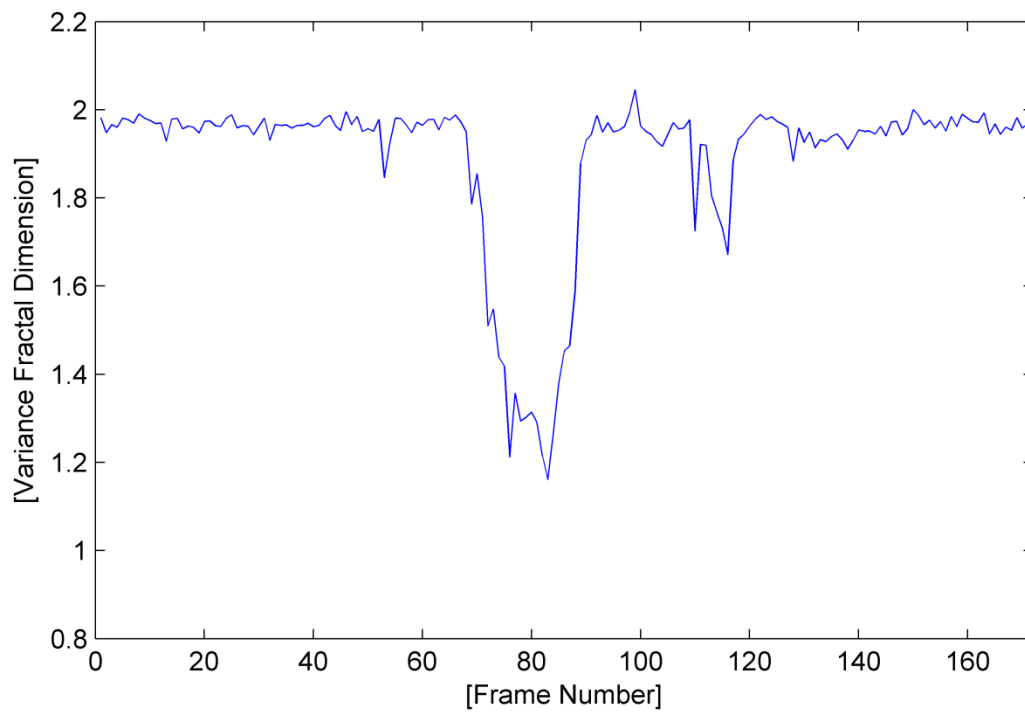


Fig. A.8. The variance fractal dimension trajectory of the utterance "test" after addition of white noise.

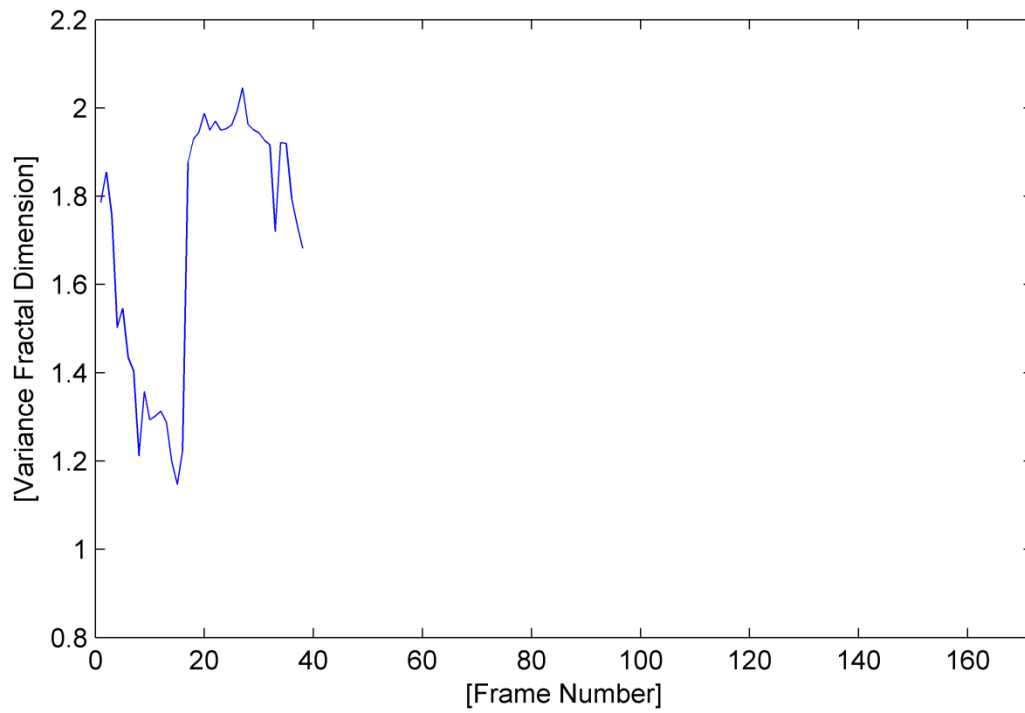


Fig. A.9. The trajectory of the utterance "test" detected by the voice activity detection algorithm.

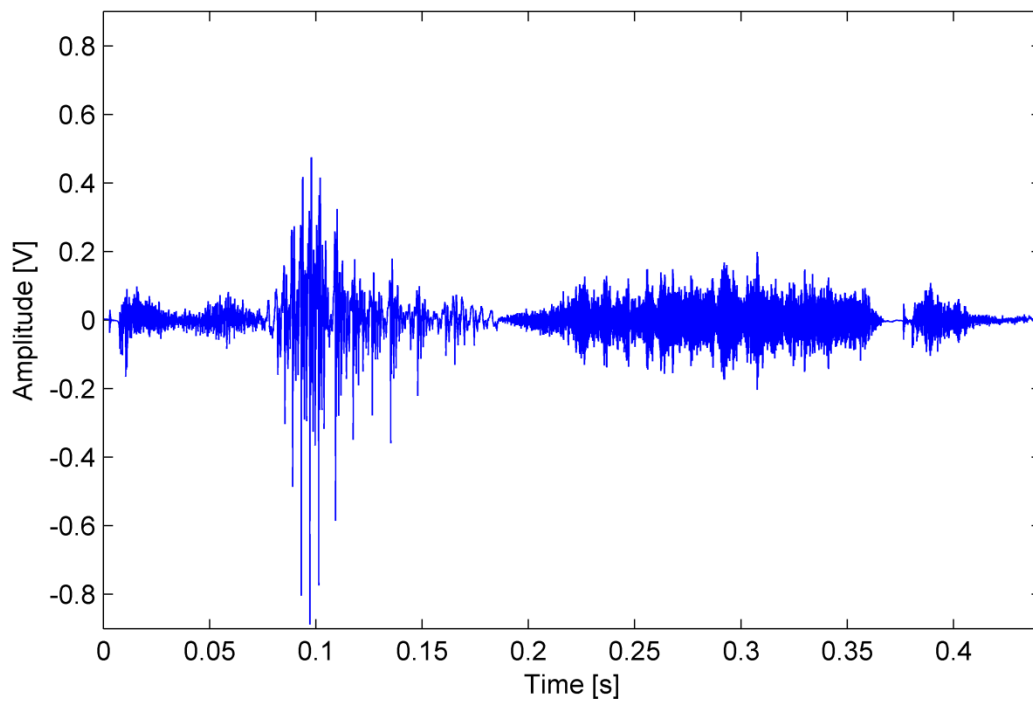


Fig. A.10. The waveform of the utterance "test" detected by the voice activity detection algorithm.

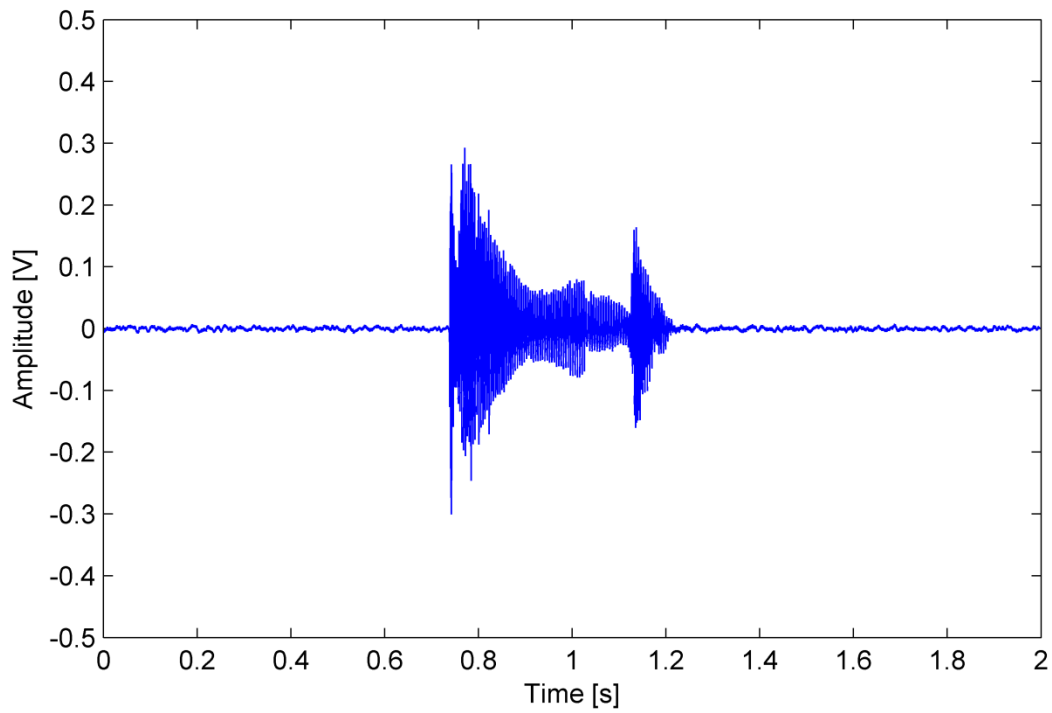


Fig. A.11. The waveform of the utterance "deed".

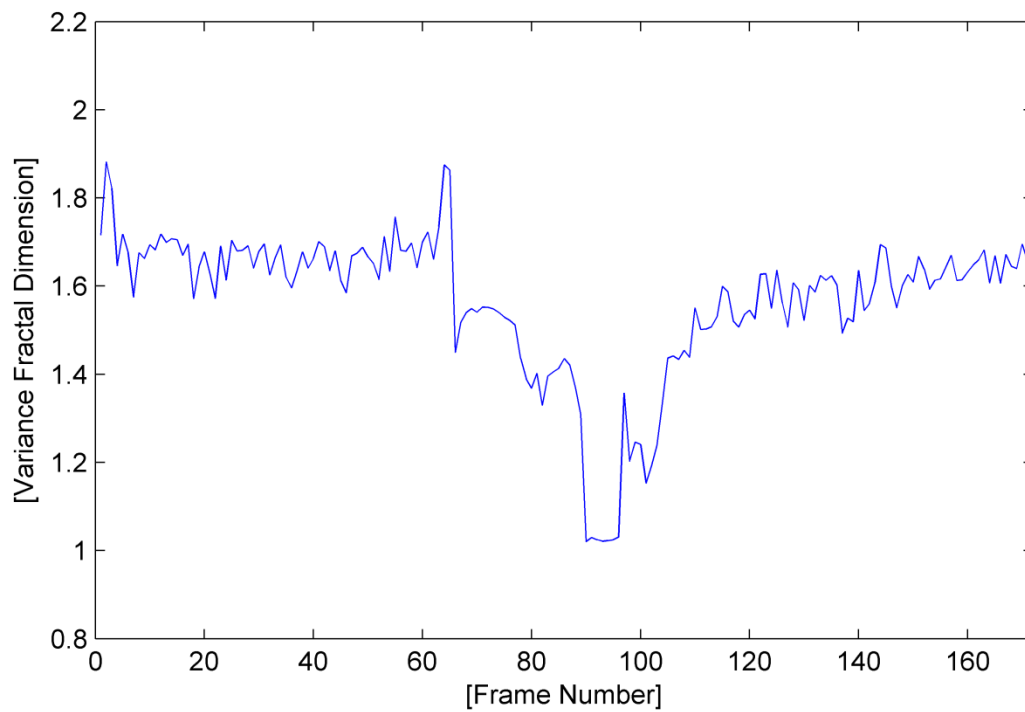


Fig. A.12. The variance fractal dimension trajectory of the utterance "deed".

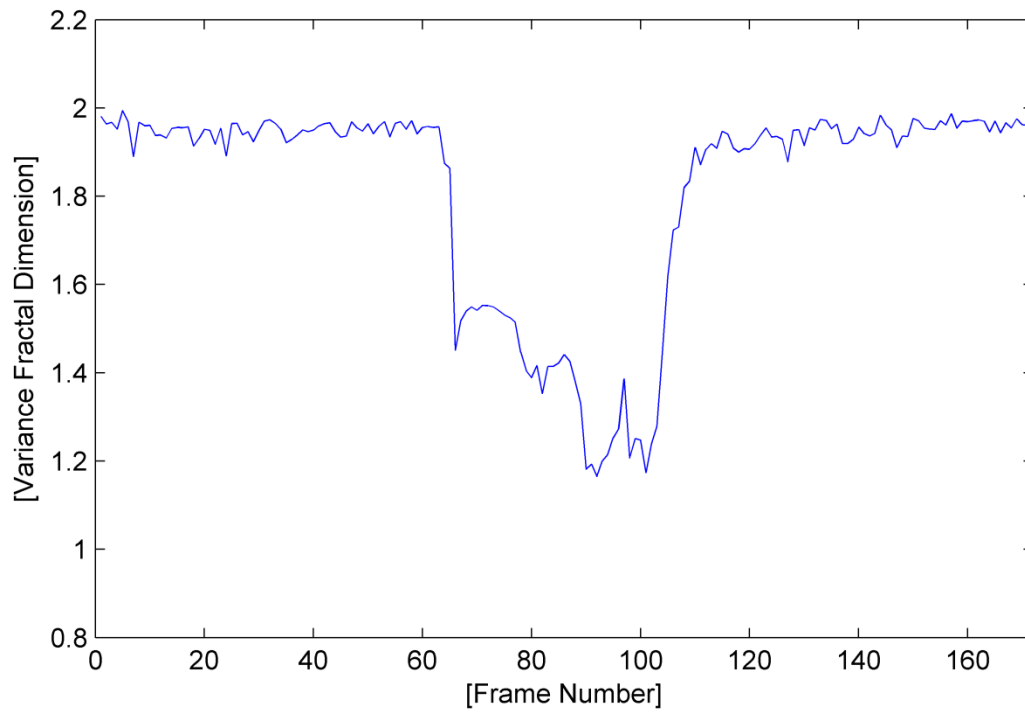


Fig. A.13. The variance fractal dimension trajectory of the utterance “deed” after addition of white noise.

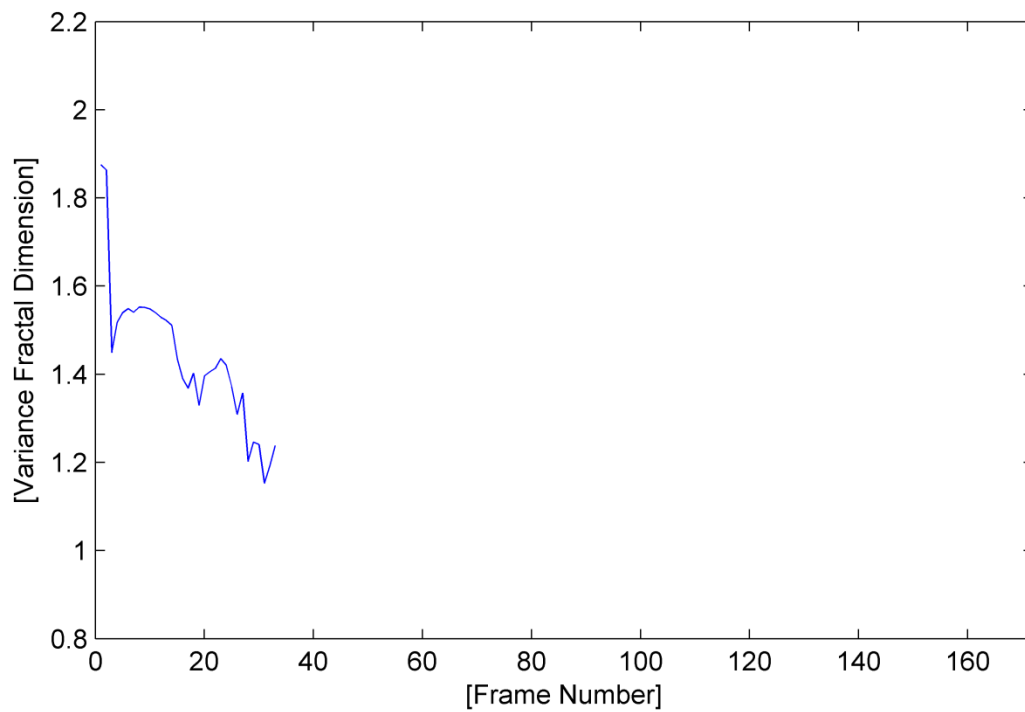


Fig. A.14. The trajectory of the utterance “deed” detected by the voice activity detection algorithm.

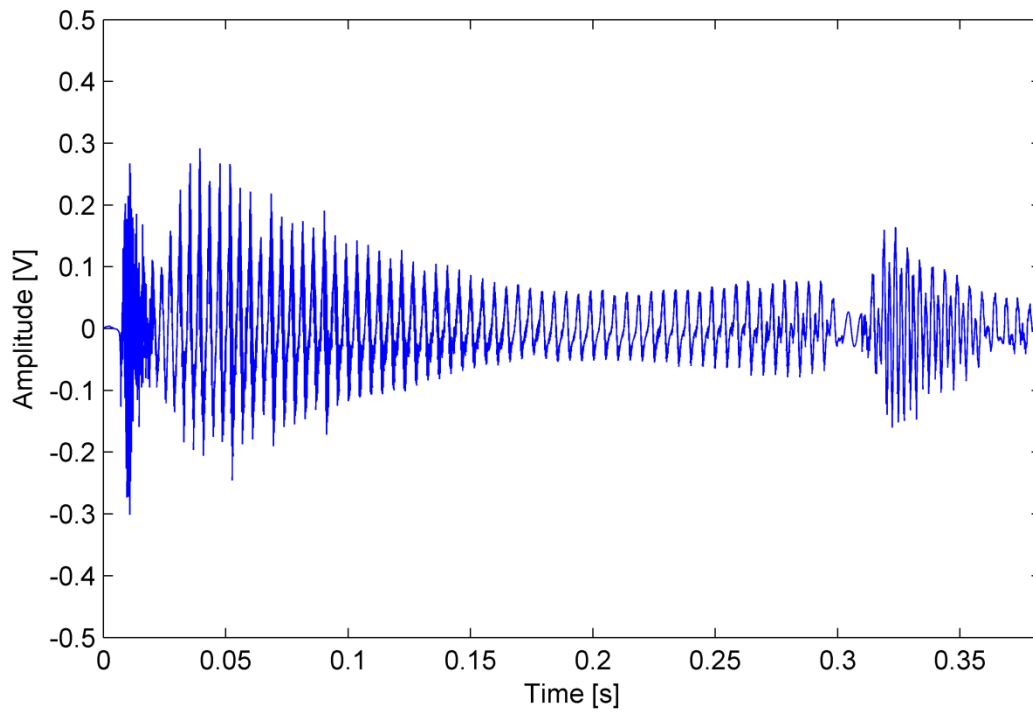


Fig. A.15. The waveform of the utterance “deed” detected by the voice activity detection algorithm.

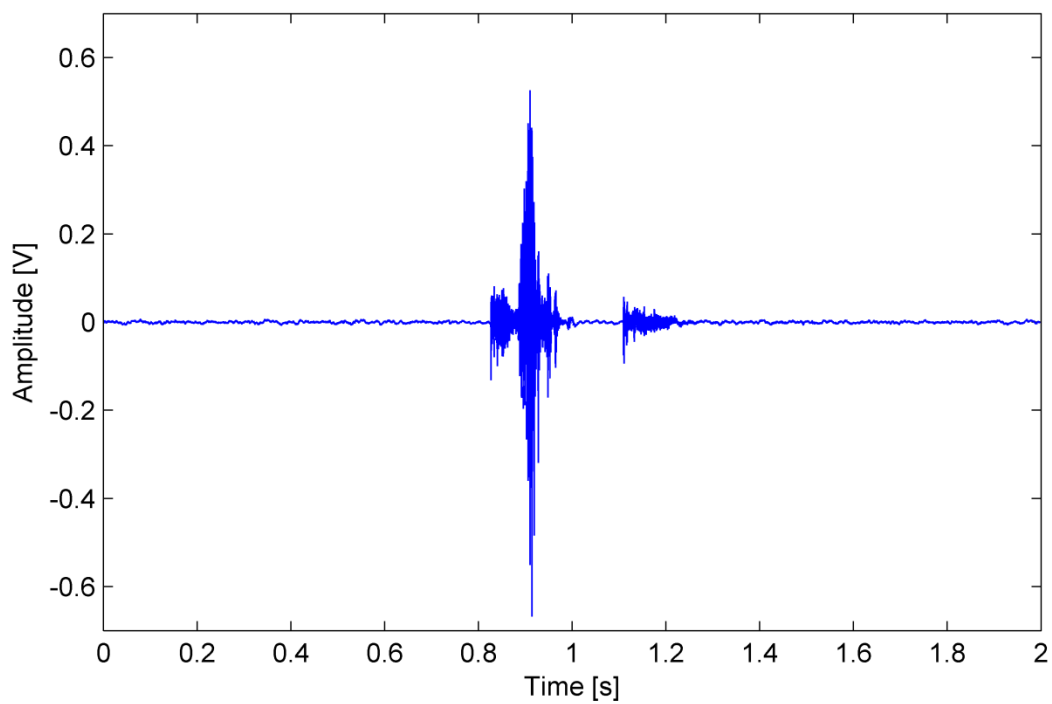


Fig. A.16. The waveform of the utterance “kick”.

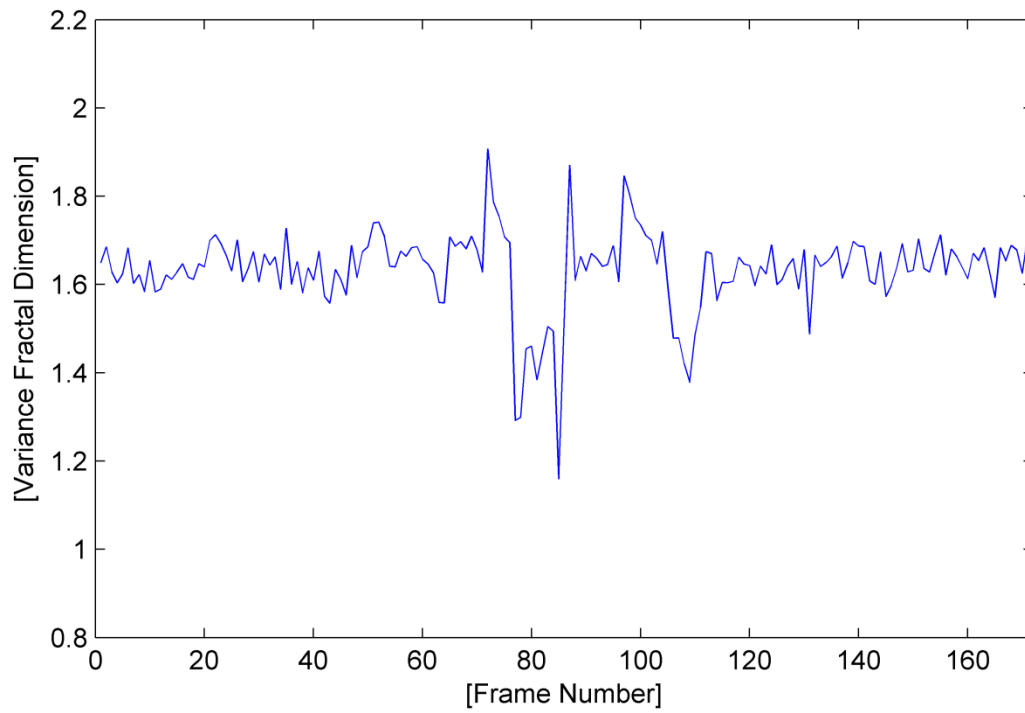


Fig. A.17. The variance fractal dimension trajectory of the utterance “kick”.

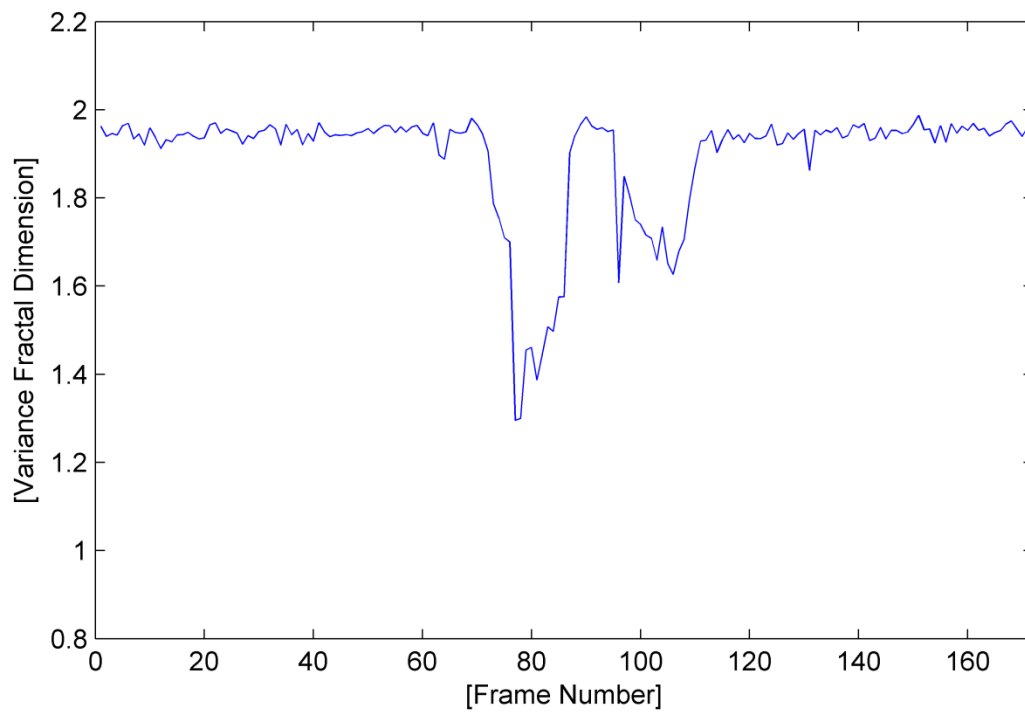


Fig. A.18. The variance fractal dimension trajectory of the utterance “kick” after addition of white noise.

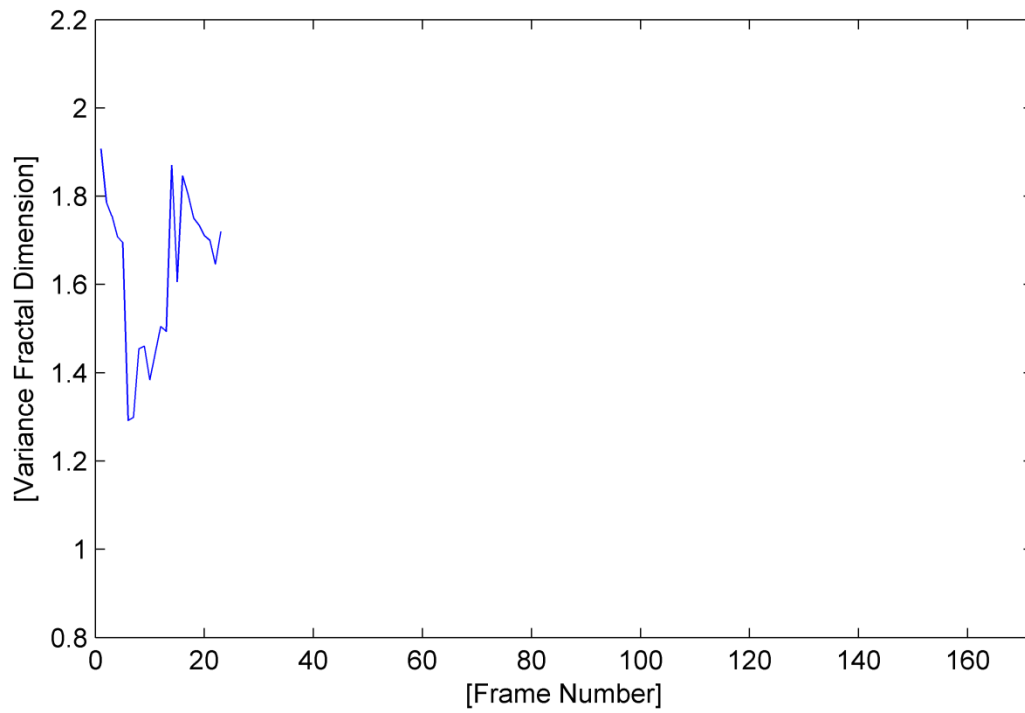


Fig. A.19. The trajectory of the utterance "kick" detected by the voice activity detection algorithm.

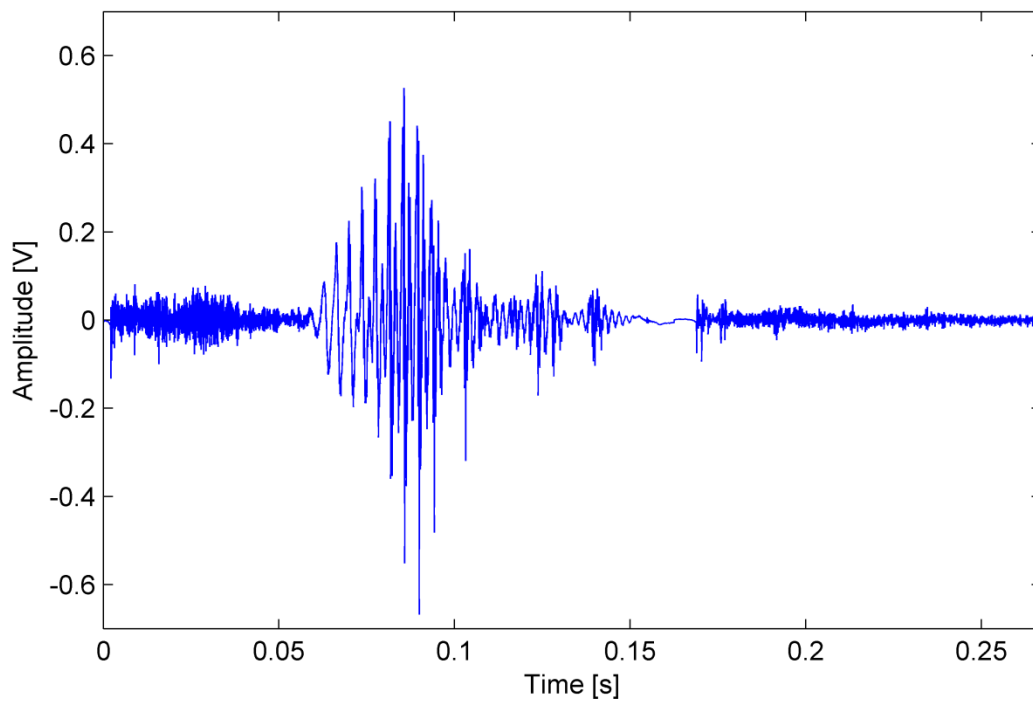


Fig. A.20. The waveform of the utterance "kick" detected by the voice activity detection algorithm.

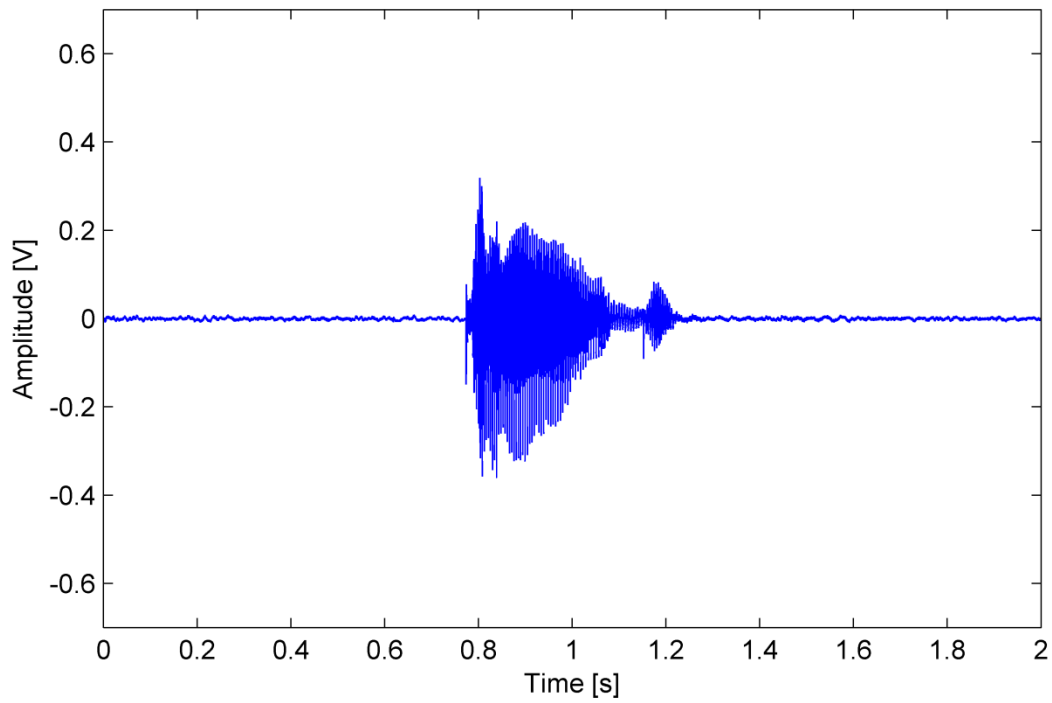


Fig. A.21. The waveform of the utterance "gag".

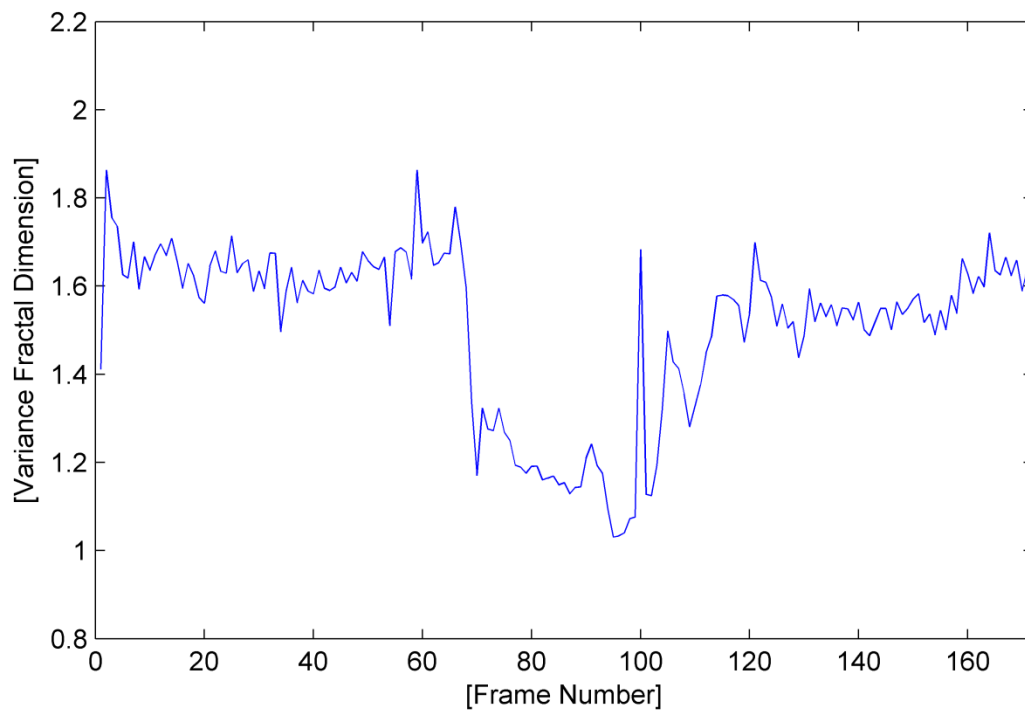


Fig. A.22. The variance fractal dimension trajectory of the utterance "gag".

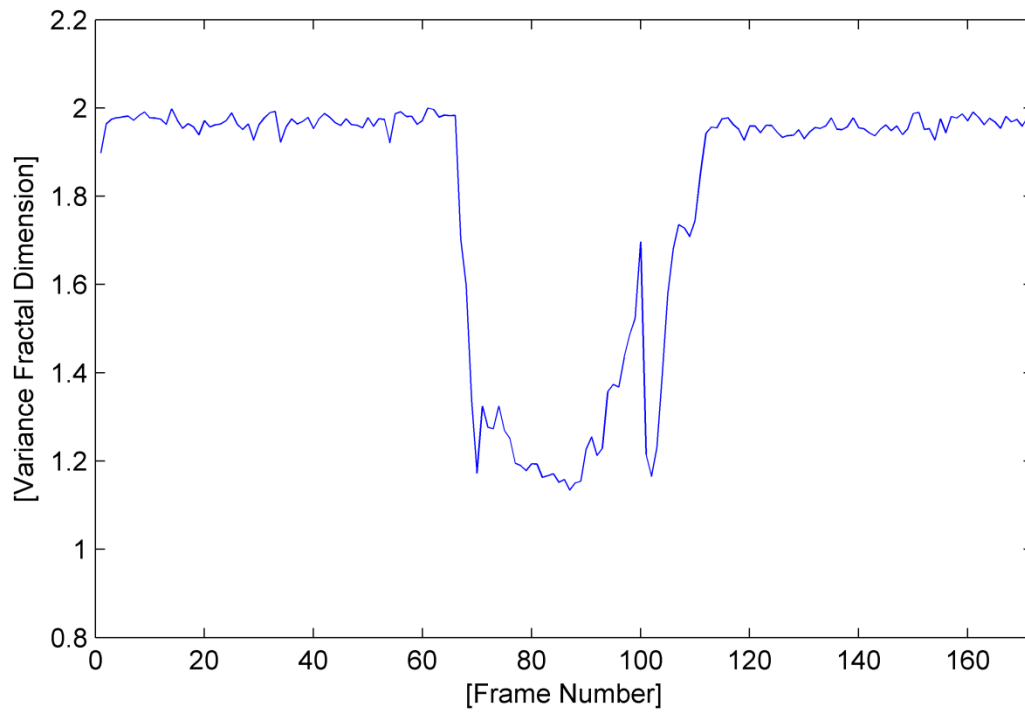


Fig. A.23. The variance fractal dimension trajectory of the utterance “gag” after addition of white noise.

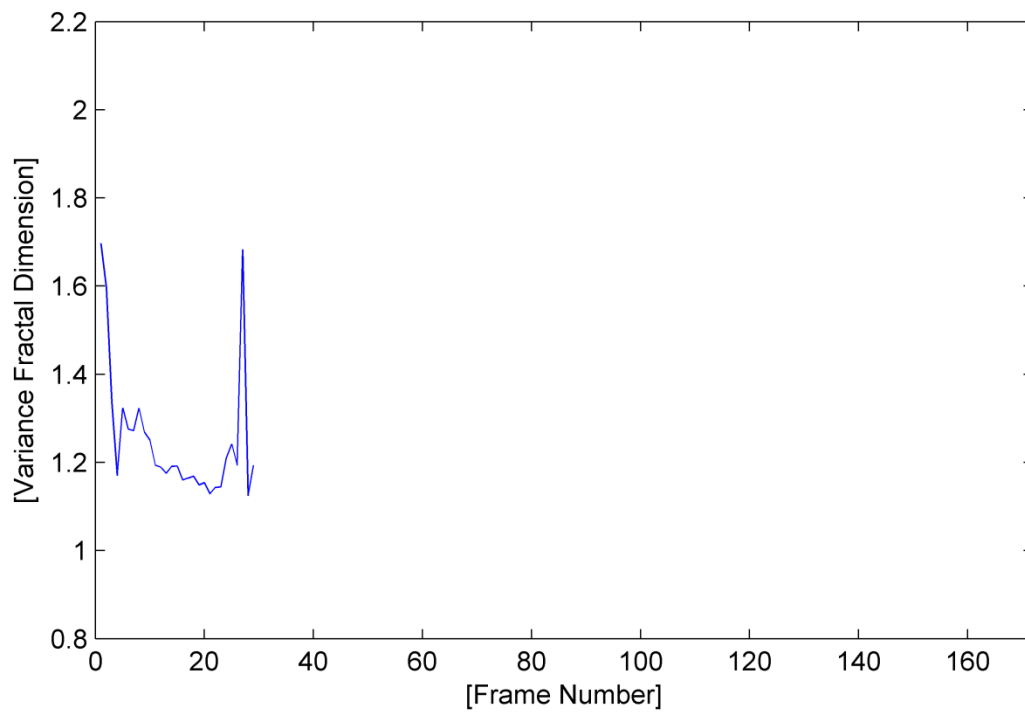


Fig. A.24. The trajectory of the utterance “gag” detected by the voice activity detection algorithm.

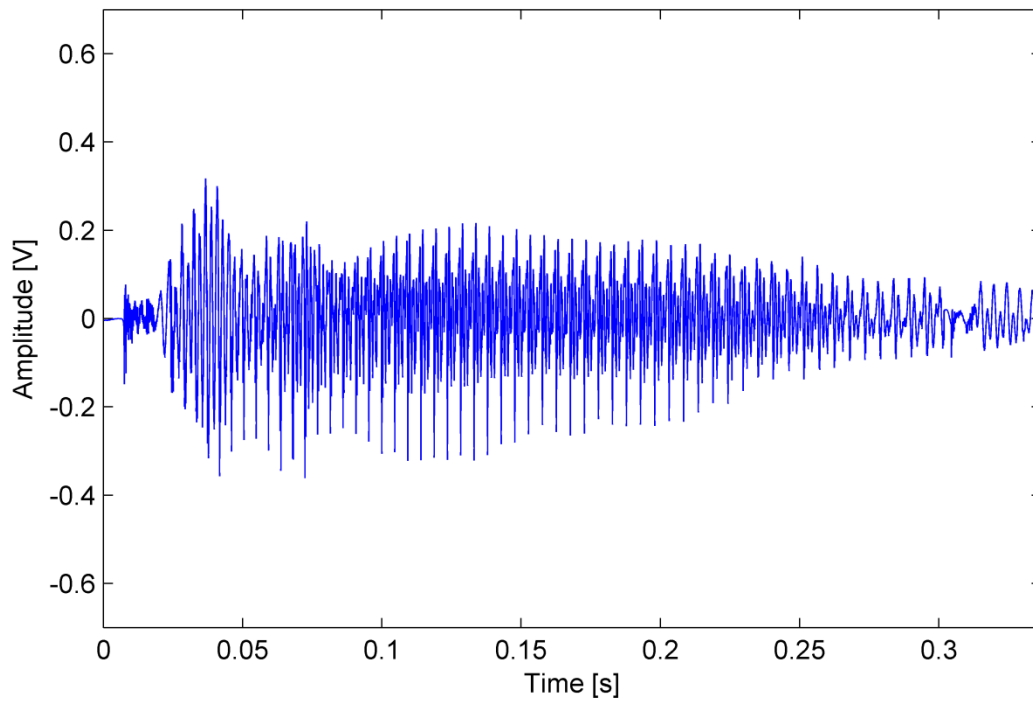


Fig. A.25. The waveform of the utterance “gag” detected by the voice activity detection algorithm.

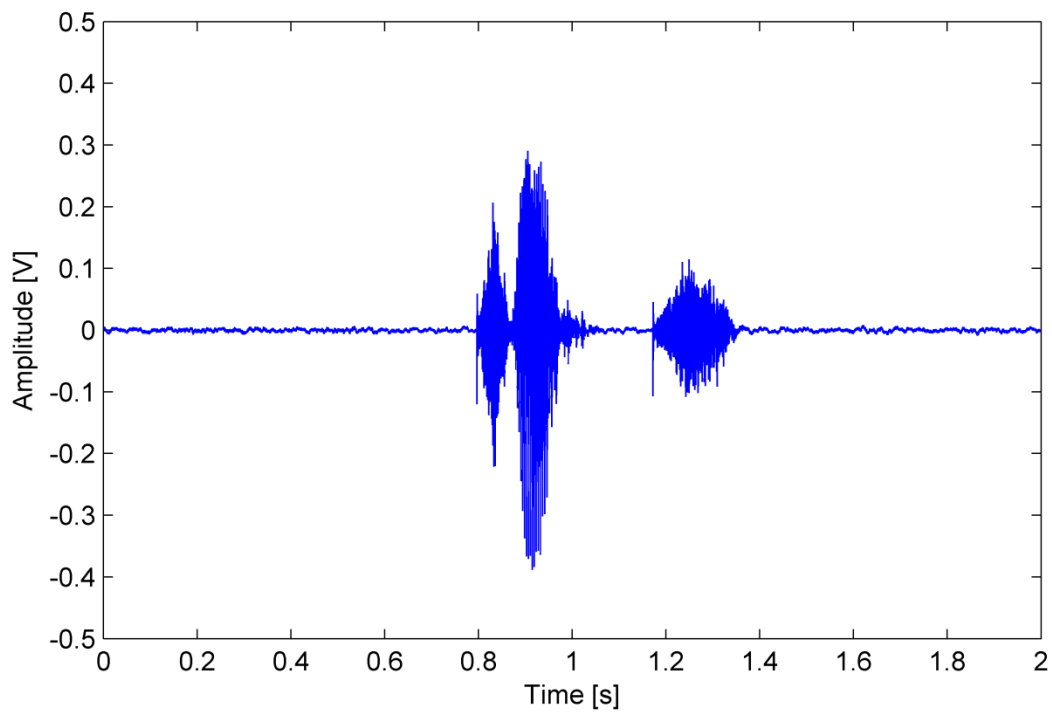


Fig. A.26. The waveform of the utterance “church”.

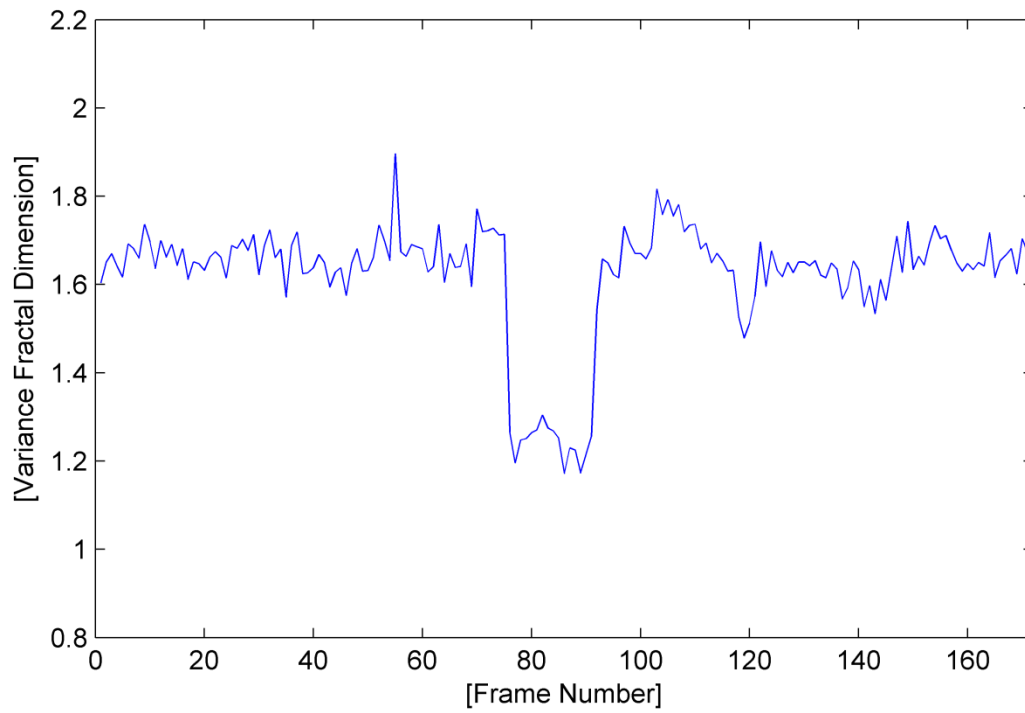


Fig. A.27. The variance fractal dimension trajectory of the utterance "church".

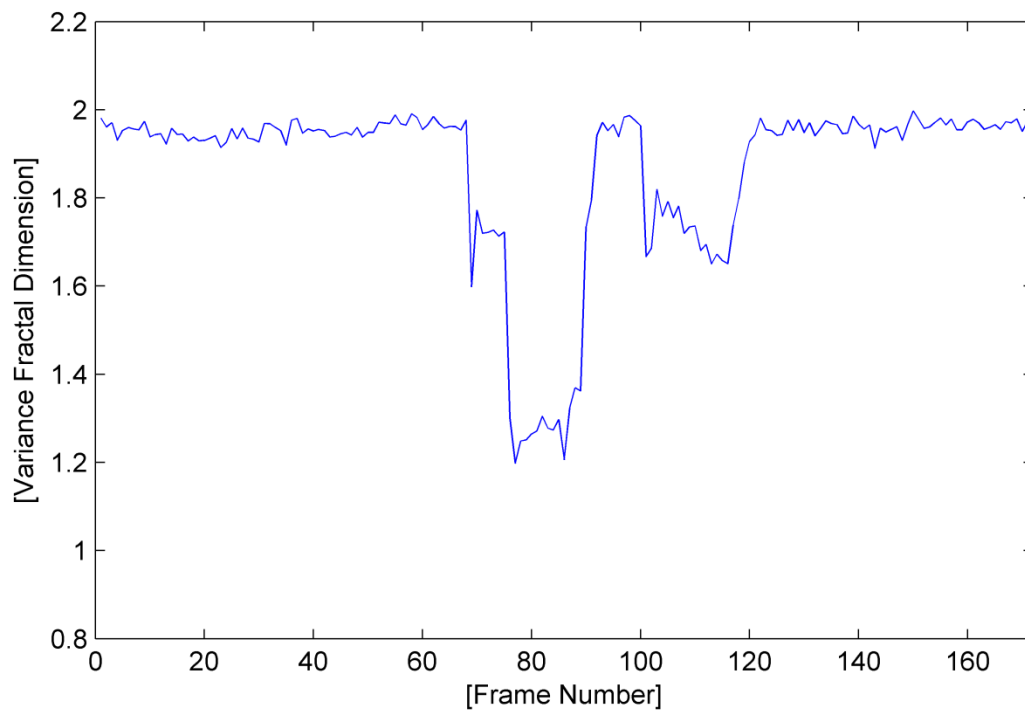


Fig. A.28. The variance fractal dimension trajectory of the utterance "church" after addition of white noise.

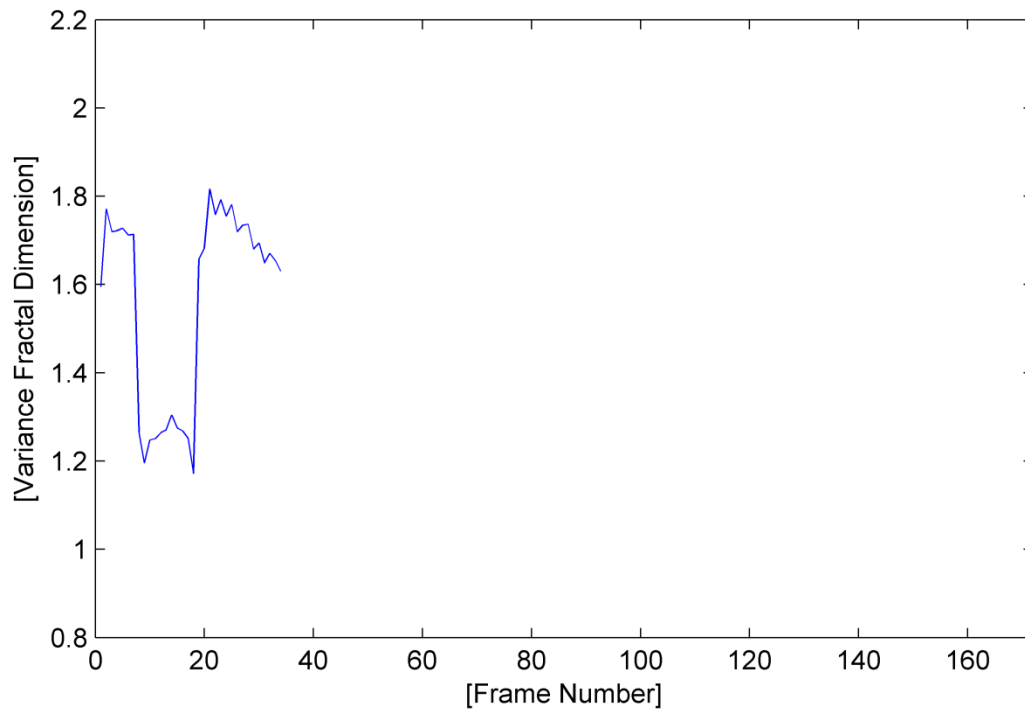


Fig. A.29. The trajectory of the utterance "church" detected by the voice activity detection algorithm.

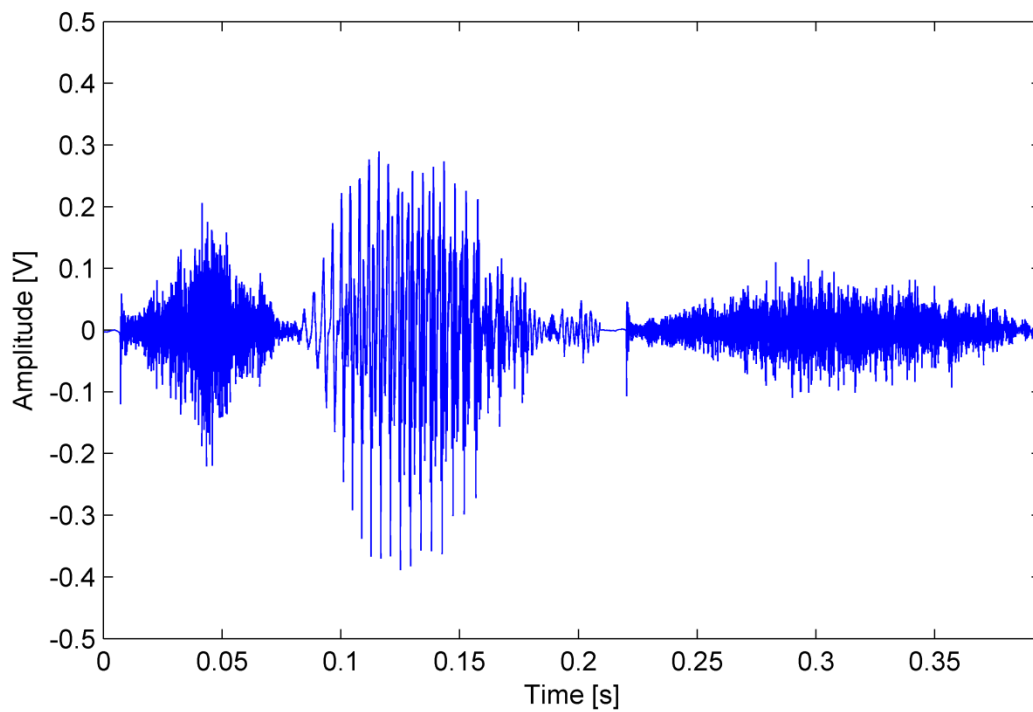


Fig. A.30. The waveform of the utterance "church" detected by the voice activity detection algorithm.

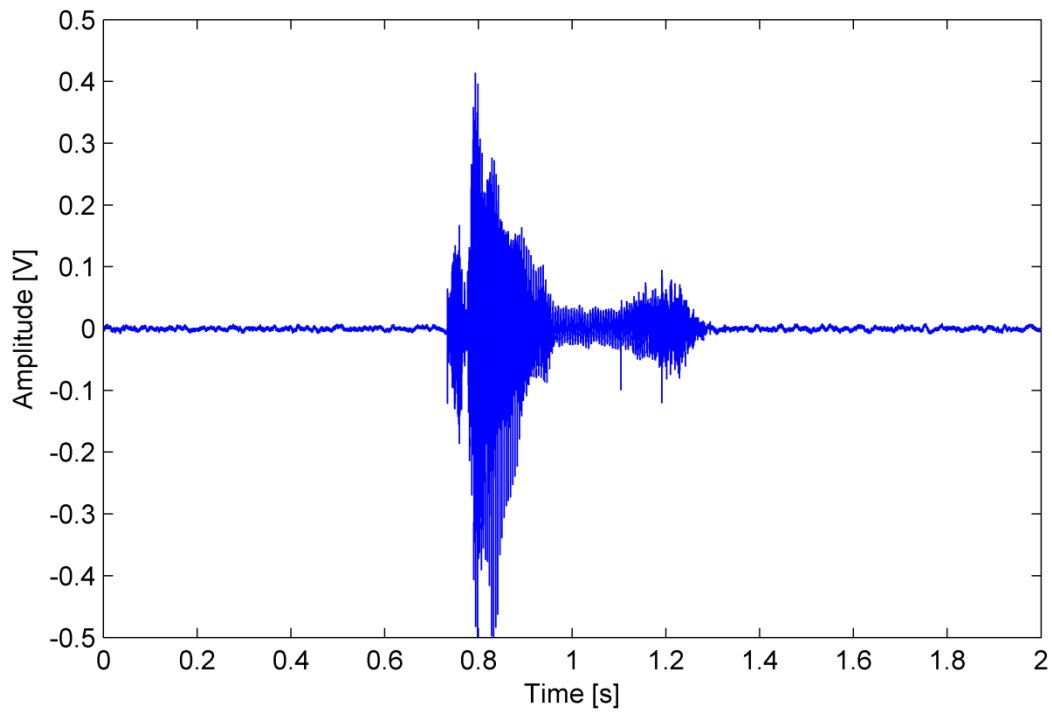


Fig. A.31. The waveform of the utterance "judge".

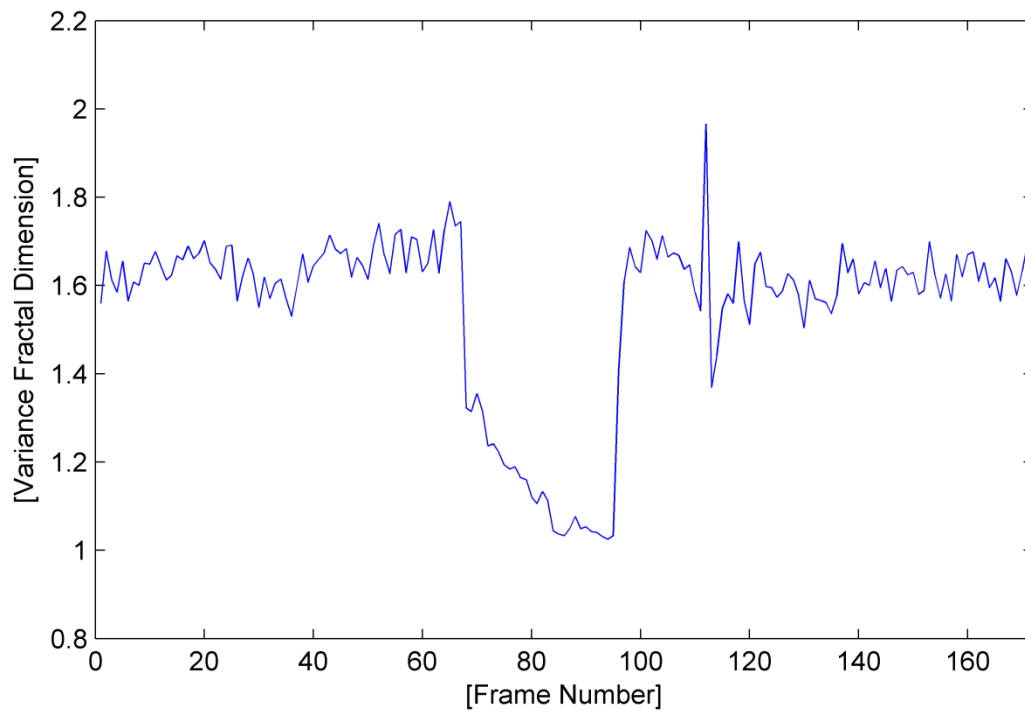


Fig. A.32. The variance fractal dimension trajectory of the utterance "judge".

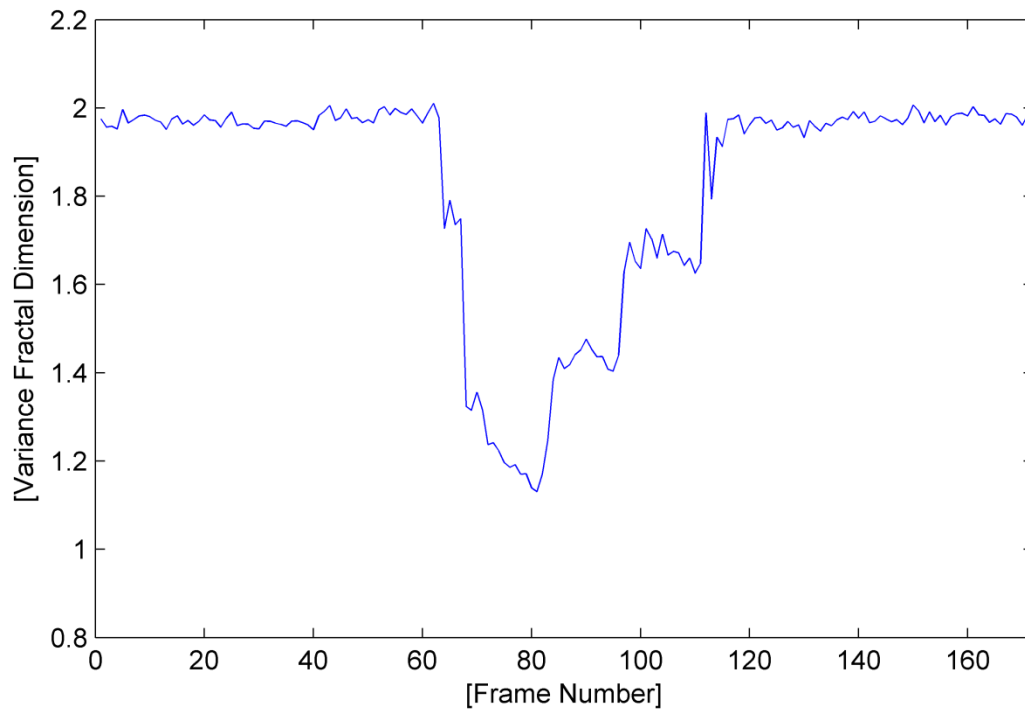


Fig. A.33. The variance fractal dimension trajectory of the utterance “judge” after addition of white noise.

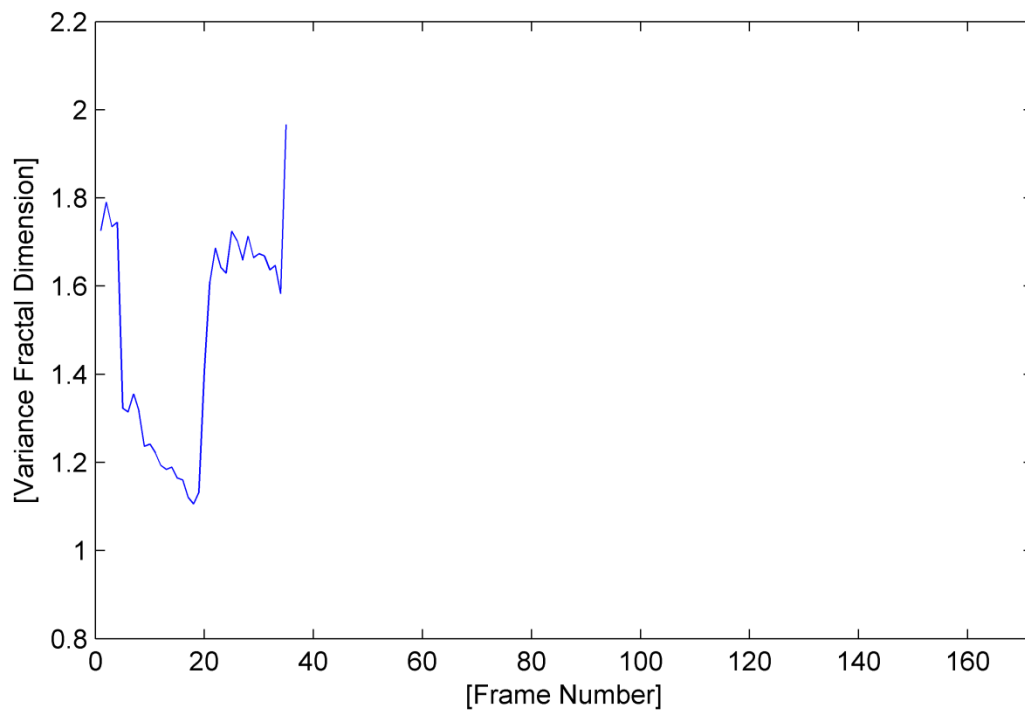


Fig. A.34. The trajectory of the utterance “judge” detected by the voice activity detection algorithm.

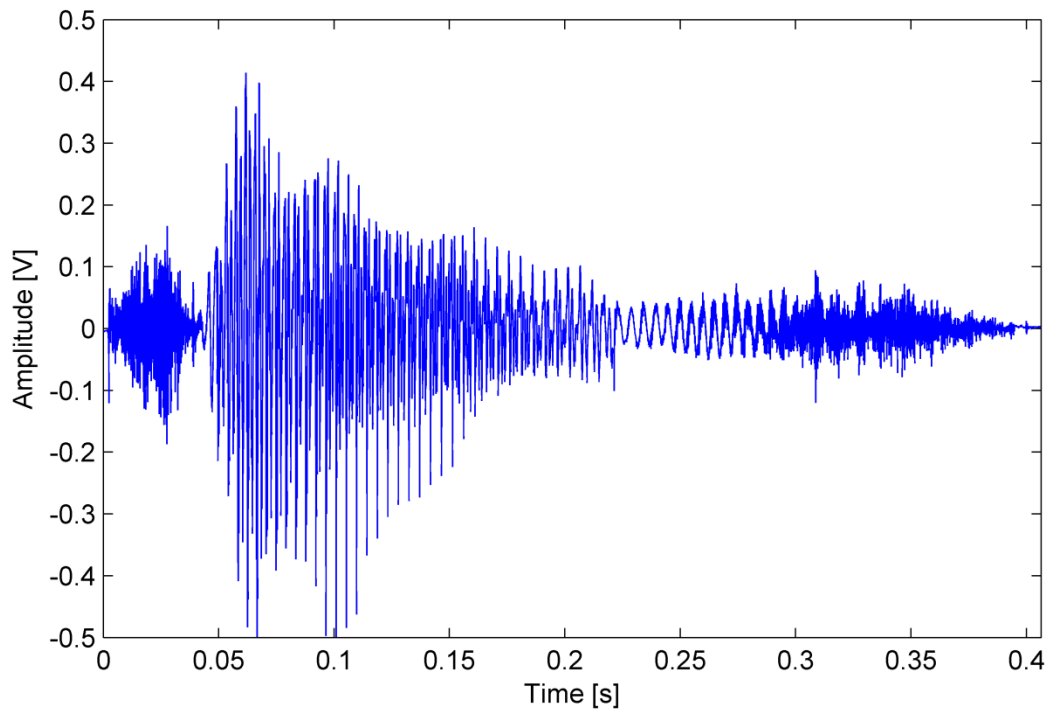


Fig. A.35. The waveform of the utterance “judge” detected by the voice activity detection algorithm.

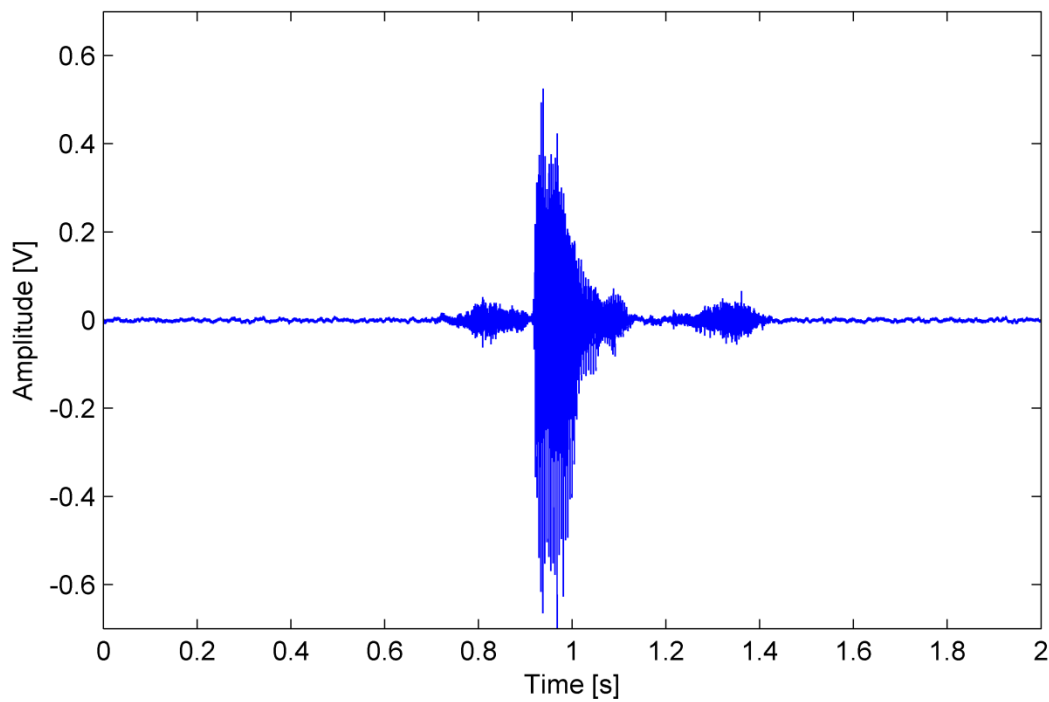


Fig. A.36. The waveform of the utterance “fife”.

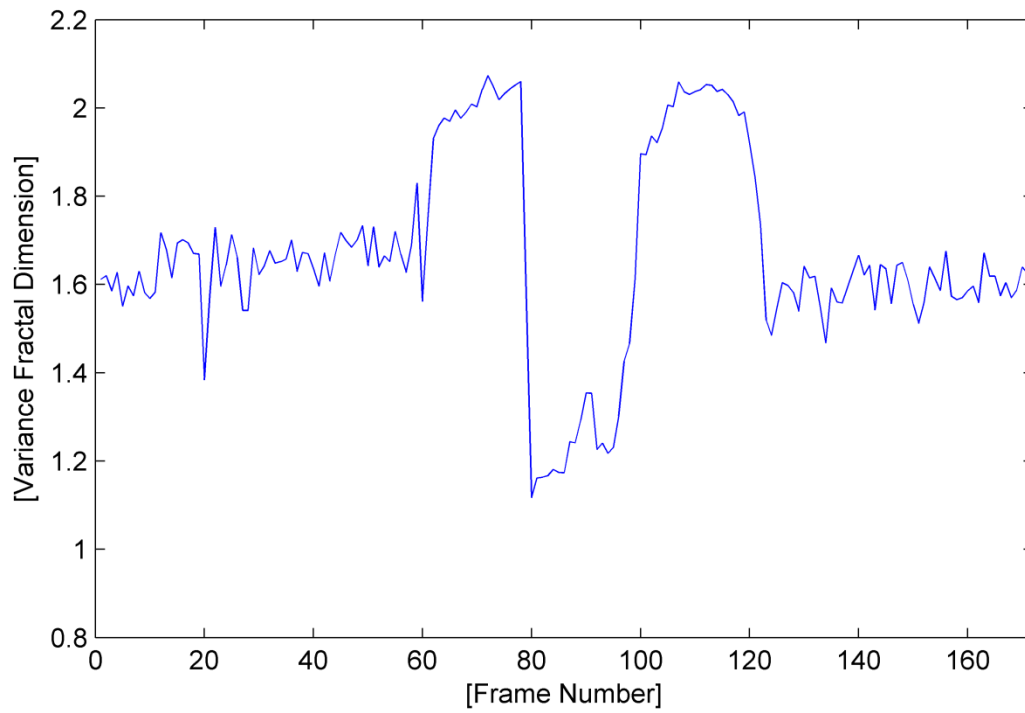


Fig. A.37. The variance fractal dimension trajectory of the utterance “fife”.

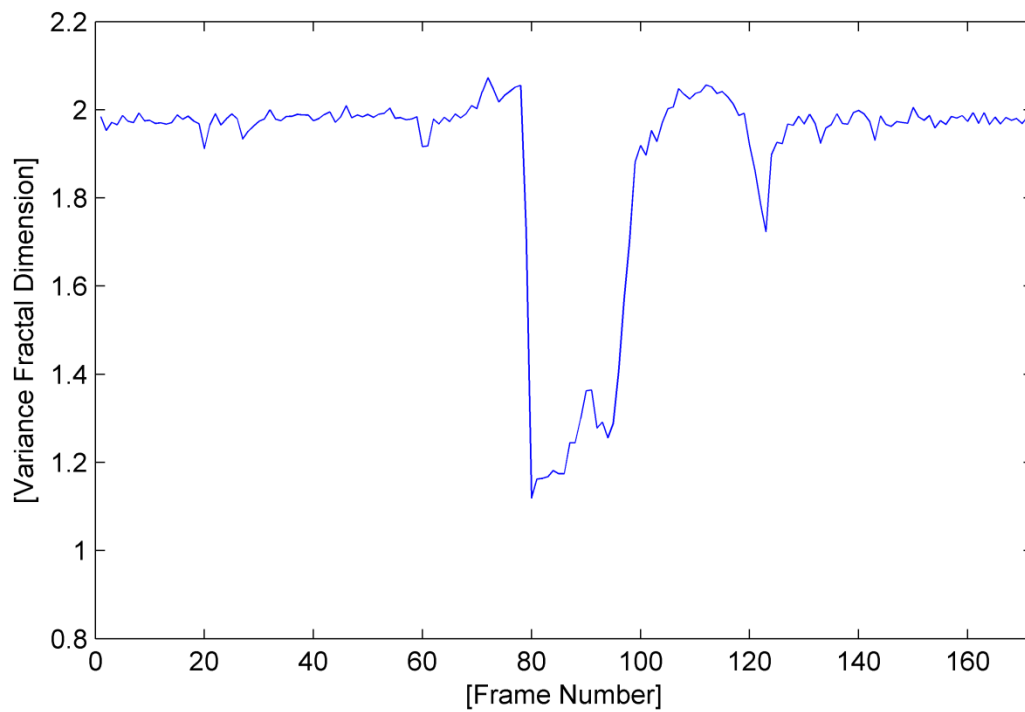


Fig. A.38. The variance fractal dimension trajectory of the utterance “fife” after addition of white noise.

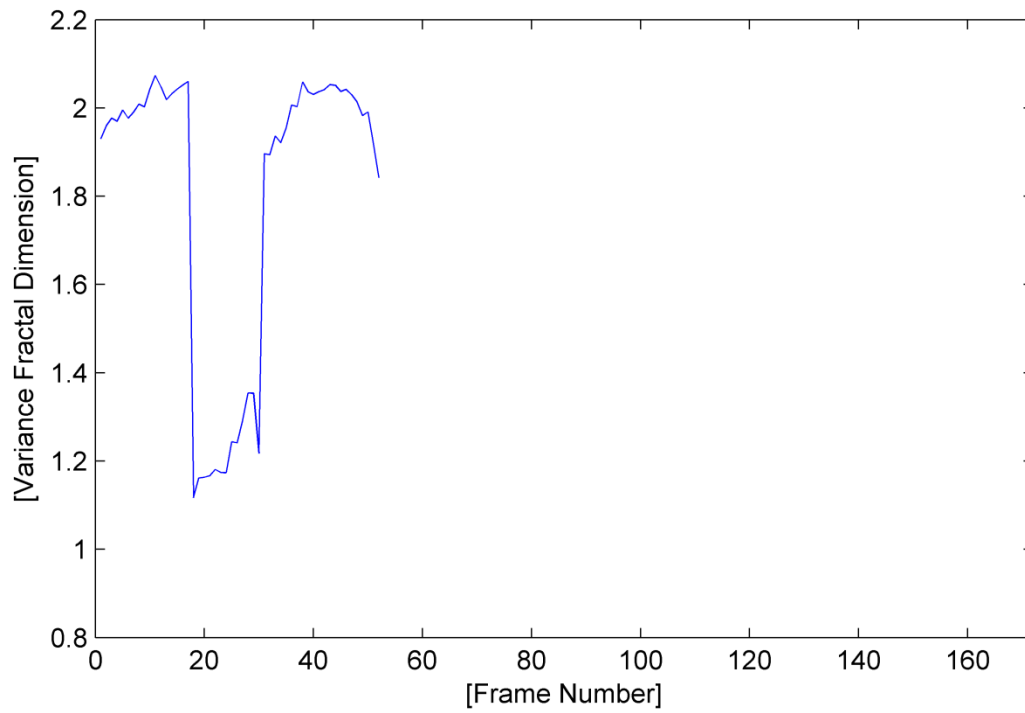


Fig. A.39. The trajectory of the utterance “fife” detected by the voice activity detection algorithm.

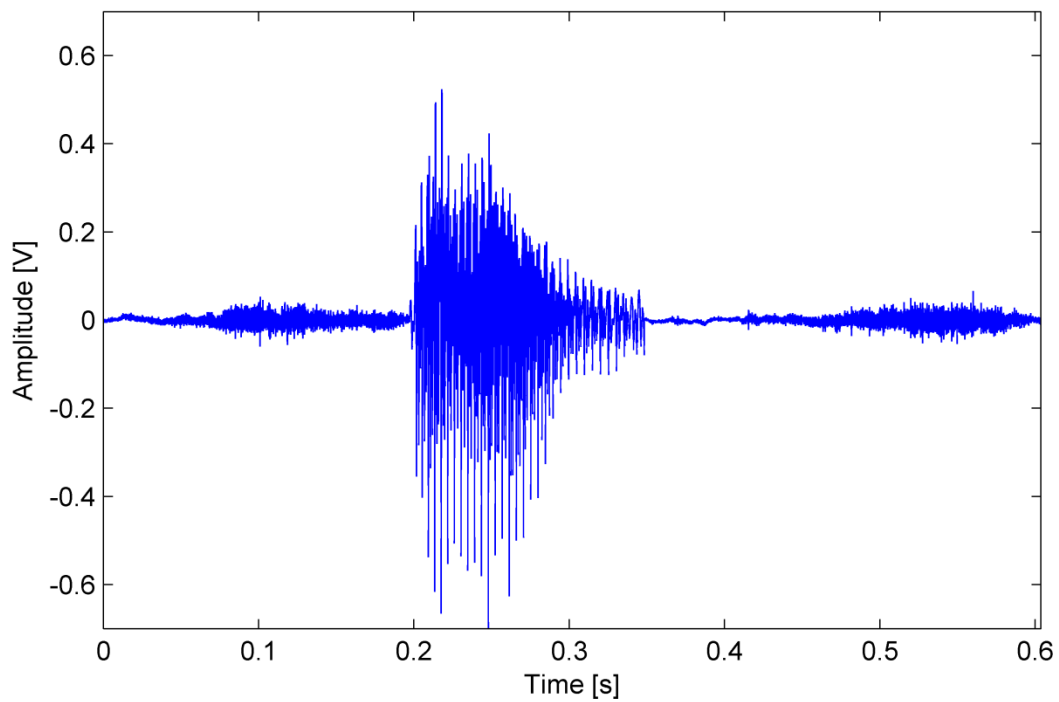


Fig. A.40. The waveform of the utterance “fife” detected by the voice activity detection algorithm.

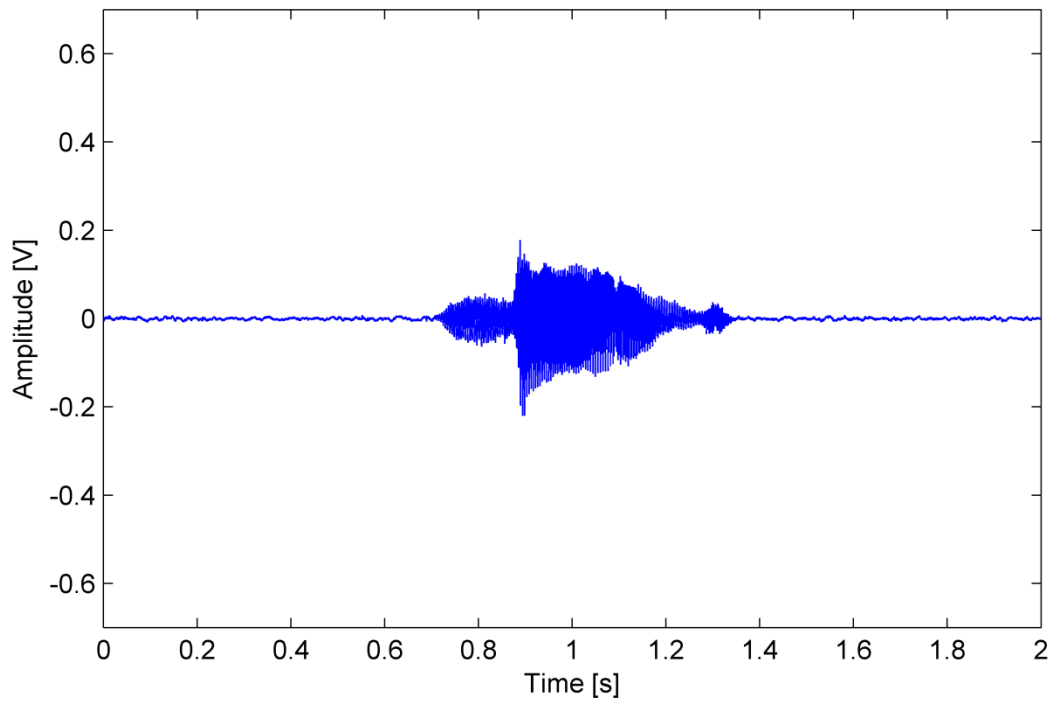


Fig. A.41. The waveform of the utterance "verve".

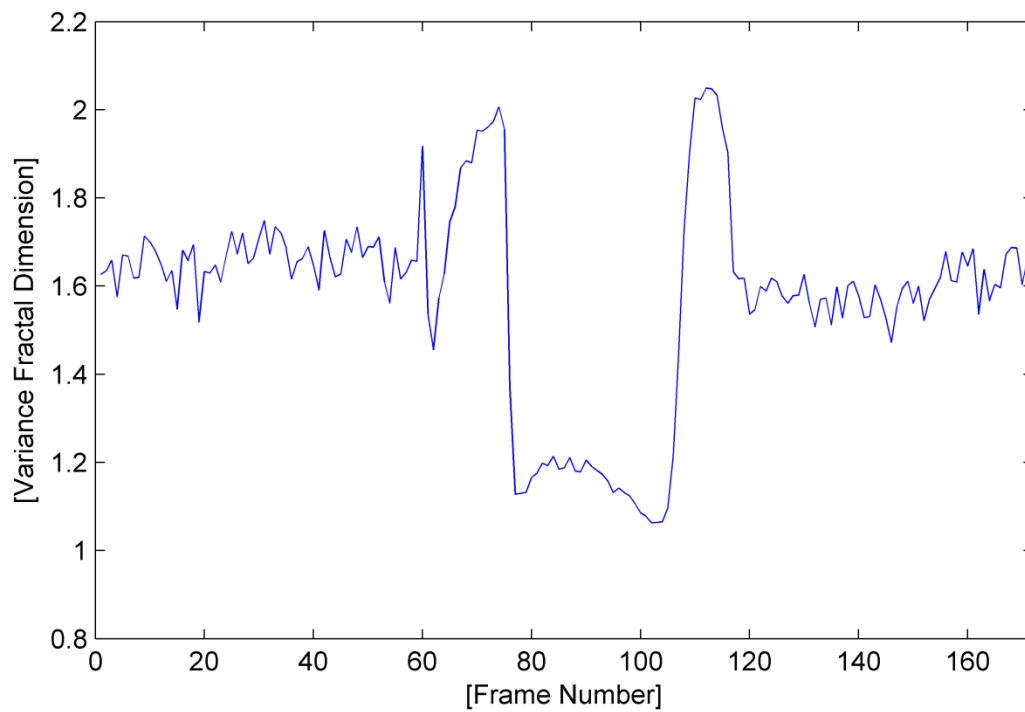


Fig. A.42. The variance fractal dimension trajectory of the utterance "verve".

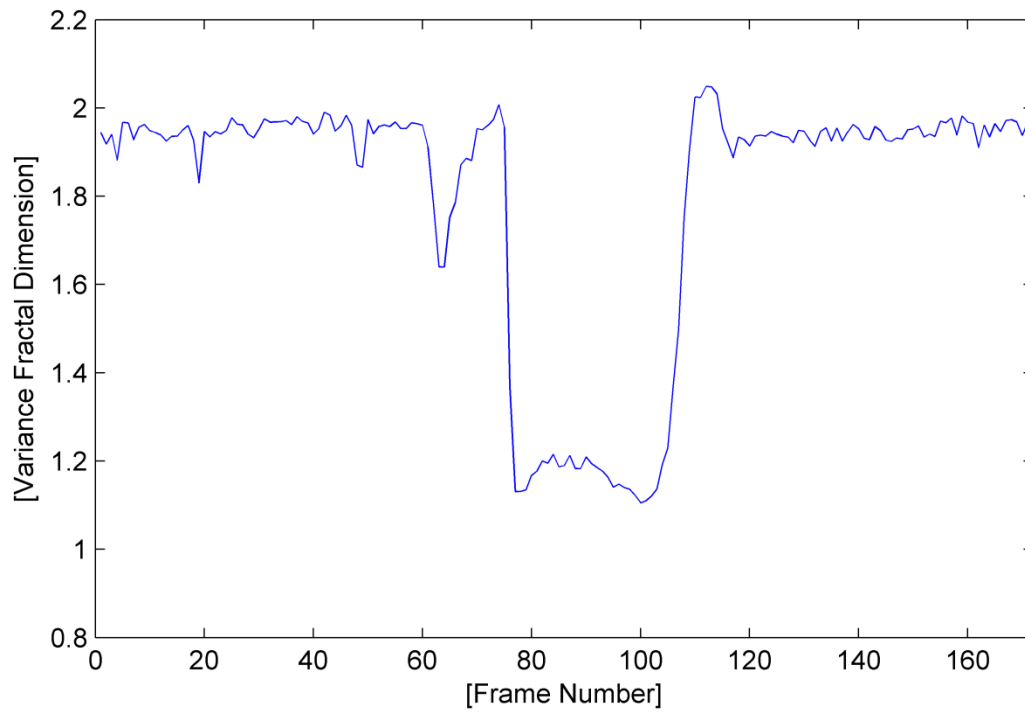


Fig. A.43. The variance fractal dimension trajectory of the utterance “verve” after addition of white noise.

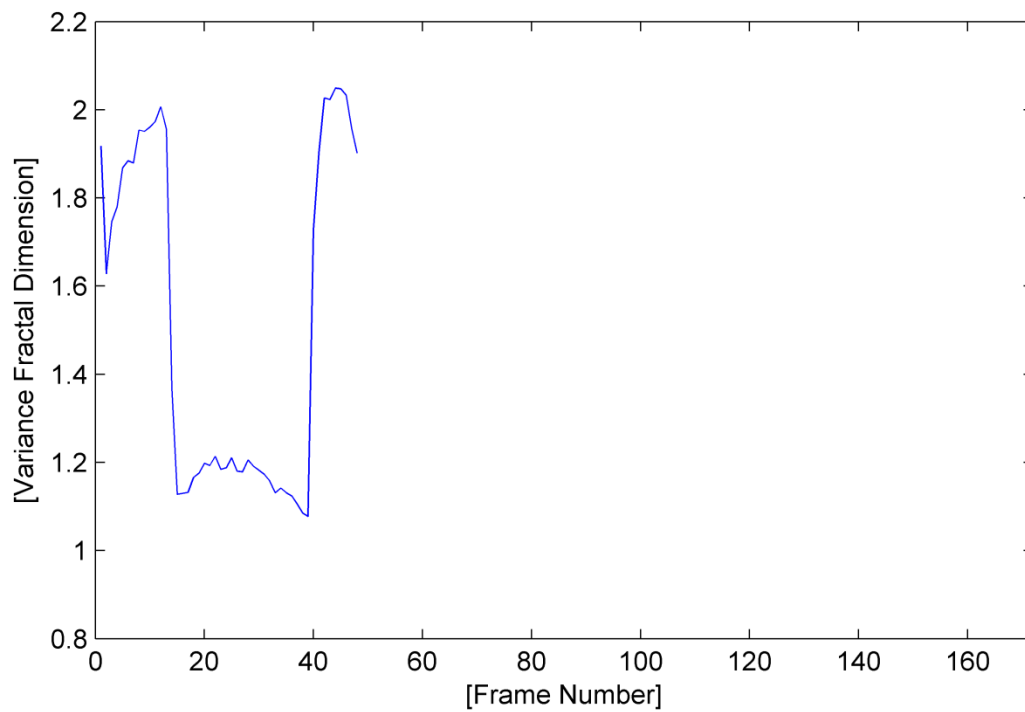


Fig. A.44. The trajectory of the utterance “verve” detected by the voice activity detection algorithm.

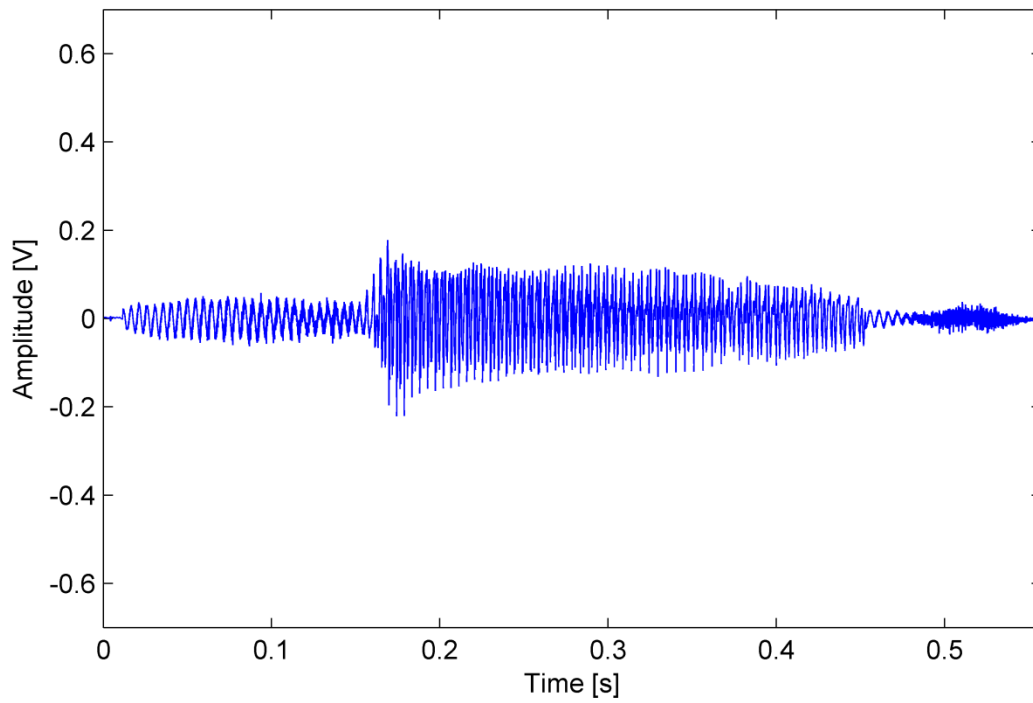


Fig. A.45. The waveform of the utterance “verve” detected by the voice activity detection algorithm.

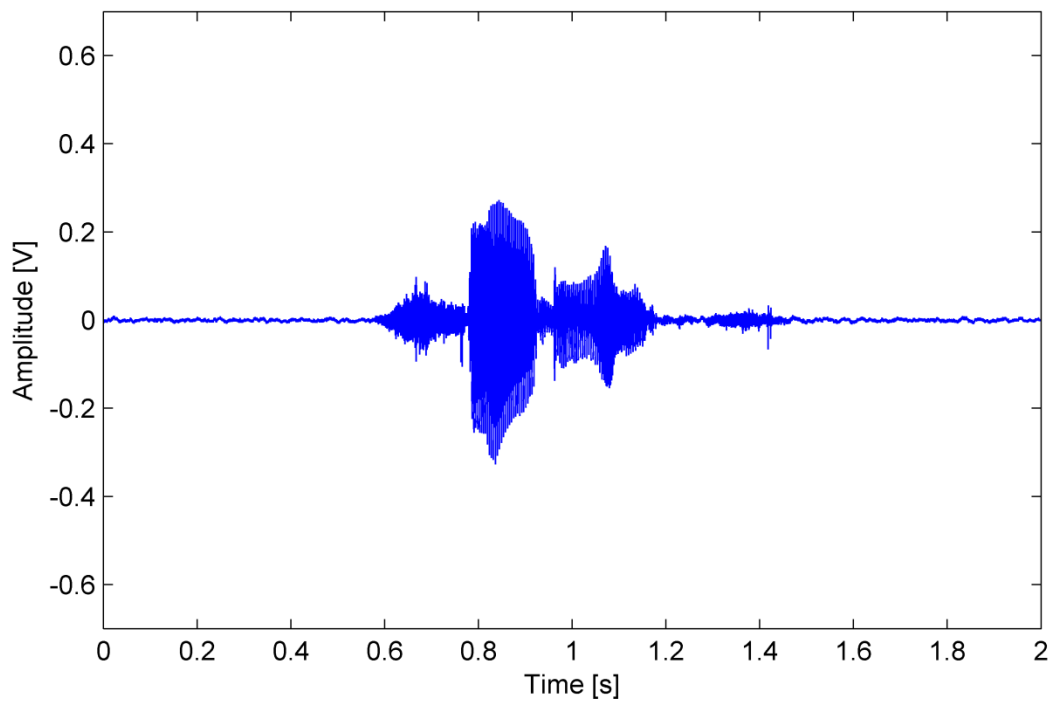


Fig. A.46. The waveform of the utterance “thirtieth”.

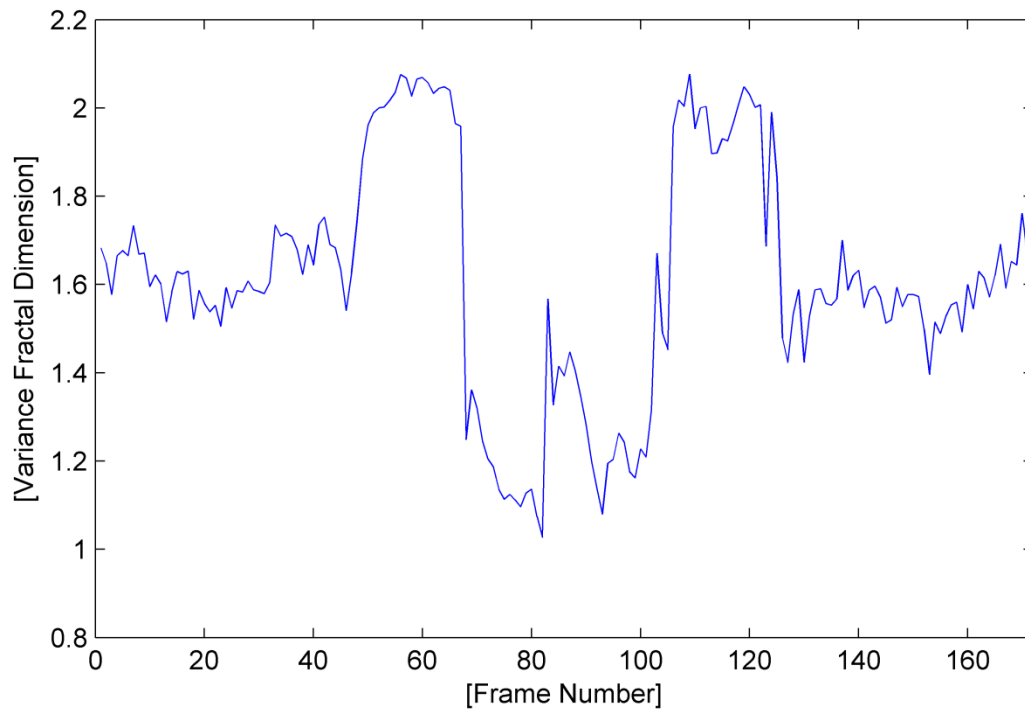


Fig. A.47. The variance fractal dimension trajectory of the utterance “thirtieth”.

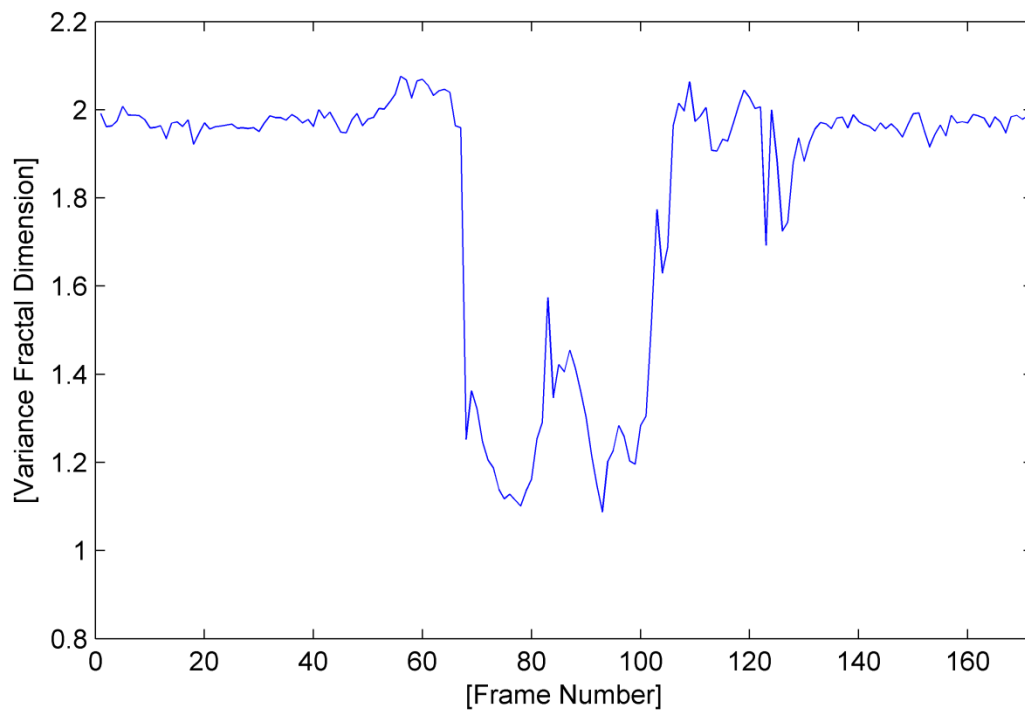


Fig. A.48. The variance fractal dimension trajectory of the utterance “thirtieth” after addition of white noise.

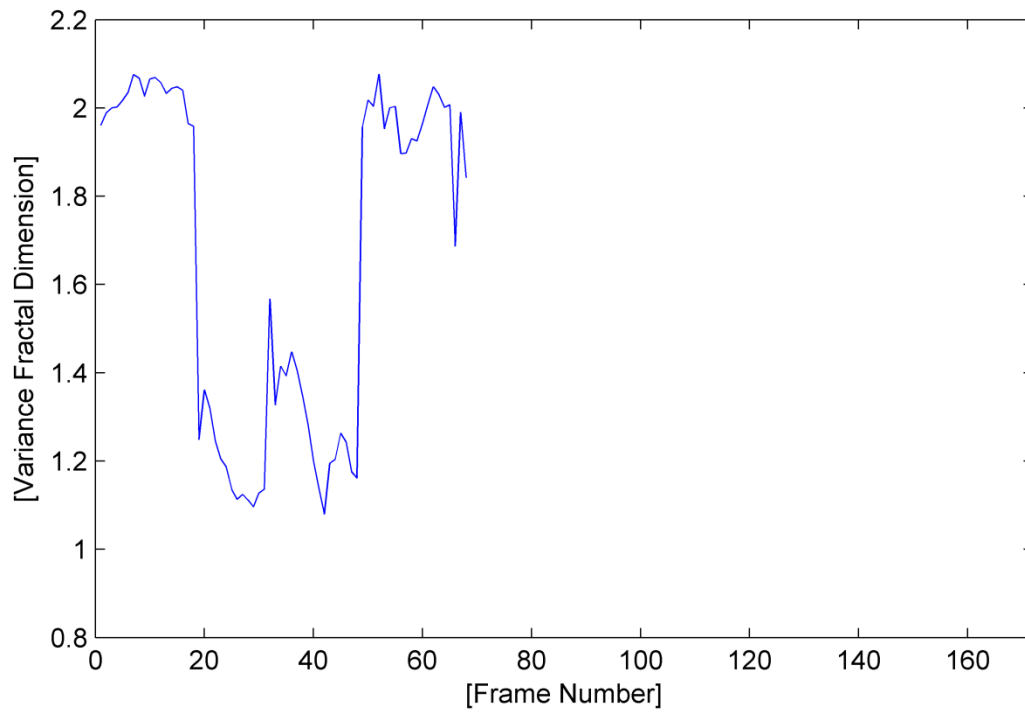


Fig. A.49. The trajectory of the utterance "thirtieth" detected by the voice activity detection algorithm.

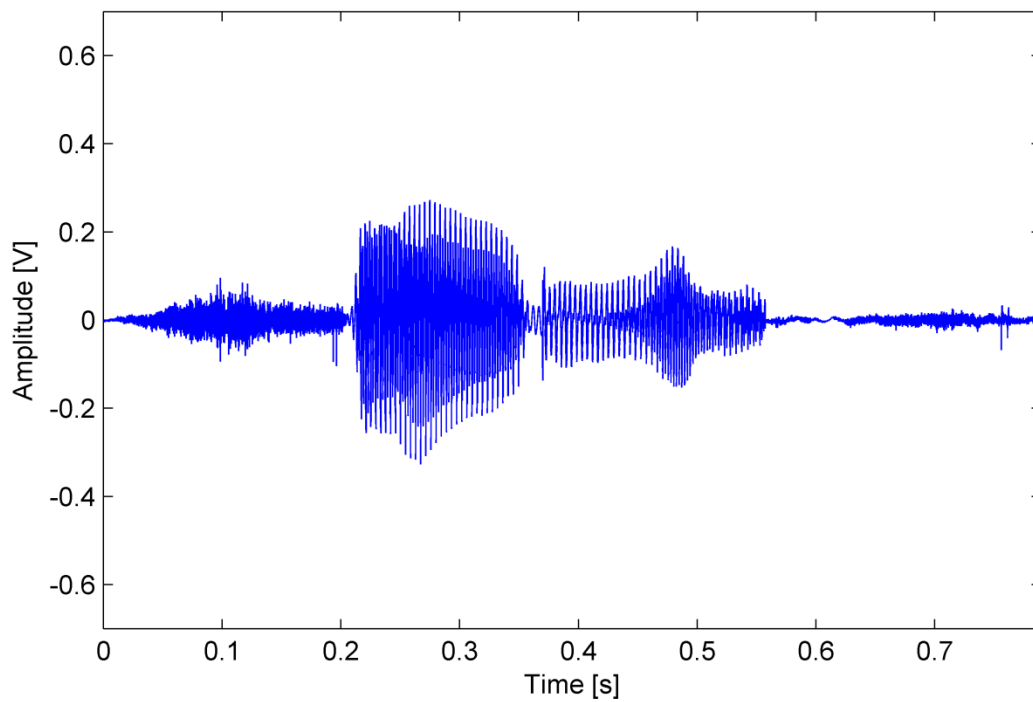


Fig. A.50. The waveform of the utterance "thirtieth" detected by the voice activity detection algorithm.

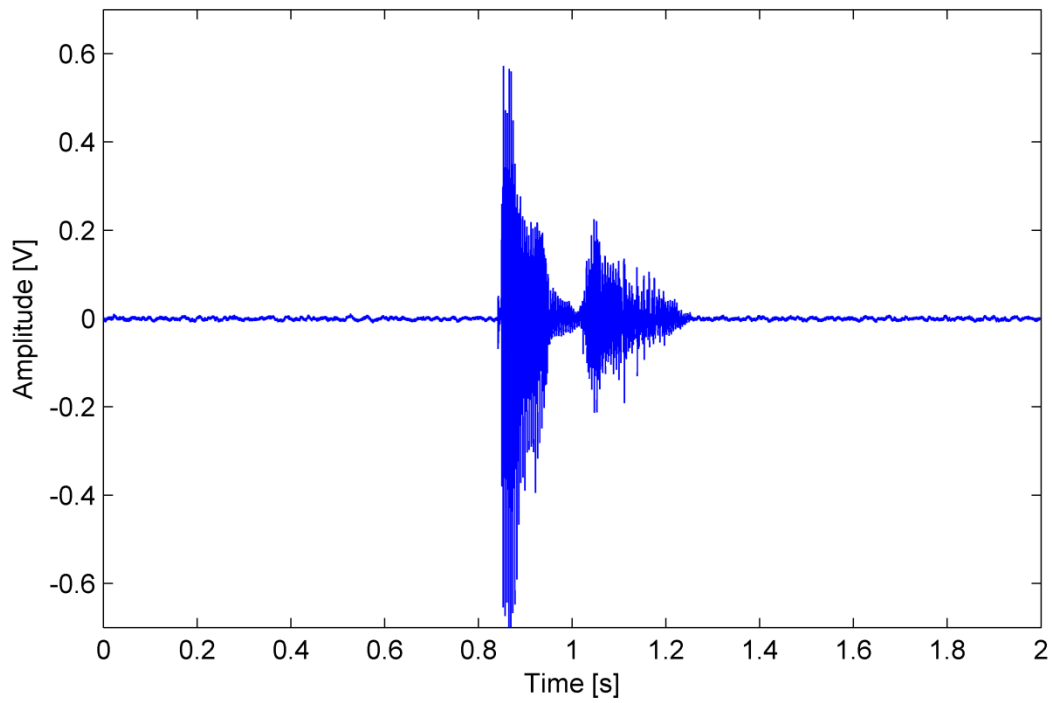


Fig. A.51. The waveform of the utterance “other”.

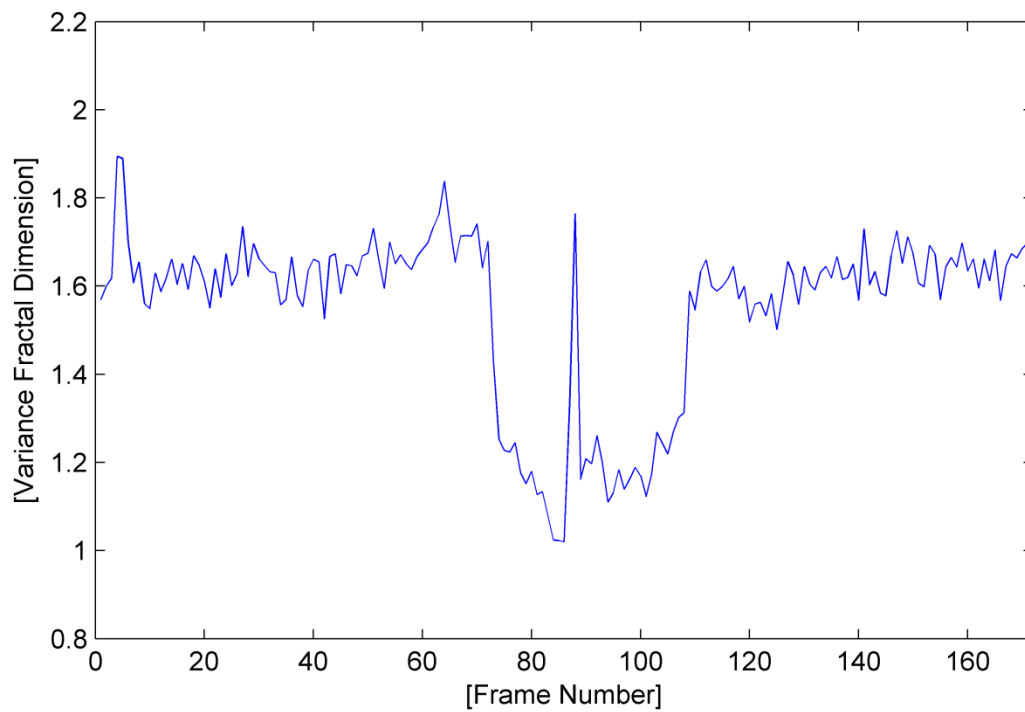


Fig. A.52. The variance fractal dimension trajectory of the utterance “other”.

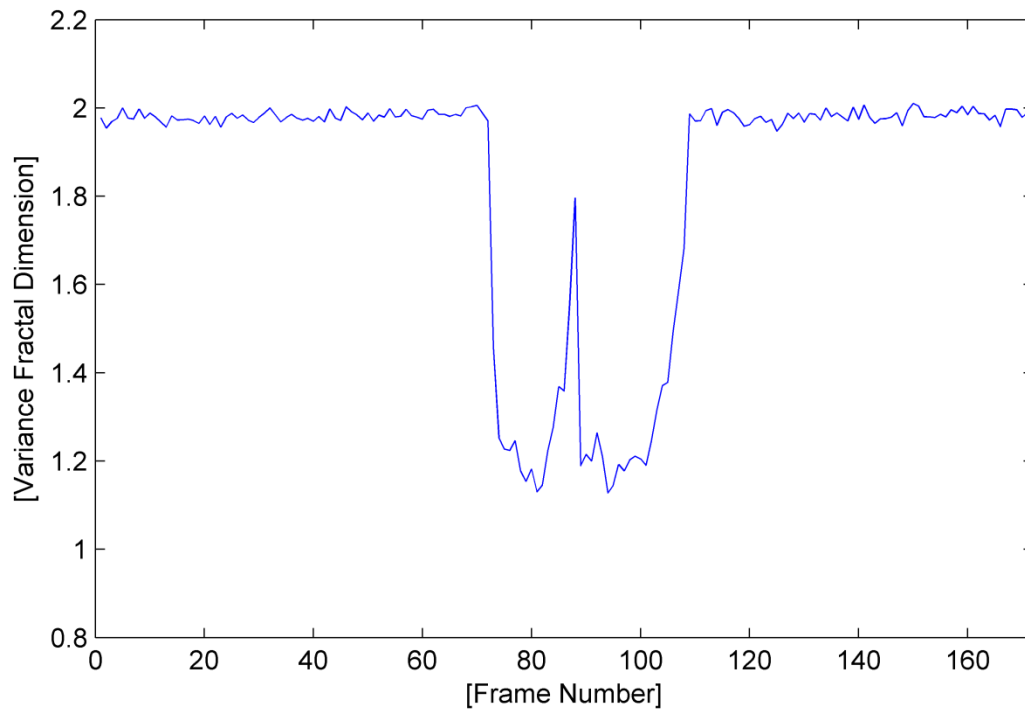


Fig. A.53. The variance fractal dimension trajectory of the utterance “other” after addition of white noise.

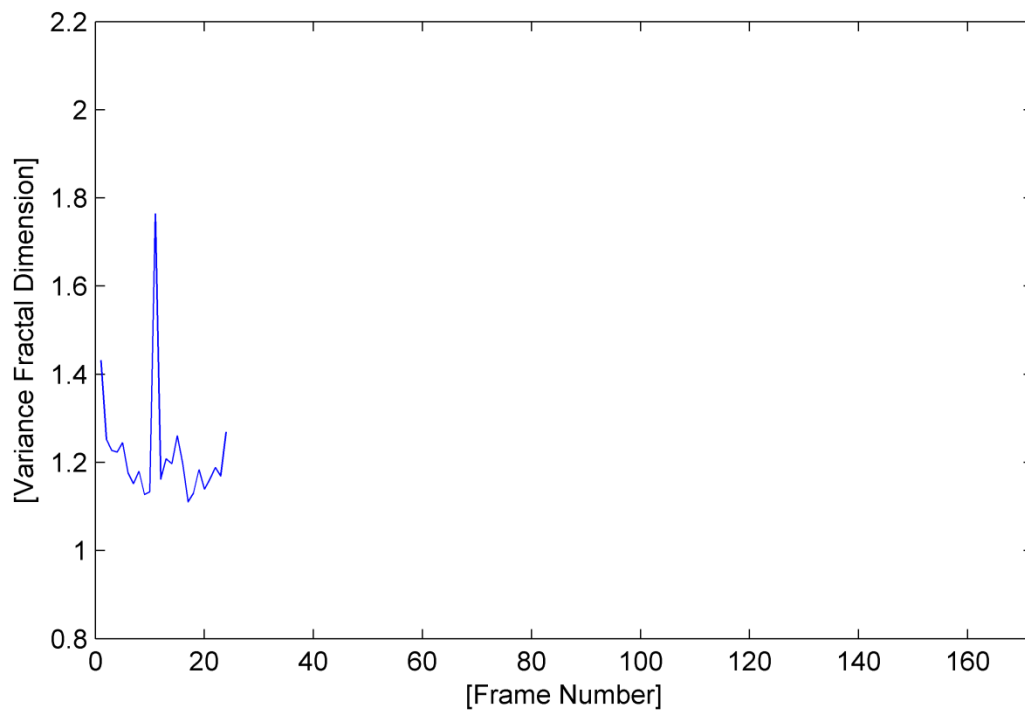


Fig. A.54. The trajectory of the utterance “other” detected by the voice activity detection algorithm.

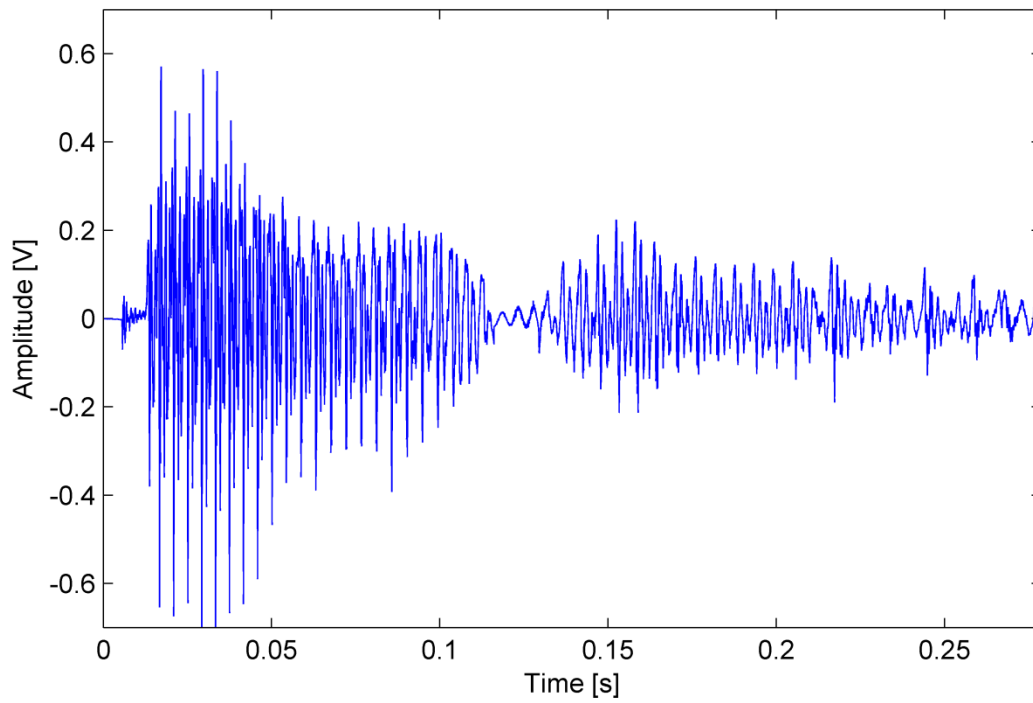


Fig. A.55. The waveform of the utterance "other" detected by the voice activity detection algorithm.

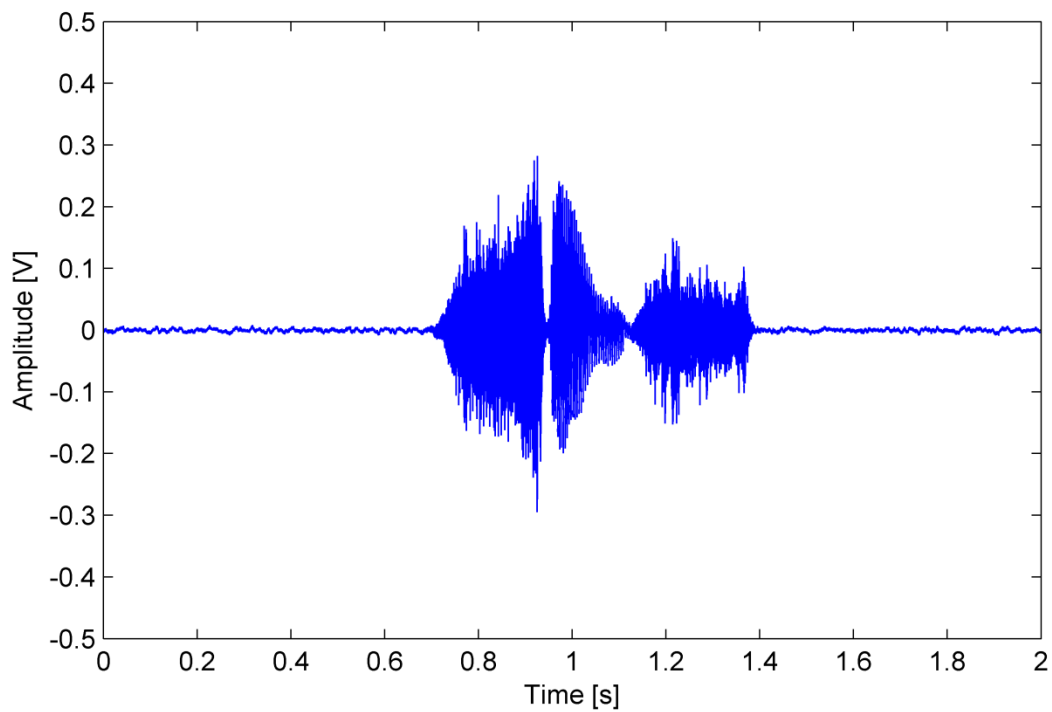


Fig. A.56. The waveform of the utterance "cease".

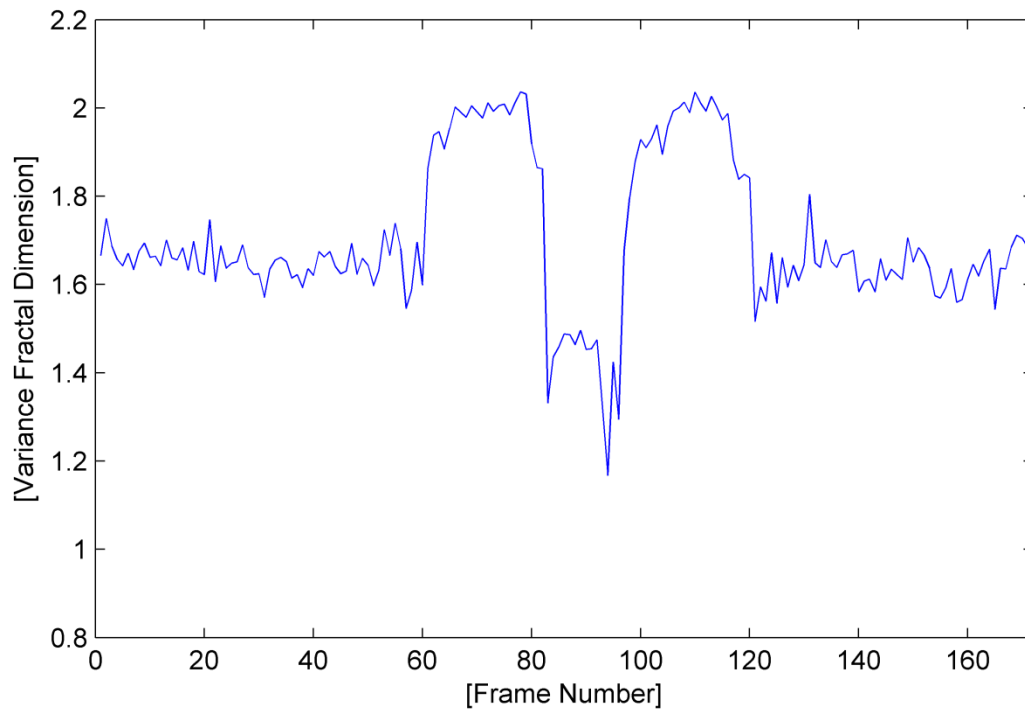


Fig. A.57. The variance fractal dimension trajectory of the utterance “cease”.

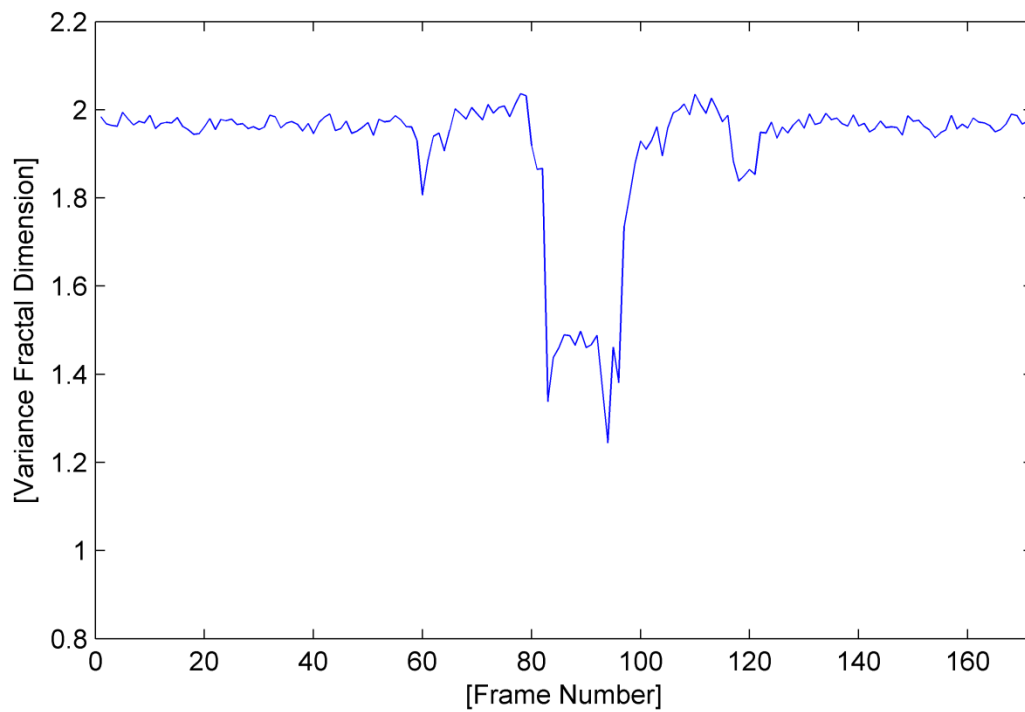


Fig. A.58. The variance fractal dimension trajectory of the utterance “cease” after addition of white noise.

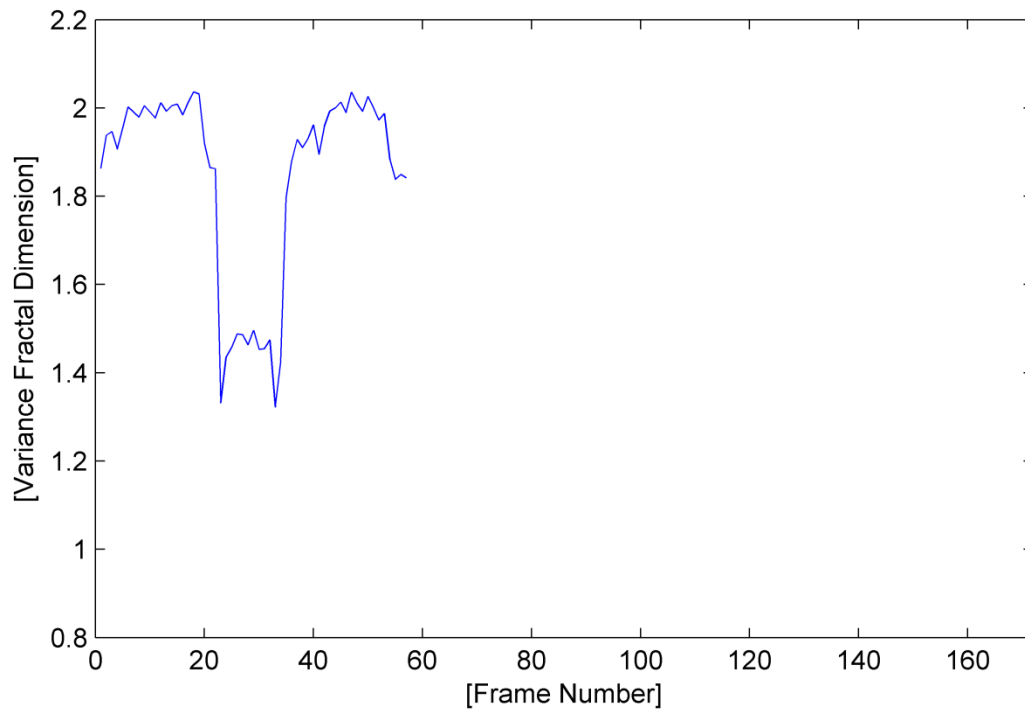


Fig. A.59. The trajectory of the utterance “cease” detected by the voice activity detection algorithm.

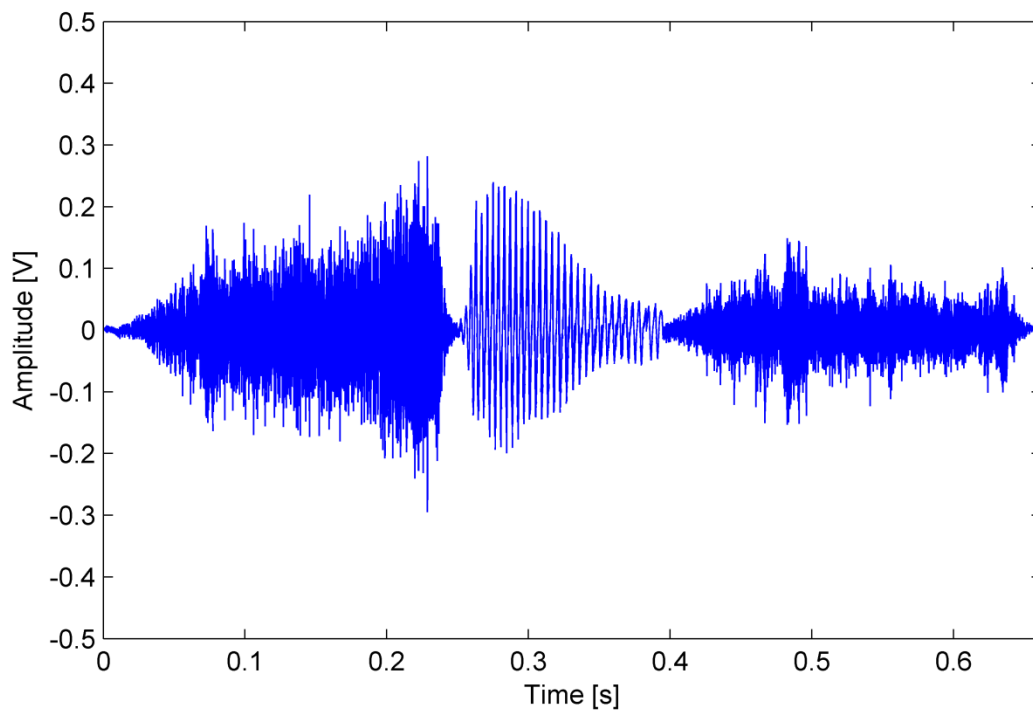


Fig. A.60. The waveform of the utterance “cease” detected by the voice activity detection algorithm.

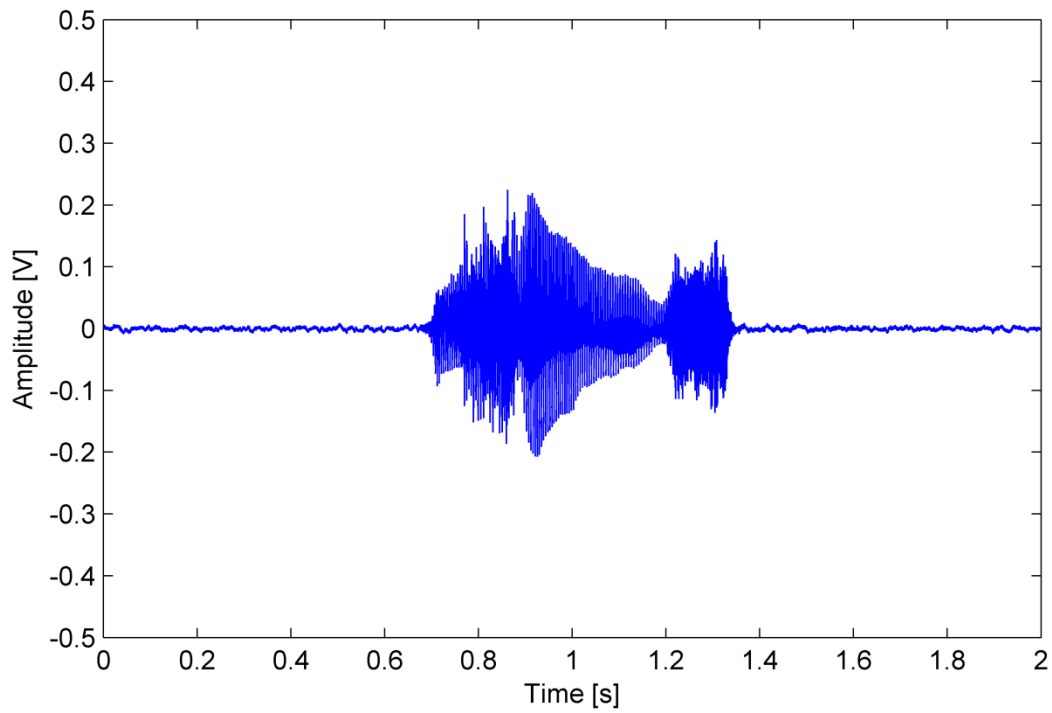


Fig. A.61. The waveform of the utterance "zoos".

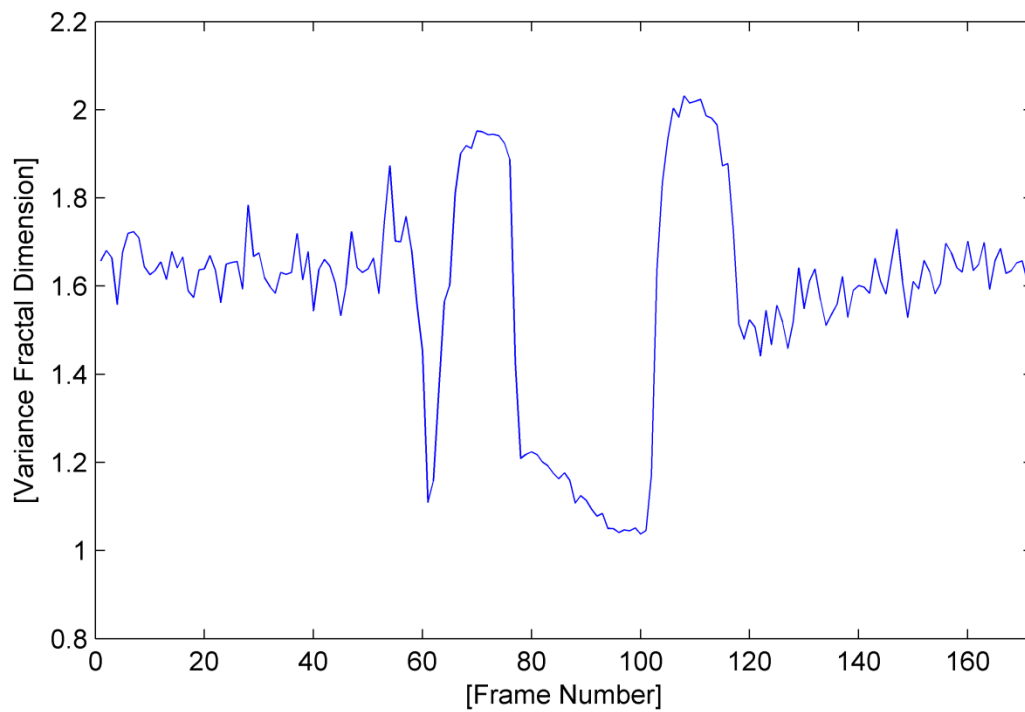


Fig. A.62. The variance fractal dimension trajectory of the utterance "zoos".

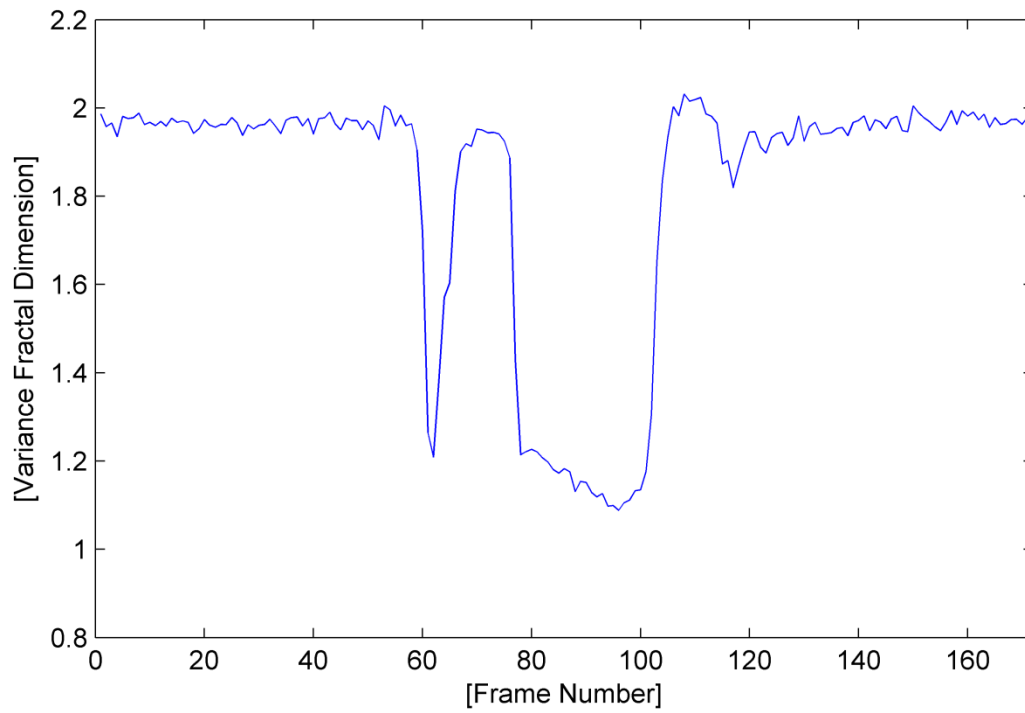


Fig. A.63. The variance fractal dimension trajectory of the utterance “zoos” after addition of white noise.

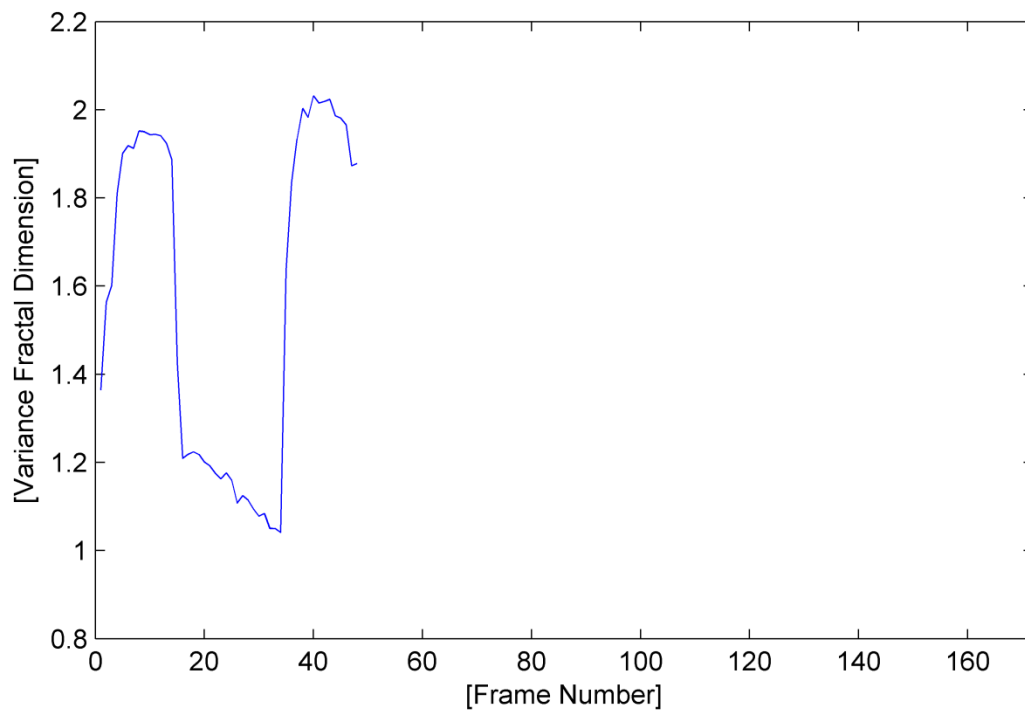


Fig. A.64. The trajectory of the utterance “zoos” detected by the voice activity detection algorithm.

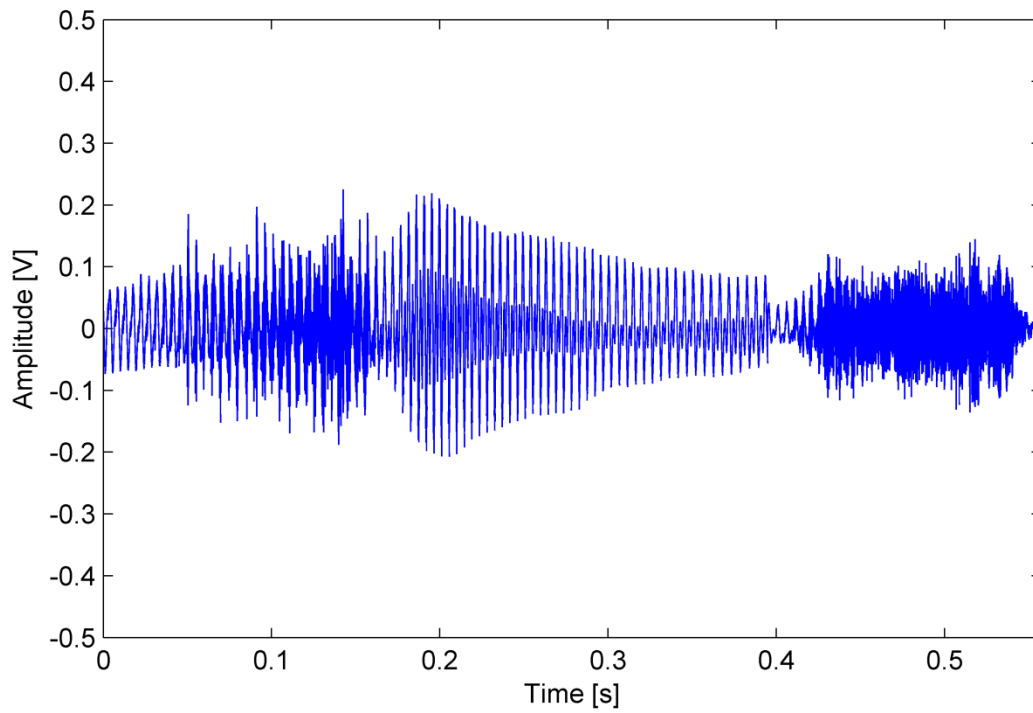


Fig. A.65. The waveform of the utterance “zoos” detected by the voice activity detection algorithm.

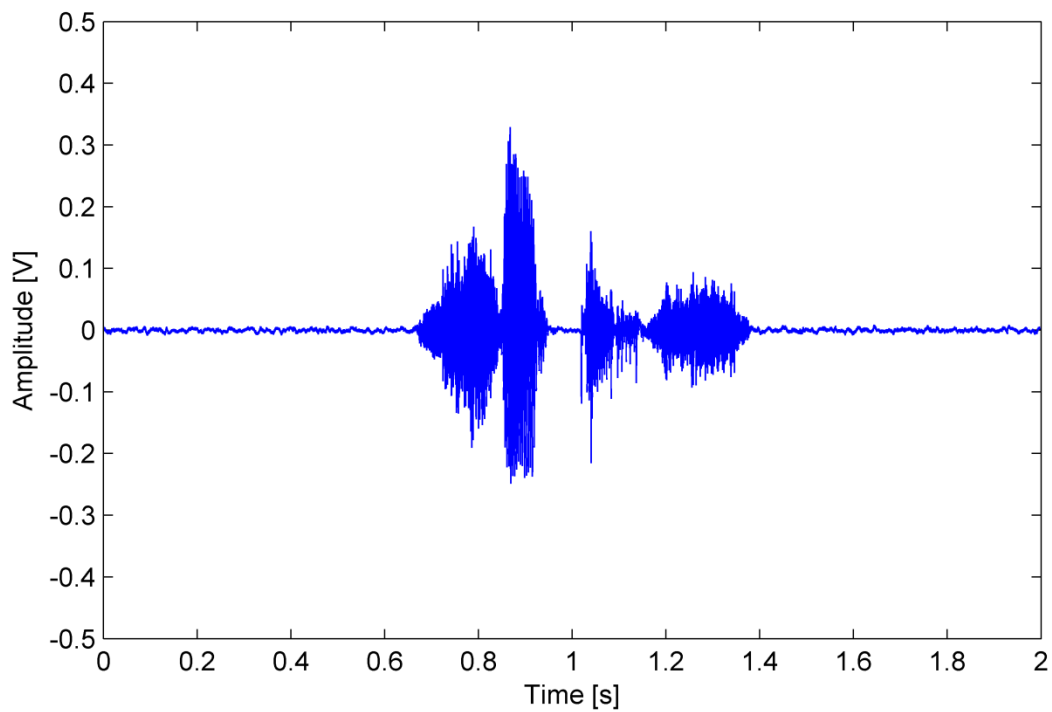


Fig. A.66. The waveform of the utterance “sheepish”.

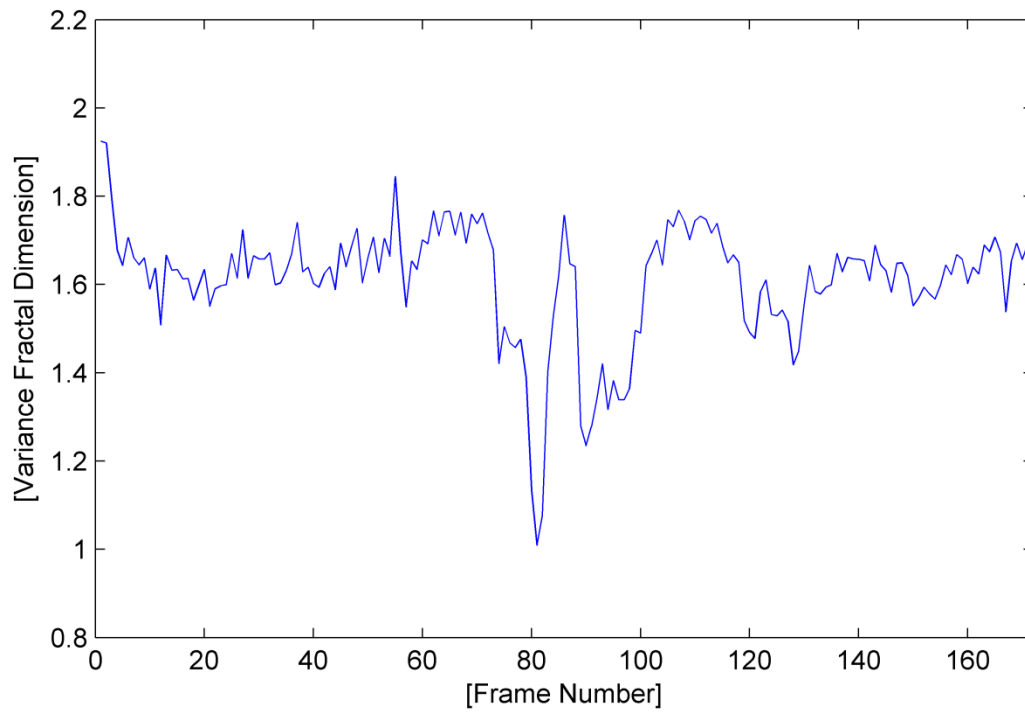


Fig. A.67. The variance fractal dimension trajectory of the utterance "sheepish".

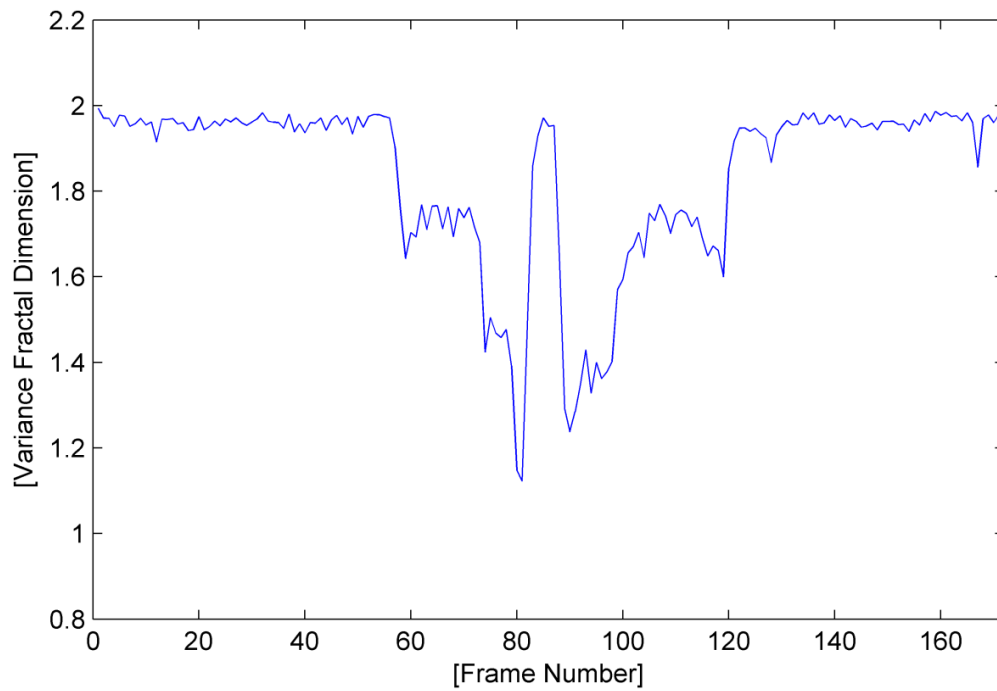


Fig. A.68. The variance fractal dimension trajectory of the utterance "sheepish" after addition of white noise.

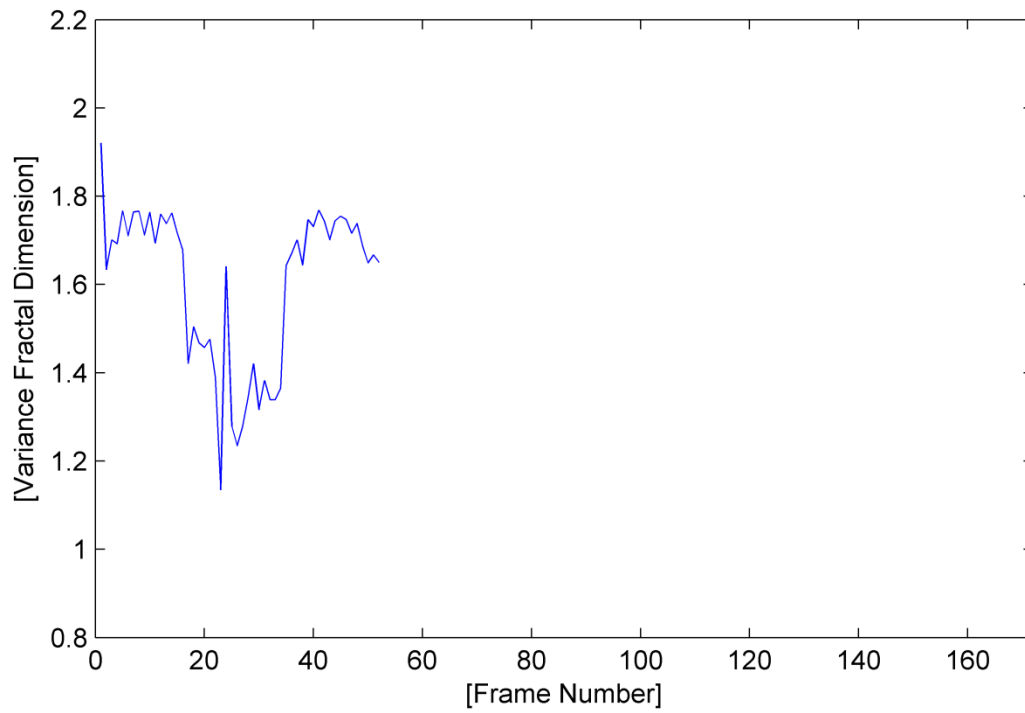


Fig. A.69. The trajectory of the utterance “sheepish” detected by the voice activity detection algorithm.

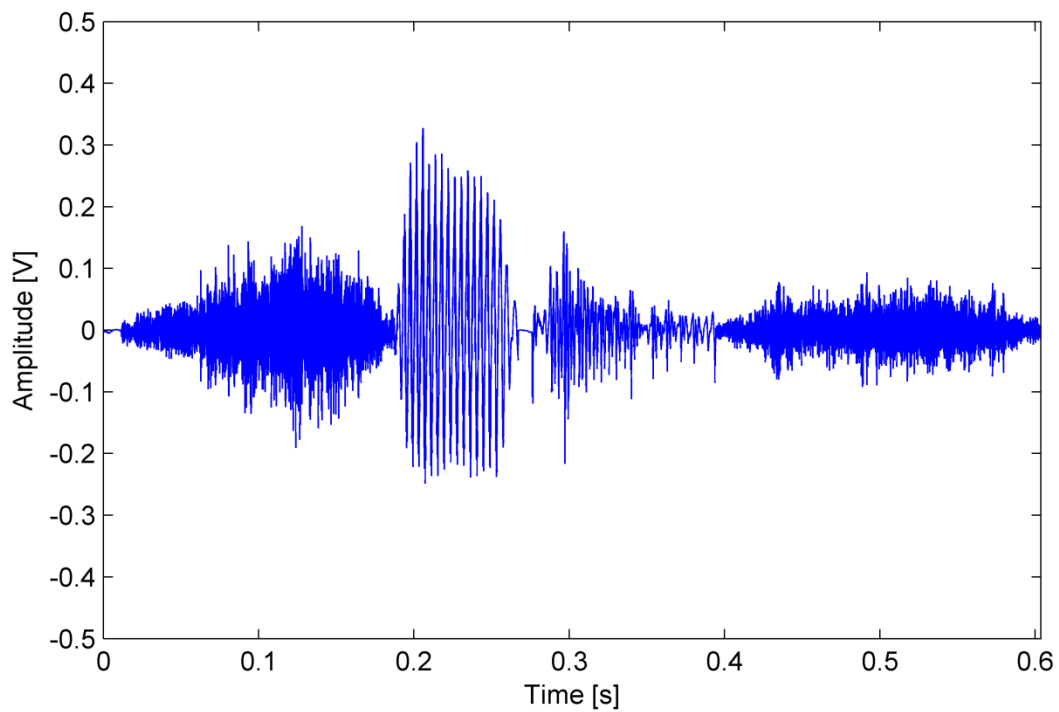


Fig. A.70. The waveform of the utterance “sheepish” detected by the voice activity detection algorithm.

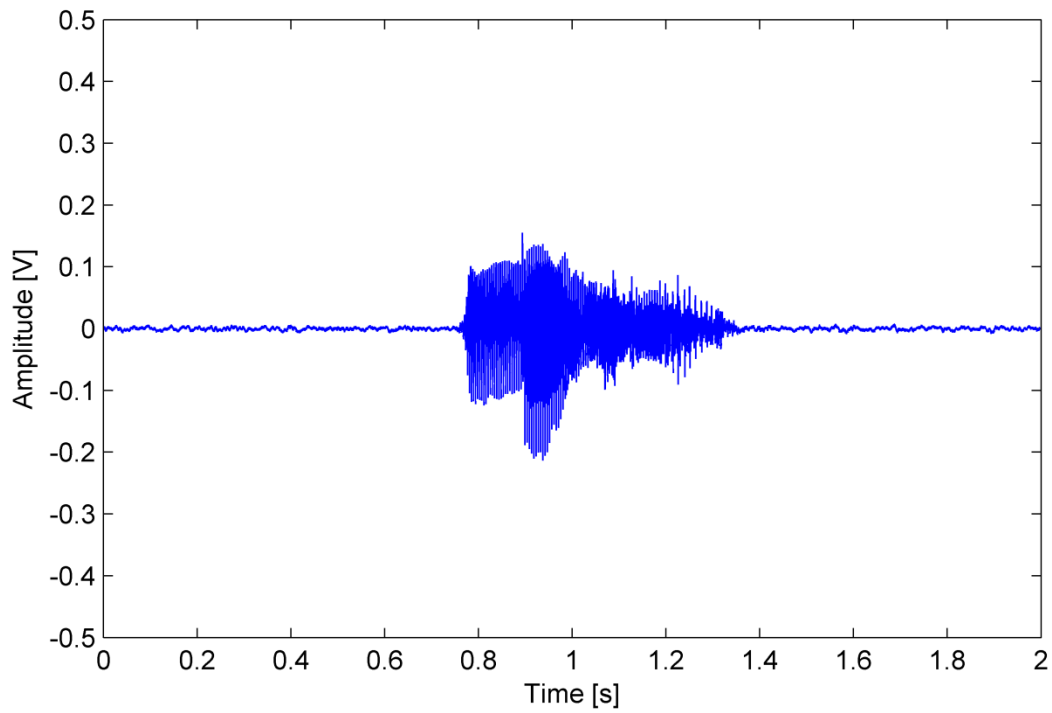


Fig. A.71. The waveform of the utterance "measure".

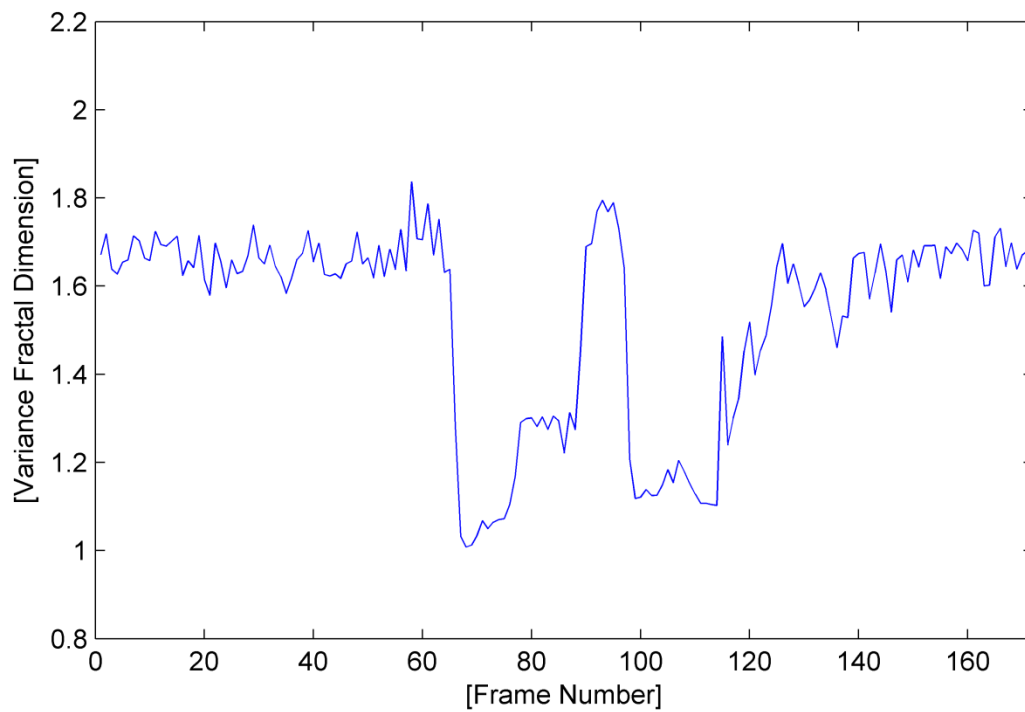


Fig. A.72. The variance fractal dimension trajectory of the utterance "measure".

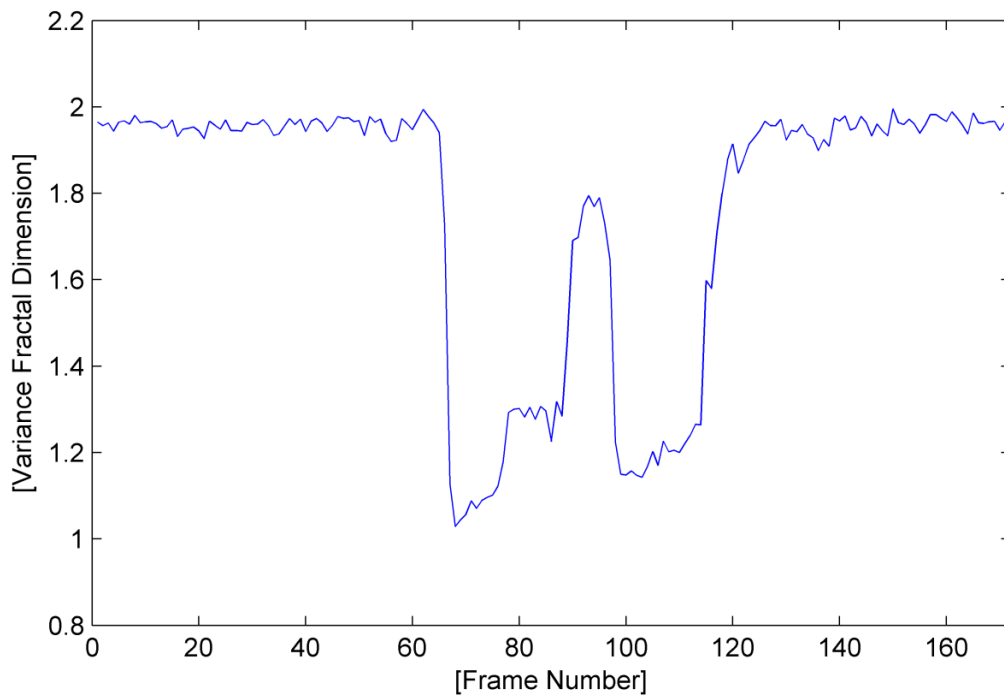


Fig. A.73. The variance fractal dimension trajectory of the utterance “measure” after addition of white noise.

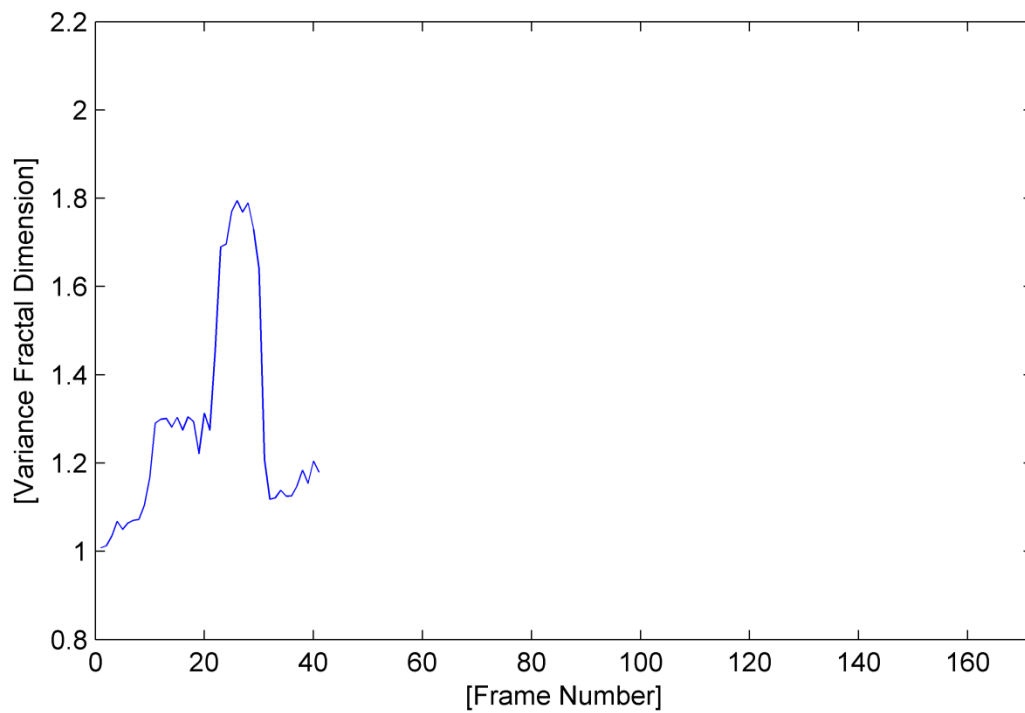


Fig. A.74. The trajectory of the utterance “measure” detected by the voice activity detection algorithm.

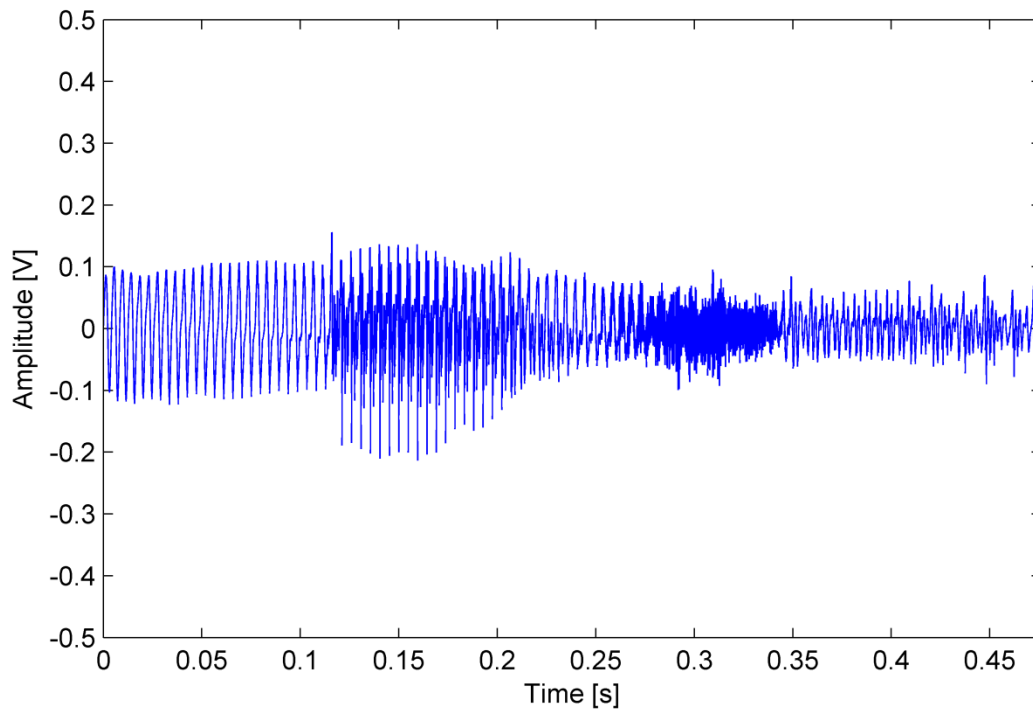


Fig. A.75. The waveform of the utterance "measure" detected by the voice activity detection algorithm.

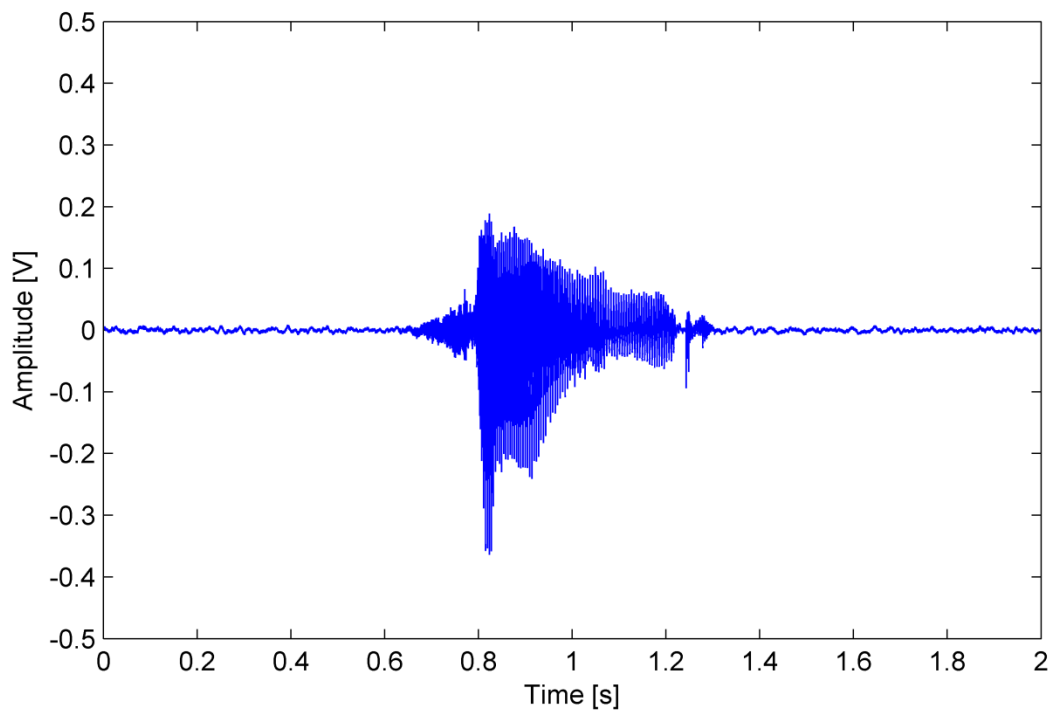


Fig. A.76. The waveform of the utterance "hand".

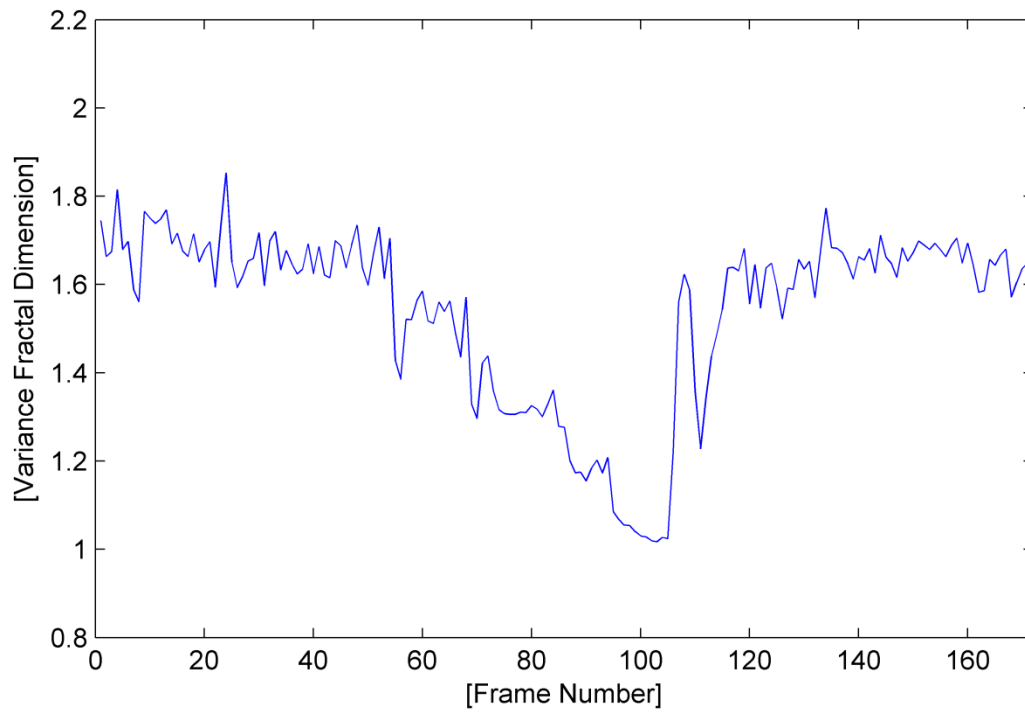


Fig. A.77. The variance fractal dimension trajectory of the utterance "hand".

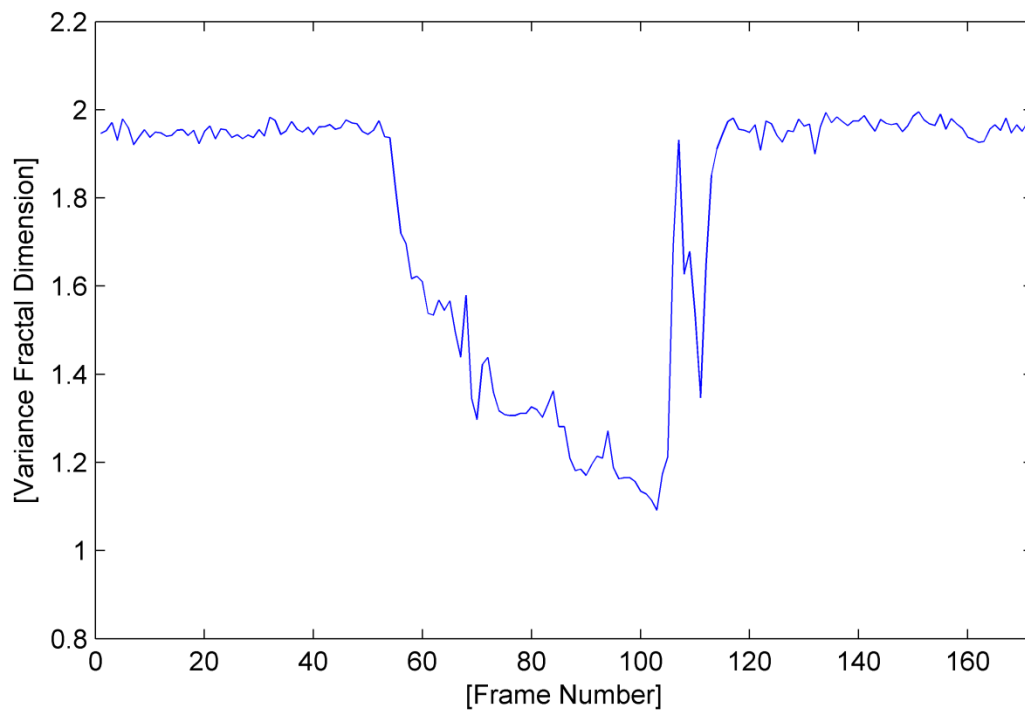


Fig. A.78. The variance fractal dimension trajectory of the utterance "hand" after addition of white noise.

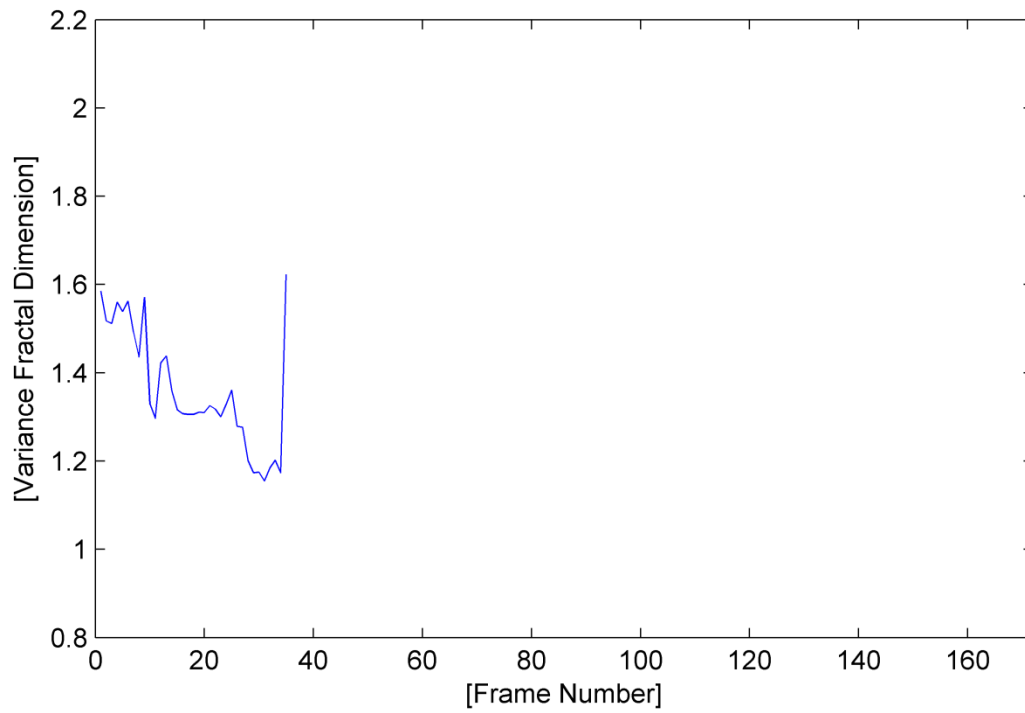


Fig. A.79. The trajectory of the utterance “hand” detected by the voice activity detection algorithm.

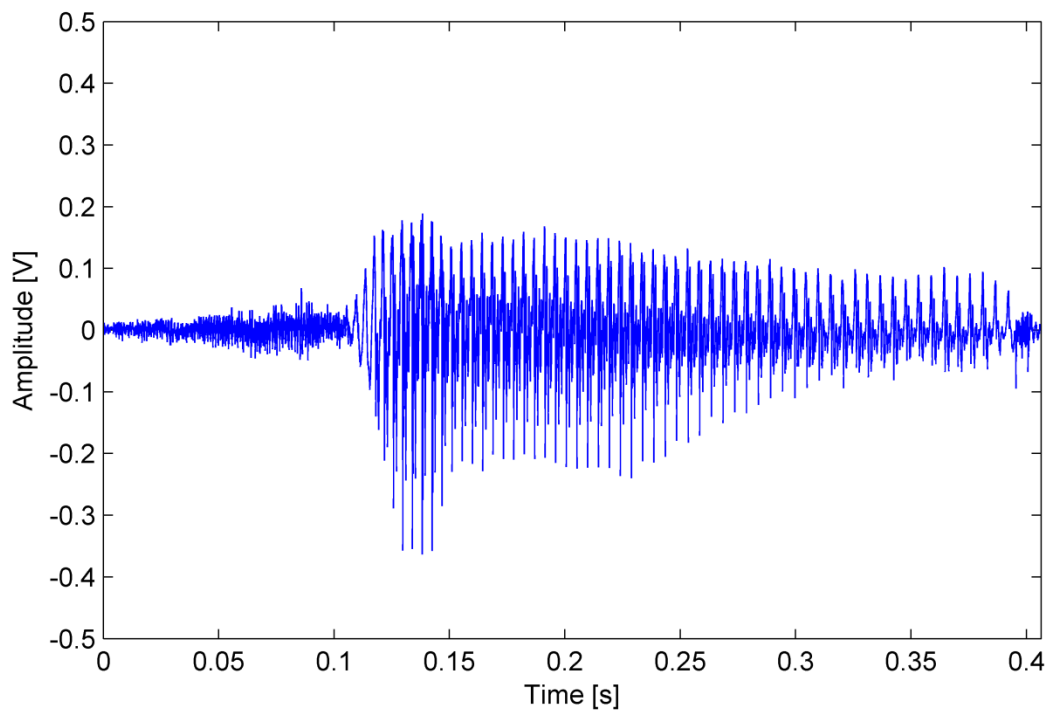


Fig. A.80. The waveform of the utterance “hand” detected by the voice activity detection algorithm.

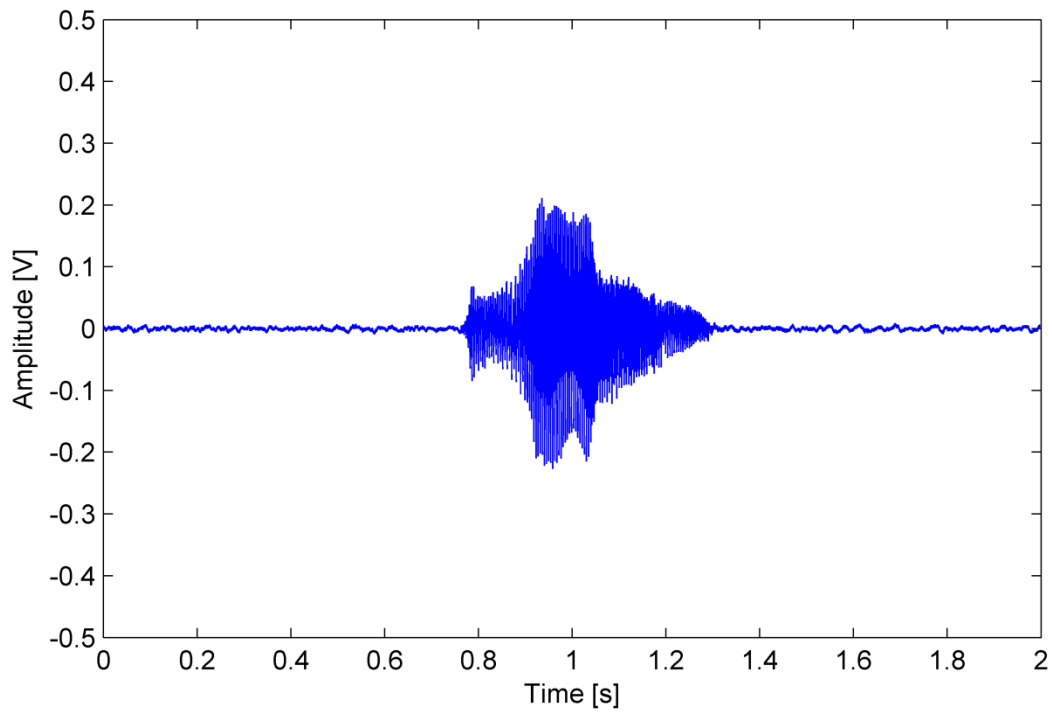


Fig. A.81. The waveform of the utterance "rear".

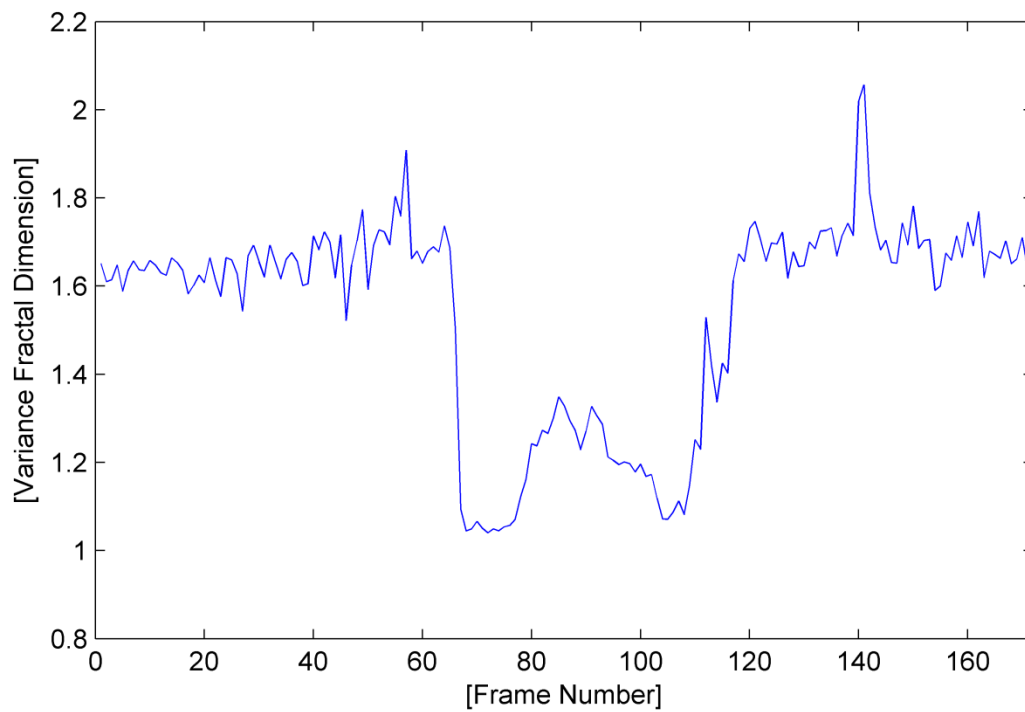


Fig. A.82. The variance fractal dimension trajectory of the utterance "rear".

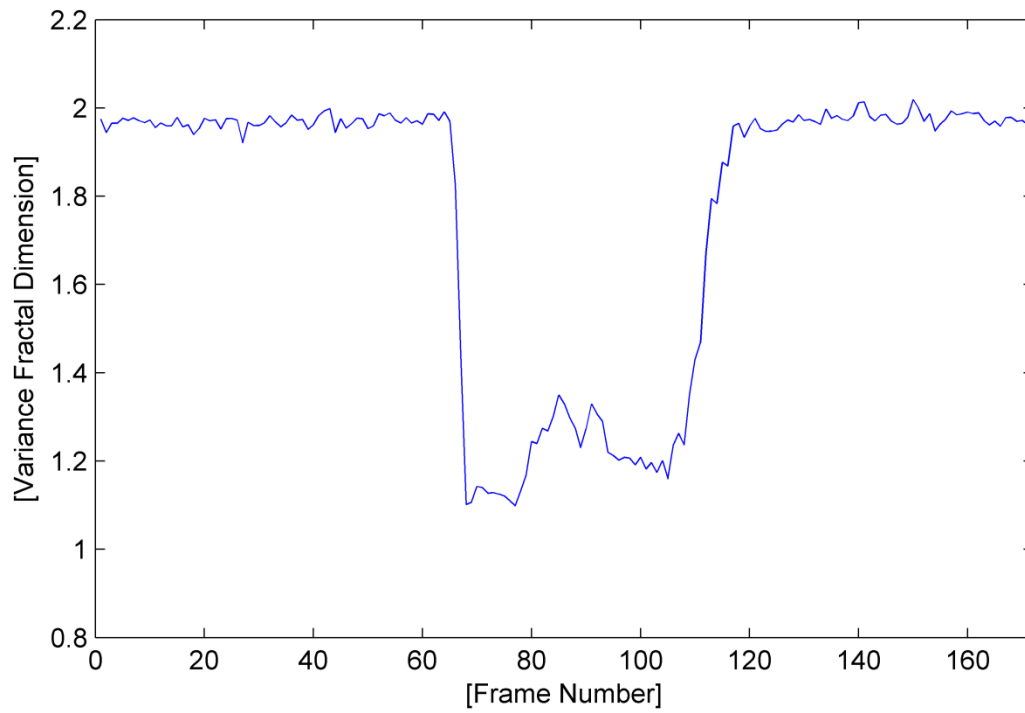


Fig. A.83. The variance fractal dimension trajectory of the utterance “rear” after addition of white noise.

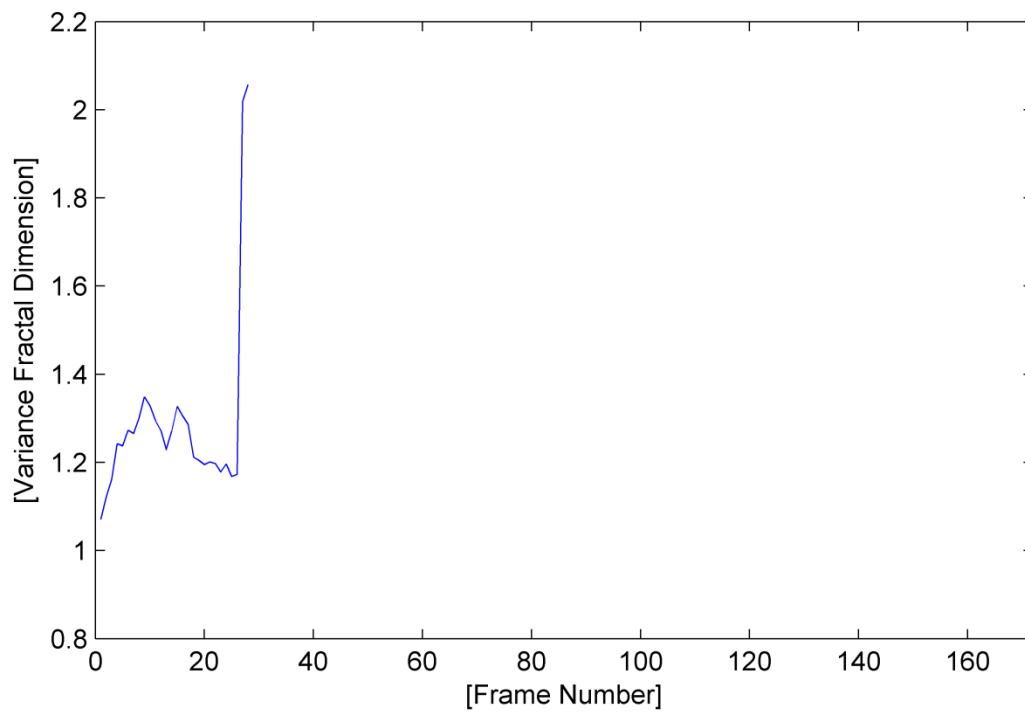


Fig. A.84. The trajectory of the utterance “rear” detected by the voice activity detection algorithm.

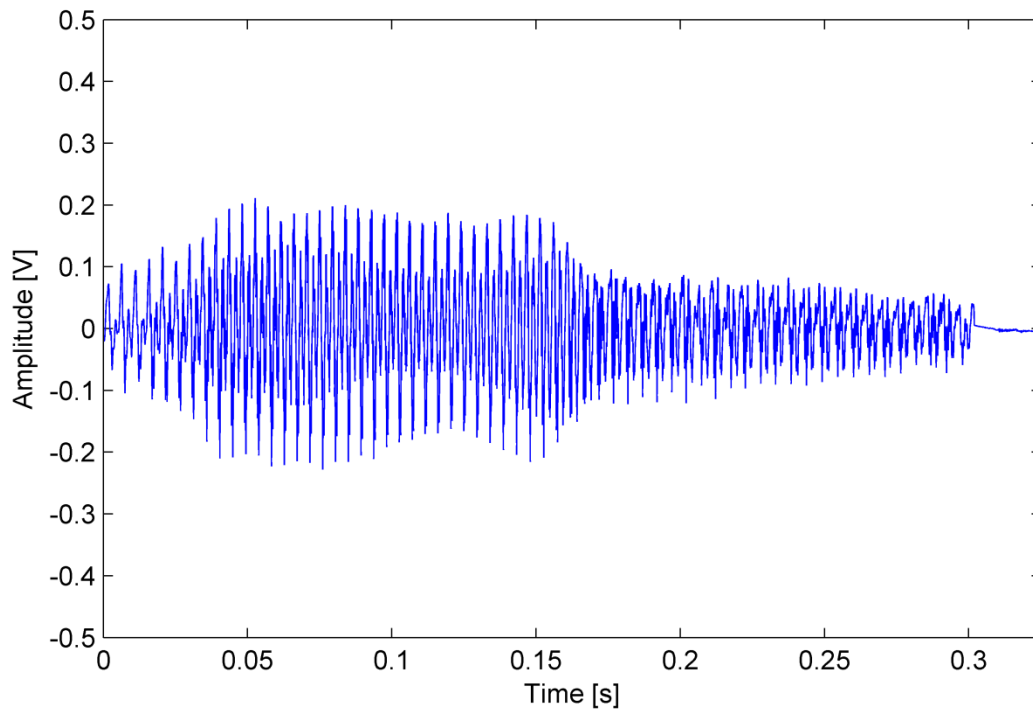


Fig. A.85. The waveform of the utterance "rear" detected by the voice activity detection algorithm.

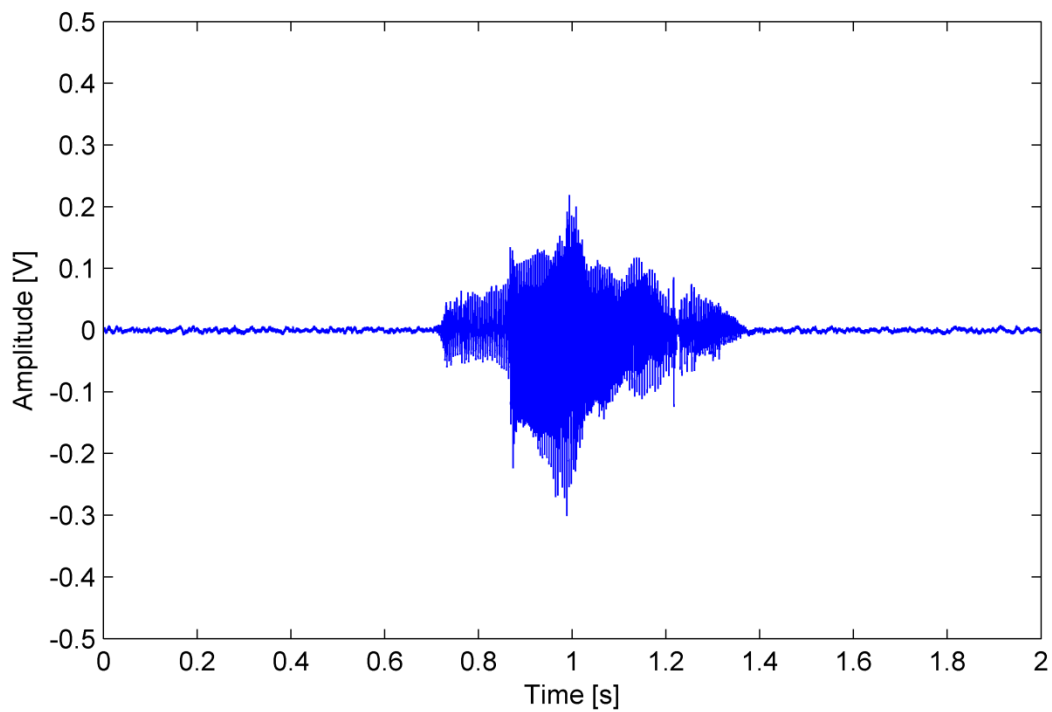


Fig. A.86. The waveform of the utterance "loyal".

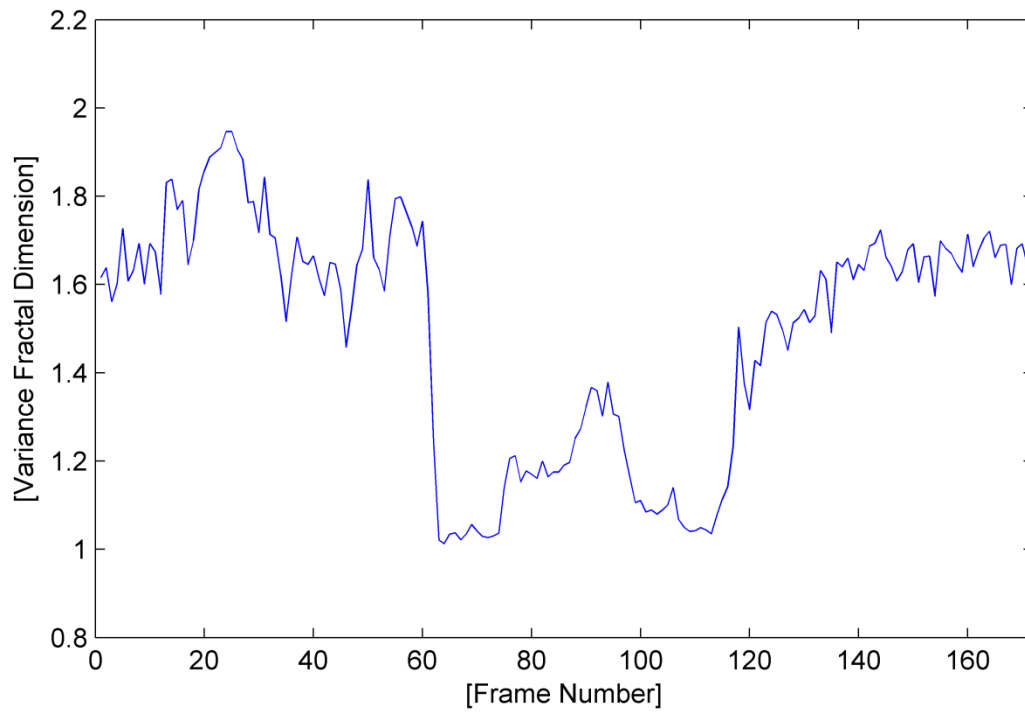


Fig. A.87. The variance fractal dimension trajectory of the utterance "loyal".

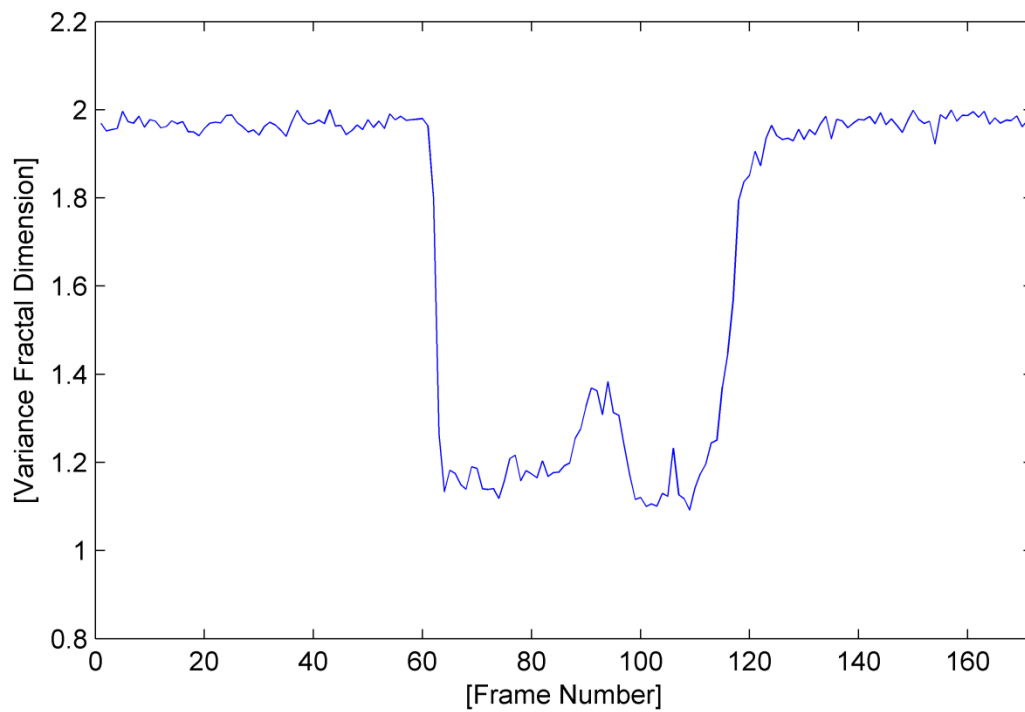


Fig. A.88. The variance fractal dimension trajectory of the utterance "loyal" after addition of white noise.

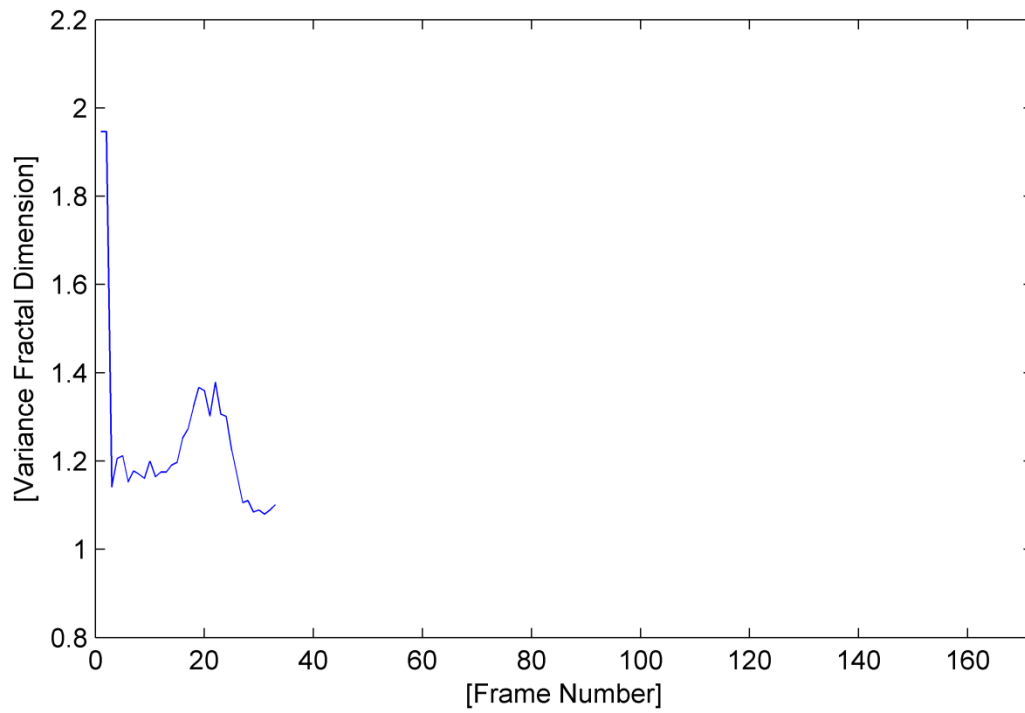


Fig. A.89. The trajectory of the utterance “loyal” detected by the voice activity detection algorithm.

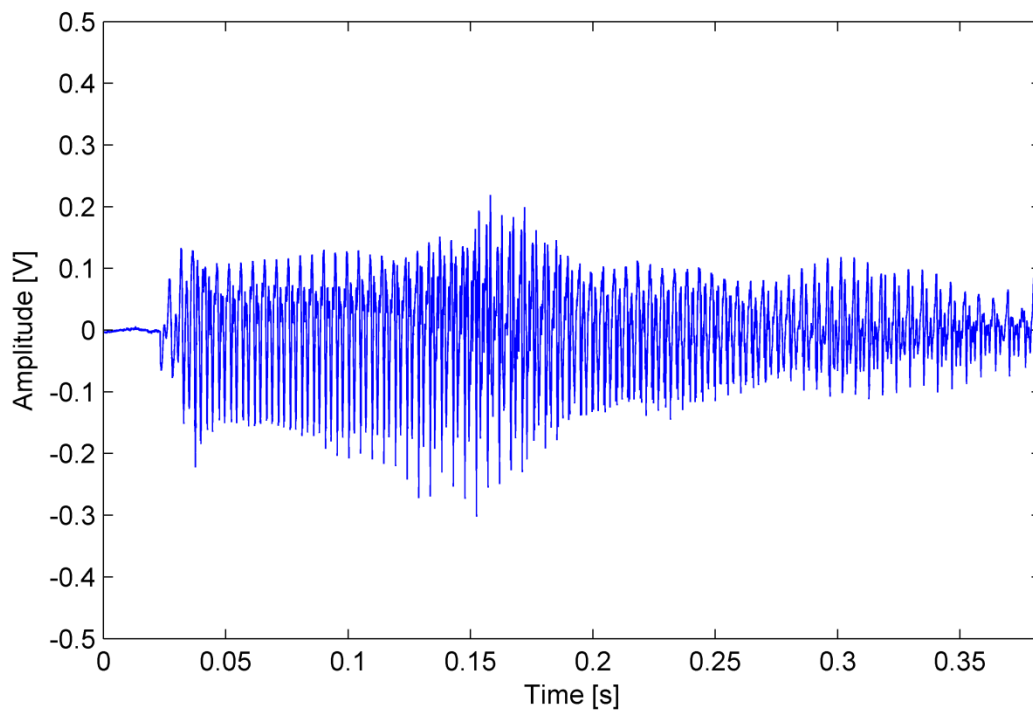


Fig. A.90. The waveform of the utterance “loyal” detected by the voice activity detection algorithm.

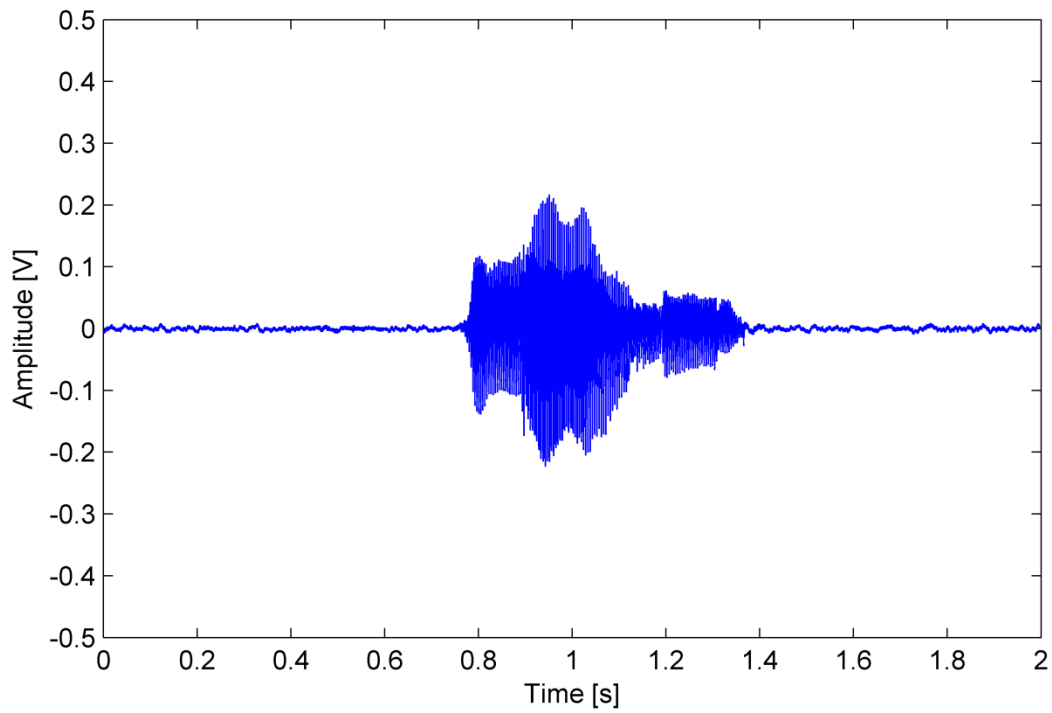


Fig. A.91. The waveform of the utterance "mime".

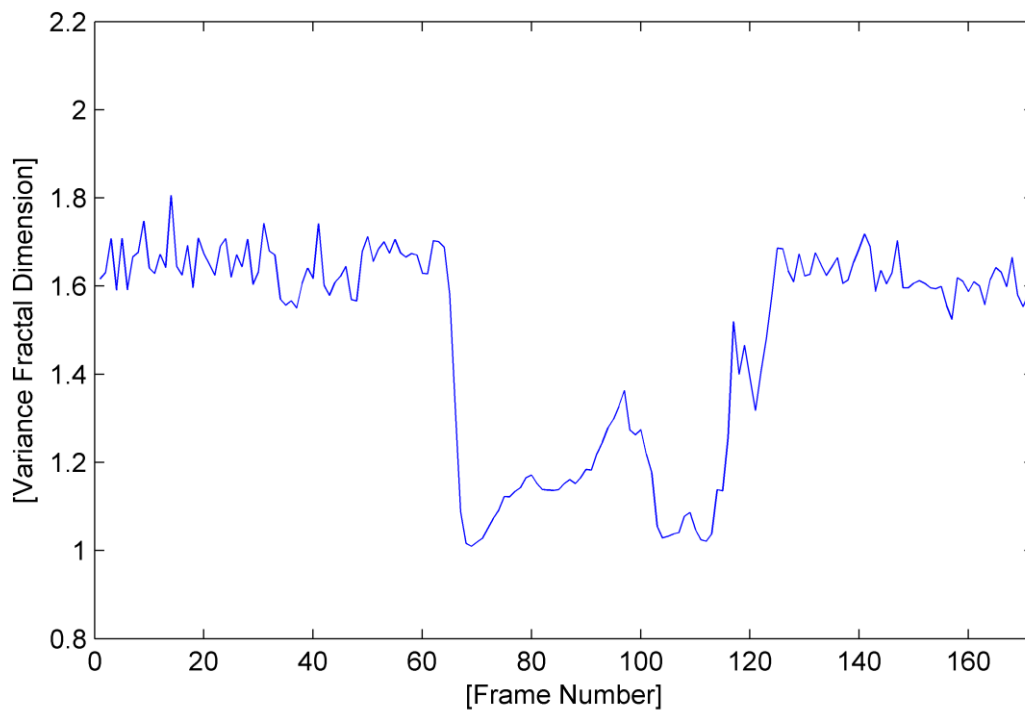


Fig. A.92. The variance fractal dimension trajectory of the utterance "mime".

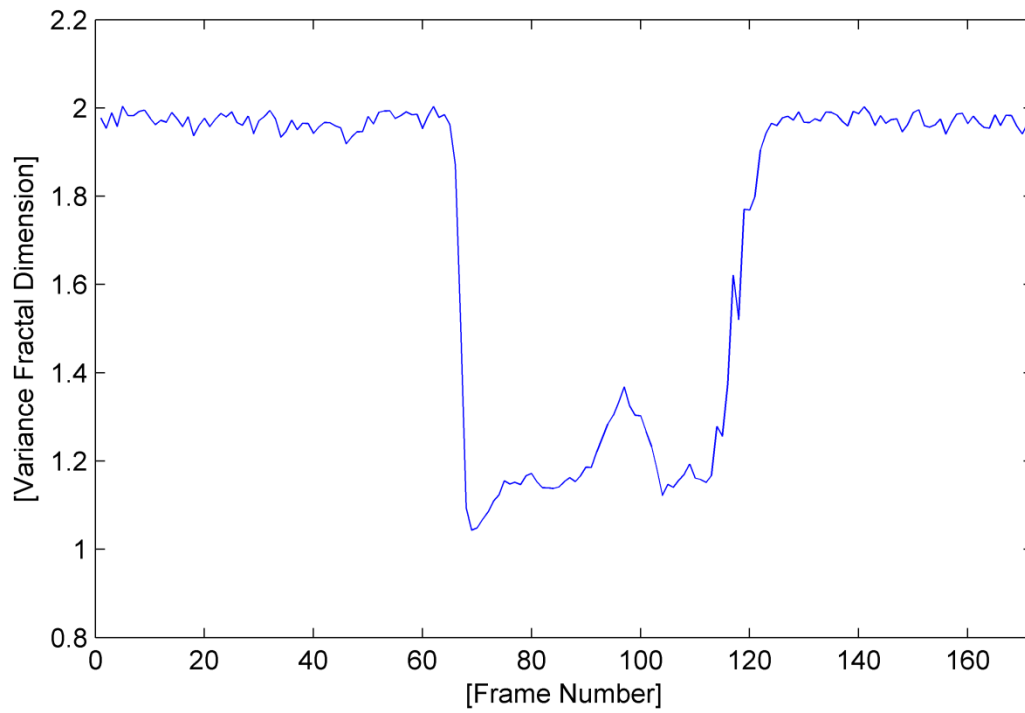


Fig. A.93. The variance fractal dimension trajectory of the utterance “mime” after addition of white noise.

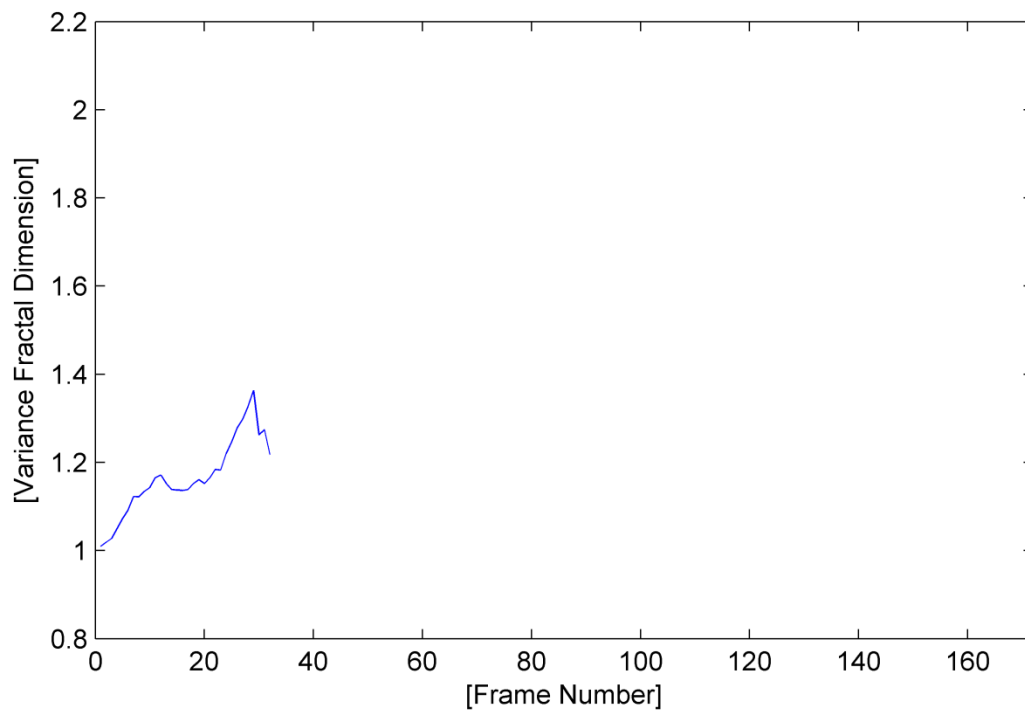


Fig. A.94. The trajectory of the utterance “mime” detected by the voice activity detection algorithm.

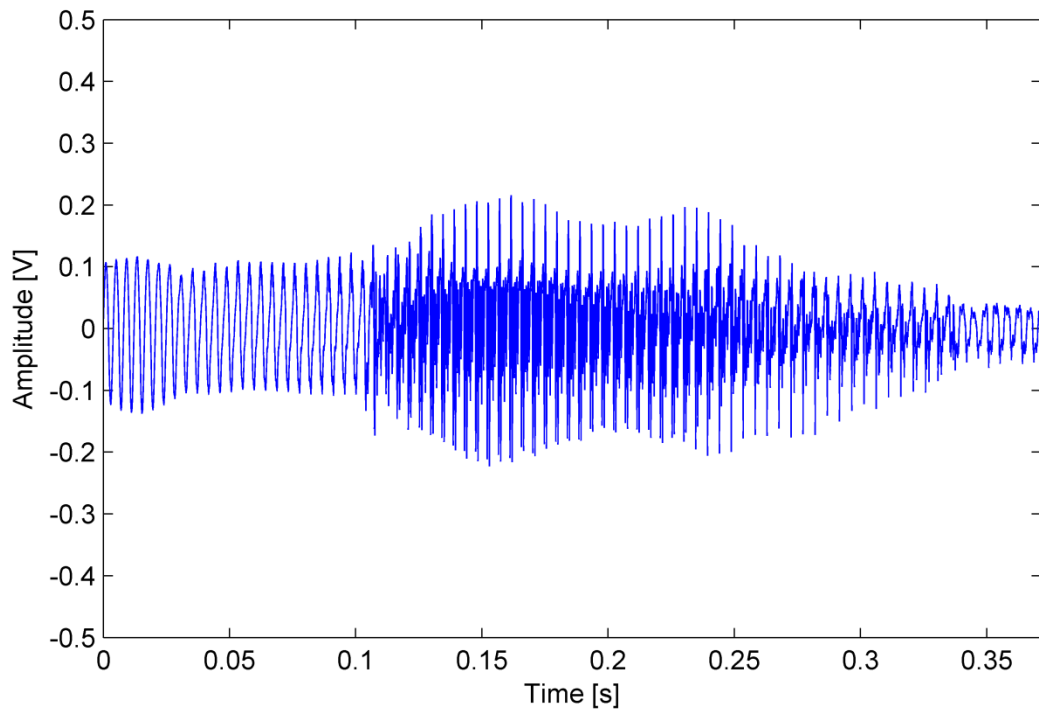


Fig. A.95. The waveform of the utterance "mime" detected by the voice activity detection algorithm.

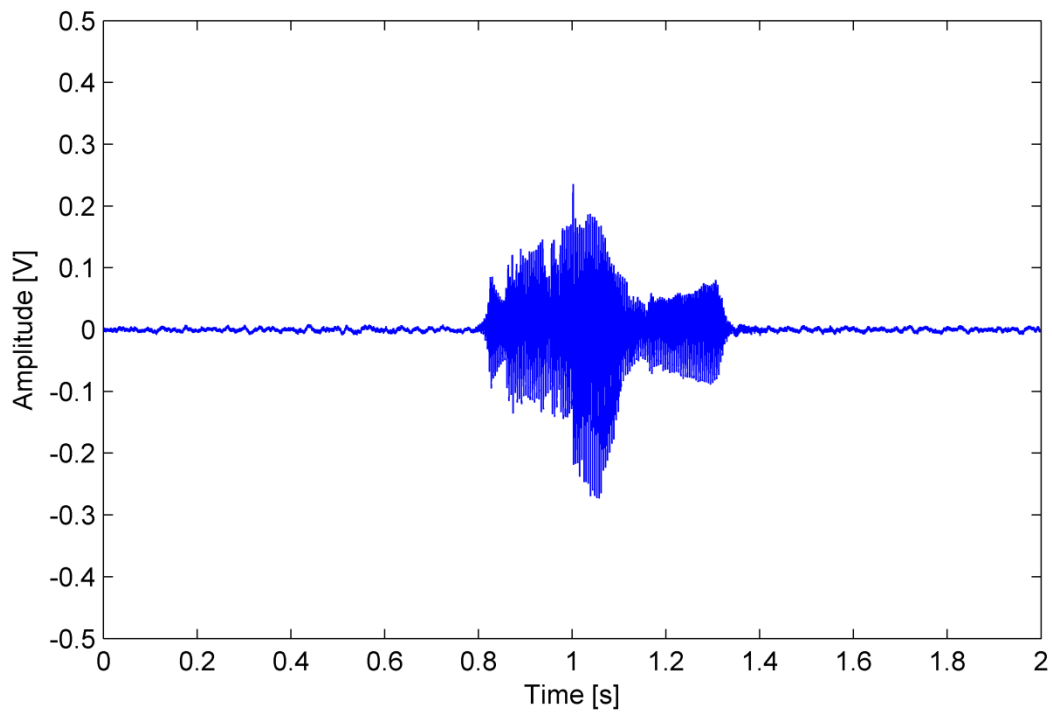


Fig. A.96. The waveform of the utterance "none".

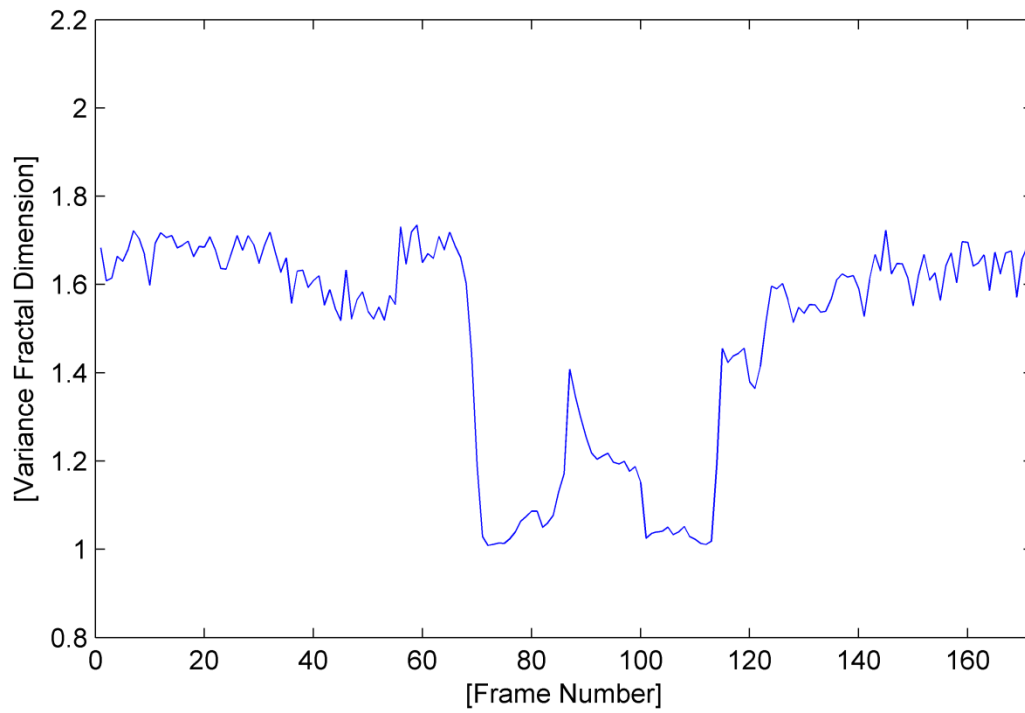


Fig. A.97. The variance fractal dimension trajectory of the utterance "none".

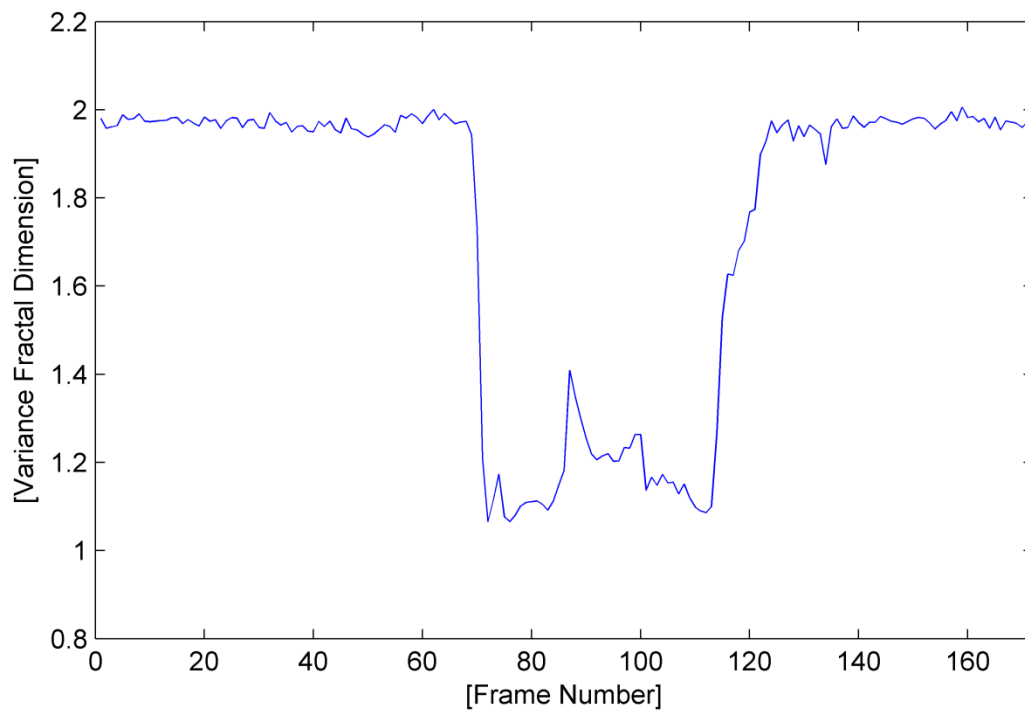


Fig. A.98. The variance fractal dimension trajectory of the utterance "none" after addition of white noise.

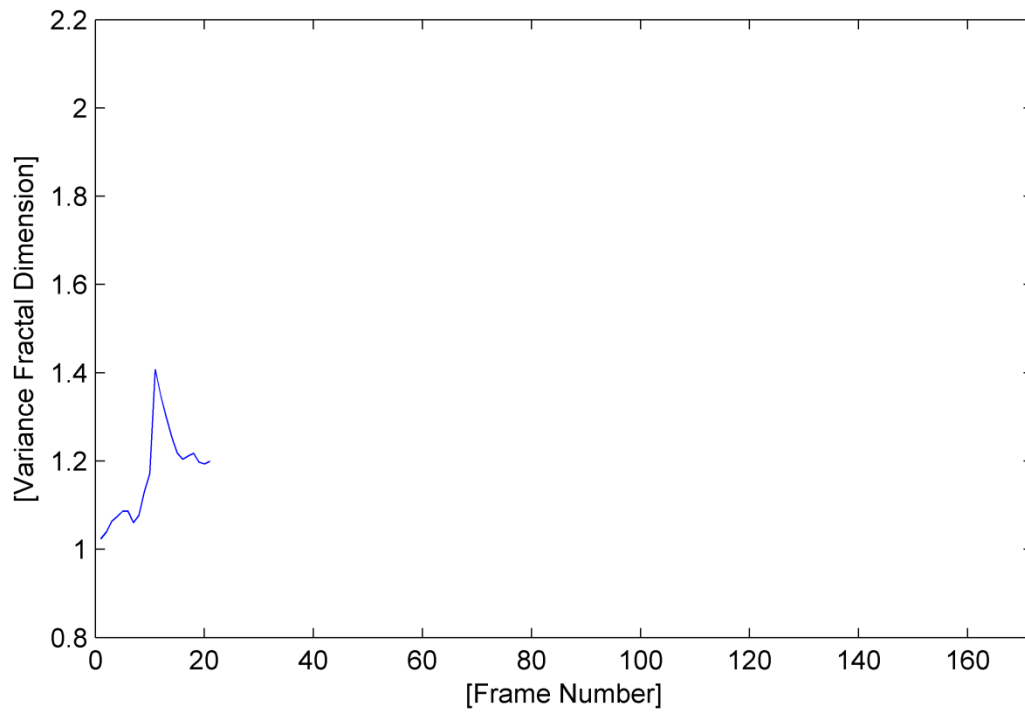


Fig. A.99. The trajectory of the utterance “none” detected by the voice activity detection algorithm.

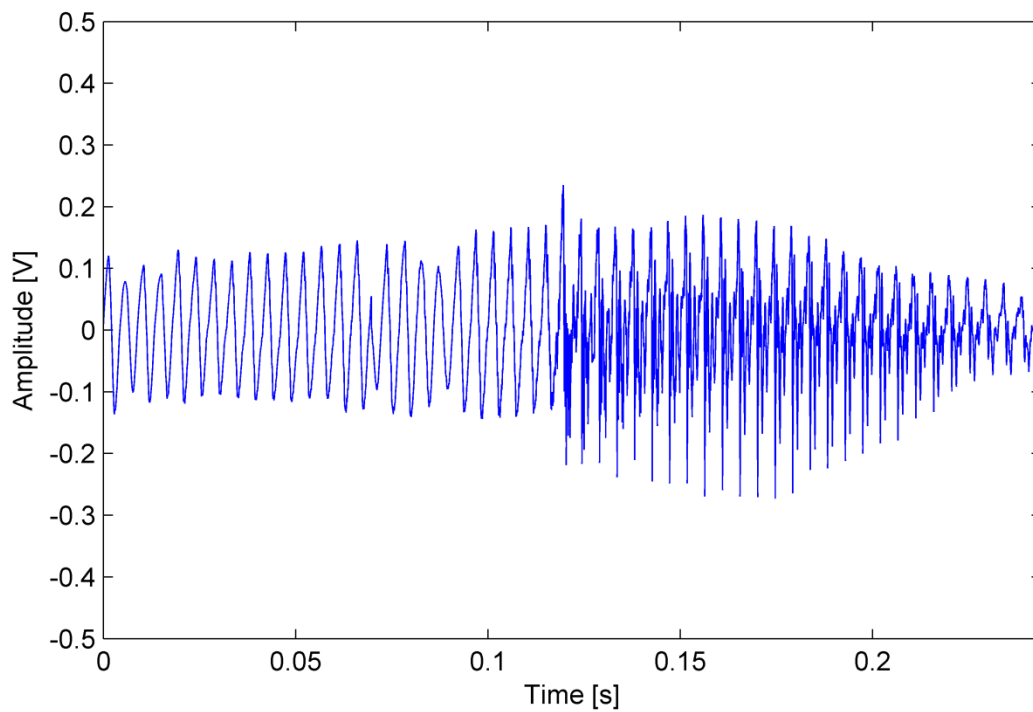


Fig. A.100. The waveform of the utterance “none” detected by the voice activity detection algorithm.

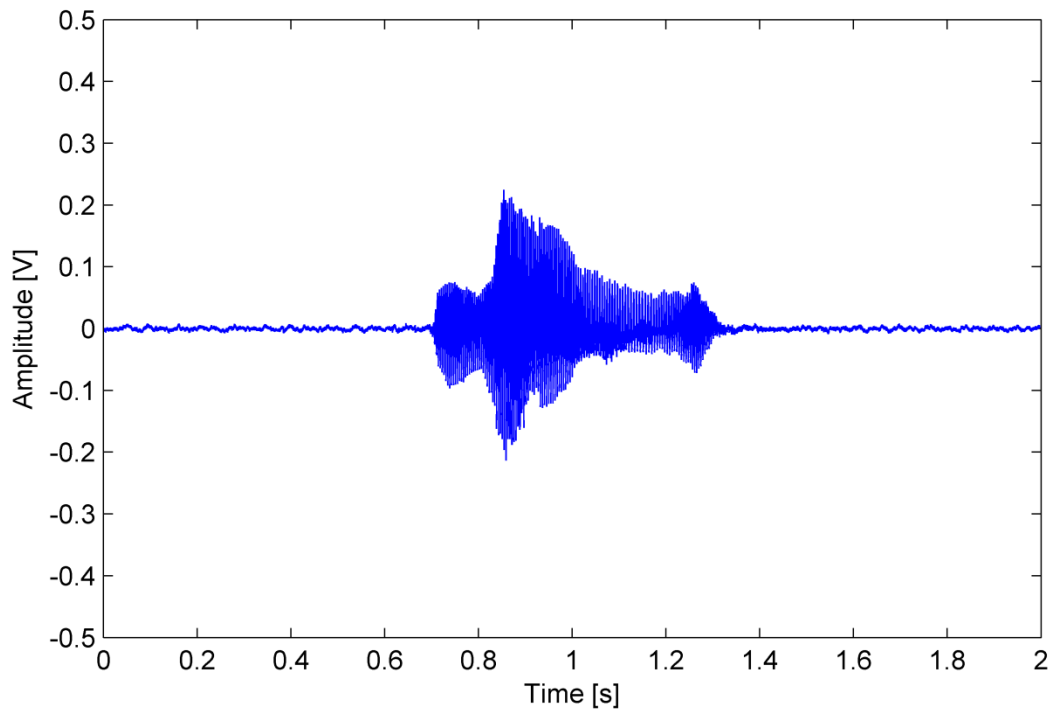


Fig. A.101. The waveform of the utterance “ringing”.

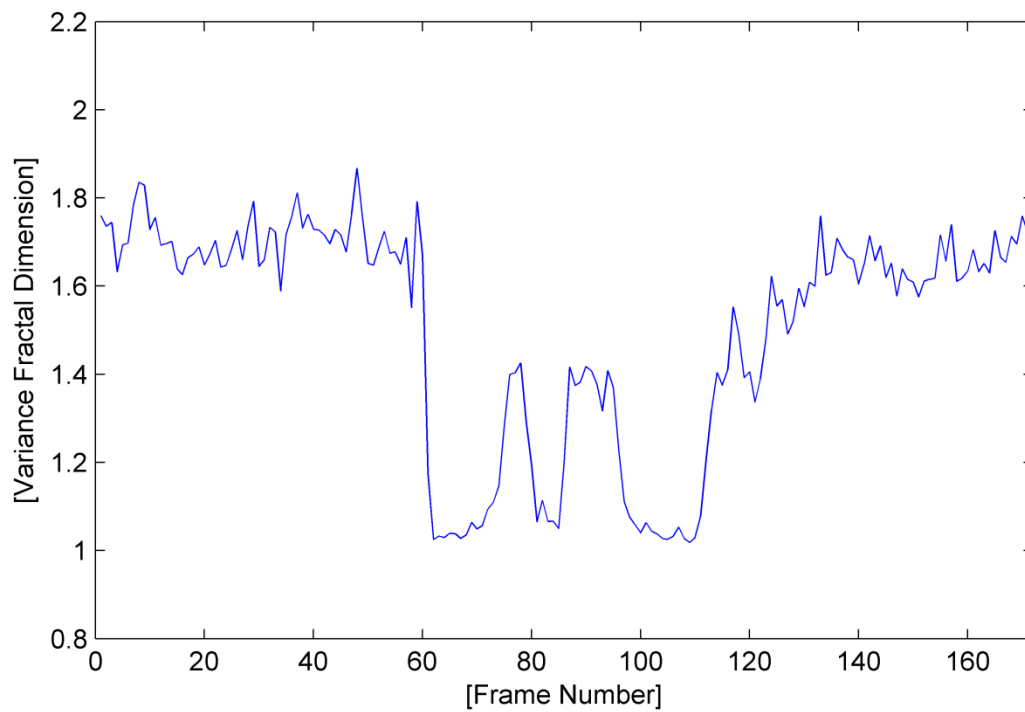


Fig. A.102. The variance fractal dimension trajectory of the utterance “ringing”.

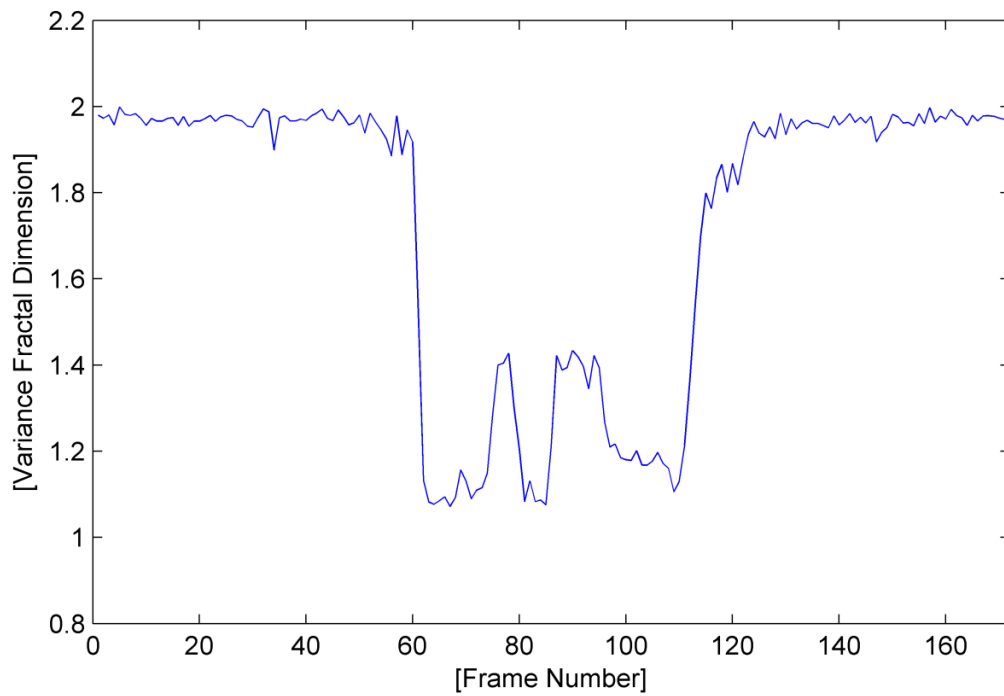


Fig. A.103. The variance fractal dimension trajectory of the utterance “ringing” after addition of white noise.

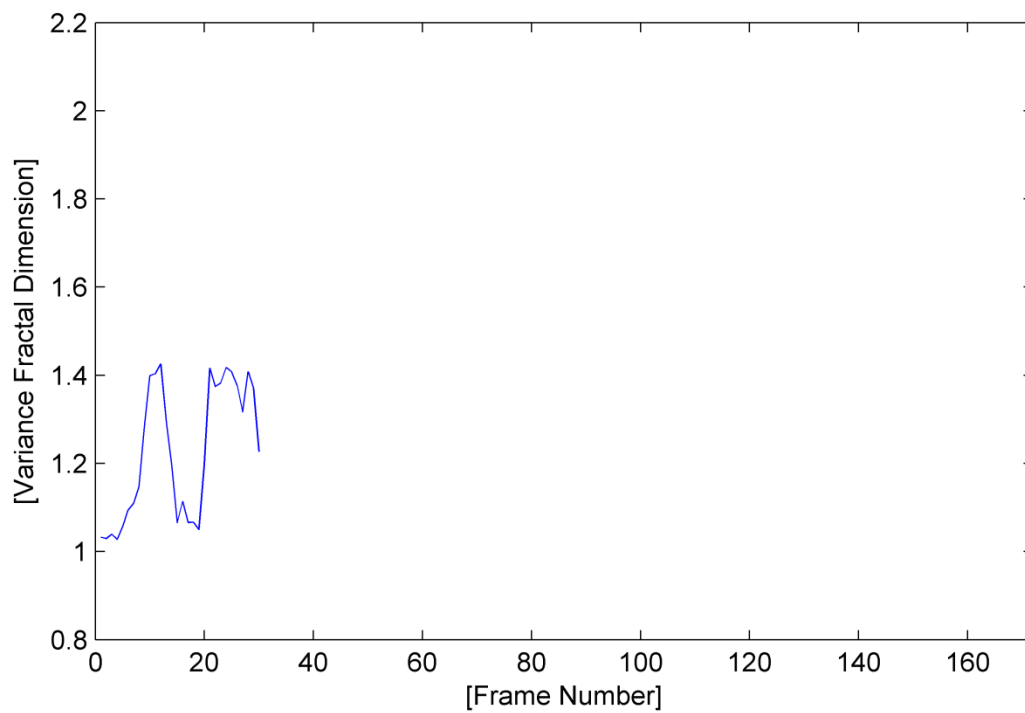


Fig. A.104. The trajectory of the utterance “ringing” detected by the voice activity detection algorithm.

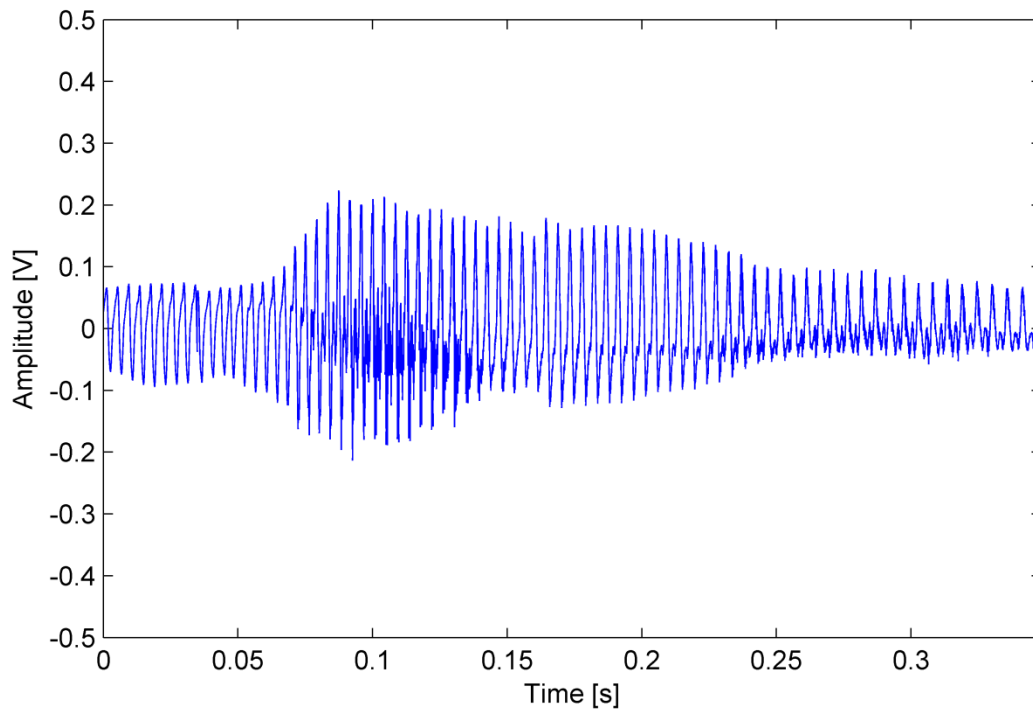


Fig. A.105. The waveform of the utterance “ringing” detected by the voice activity detection algorithm.

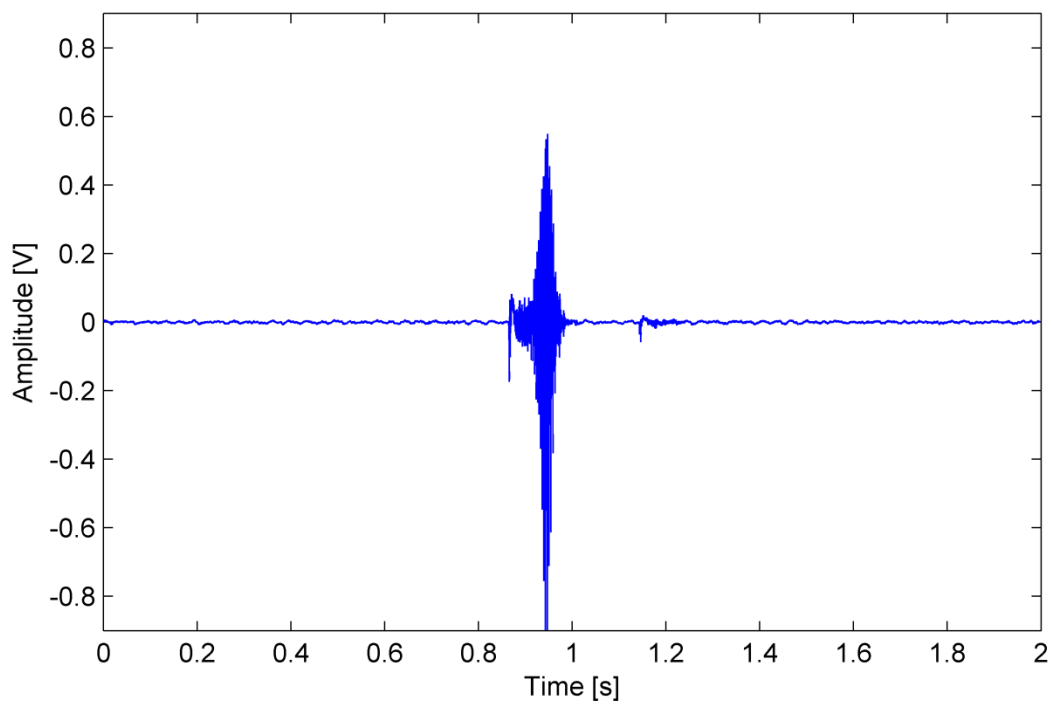


Fig. A.106. The waveform of the utterance “pip”.

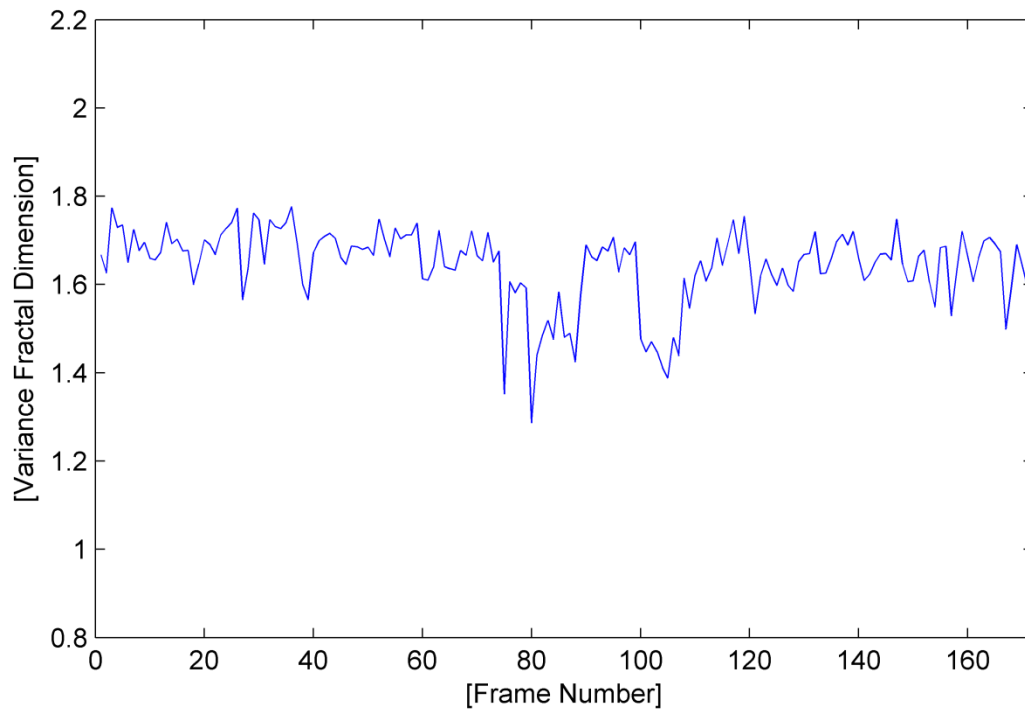


Fig. A.107. The variance fractal dimension trajectory of the utterance “pip”.

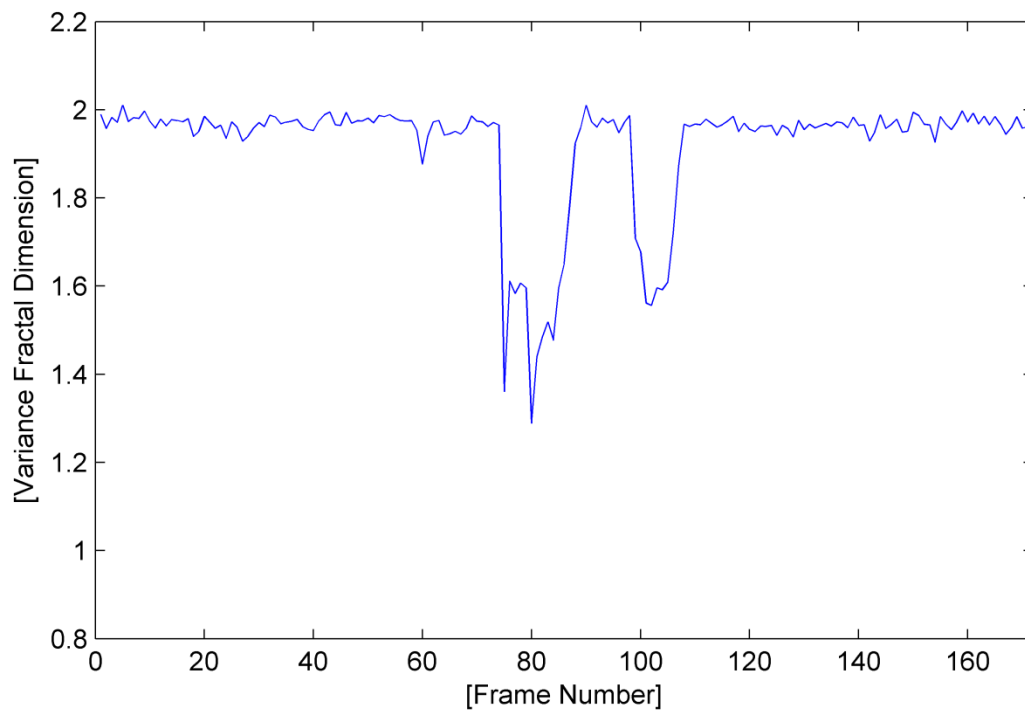


Fig. A.108. The variance fractal dimension trajectory of the utterance “pip” after addition of white noise.

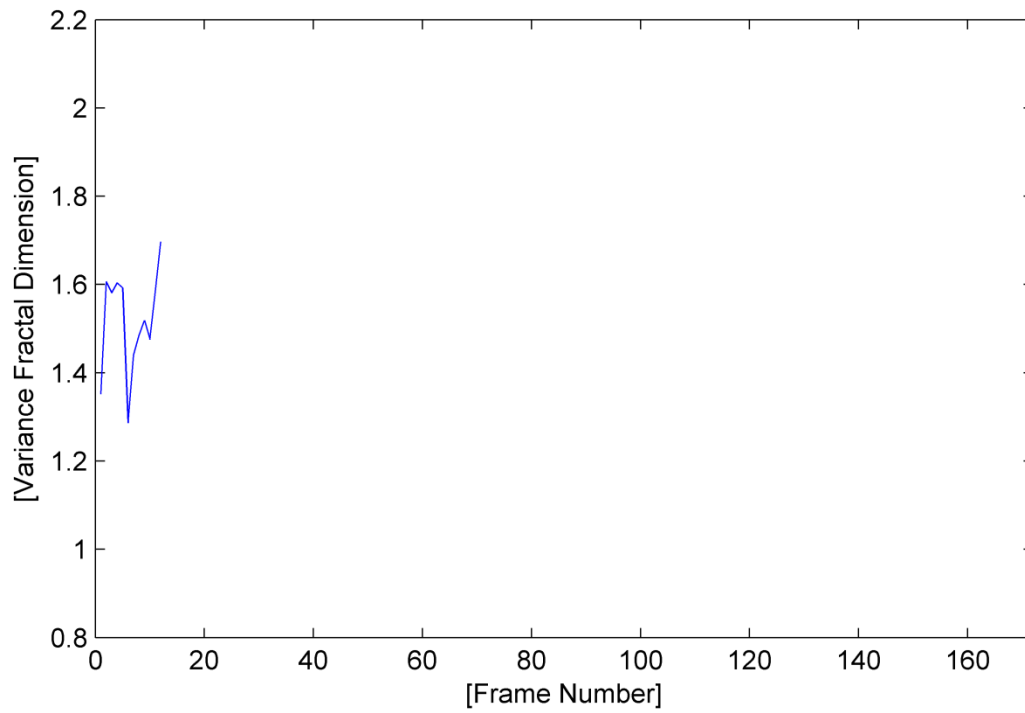


Fig. A.109. The trajectory of the utterance “pip” detected by the voice activity detection algorithm.

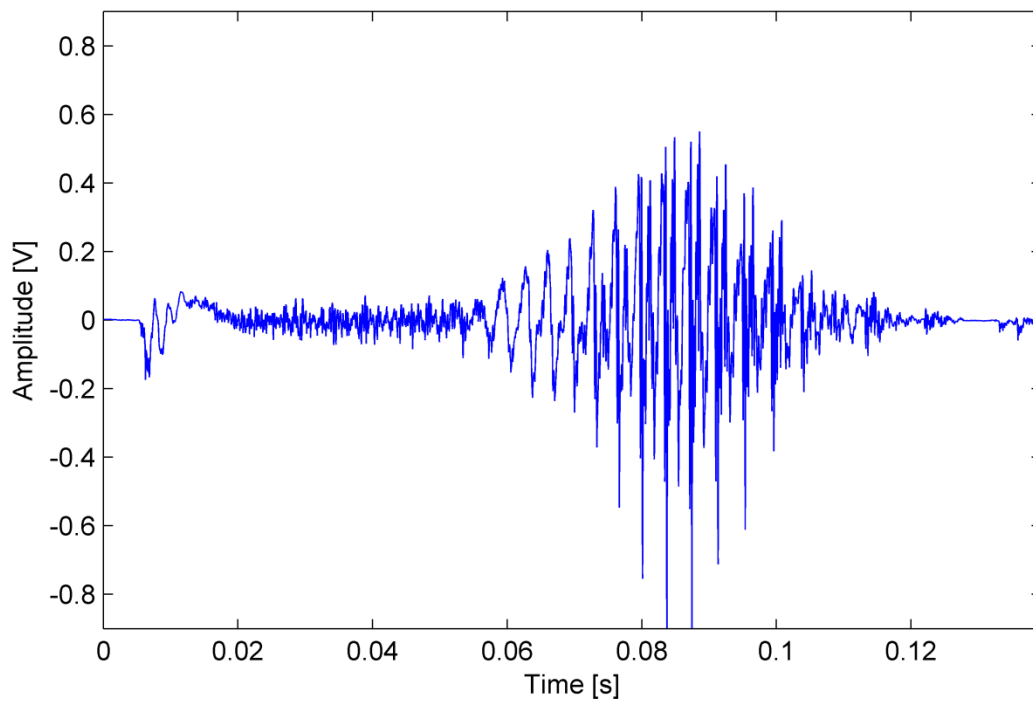


Fig. A.110. The waveform of the utterance “pip” detected by the voice activity detection algorithm.

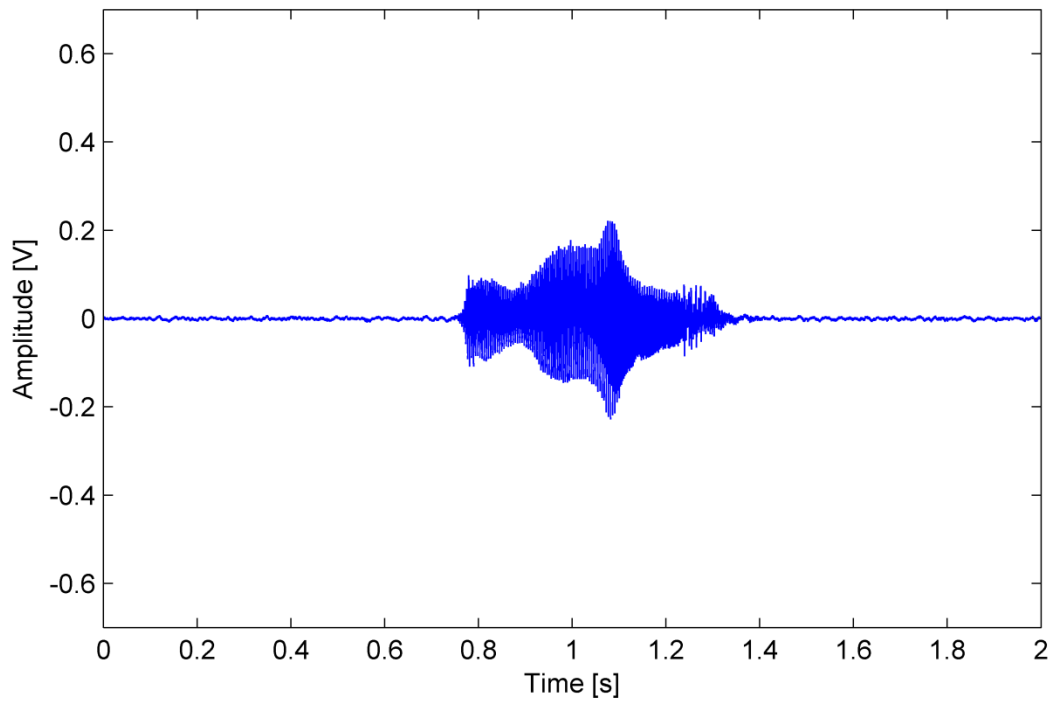


Fig. A.111. The waveform of the utterance “year”.

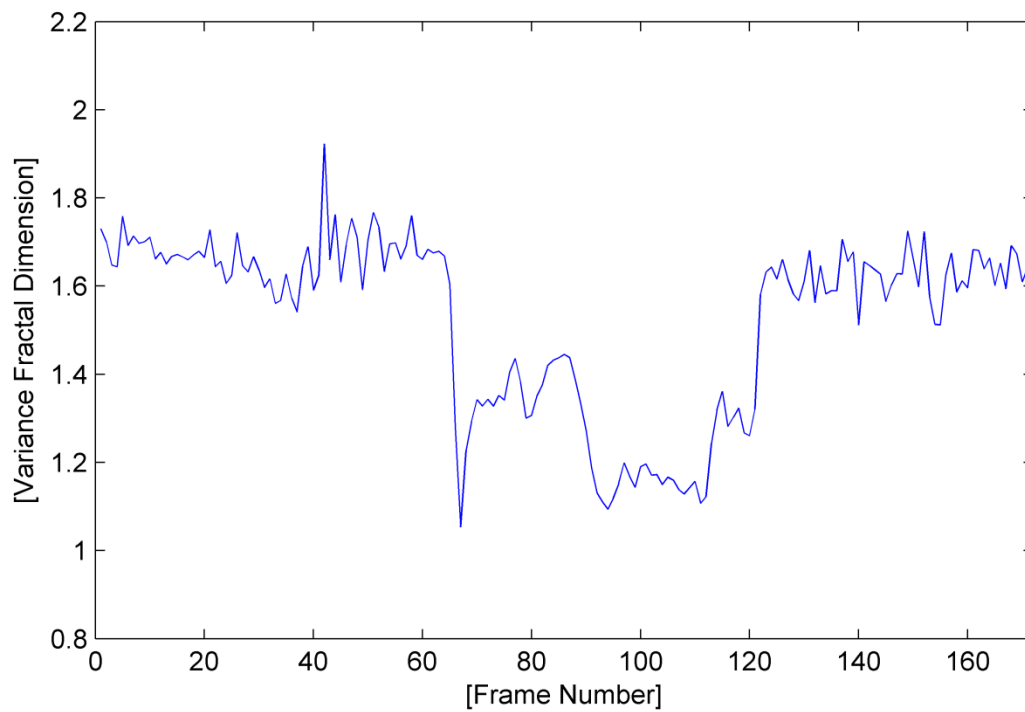


Fig. A.112. The variance fractal dimension trajectory of the utterance “year”.

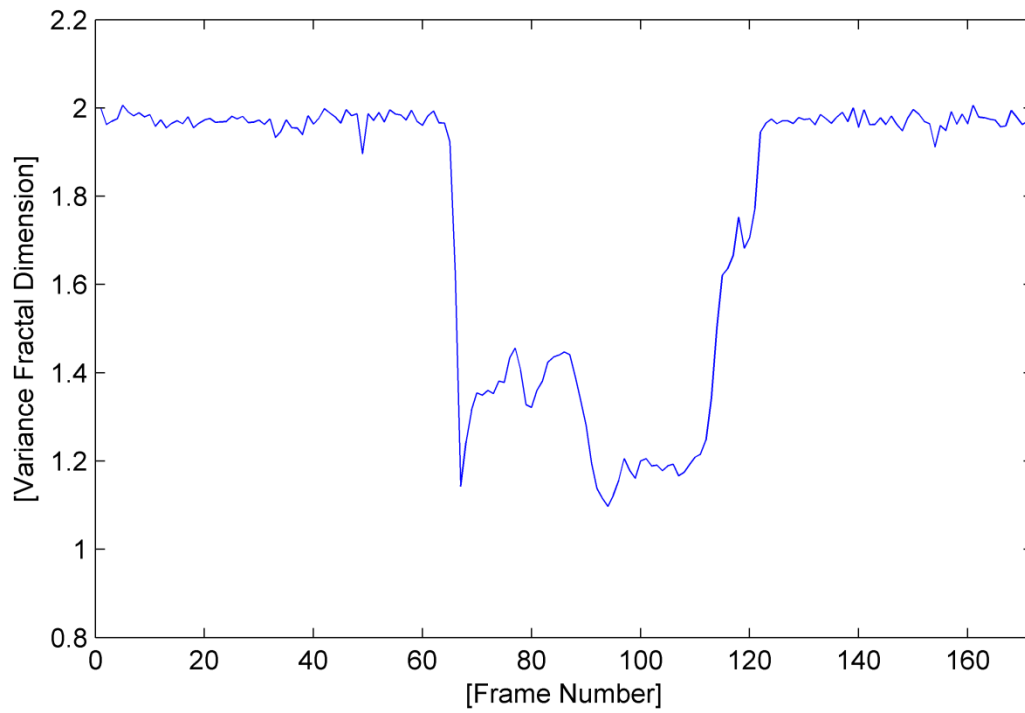


Fig. A.113. The variance fractal dimension trajectory of the utterance “year” after addition of white noise.

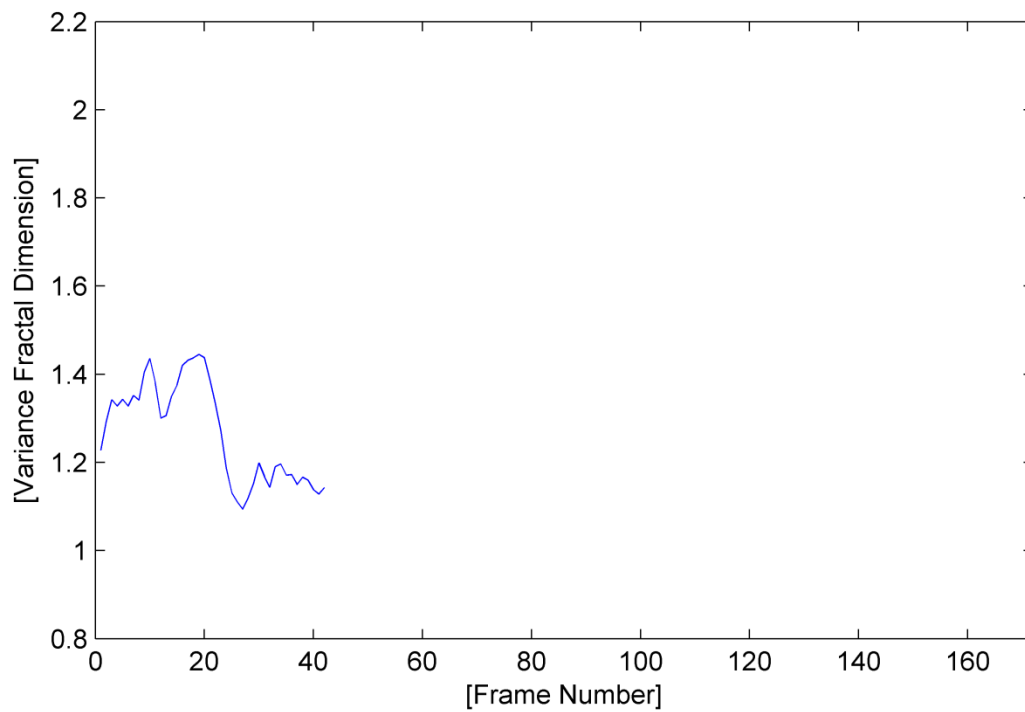


Fig. A.114. The trajectory of the utterance “year” detected by the voice activity detection algorithm.

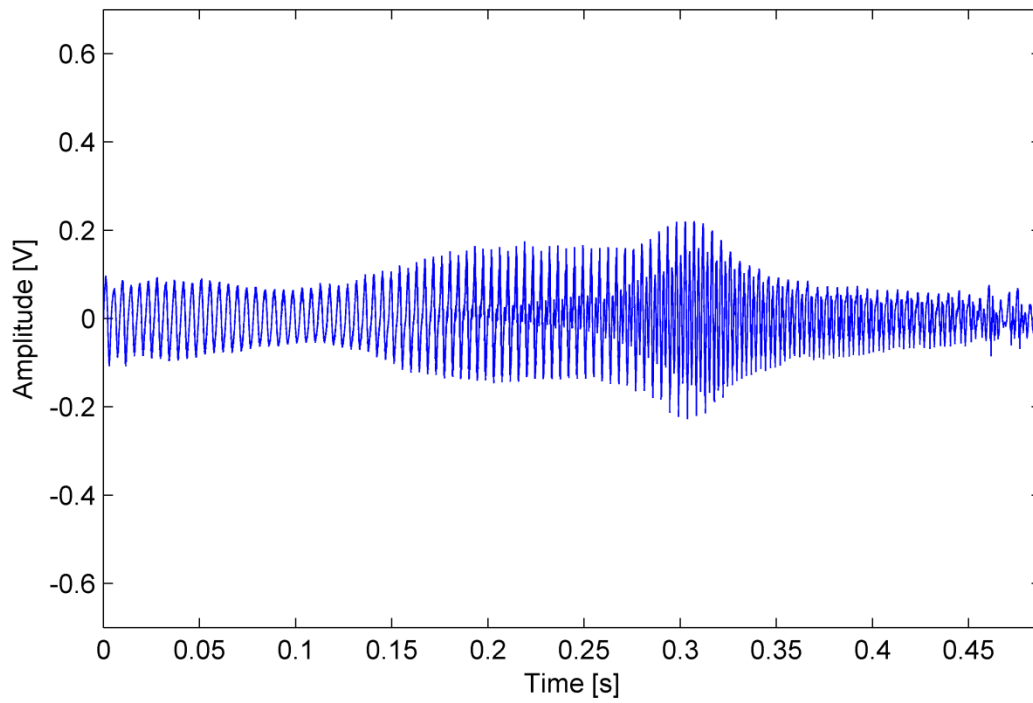


Fig. A.115. The waveform of the utterance "year" detected by the voice activity detection algorithm.

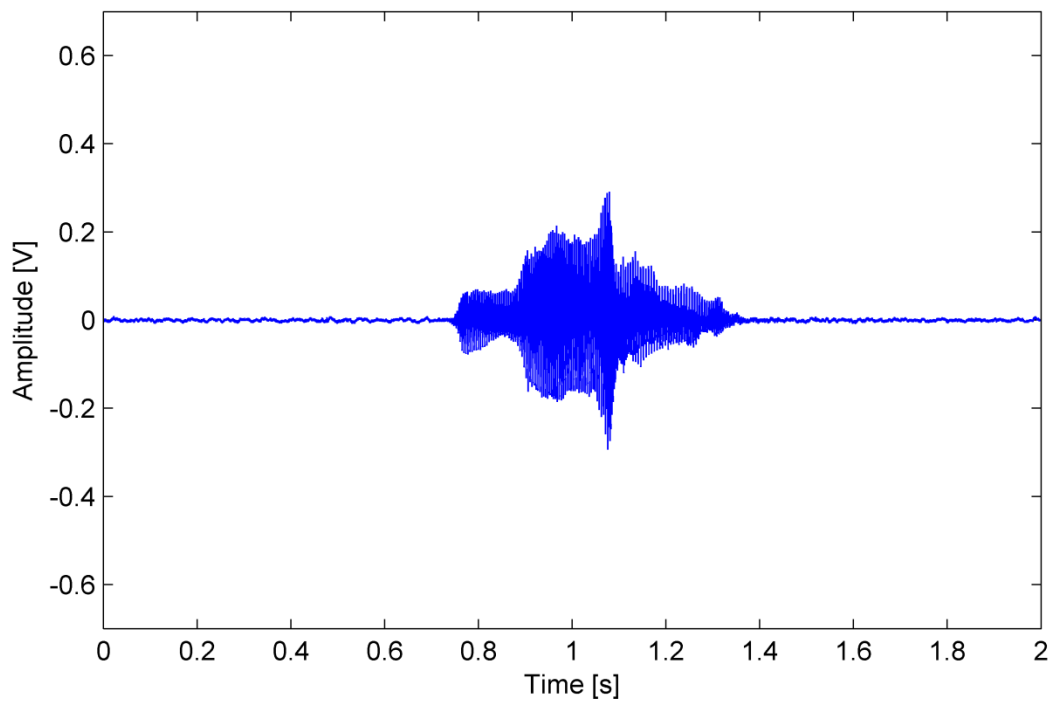


Fig. A.116. The waveform of the utterance "weal".

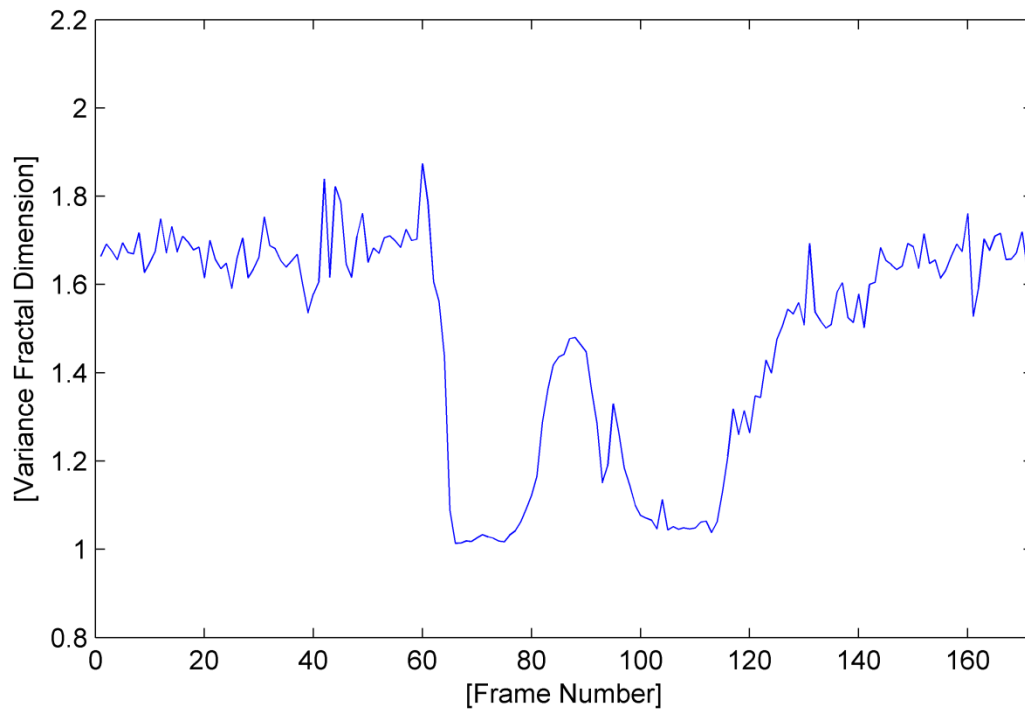


Fig. A.117. The variance fractal dimension trajectory of the utterance "weal".

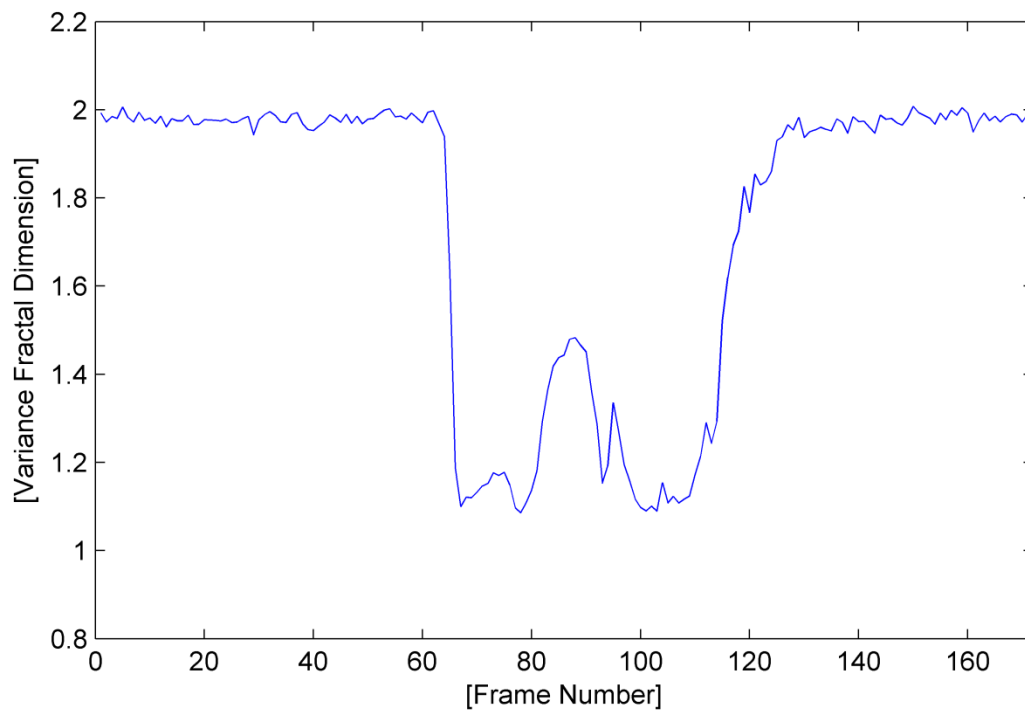


Fig. A.118. The variance fractal dimension trajectory of the utterance "weal" after addition of white noise.

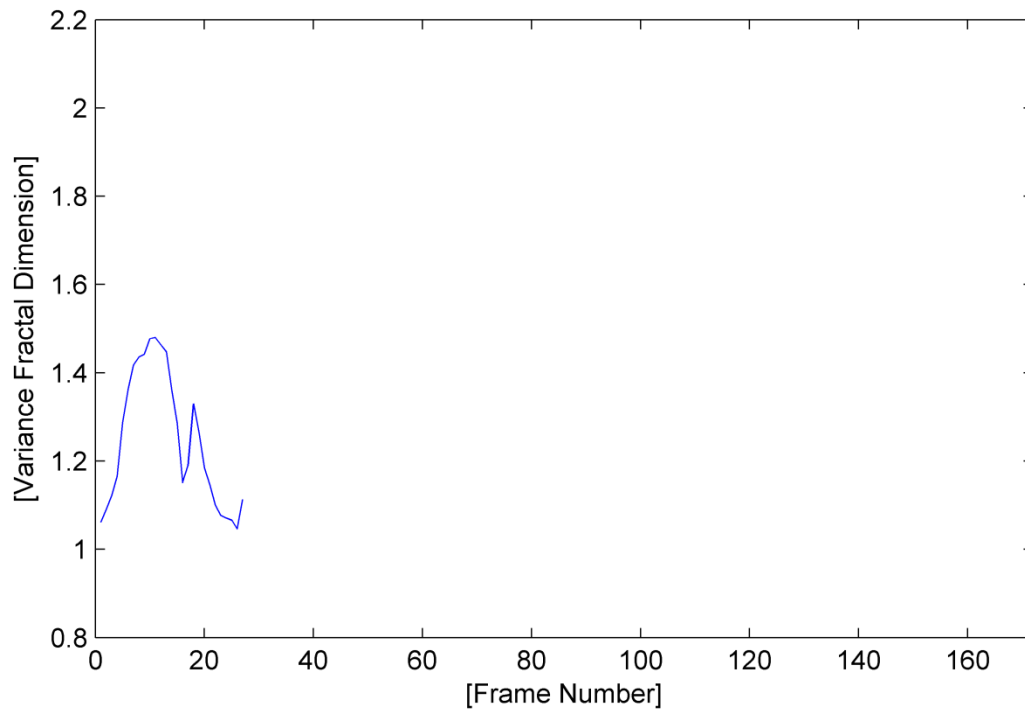


Fig. A.119. The trajectory of the utterance “weal” detected by the voice activity detection algorithm.

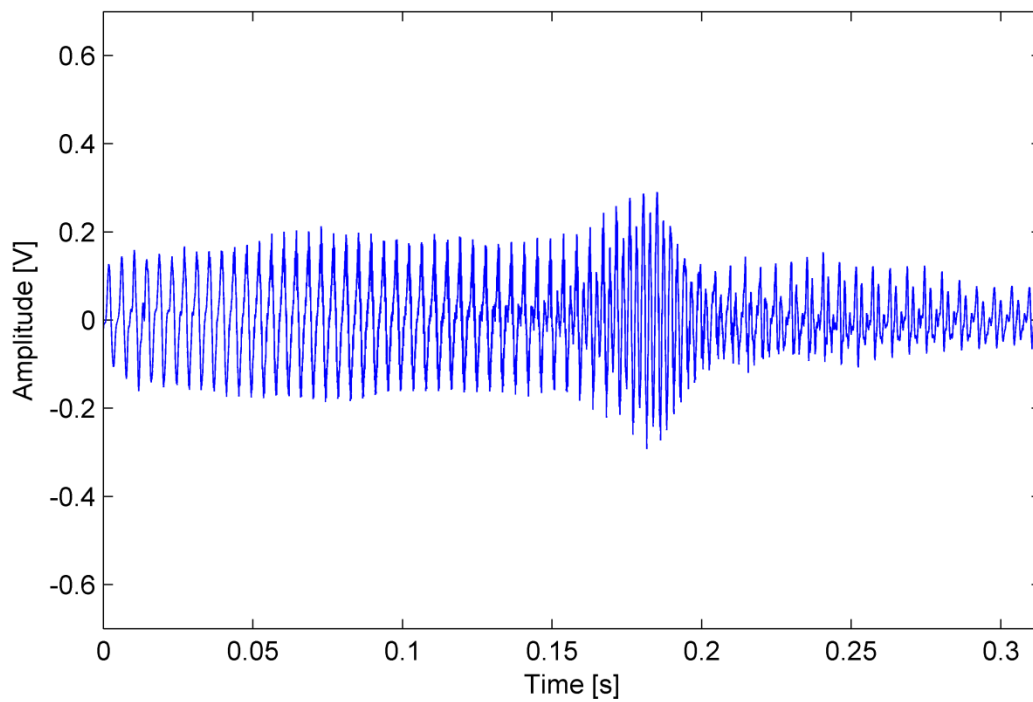


Fig. A.120. The waveform of the utterance “weal” detected by the voice activity detection algorithm.

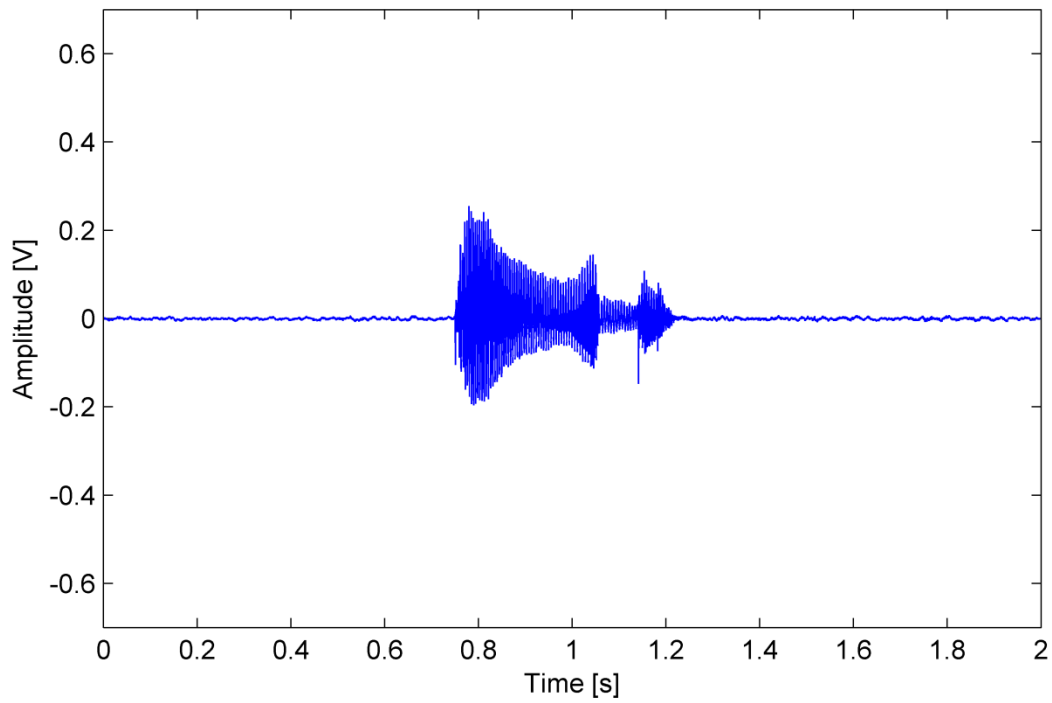


Fig. A.121. The waveform of the utterance "bead".

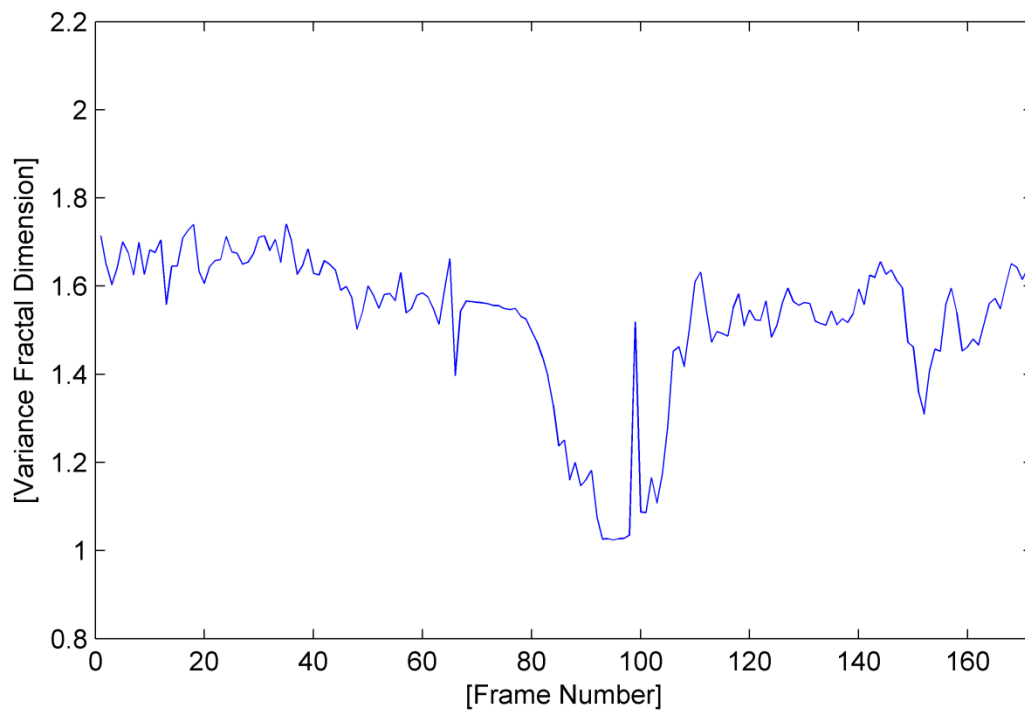


Fig. A.122. The variance fractal dimension trajectory of the utterance "bead".

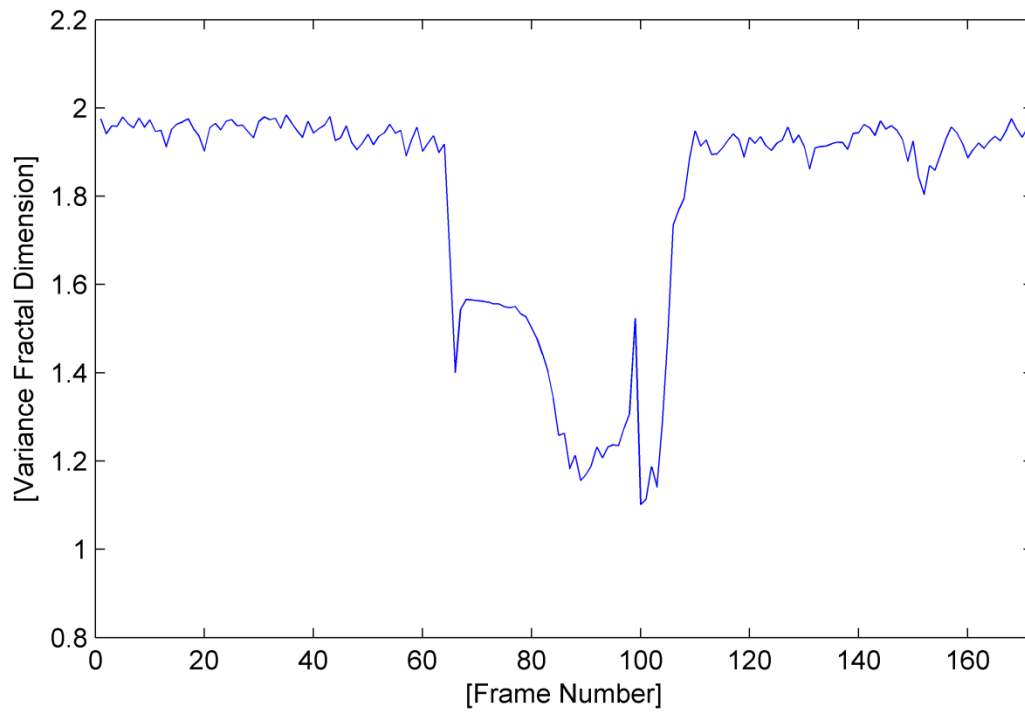


Fig. A.123. The variance fractal dimension trajectory of the utterance “bead” after addition of white noise.

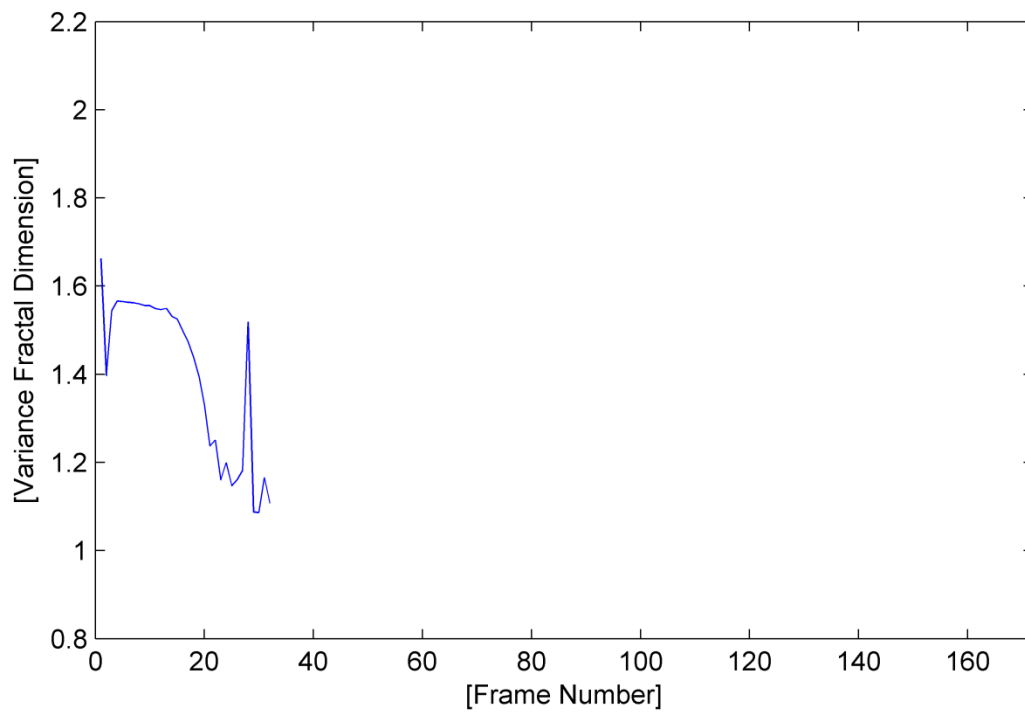


Fig. A.124. The trajectory of the utterance “bead” detected by the voice activity detection algorithm.

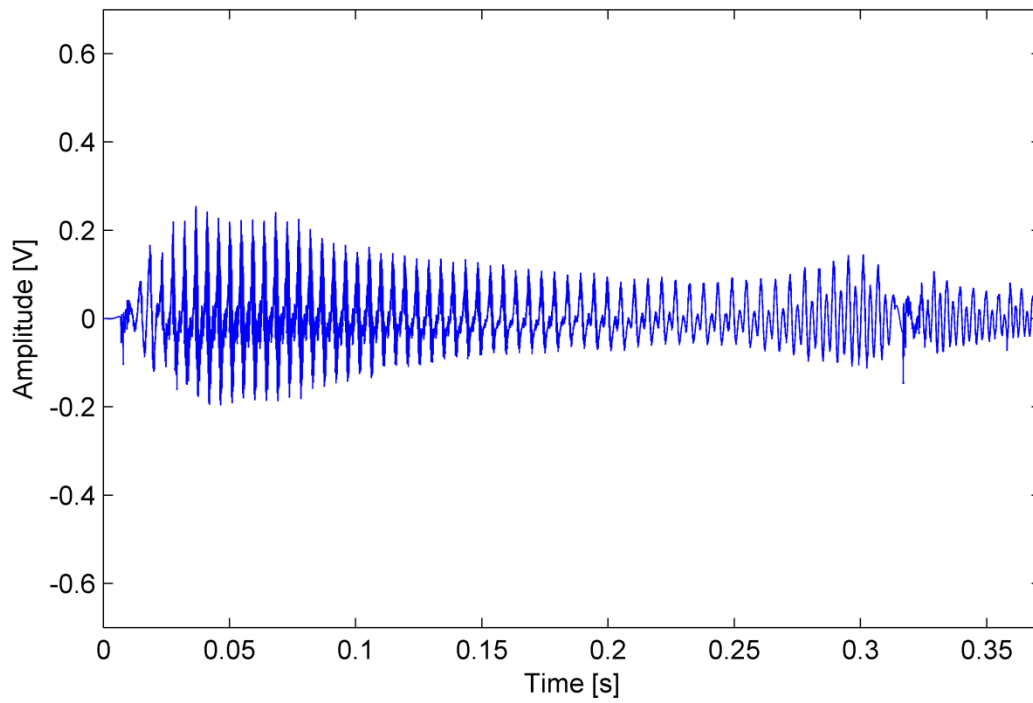


Fig. A.125. The waveform of the utterance "bead" detected by the voice activity detection algorithm.

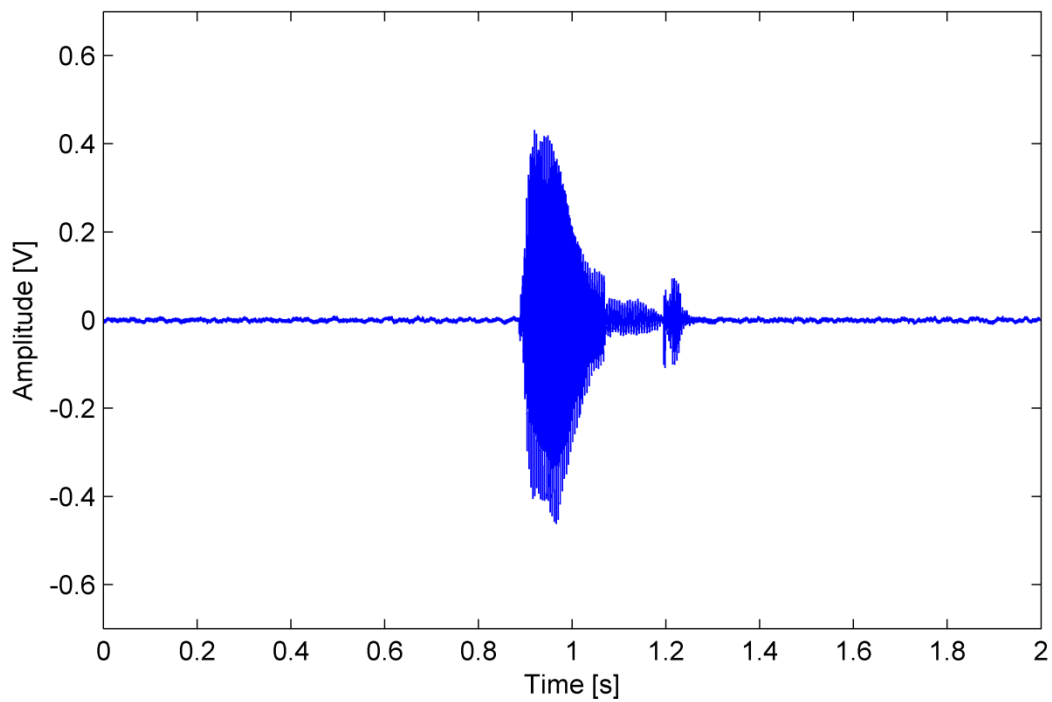


Fig. A.126. The waveform of the utterance "bid".

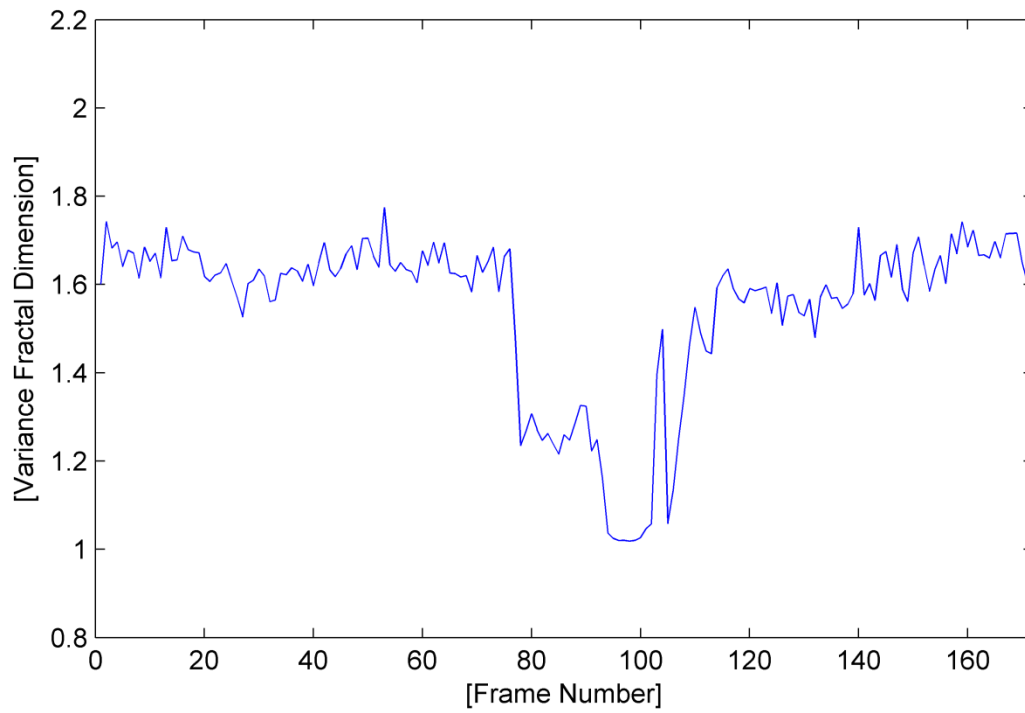


Fig. A.127. The variance fractal dimension trajectory of the utterance "bid".

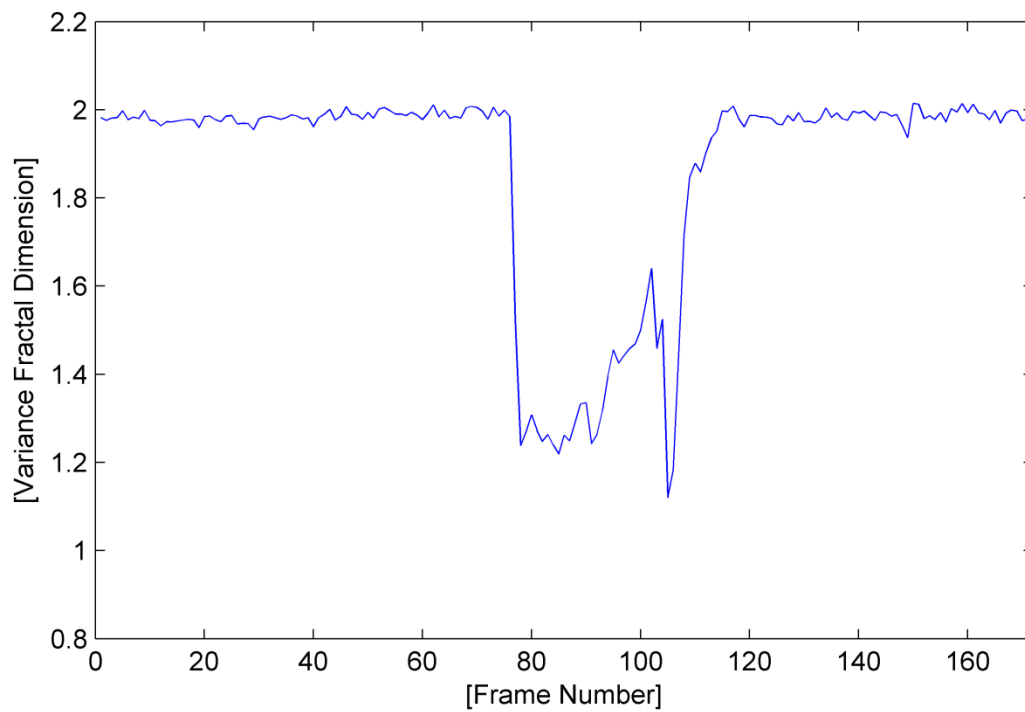


Fig. A.128. The variance fractal dimension trajectory of the utterance "bid" after addition of white noise.

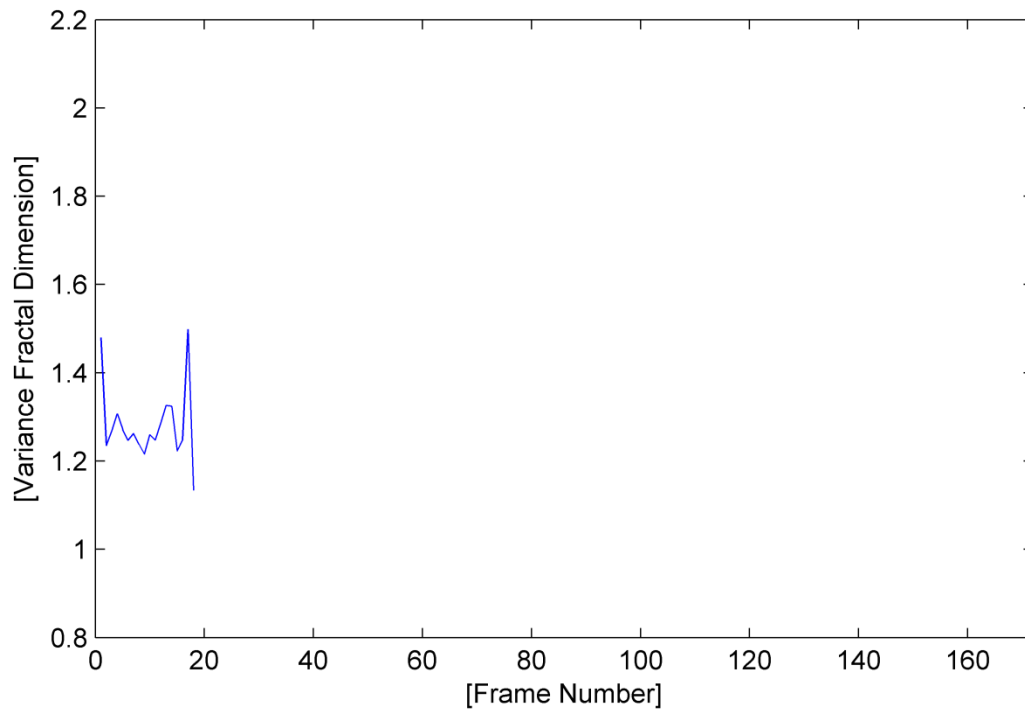


Fig. A.129. The trajectory of the utterance “bid” detected by the voice activity detection algorithm.

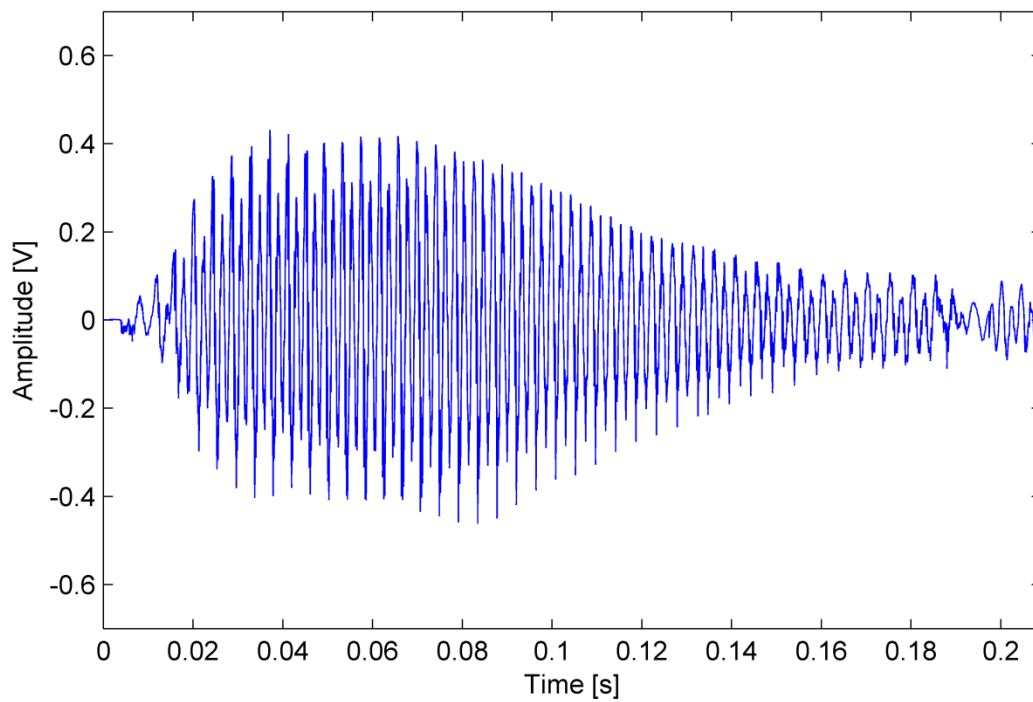


Fig. A.130. The waveform of the utterance “bid” detected by the voice activity detection algorithm.

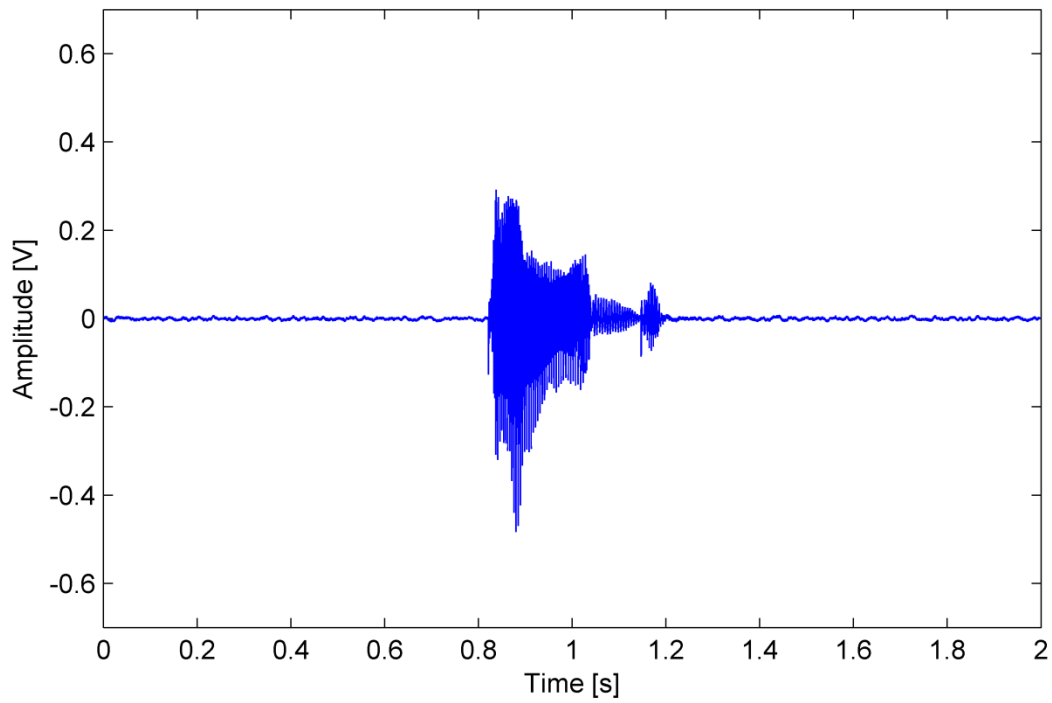


Fig. A.131. The waveform of the utterance "bed".

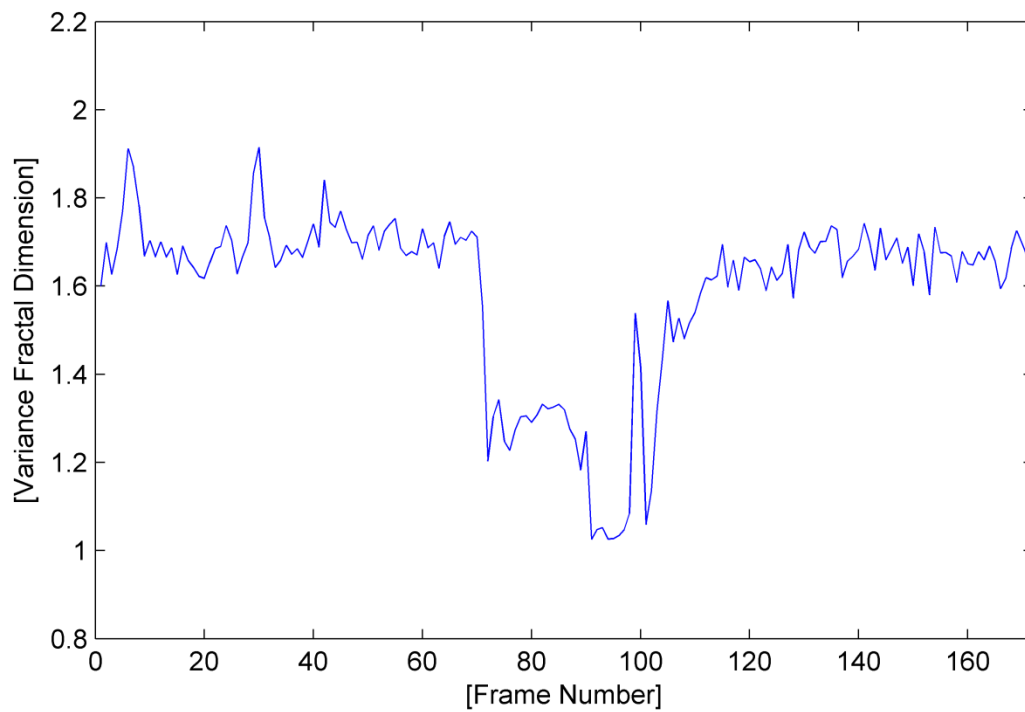


Fig. A.132. The variance fractal dimension trajectory of the utterance "bed".

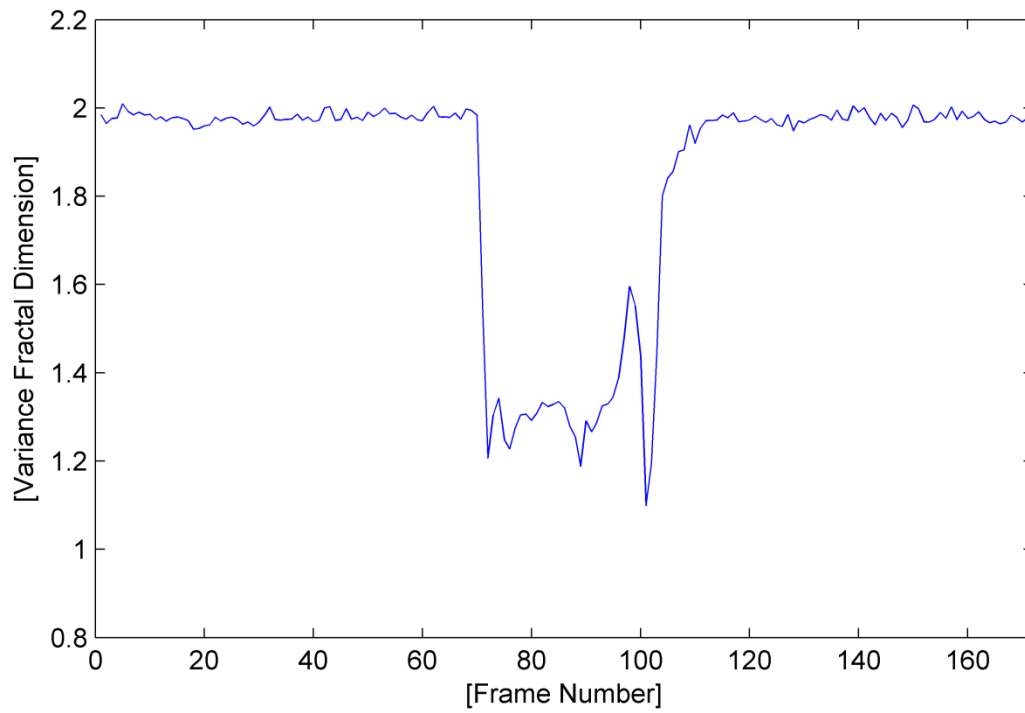


Fig. A.133. The variance fractal dimension trajectory of the utterance "bed" after addition of white noise.

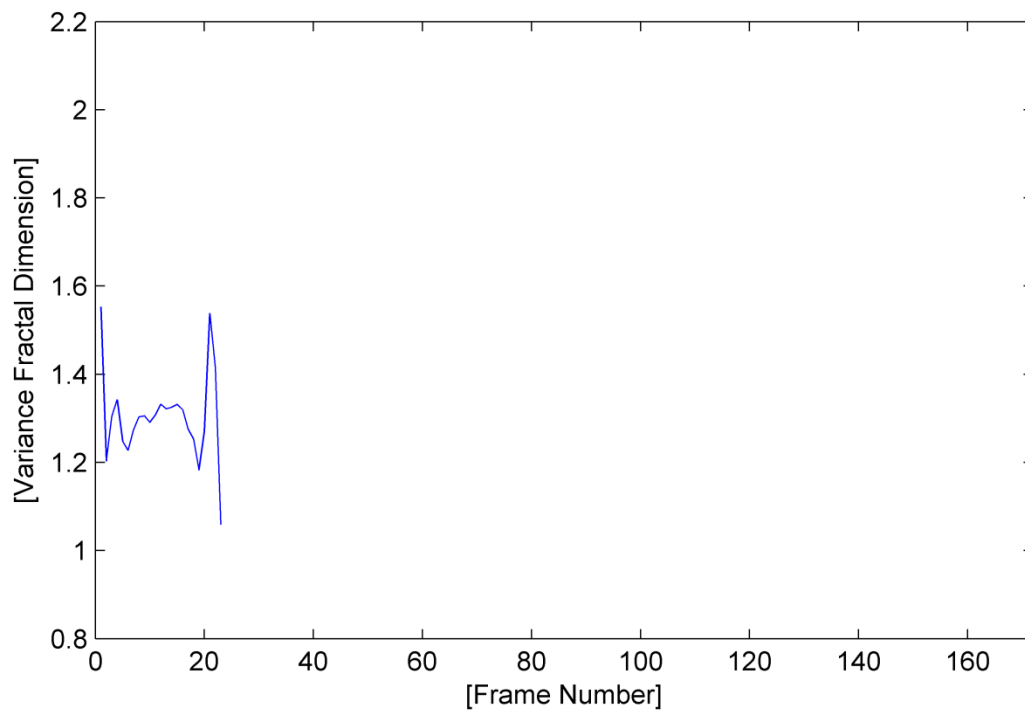


Fig. A.134. The trajectory of the utterance "bed" detected by the voice activity detection algorithm.

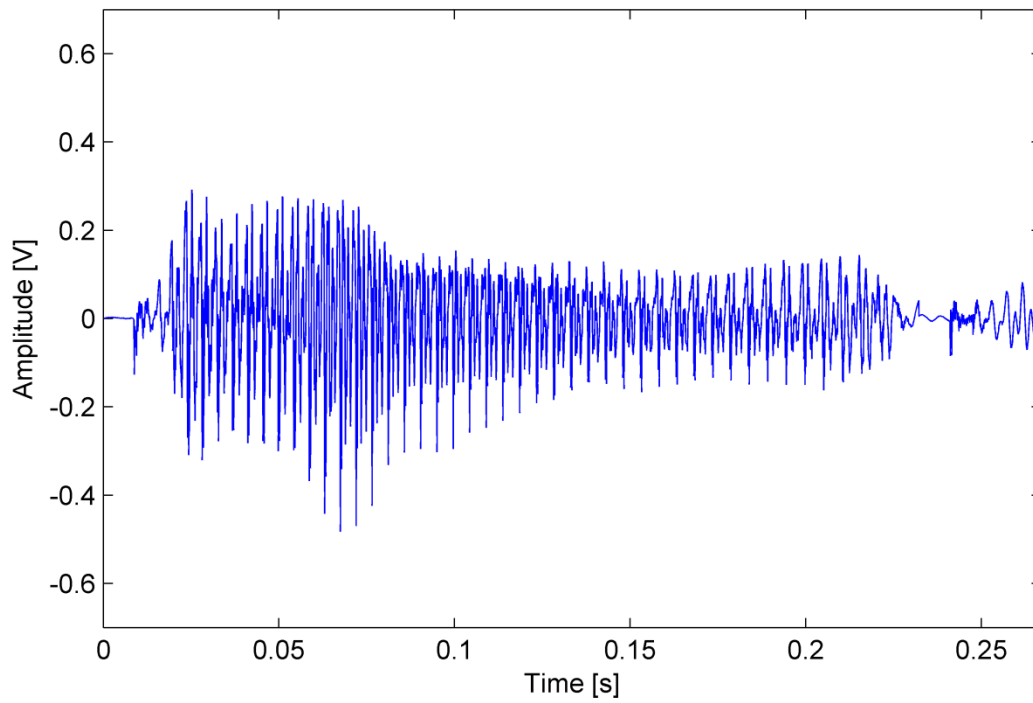


Fig. A.135. The waveform of the utterance “bed” detected by the voice activity detection algorithm.

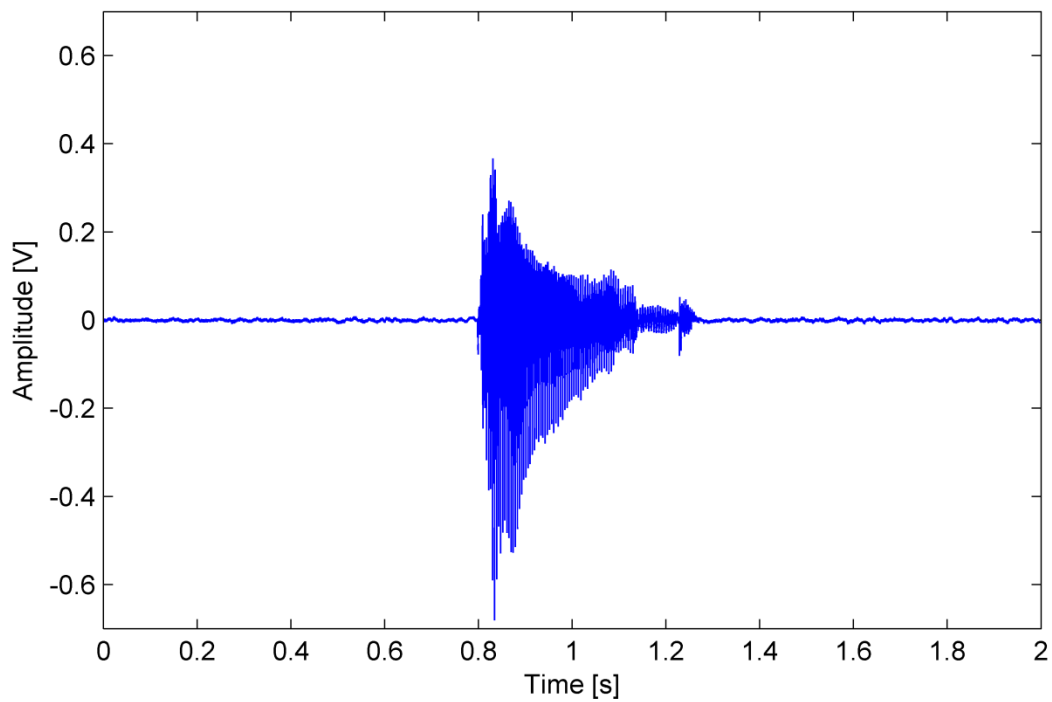


Fig. A.136. The waveform of the utterance “bad”.

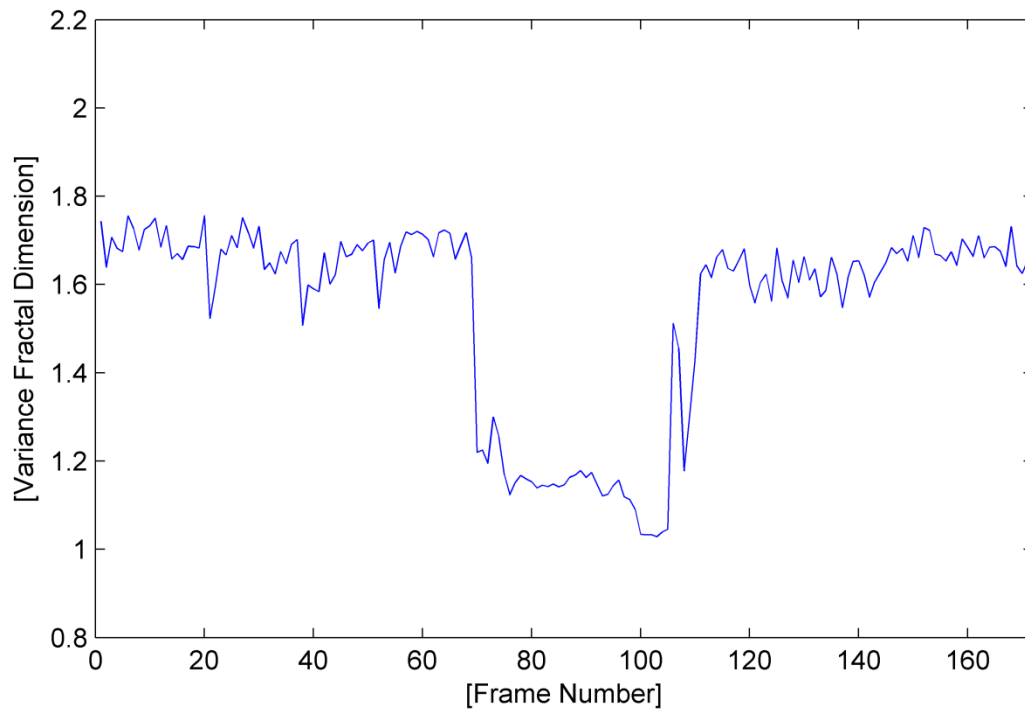


Fig. A.137. The variance fractal dimension trajectory of the utterance "bad".

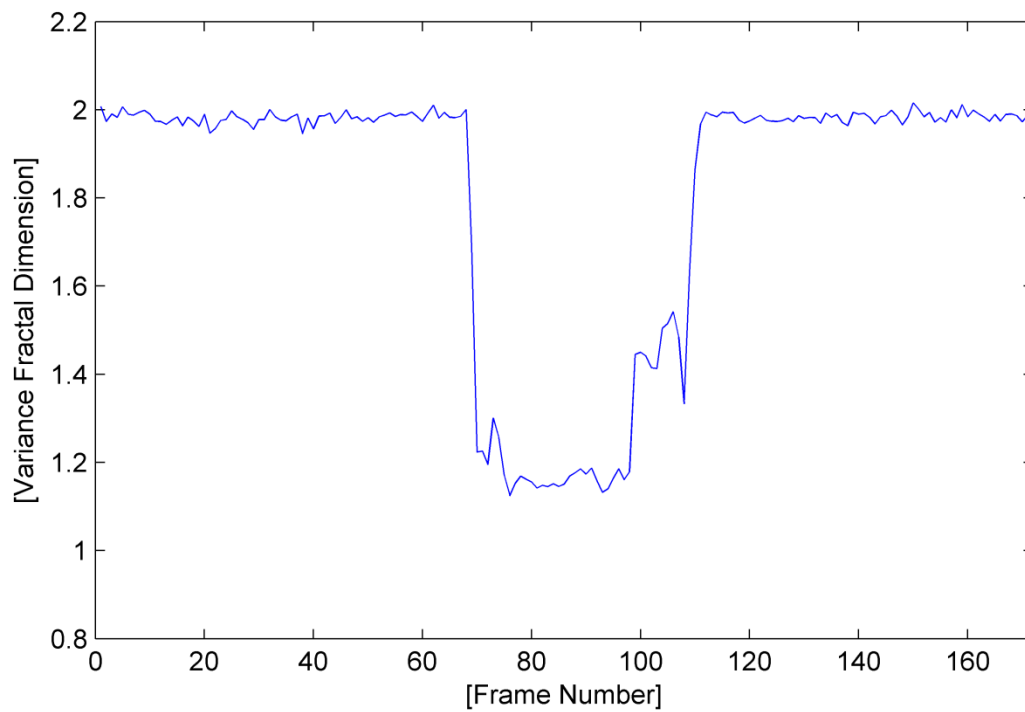


Fig. A.138. The variance fractal dimension trajectory of the utterance "bad" after addition of white noise.

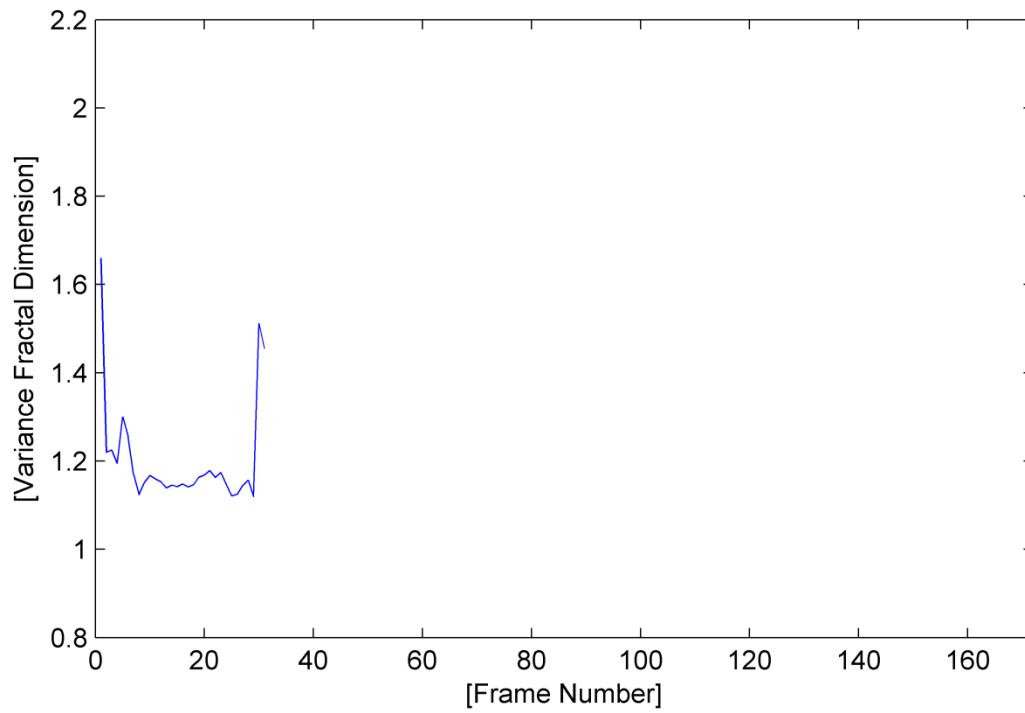


Fig. A.139. The trajectory of the utterance “bad” detected by the voice activity detection algorithm.

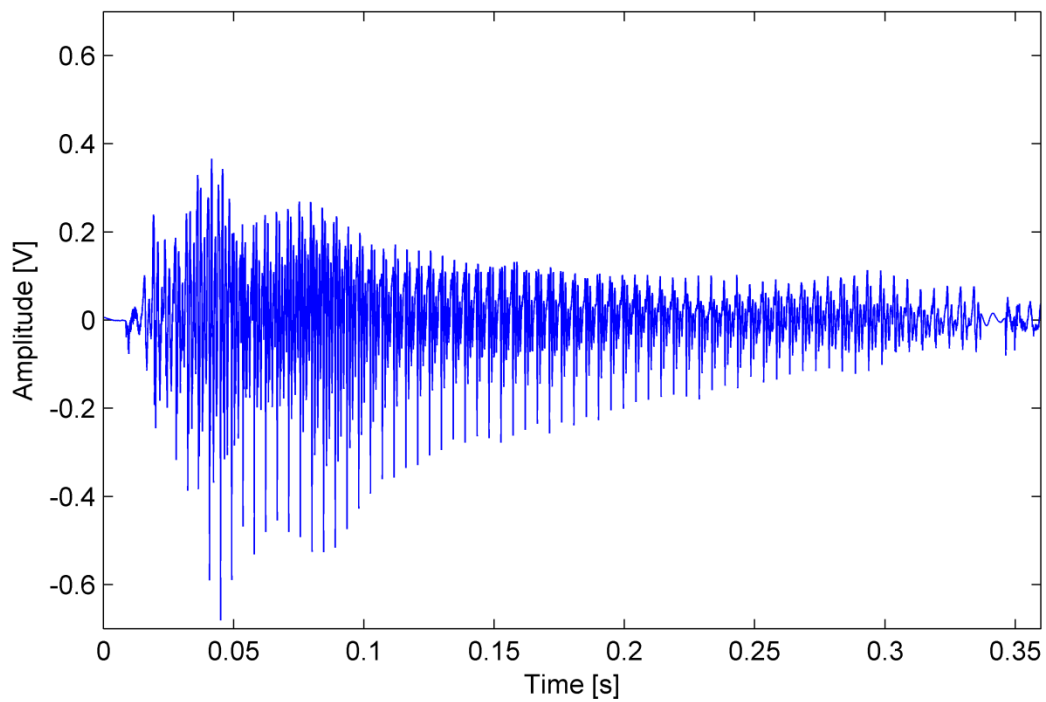


Fig. A.140. The waveform of the utterance “bad” detected by the voice activity detection algorithm.

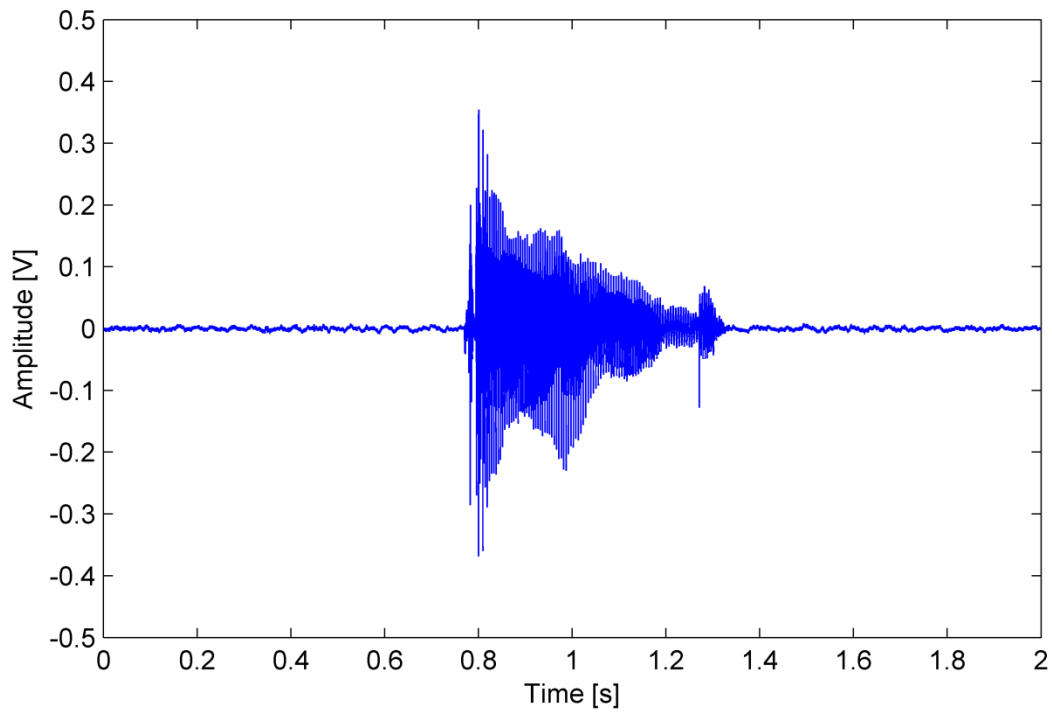


Fig. A.141. The waveform of the utterance "bard".

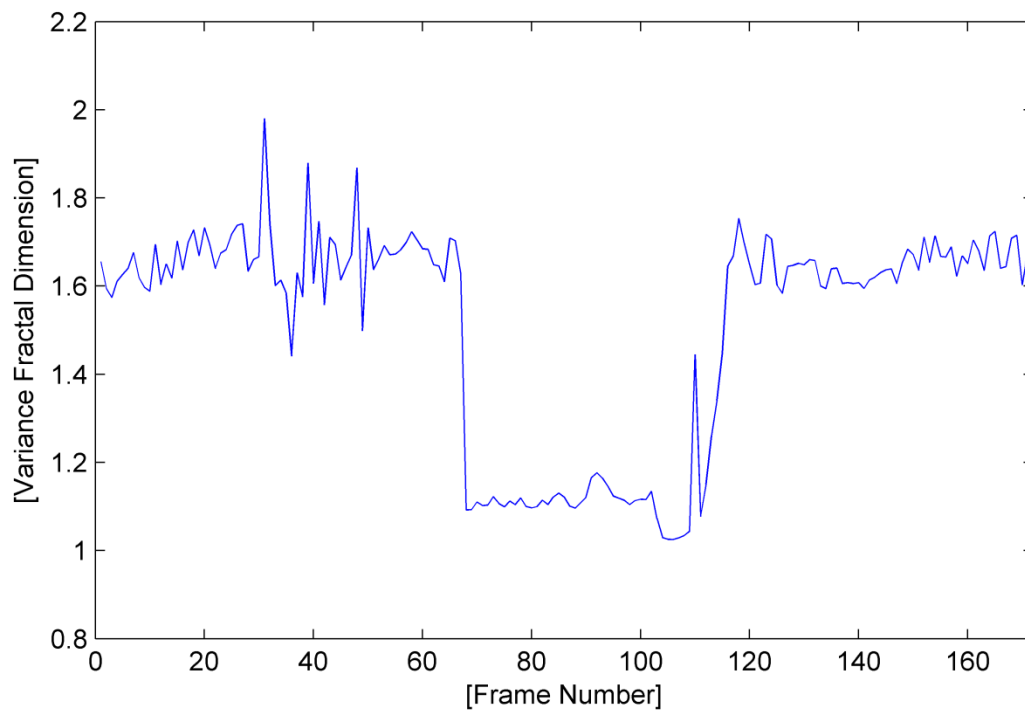


Fig. A.142. The variance fractal dimension trajectory of the utterance "bard".

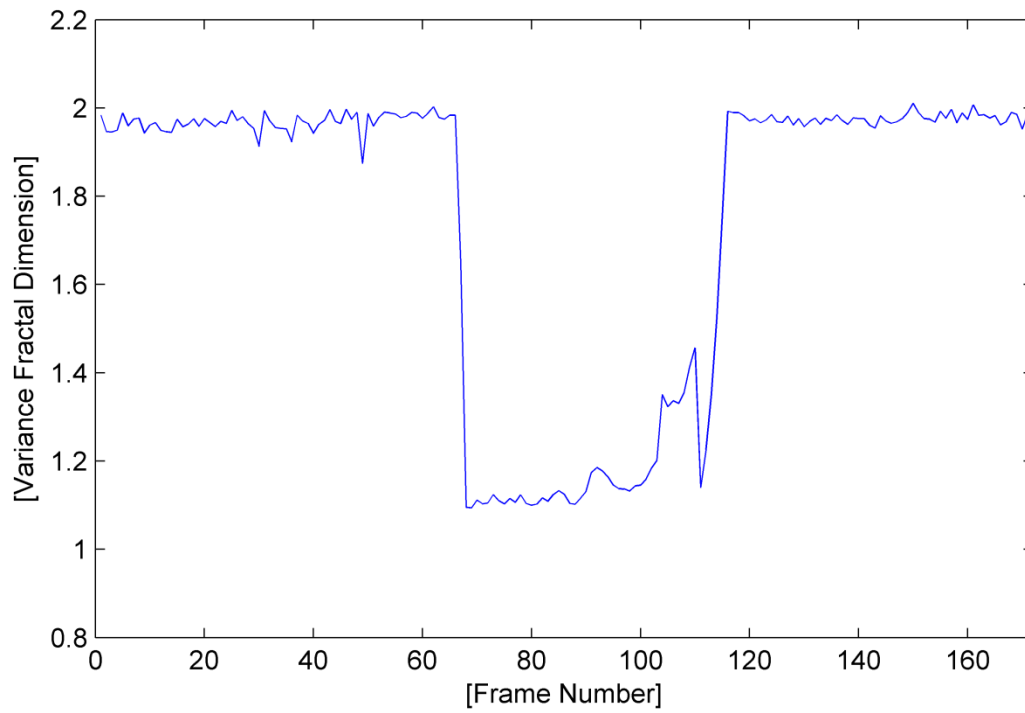


Fig. A.143. The variance fractal dimension trajectory of the utterance “bard” after addition of white noise.

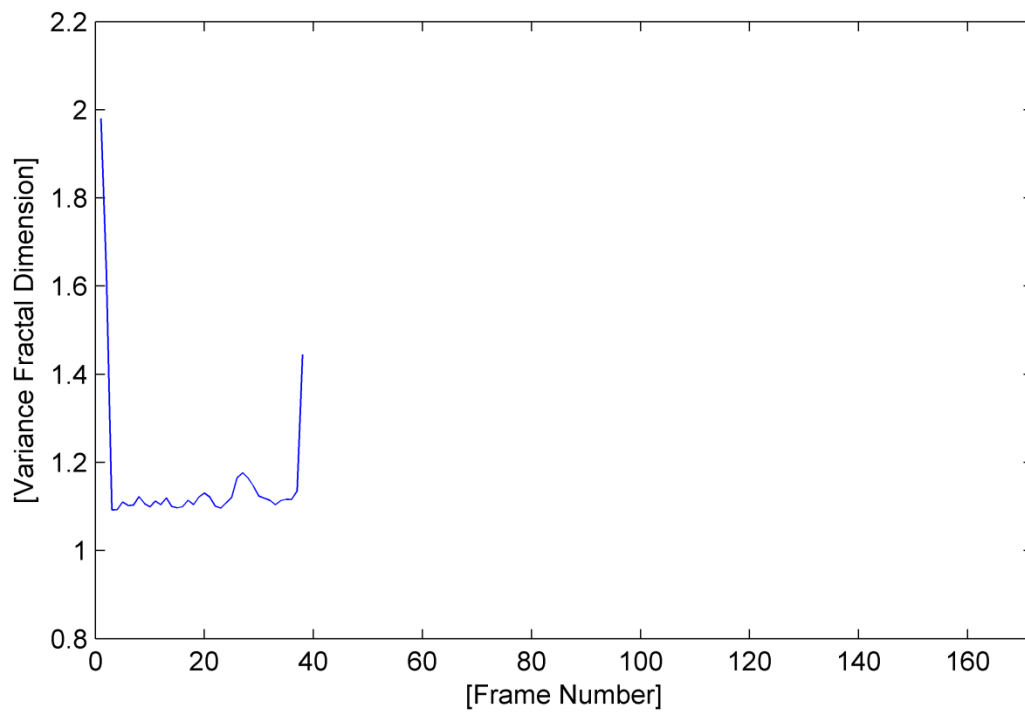


Fig. A.144. The trajectory of the utterance “bard” detected by the voice activity detection algorithm.

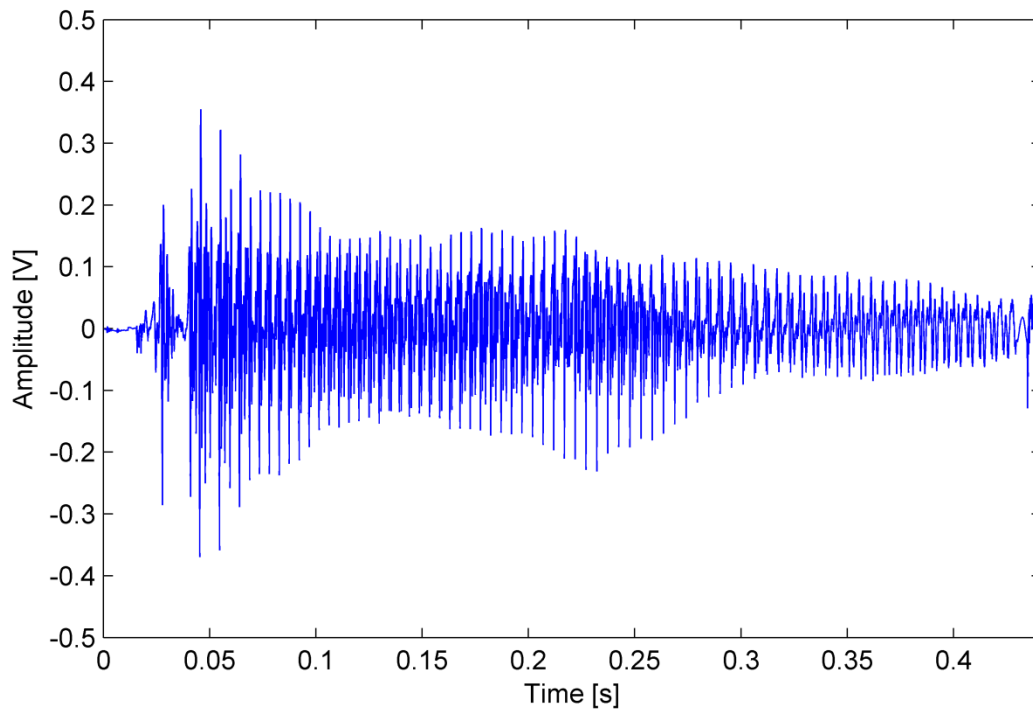


Fig. A.145. The waveform of the utterance “bard” detected by the voice activity detection algorithm.

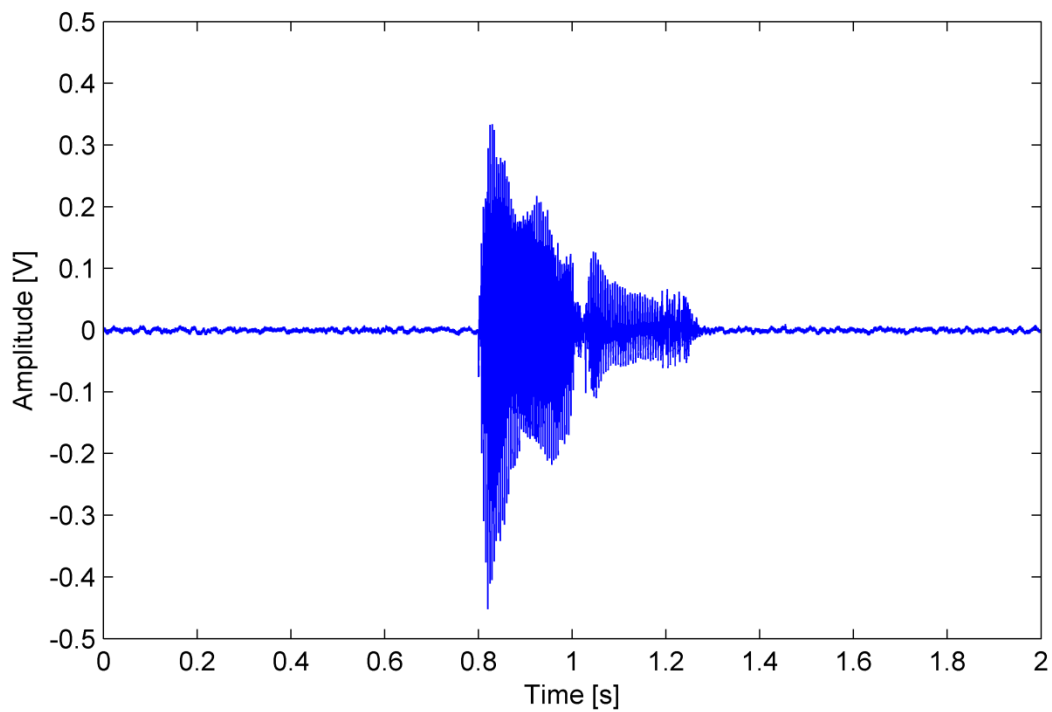


Fig. A.146. The waveform of the utterance “body”.

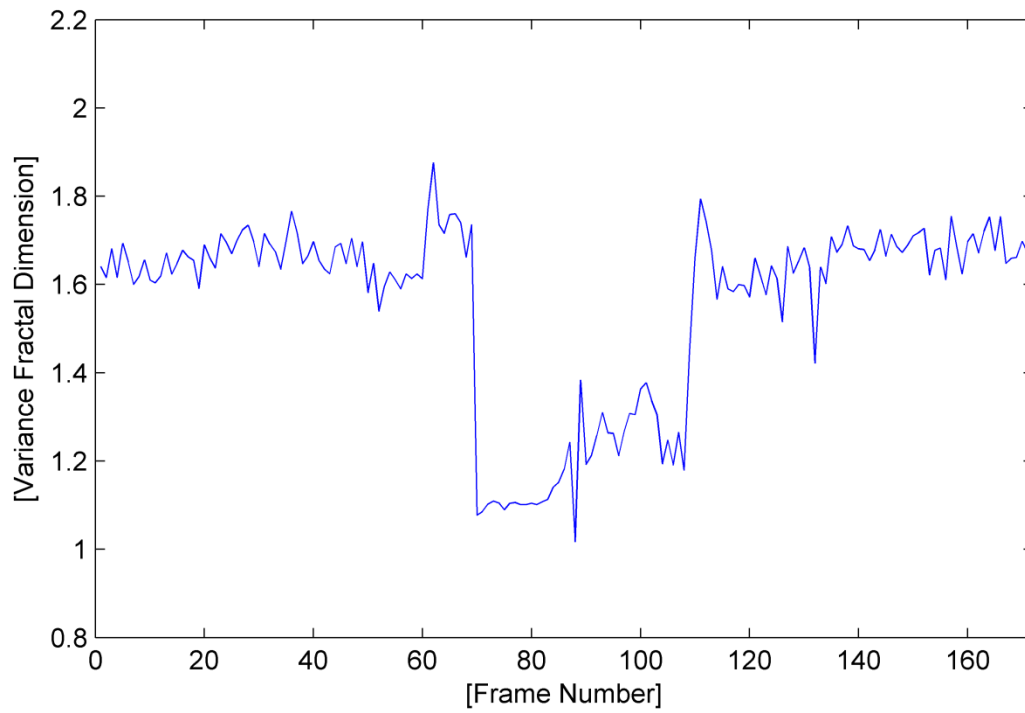


Fig. A.147. The variance fractal dimension trajectory of the utterance "body".

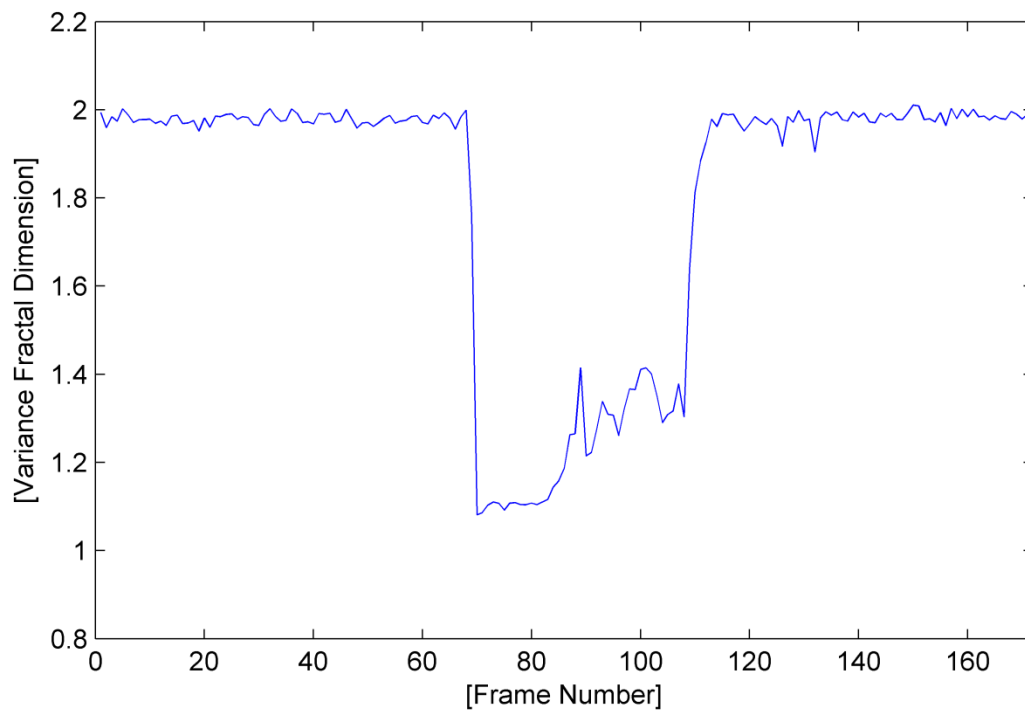


Fig. A.148. The variance fractal dimension trajectory of the utterance "body" after addition of white noise.

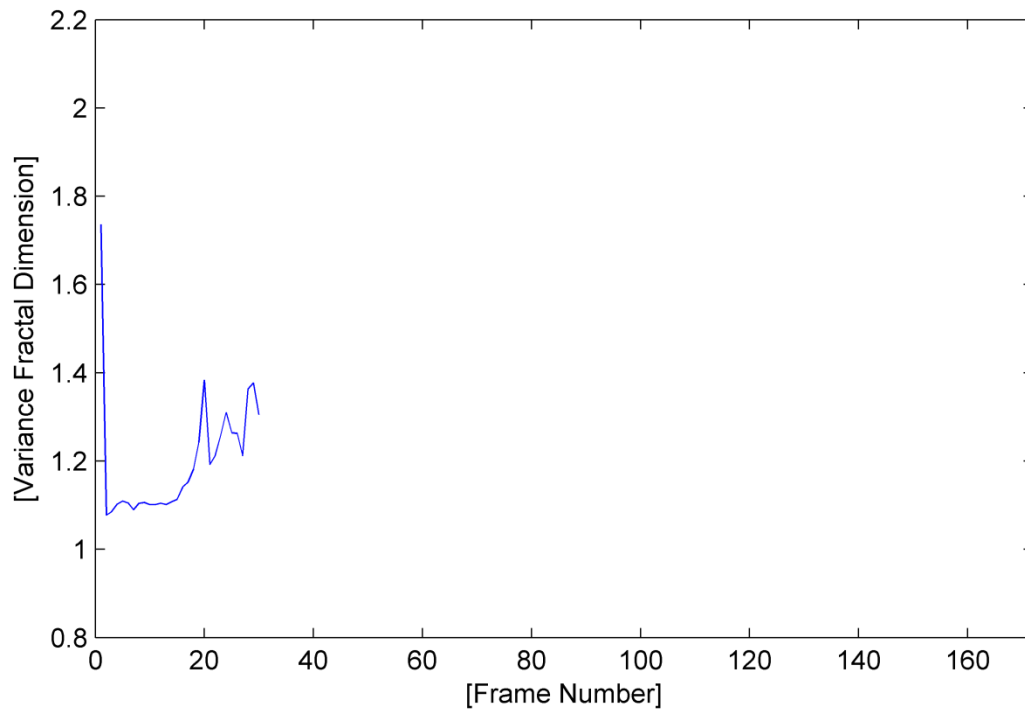


Fig. A.149. The trajectory of the utterance “body” detected by the voice activity detection algorithm.

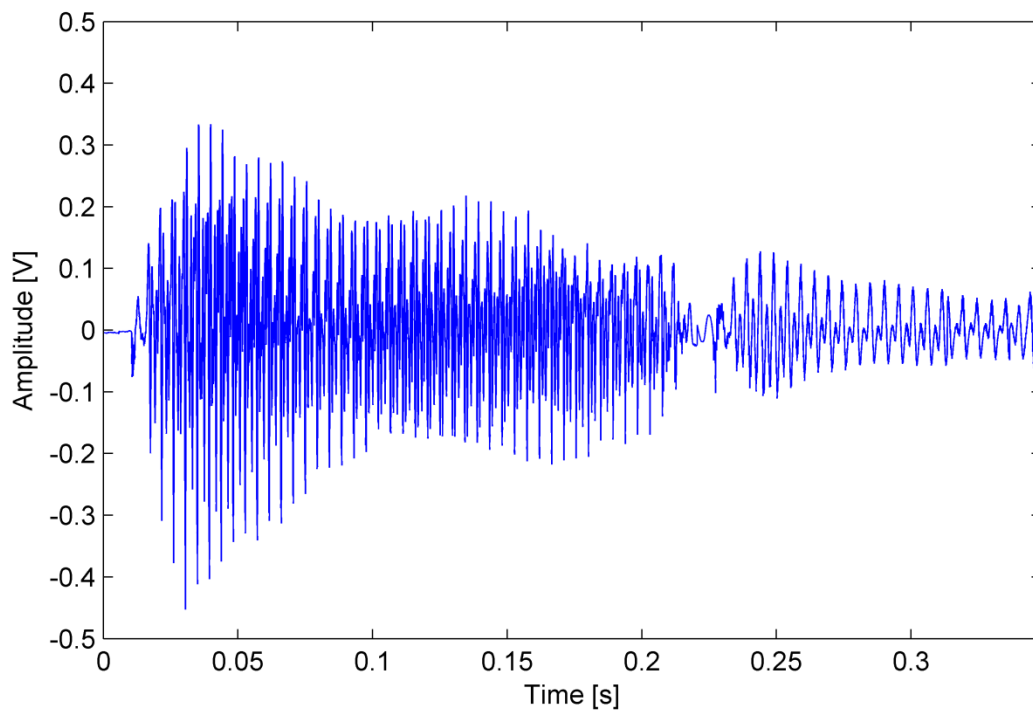


Fig. A.150. The waveform of the utterance “body” detected by the voice activity detection algorithm.

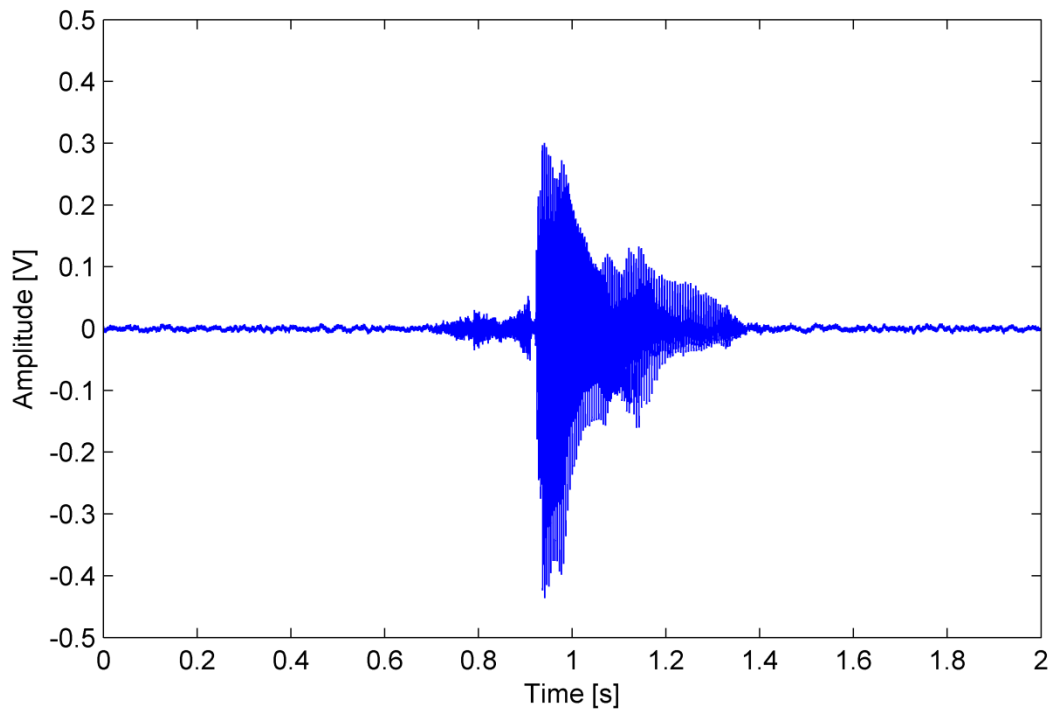


Fig. A.151. The waveform of the utterance "for".

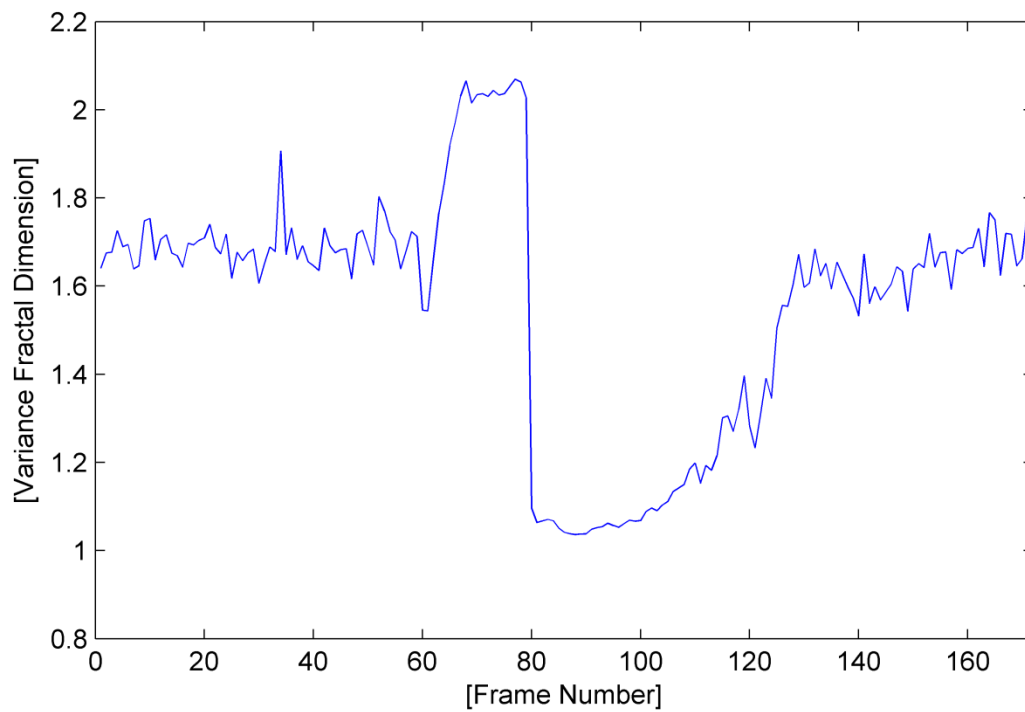


Fig. A.152. The variance fractal dimension trajectory of the utterance "for".

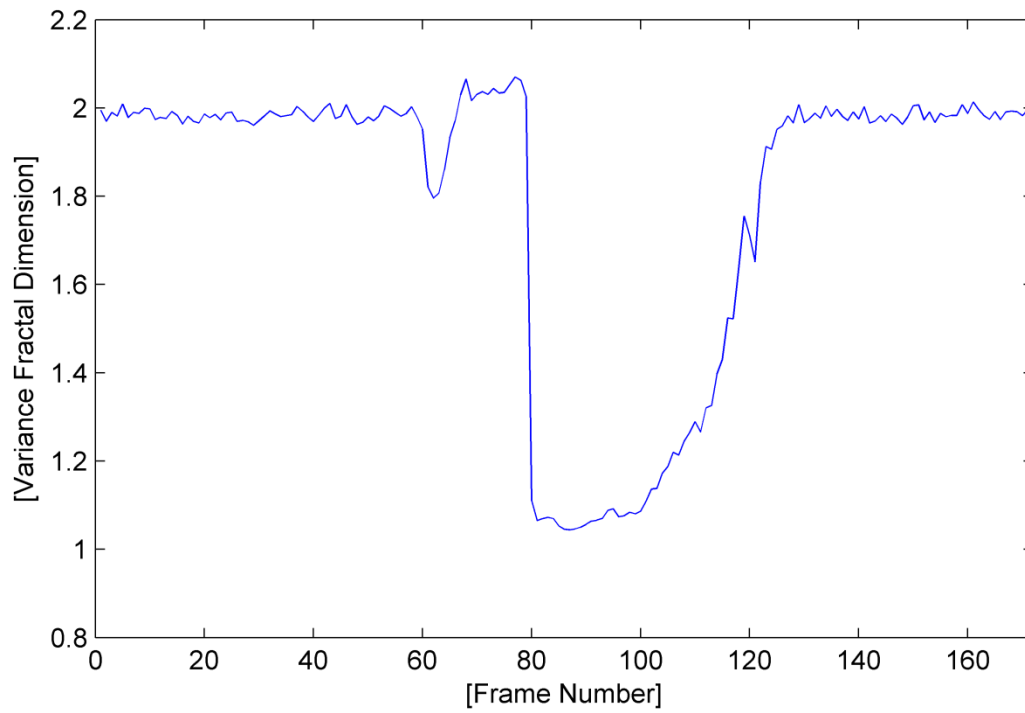


Fig. A.153. The variance fractal dimension trajectory of the utterance “for” after addition of white noise.

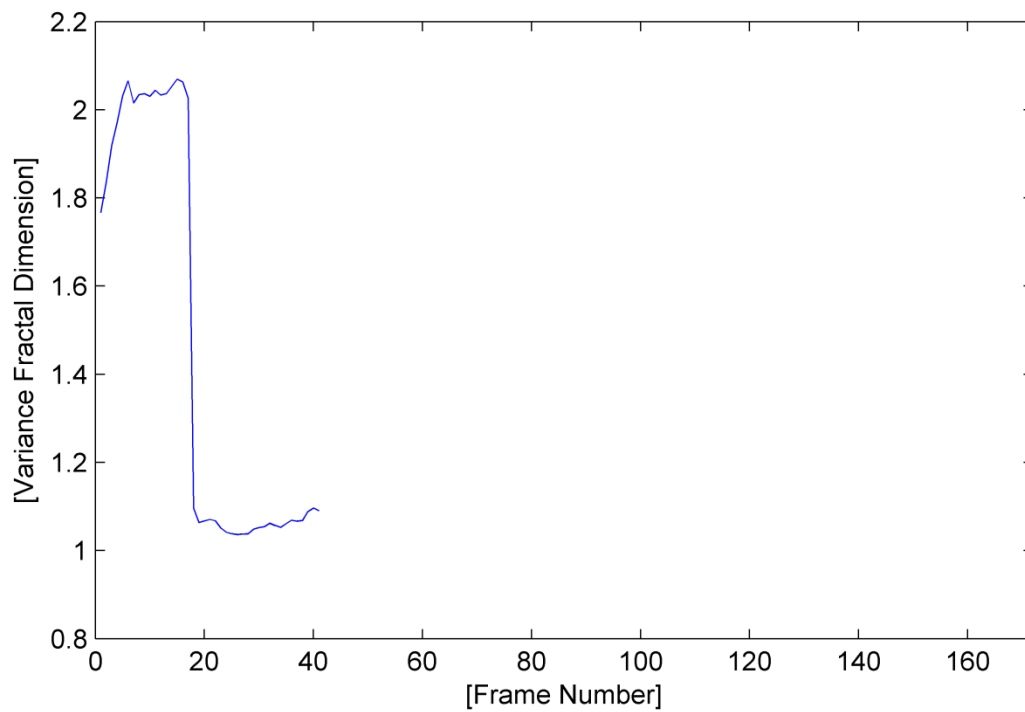


Fig. A.154. The trajectory of the utterance “for” detected by the voice activity detection algorithm.

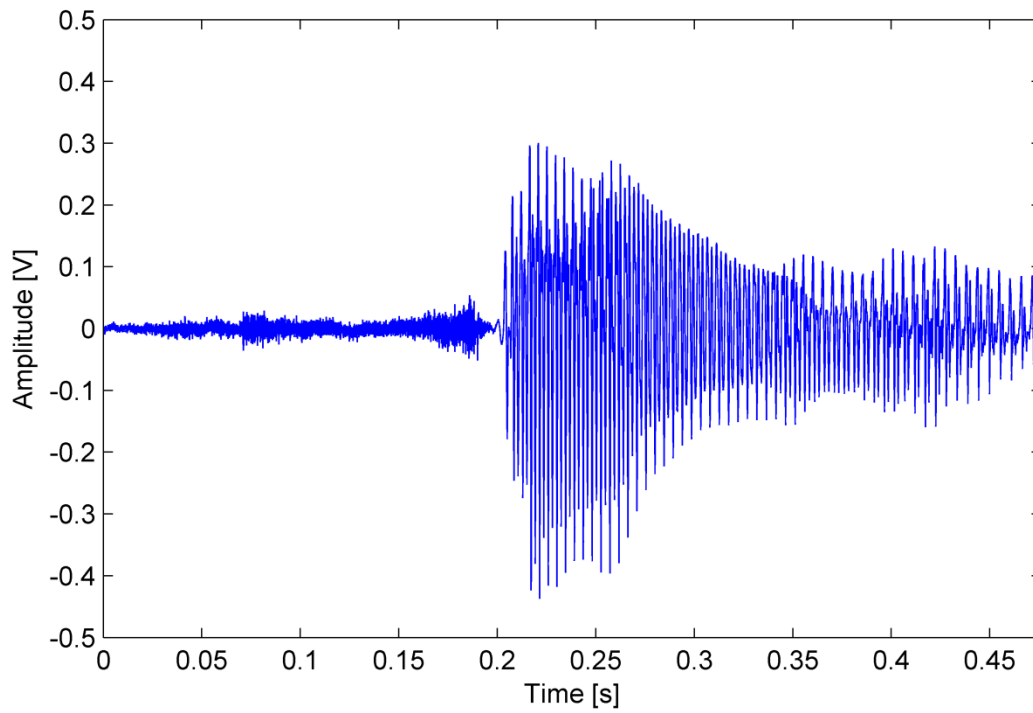


Fig. A.155. The waveform of the utterance “for” detected by the voice activity detection algorithm.

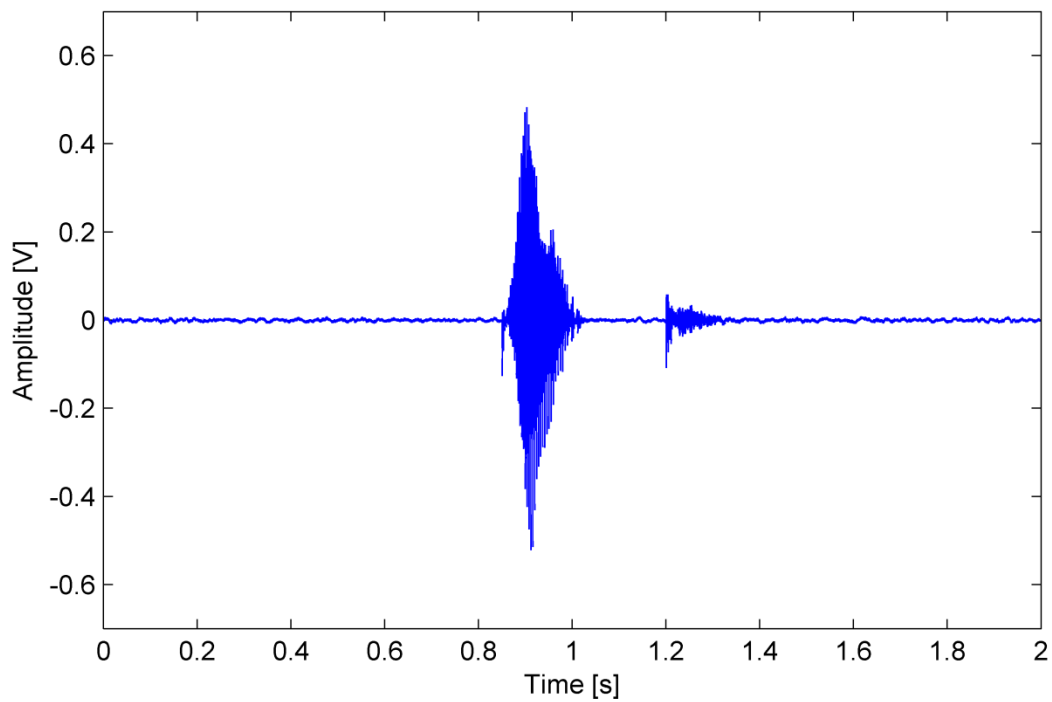


Fig. A.156. The waveform of the utterance “book”.

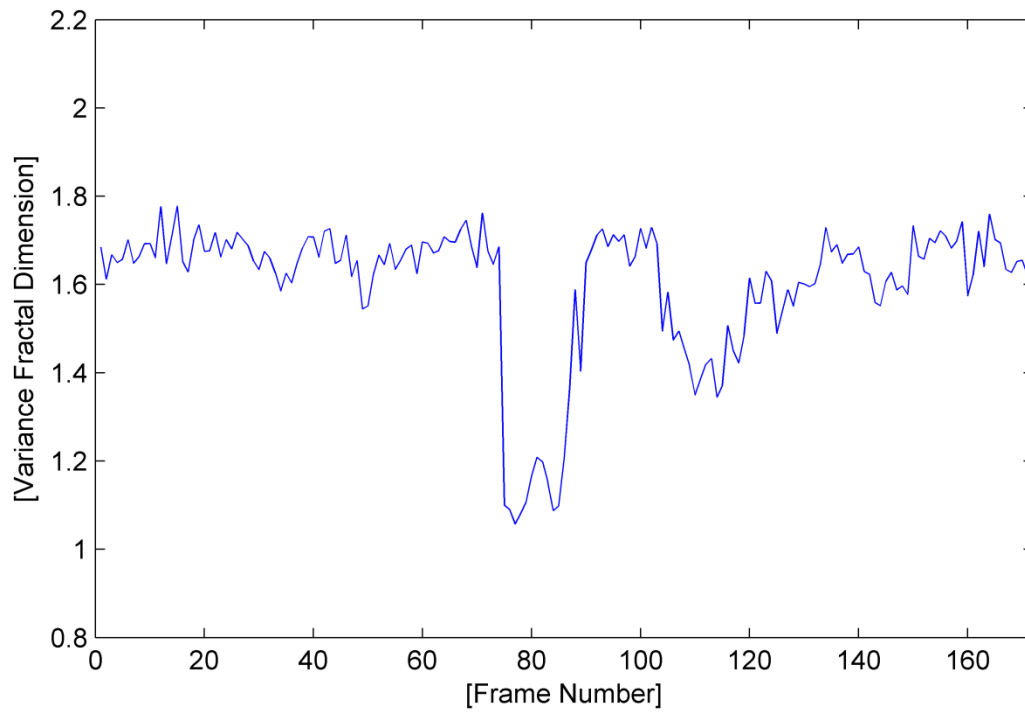


Fig. A.157. The variance fractal dimension trajectory of the utterance "book".

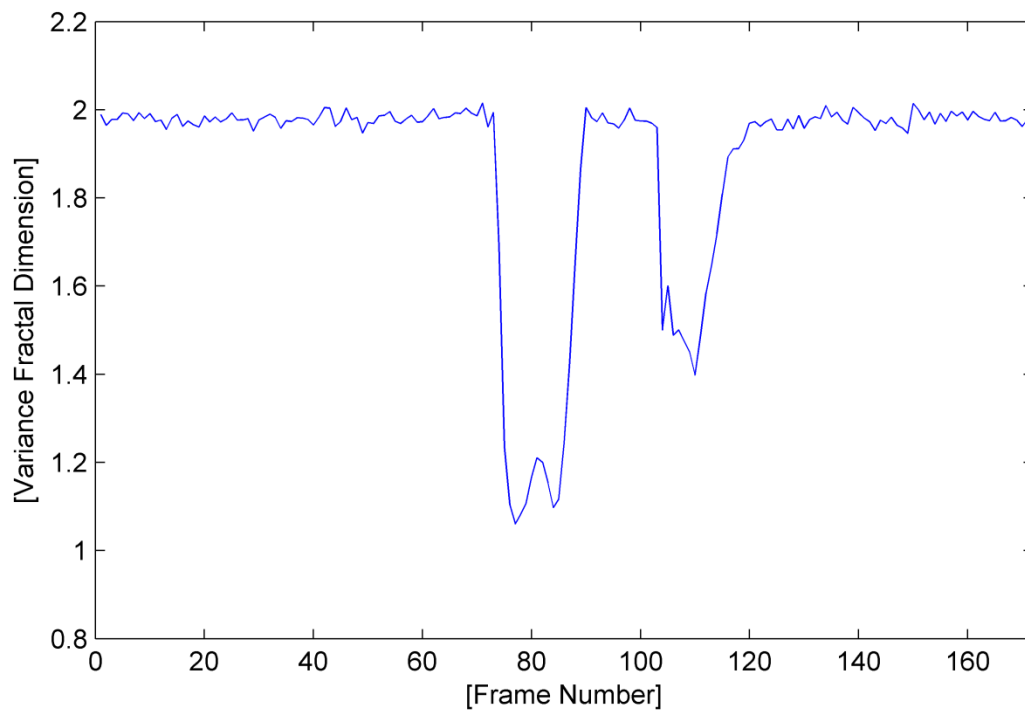


Fig. A.158. The variance fractal dimension trajectory of the utterance "book" after addition of white noise.

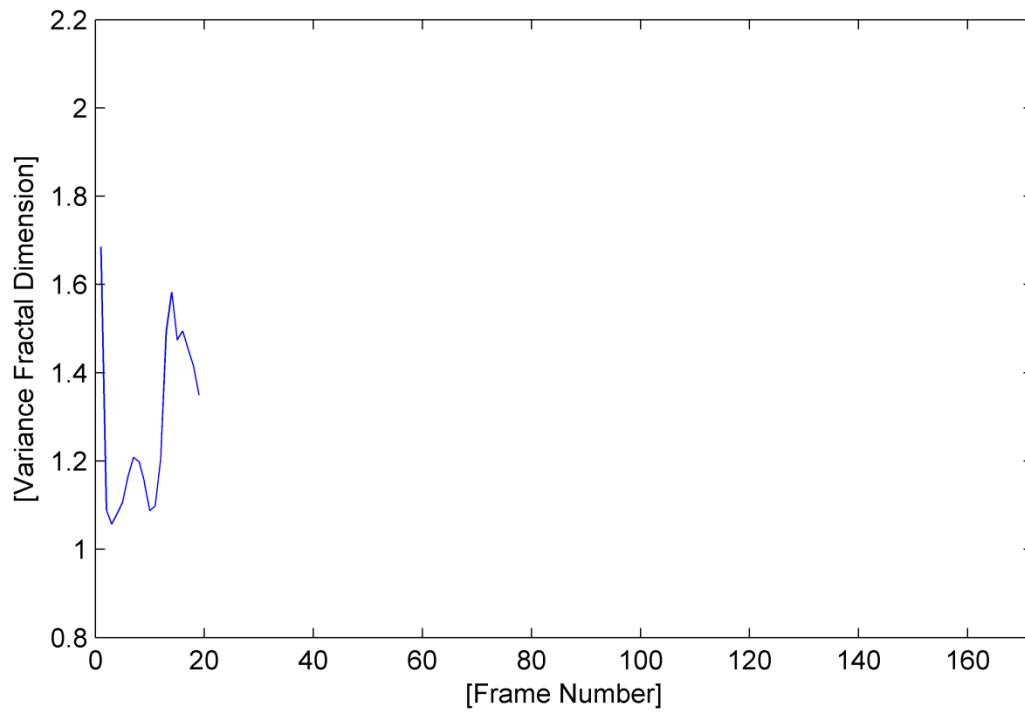


Fig. A.159. The trajectory of the utterance “book” detected by the voice activity detection algorithm.

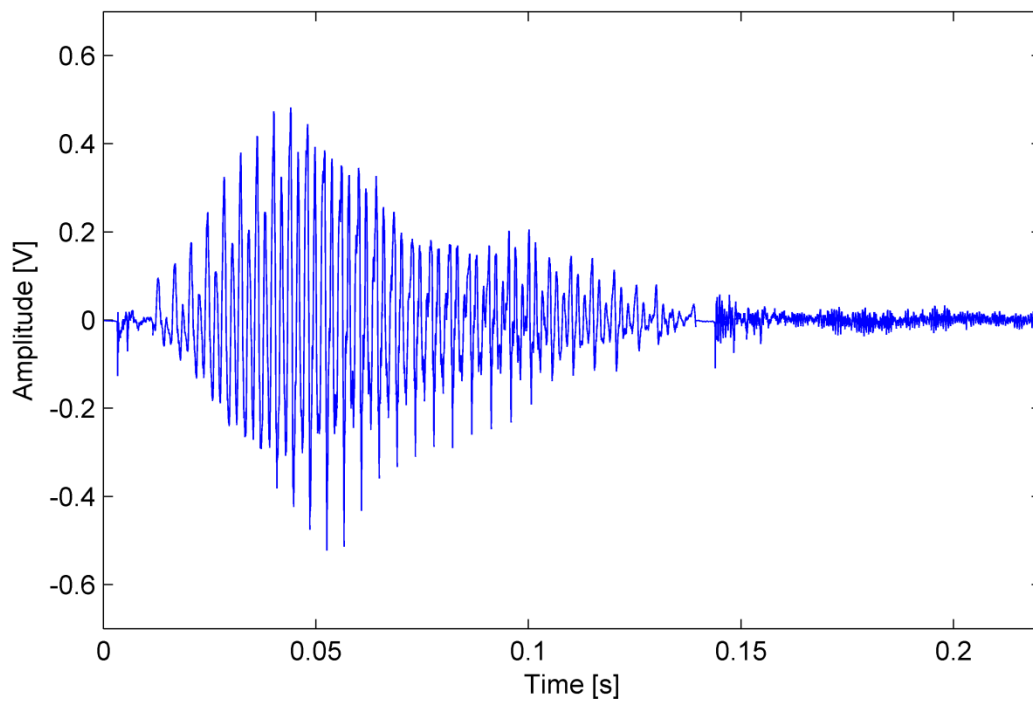


Fig. A.160. The waveform of the utterance “book” detected by the voice activity detection algorithm.

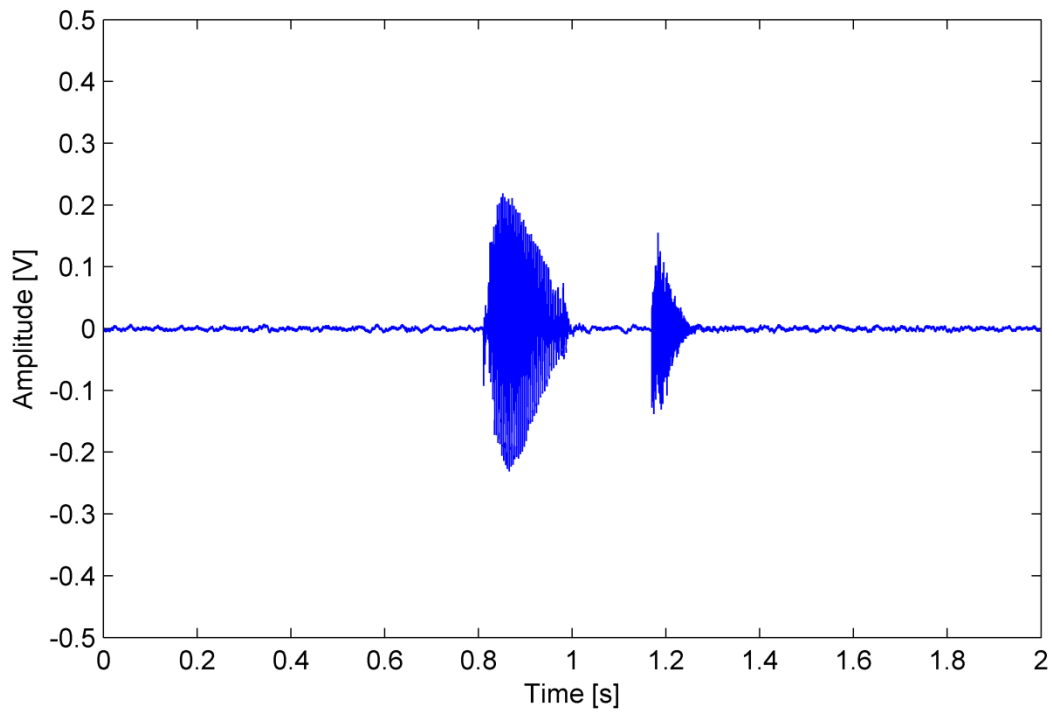


Fig. A.161. The waveform of the utterance "boot".

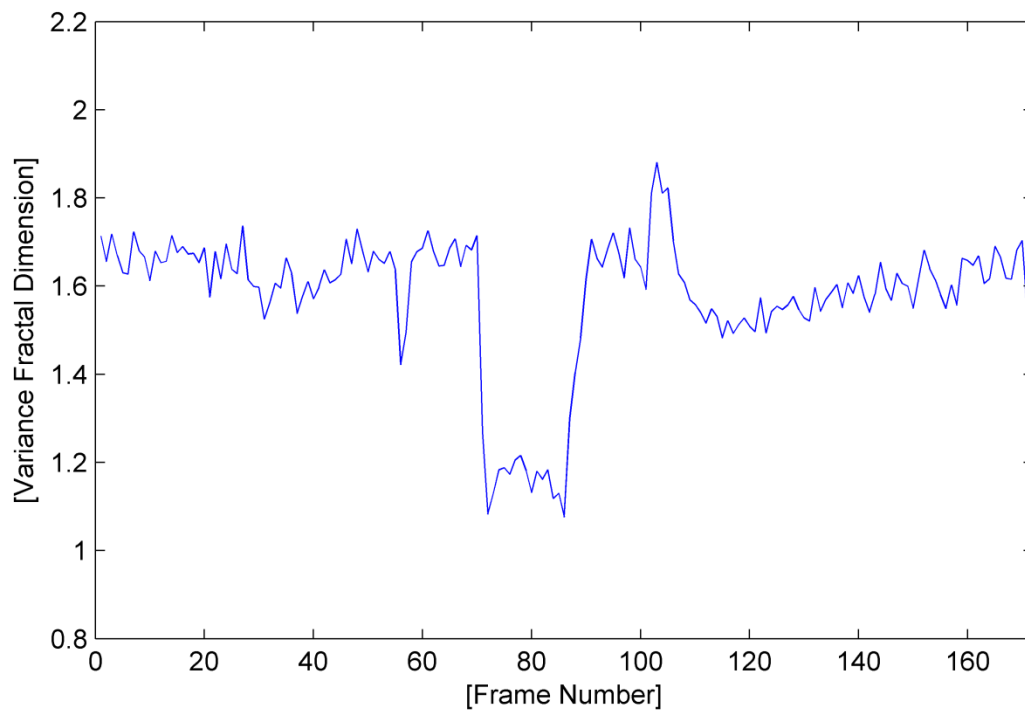


Fig. A.162. The variance fractal dimension trajectory of the utterance "boot".

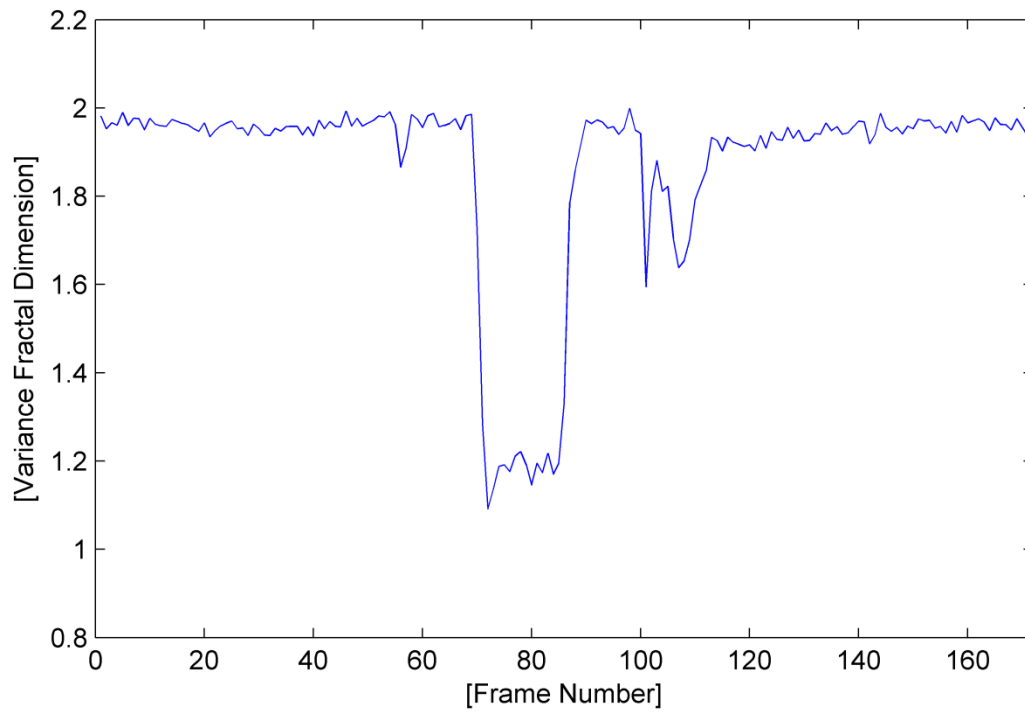


Fig. A.163. The variance fractal dimension trajectory of the utterance “boot” after addition of white noise.

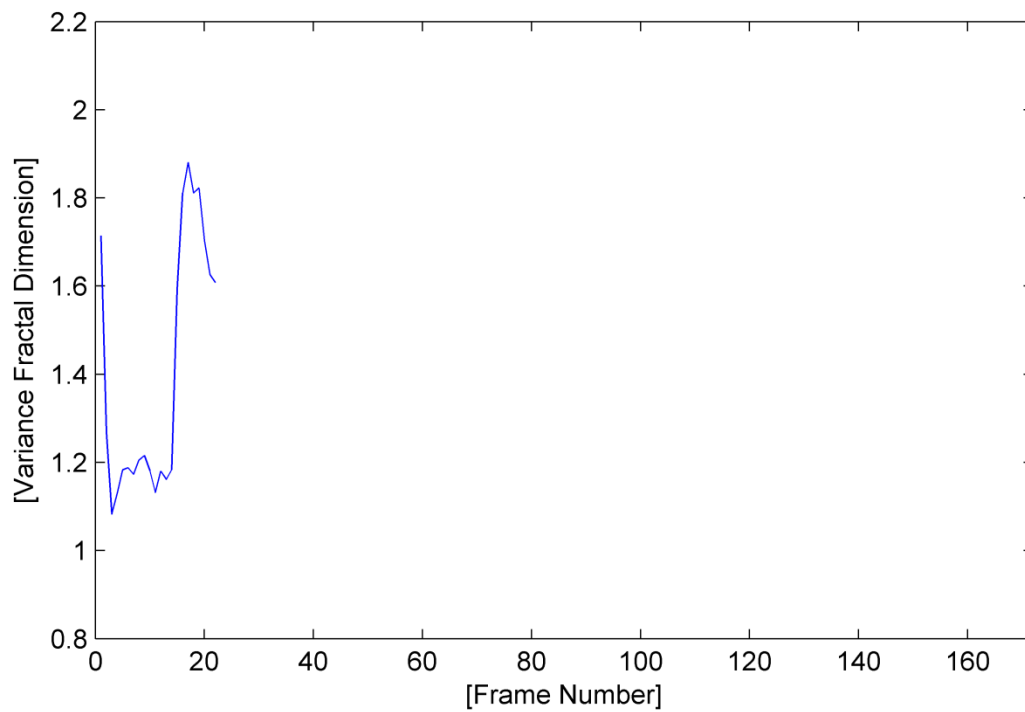


Fig. A.164. The trajectory of the utterance “boot” detected by the voice activity detection algorithm.

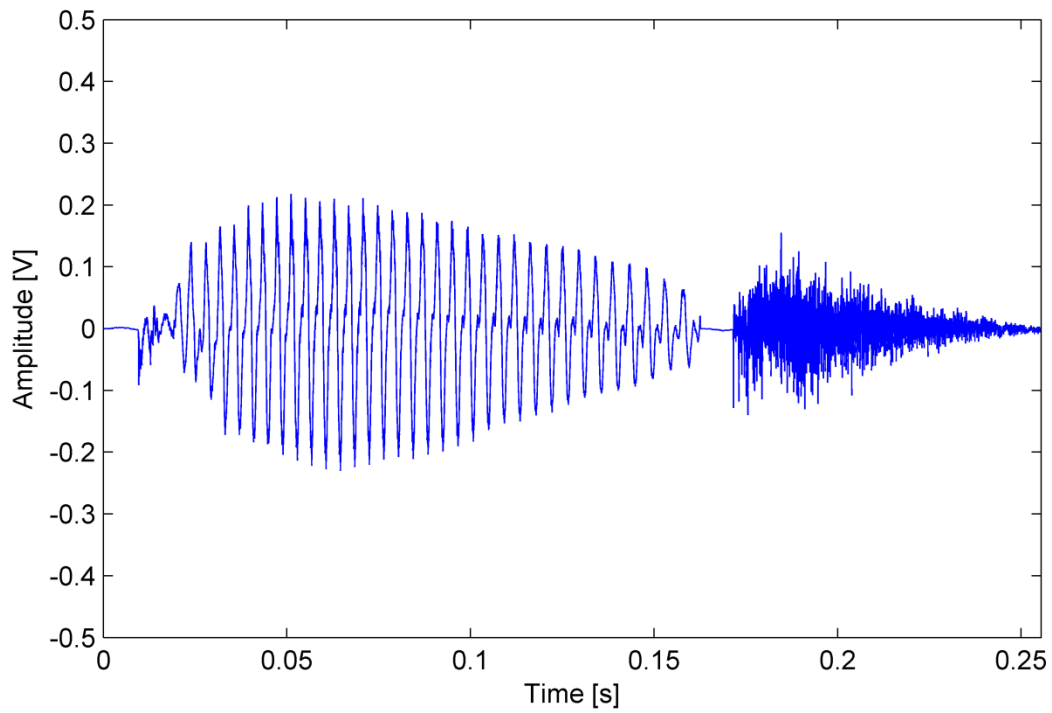


Fig. A.165. The waveform of the utterance "boot" detected by the voice activity detection algorithm.

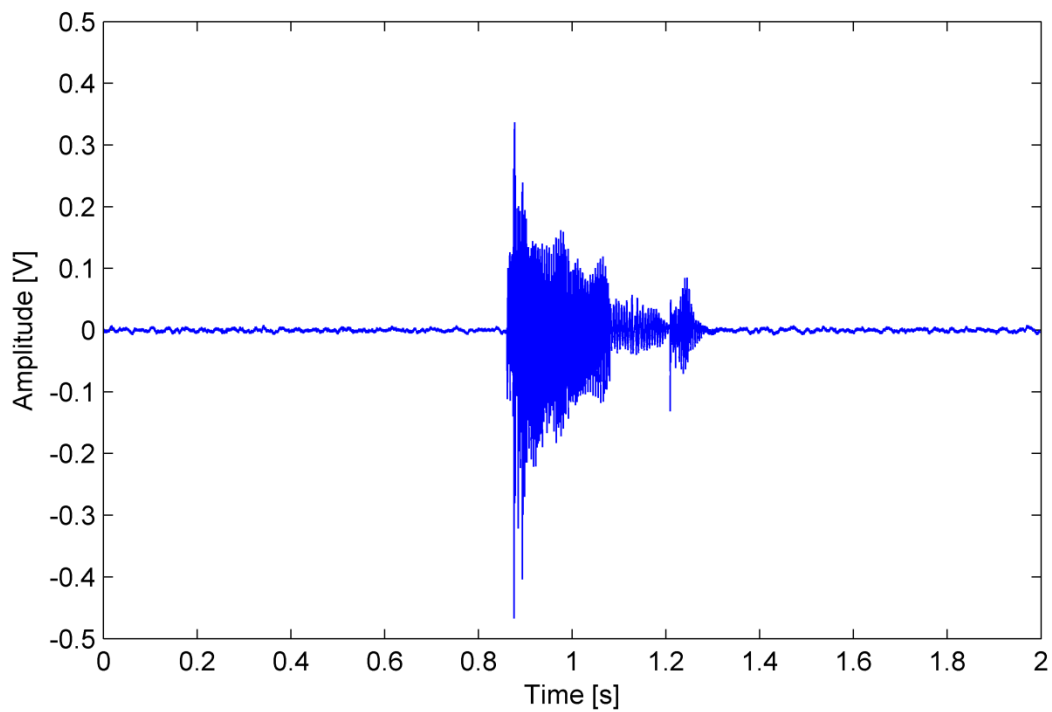


Fig. A.166. The waveform of the utterance "bud".

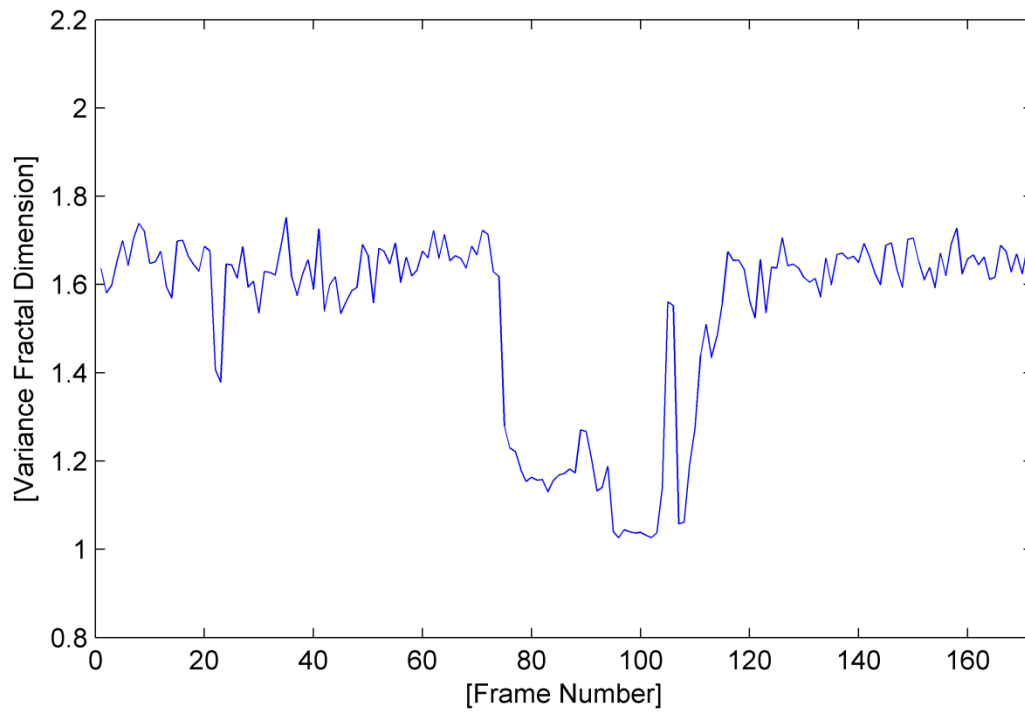


Fig. A.167. The variance fractal dimension trajectory of the utterance "bud".

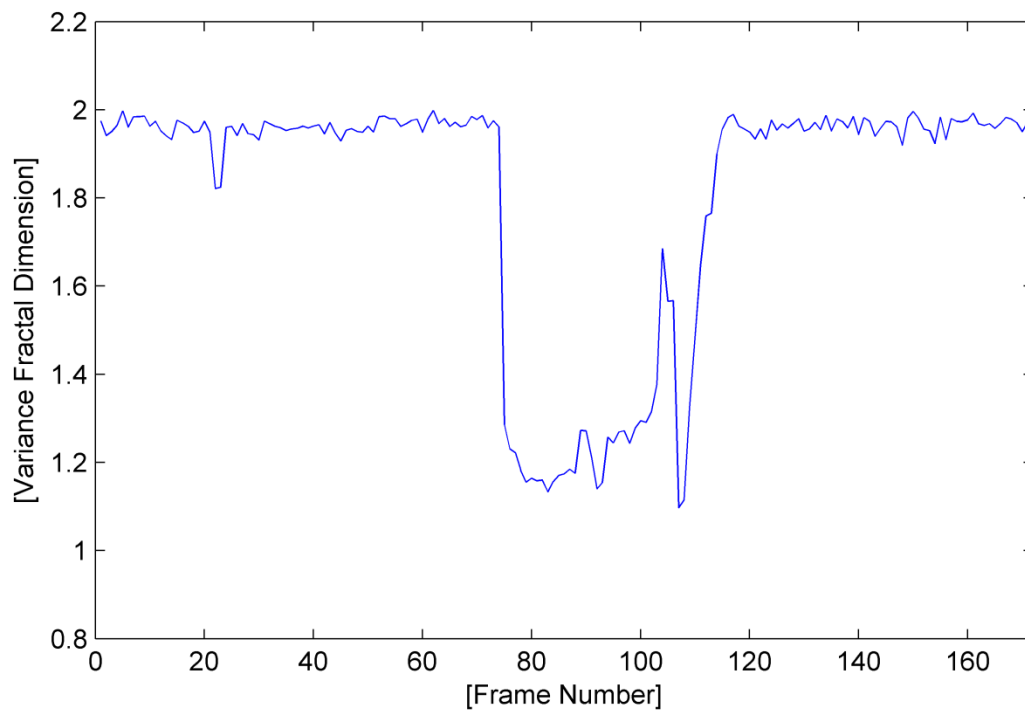


Fig. A.168. The variance fractal dimension trajectory of the utterance "bud" after addition of white noise.

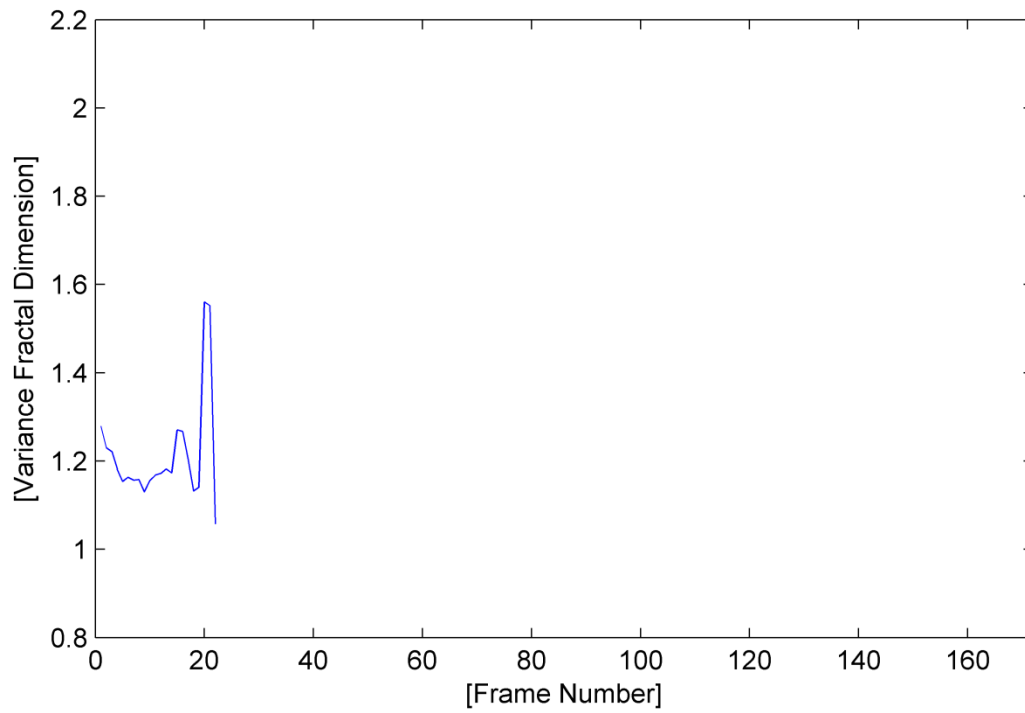


Fig. A.169. The trajectory of the utterance “bud” detected by the voice activity detection algorithm.

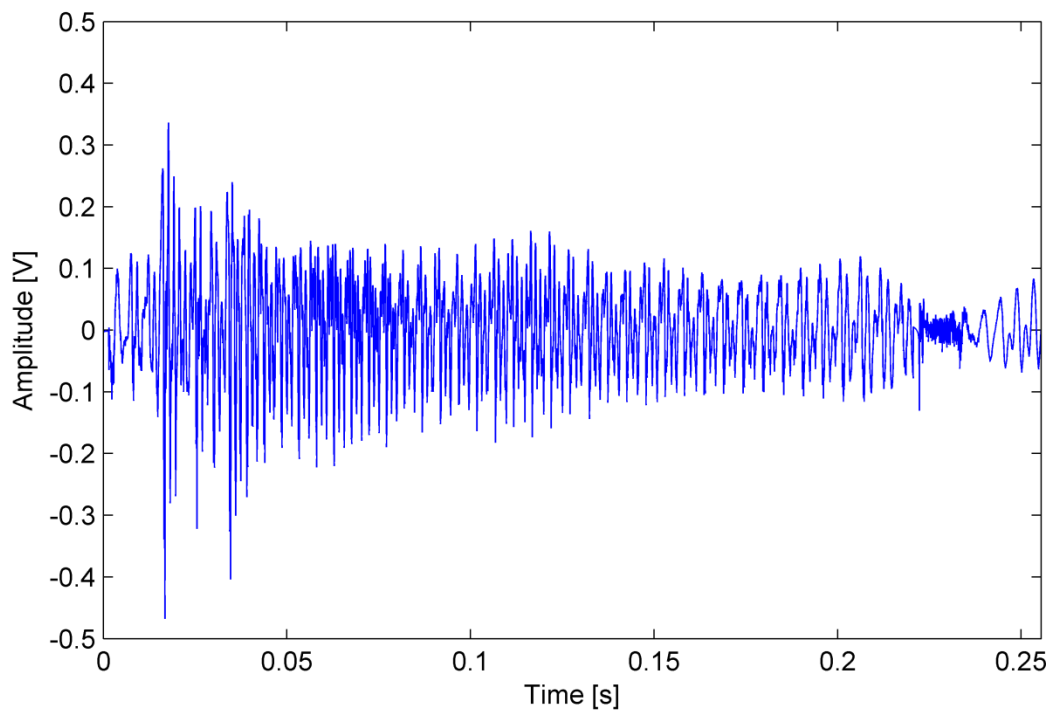


Fig. A.170. The waveform of the utterance “bud” detected by the voice activity detection algorithm.

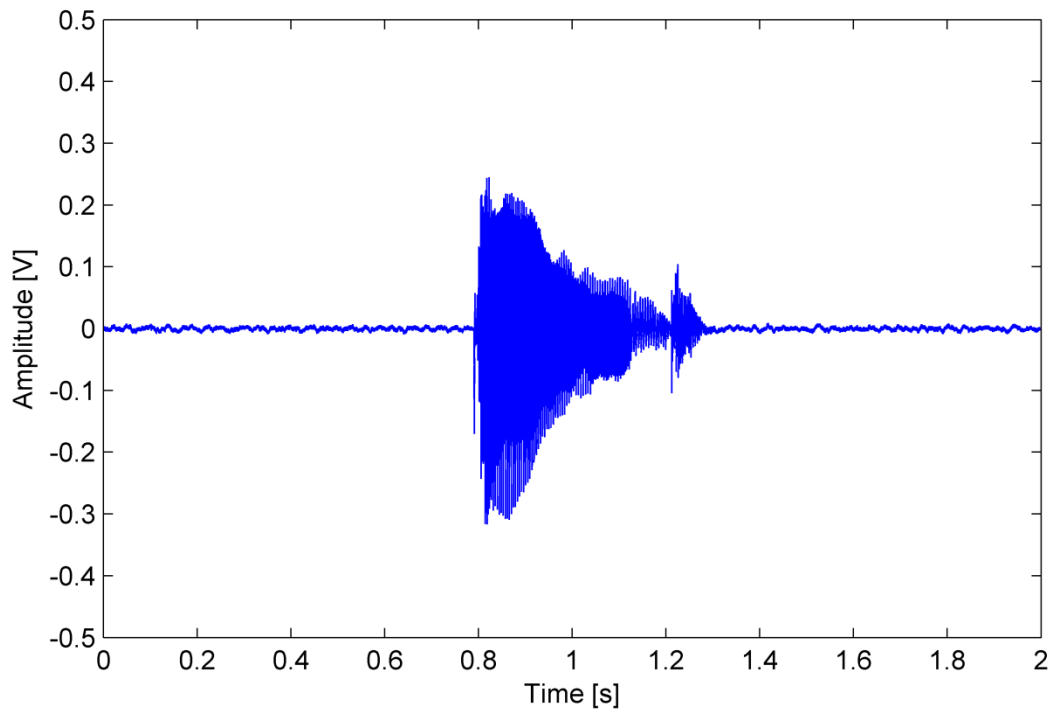


Fig. A.171. The waveform of the utterance "bird".

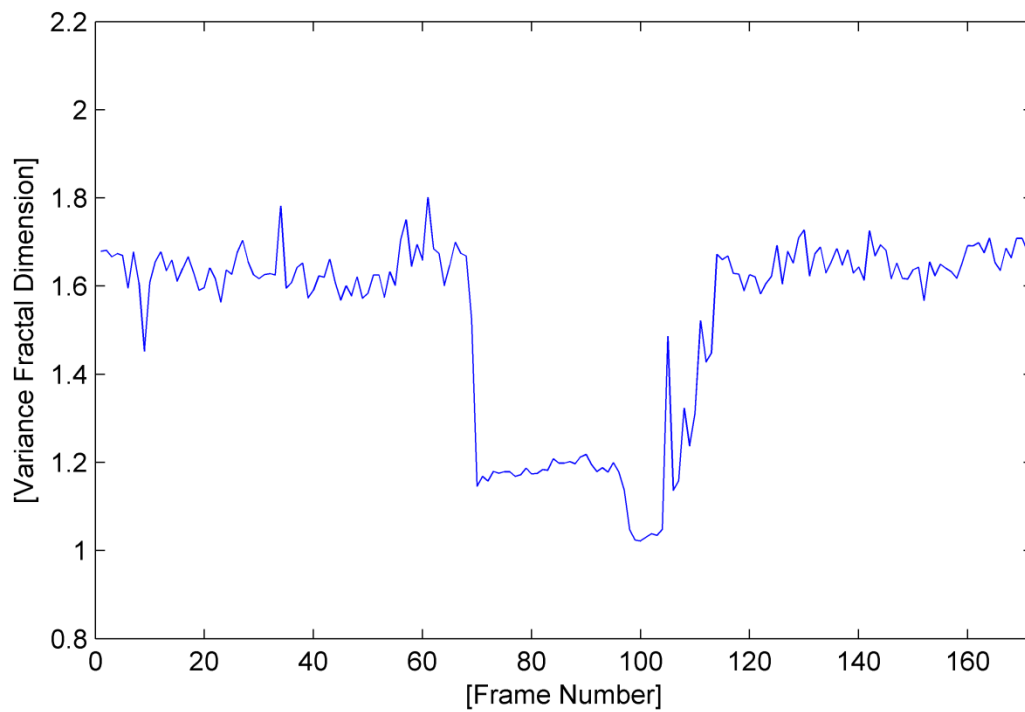


Fig. A.172. The variance fractal dimension trajectory of the utterance "bird".

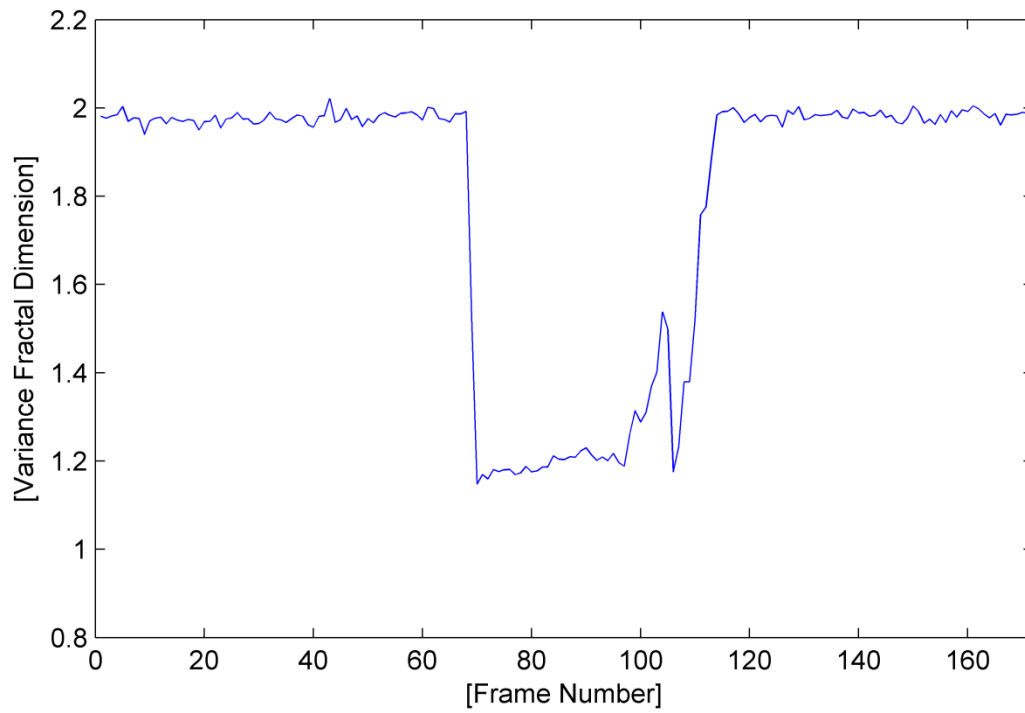


Fig. A.173. The variance fractal dimension trajectory of the utterance "bird" after addition of white noise.

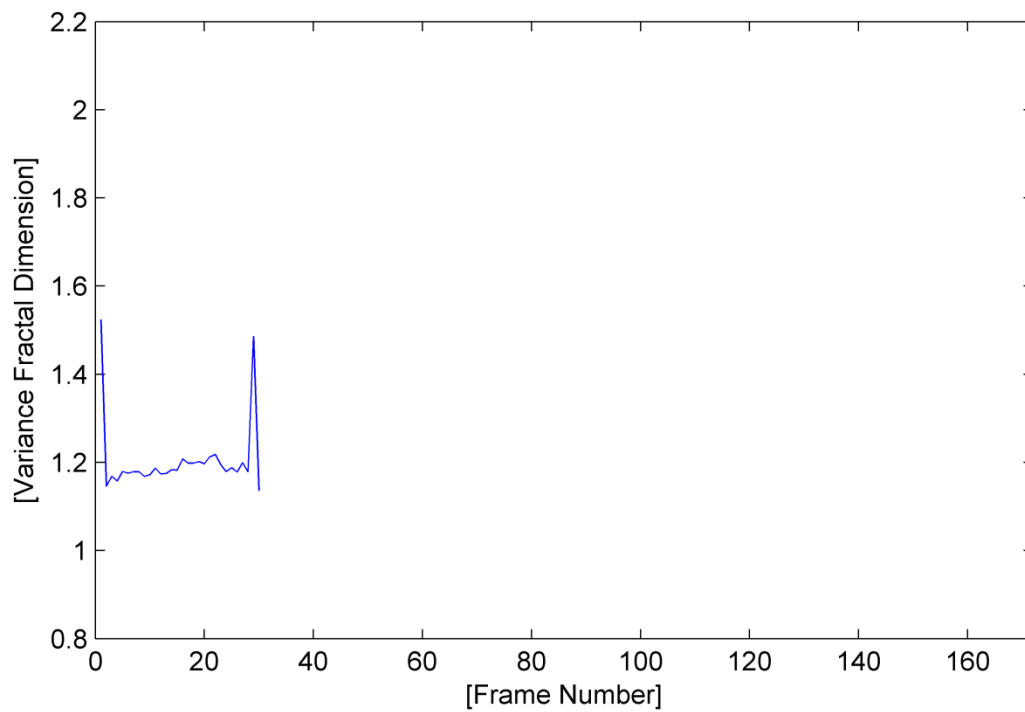


Fig. A.174. The trajectory of the utterance "bird" detected by the voice activity detection algorithm.

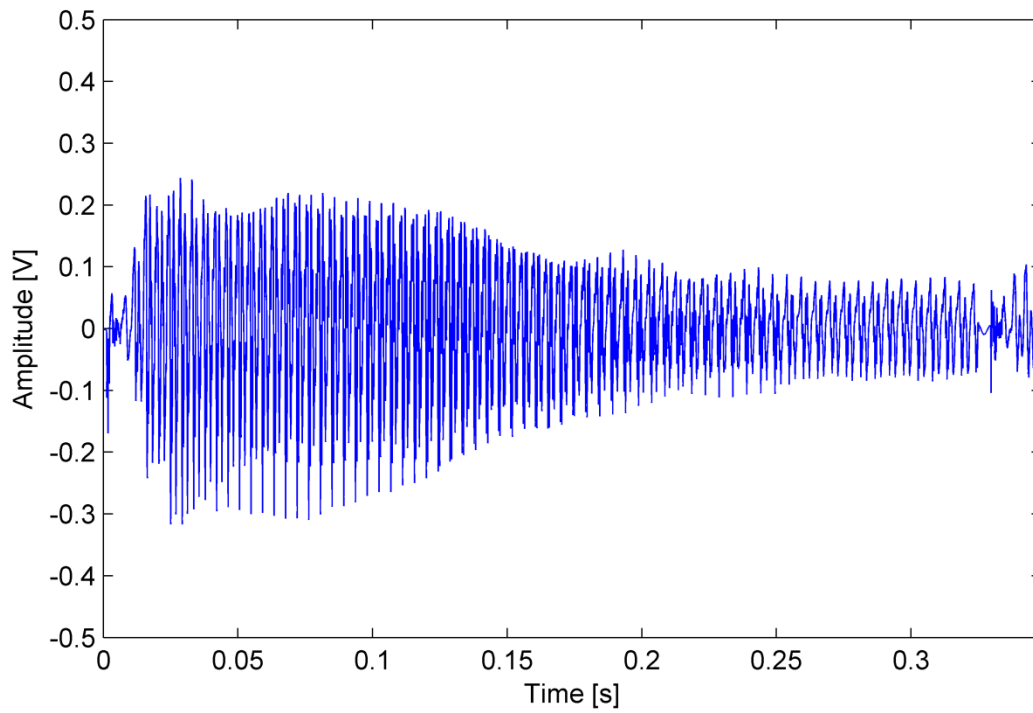


Fig. A.175. The waveform of the utterance "bird" detected by the voice activity detection algorithm.

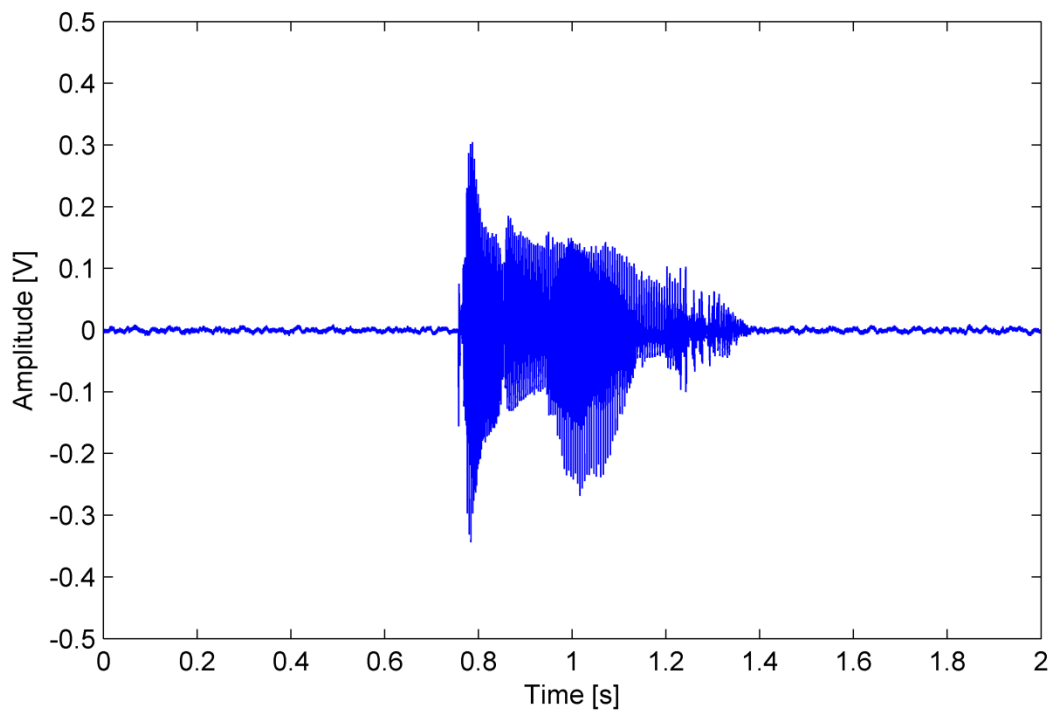


Fig. A.176. The waveform of the utterance "banana".

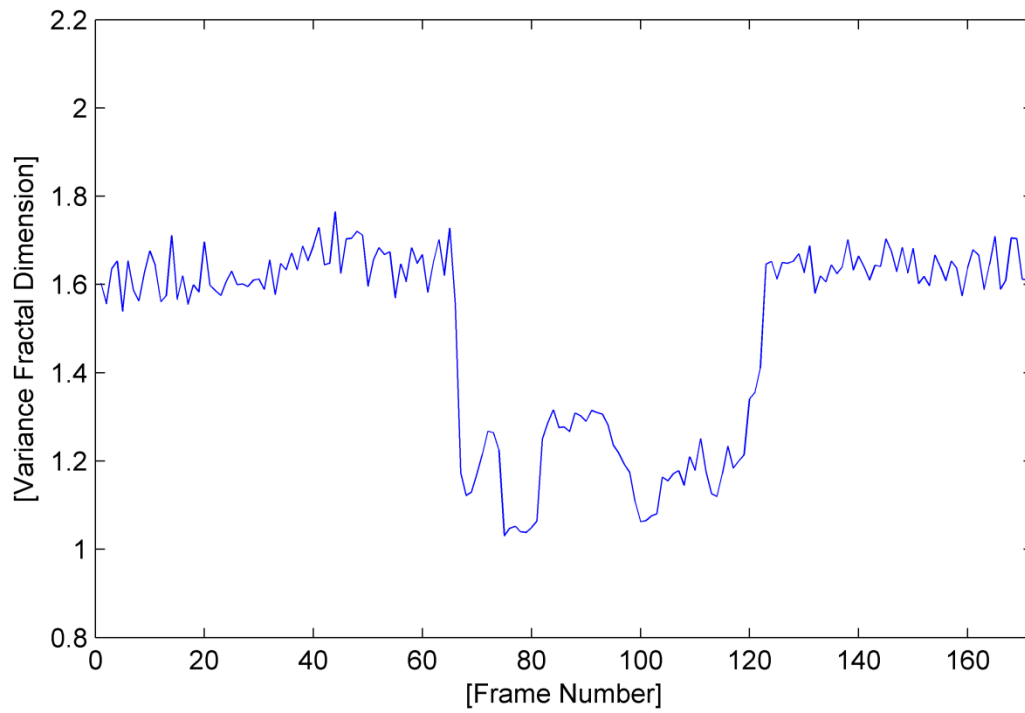


Fig. A.177. The variance fractal dimension trajectory of the utterance “banana”.

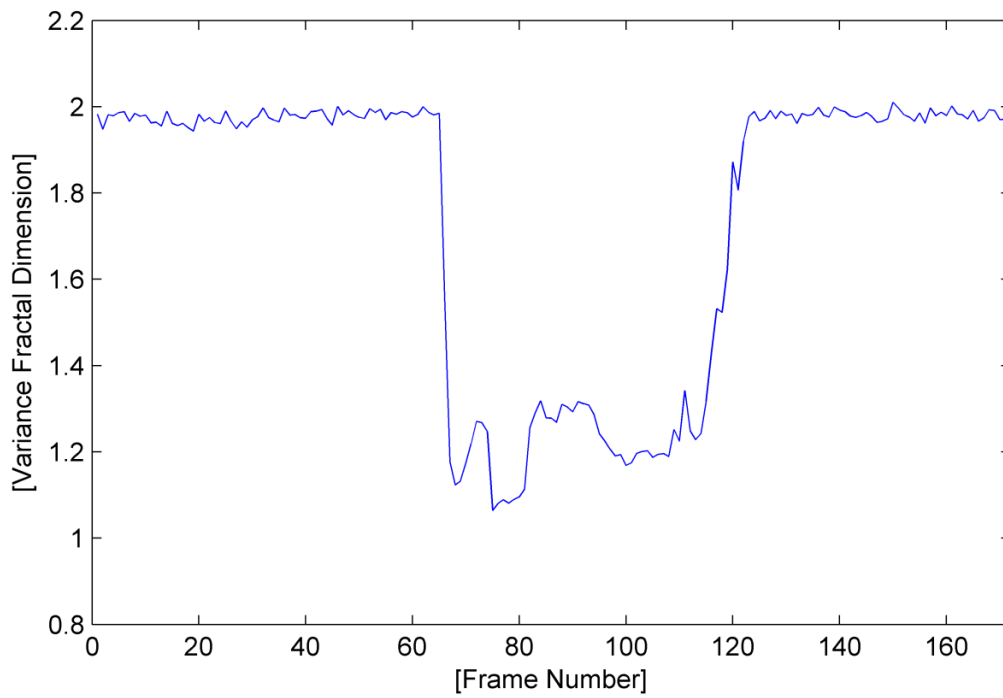


Fig. A.178. The variance fractal dimension trajectory of the utterance “banana” after addition of white noise.

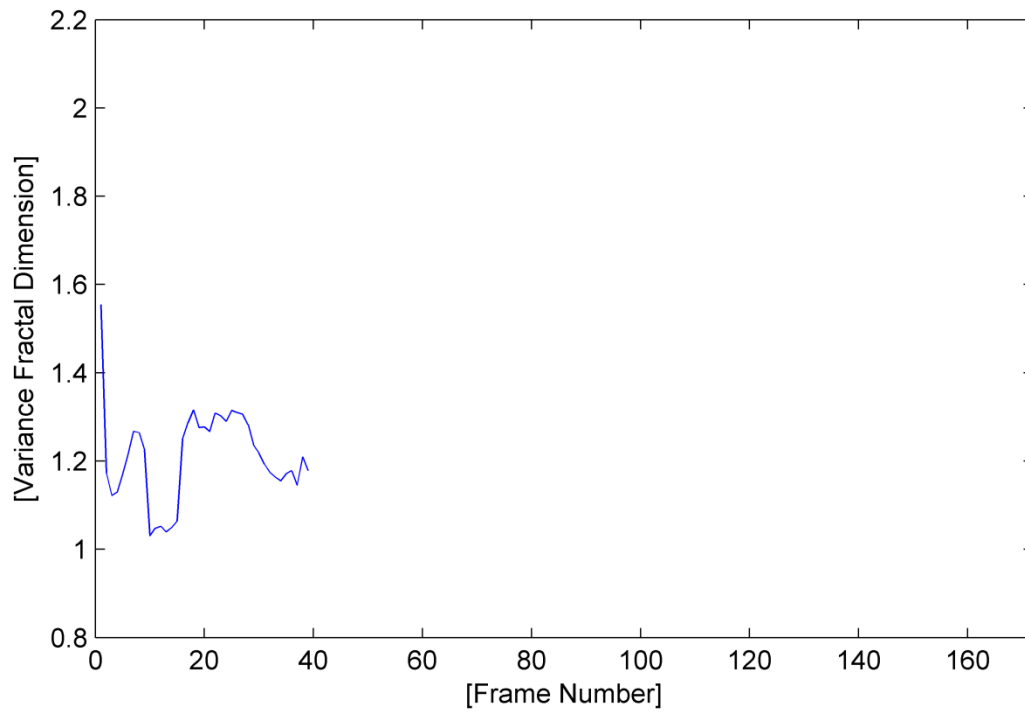


Fig. A.179. The trajectory of the utterance “banana” detected by the voice activity detection algorithm.

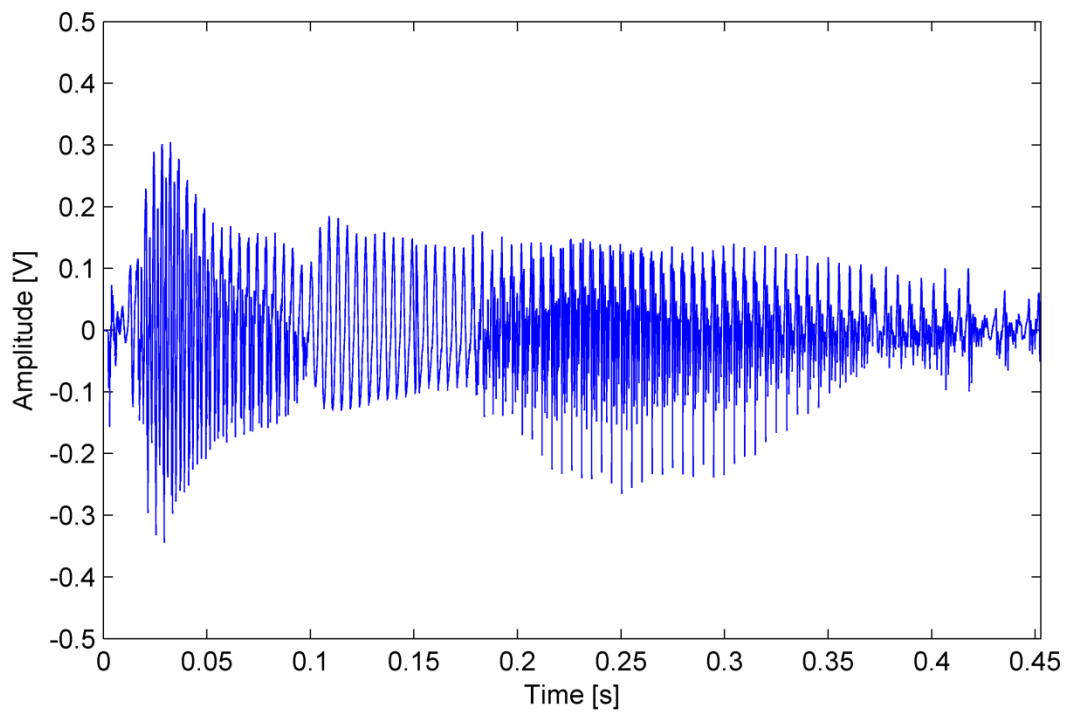


Fig. A.180. The waveform of the utterance “banana” detected by the voice activity detection algorithm.

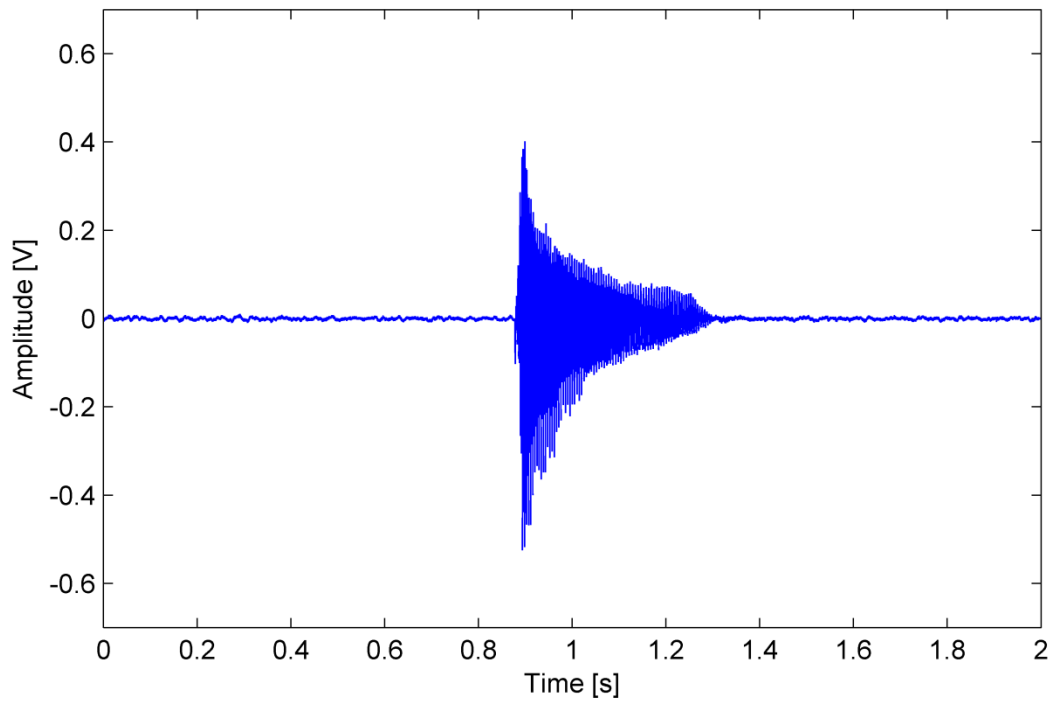


Fig. A.181. The waveform of the utterance "bay".

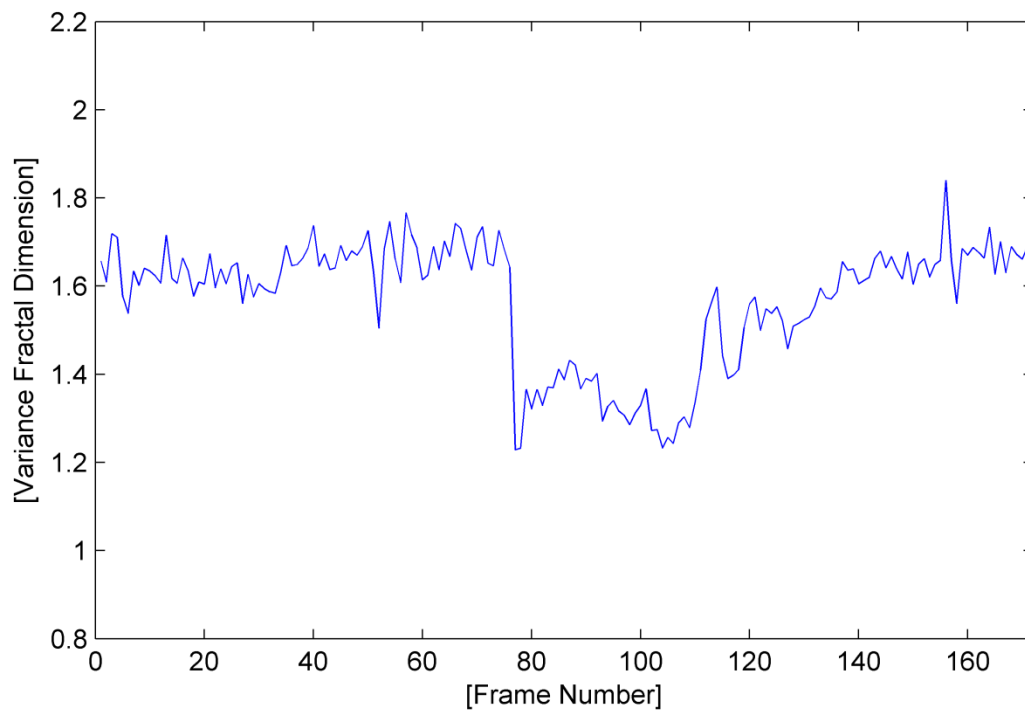


Fig. A.182. The variance fractal dimension trajectory of the utterance "bay".

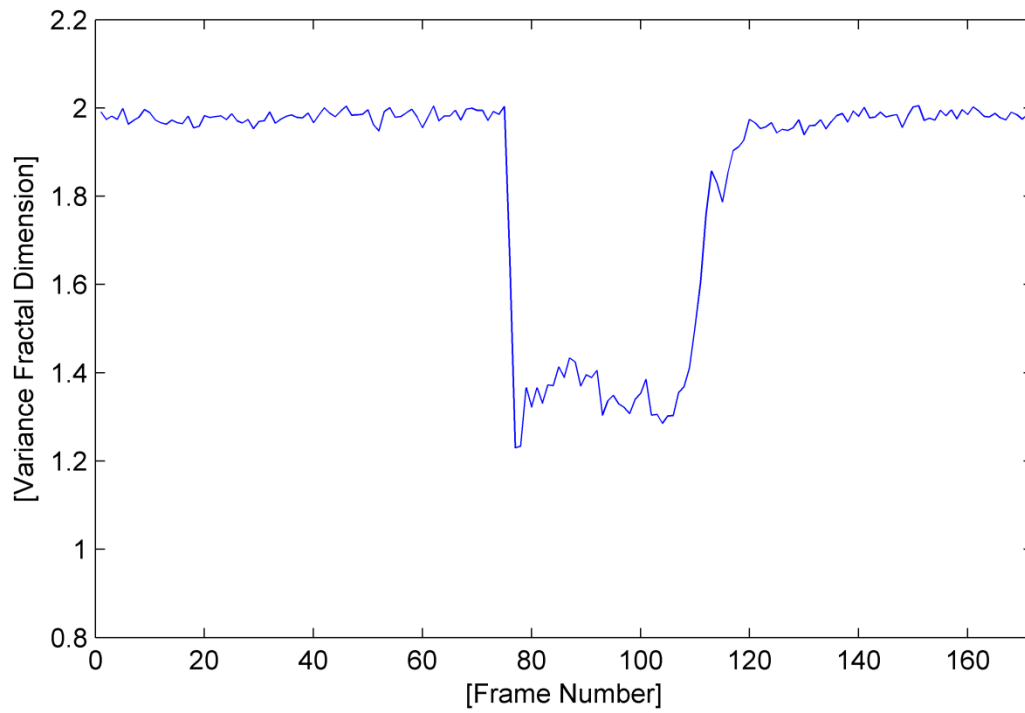


Fig. A.183. The variance fractal dimension trajectory of the utterance “bay” after addition of white noise.

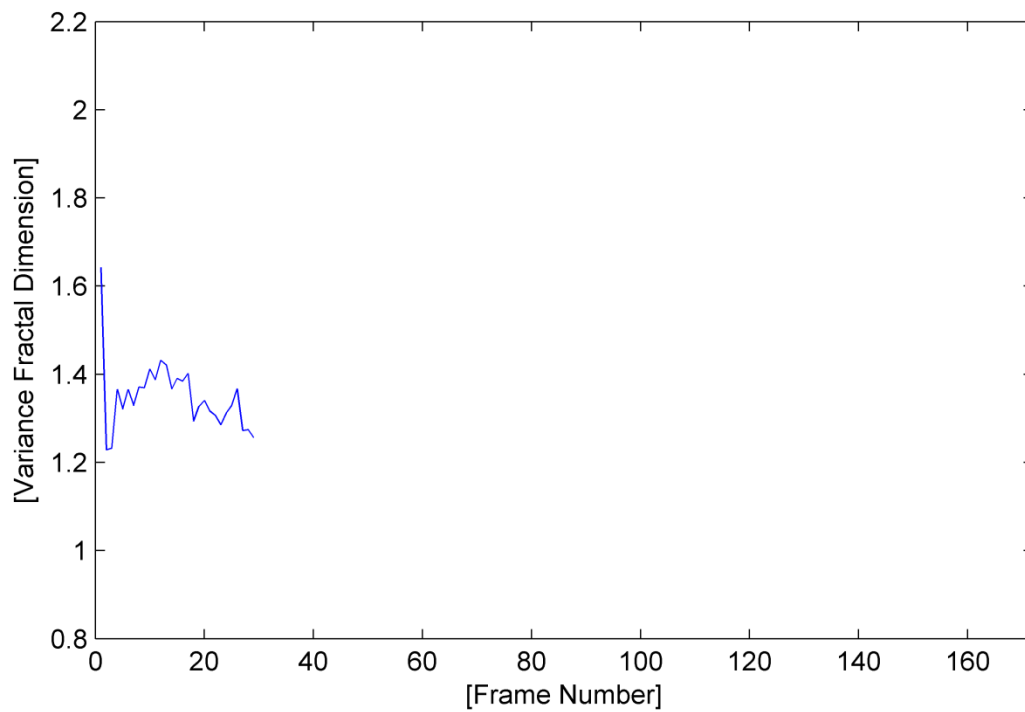


Fig. A.184. The trajectory of the utterance “bay” detected by the voice activity detection algorithm.

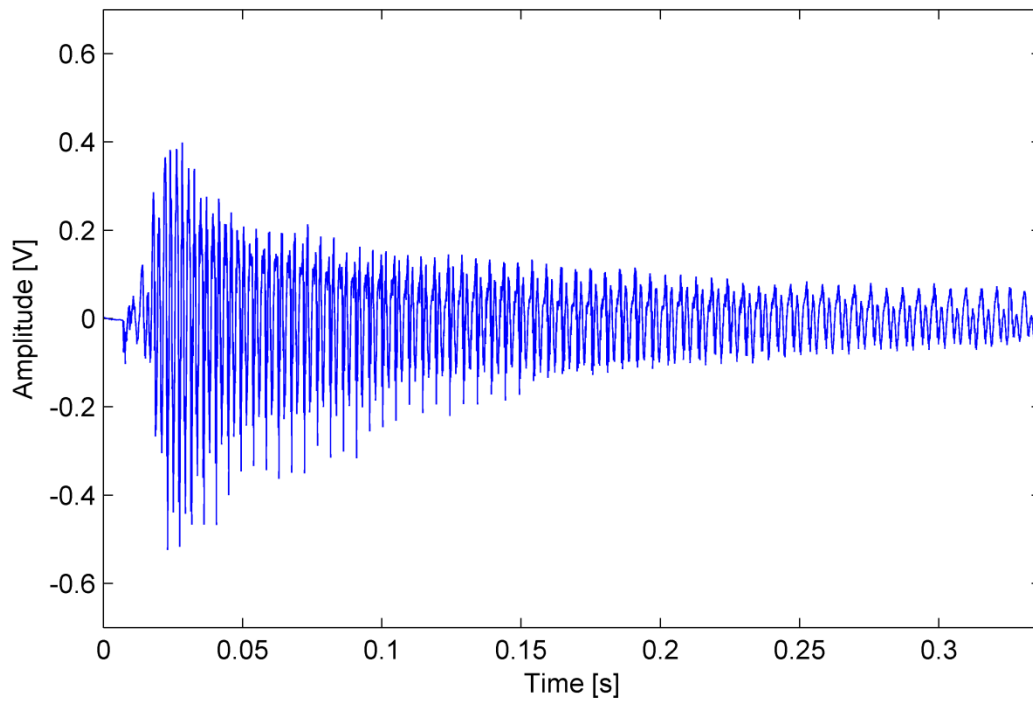


Fig. A.185. The waveform of the utterance "bay" detected by the voice activity detection algorithm.

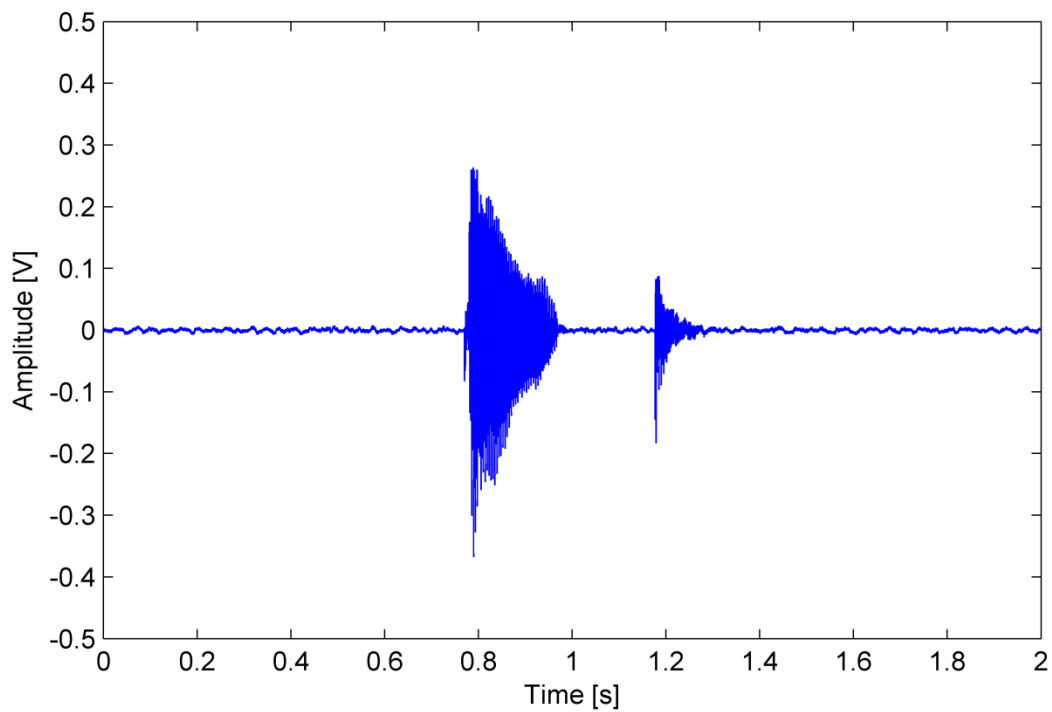


Fig. A.186. The waveform of the utterance "boat".

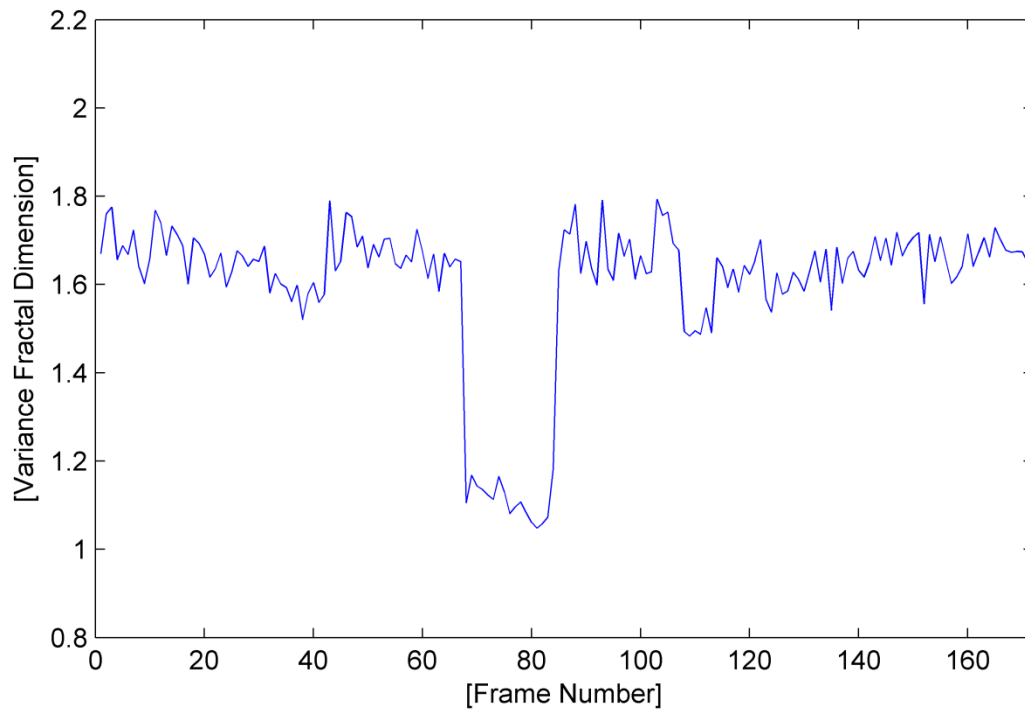


Fig. A.187. The variance fractal dimension trajectory of the utterance "boat".

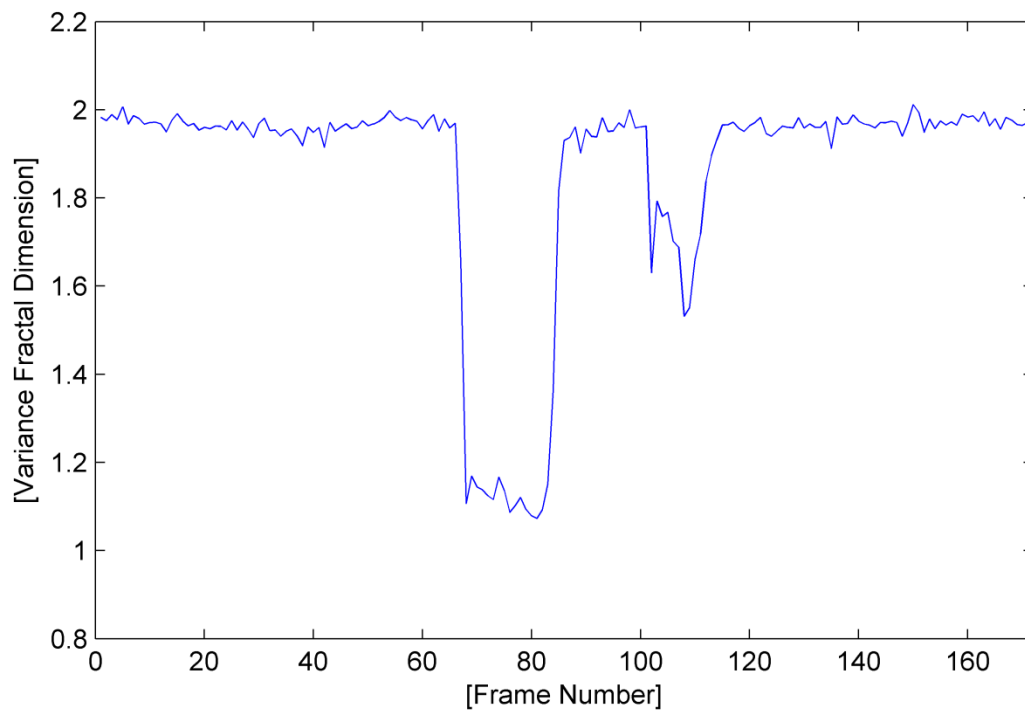


Fig. A.188. The variance fractal dimension trajectory of the utterance "boat" after addition of white noise.

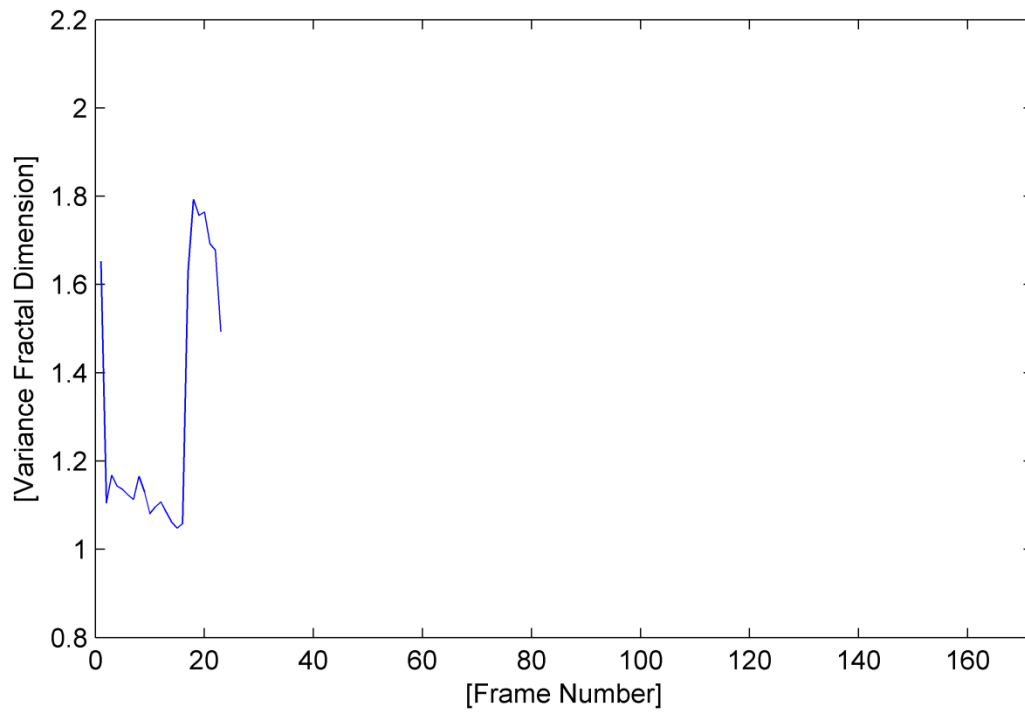


Fig. A.189. The trajectory of the utterance “boat” detected by the voice activity detection algorithm.

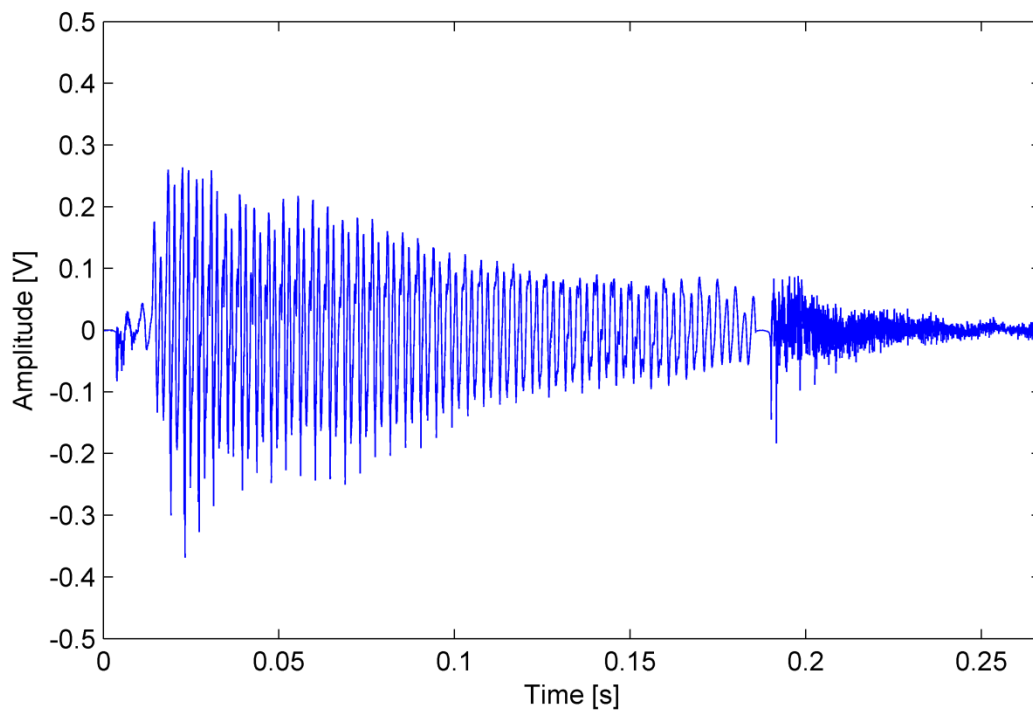


Fig. A.190. The waveform of the utterance “boat” detected by the voice activity detection algorithm.

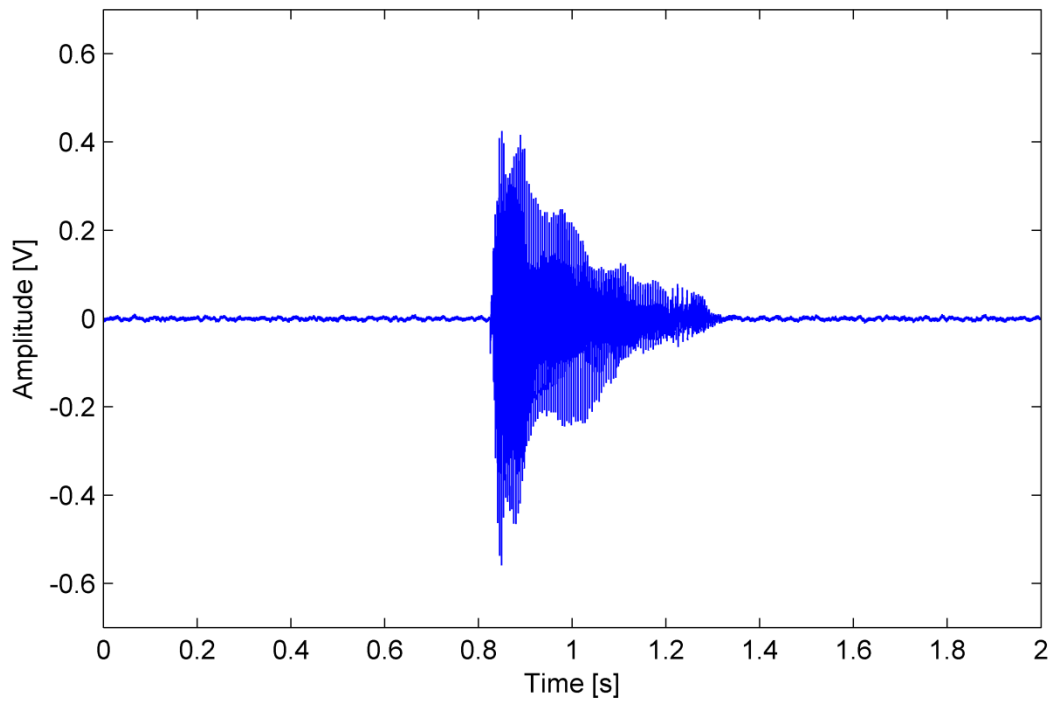


Fig. A.191. The waveform of the utterance "buy".

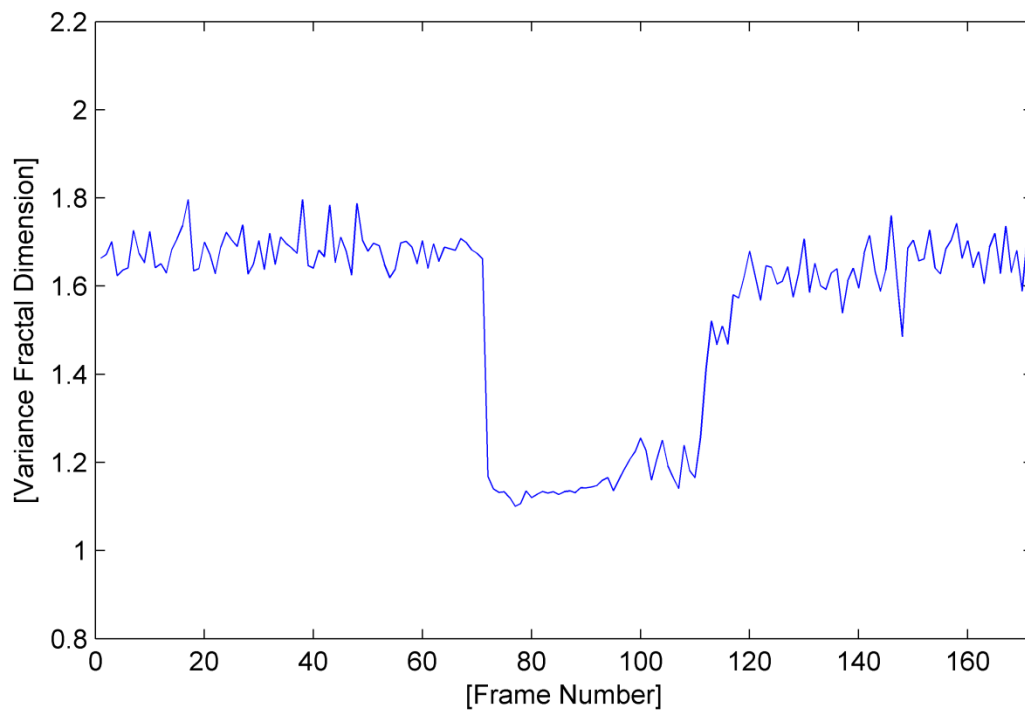


Fig. A.192. The variance fractal dimension trajectory of the utterance "buy".

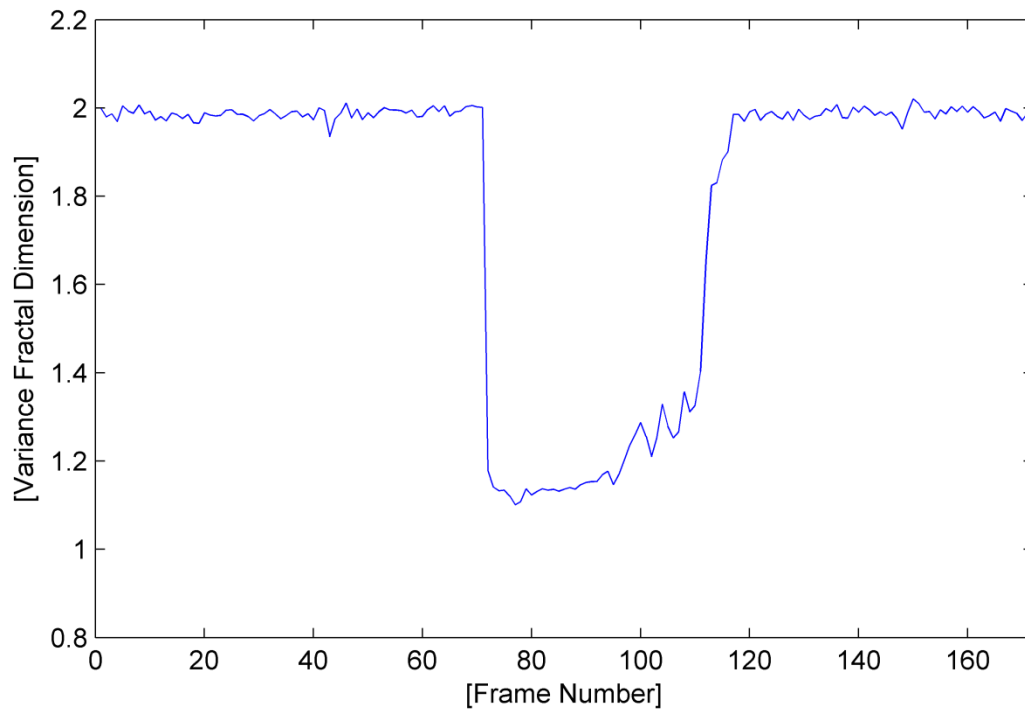


Fig. A.193. The variance fractal dimension trajectory of the utterance “buy” after addition of white noise.

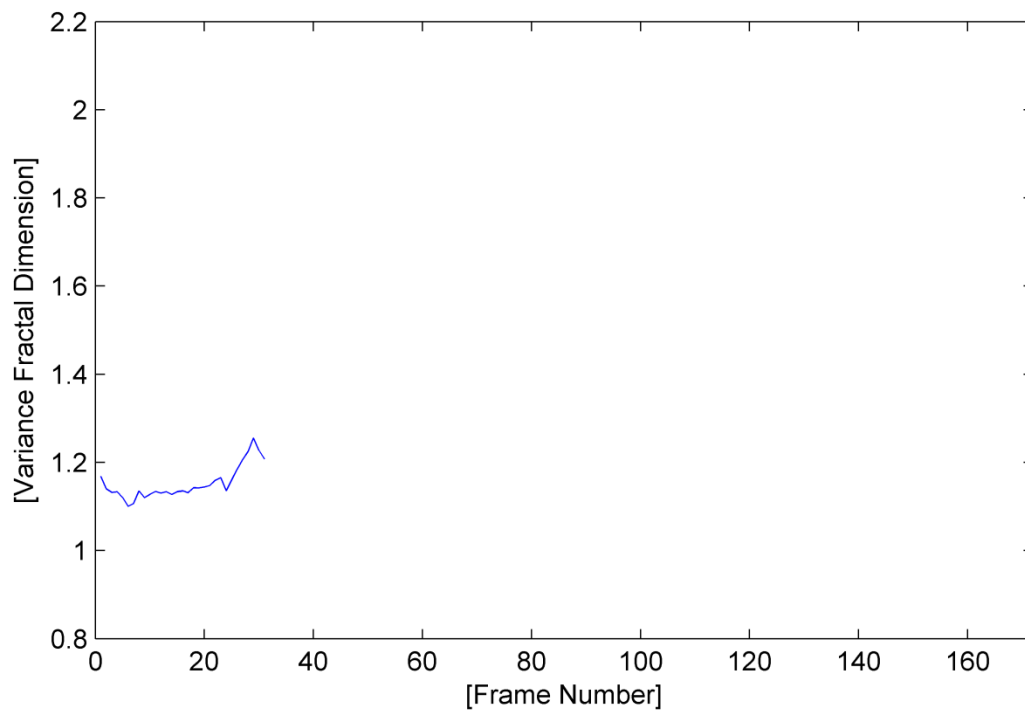


Fig. A.194. The trajectory of the utterance “buy” detected by the voice activity detection algorithm.

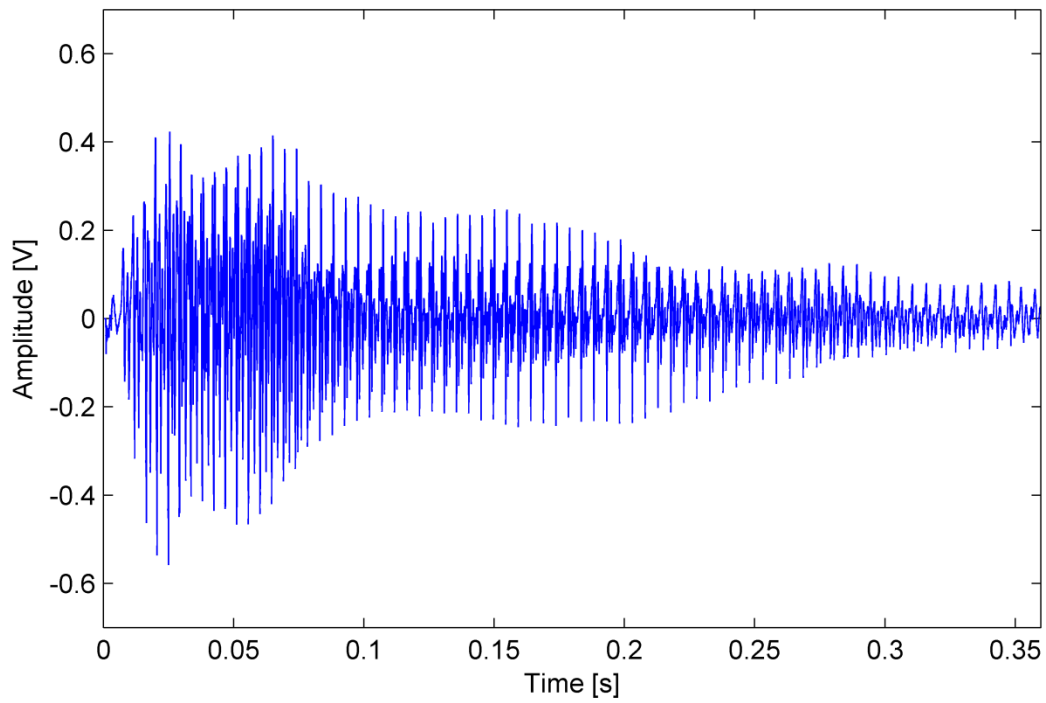


Fig. A.195. The waveform of the utterance "buy" detected by the voice activity detection algorithm.

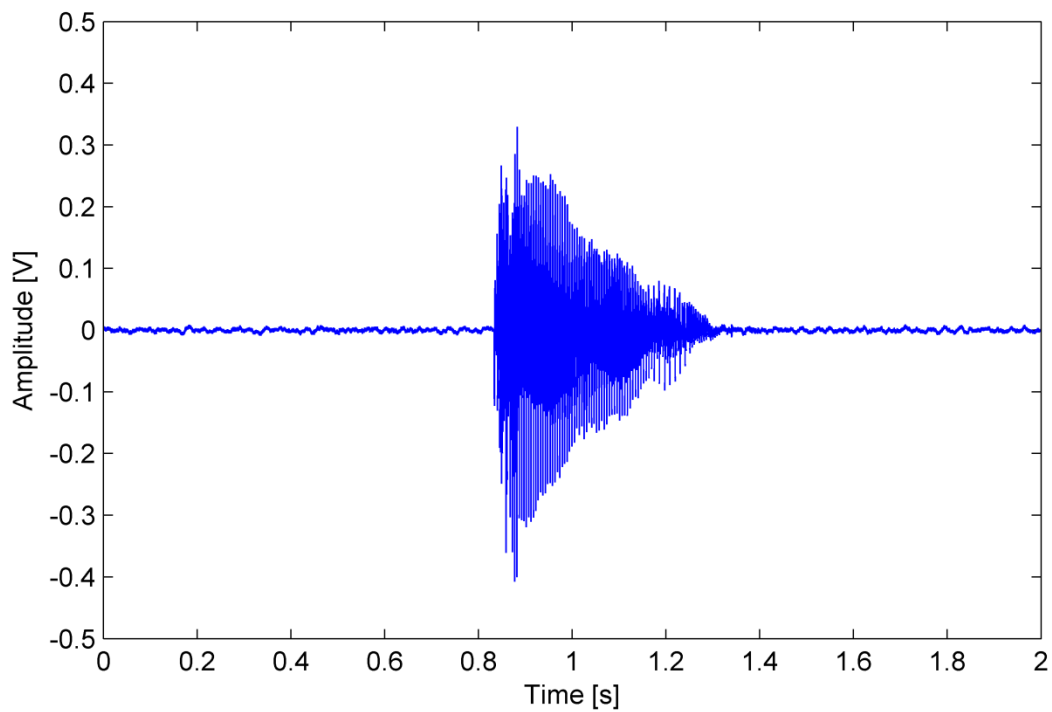


Fig. A.196. The waveform of the utterance "bough".

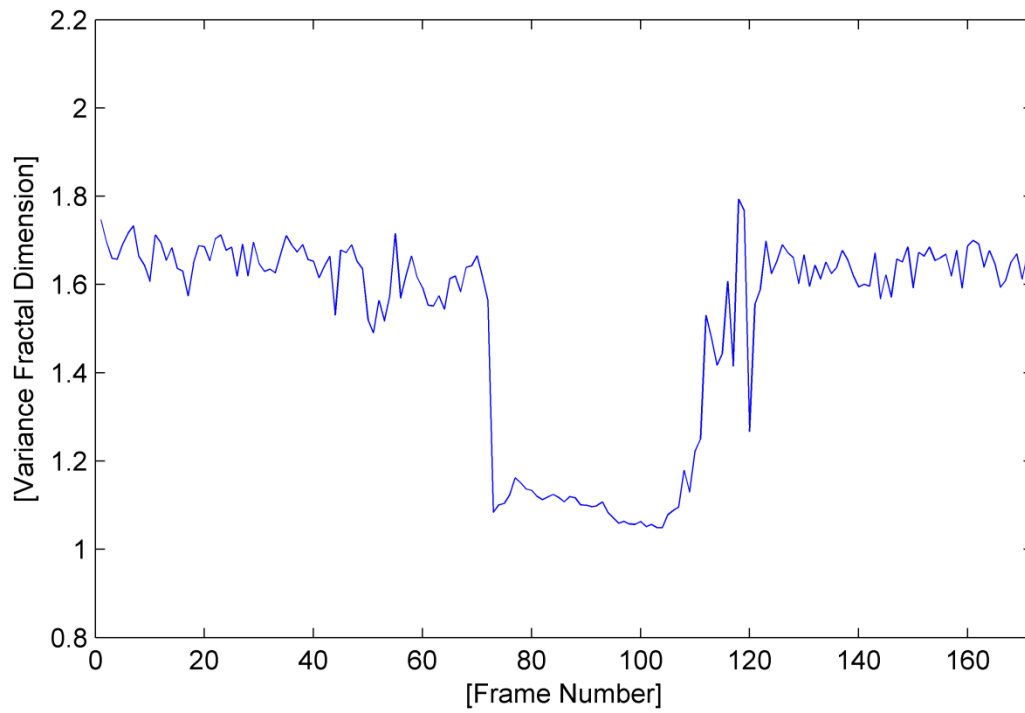


Fig. A.197. The variance fractal dimension trajectory of the utterance "bough".

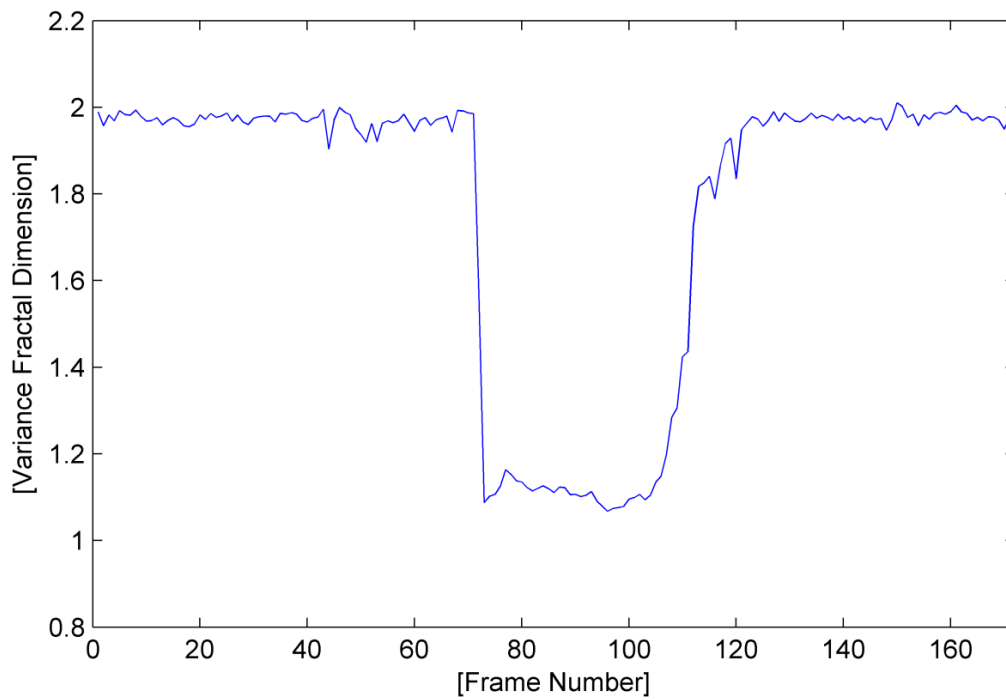


Fig. A.198. The variance fractal dimension trajectory of the utterance "bough" after addition of white noise.

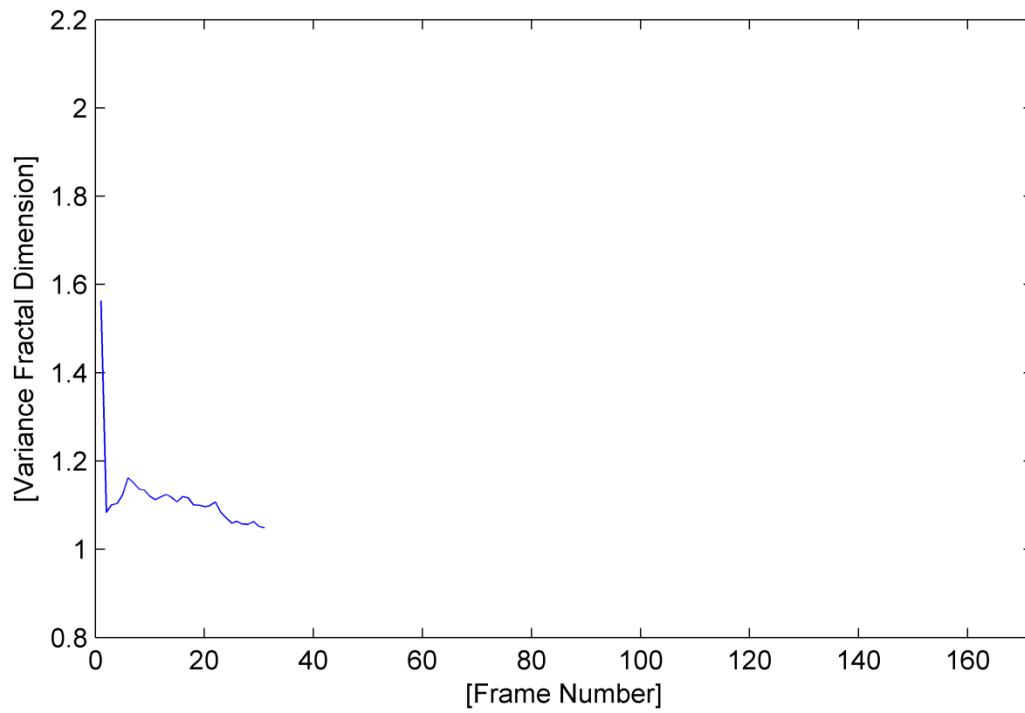


Fig. A.199. The trajectory of the utterance “bough” detected by the voice activity detection algorithm.

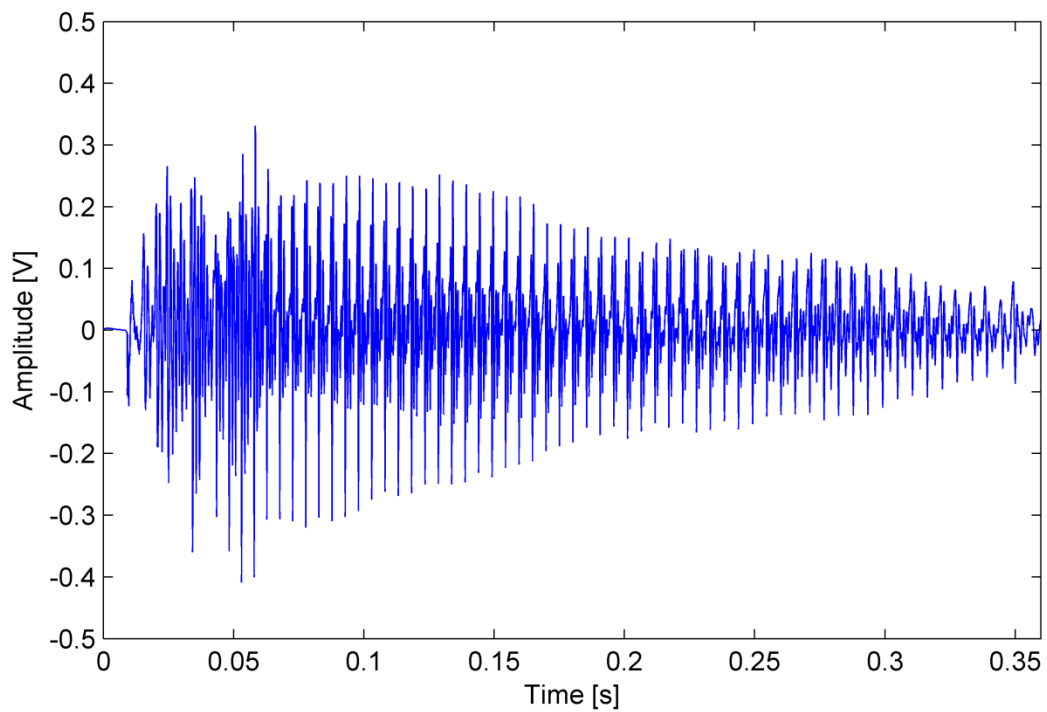


Fig. A.200. The waveform of the utterance “bough” detected by the voice activity detection algorithm.

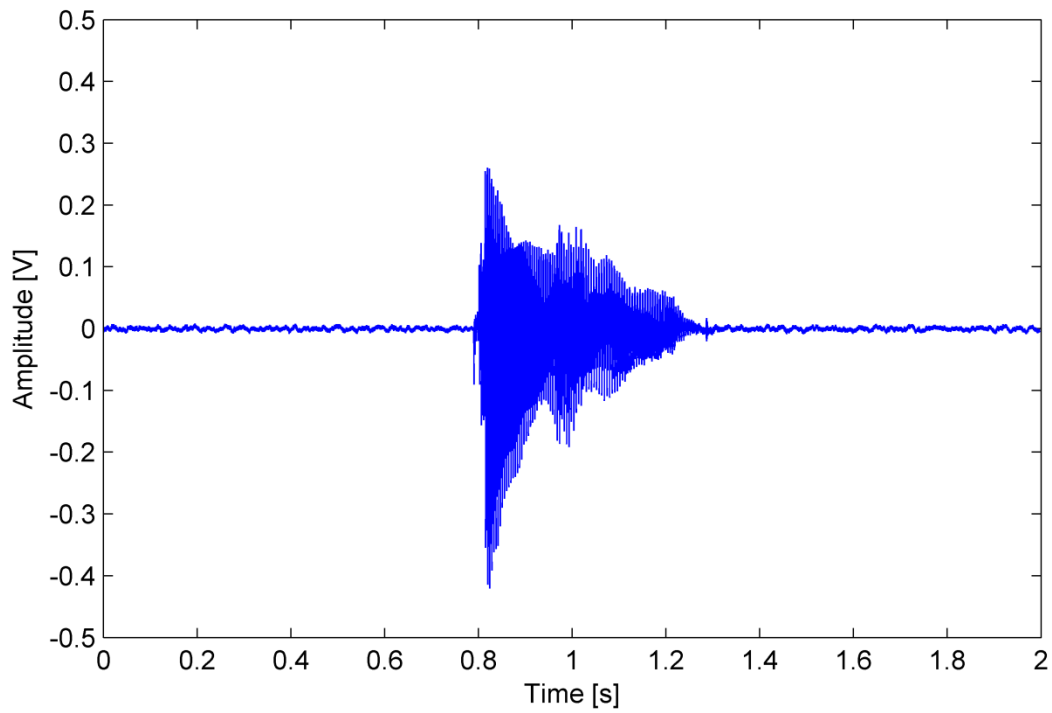


Fig. A.201. The waveform of the utterance “boy”.

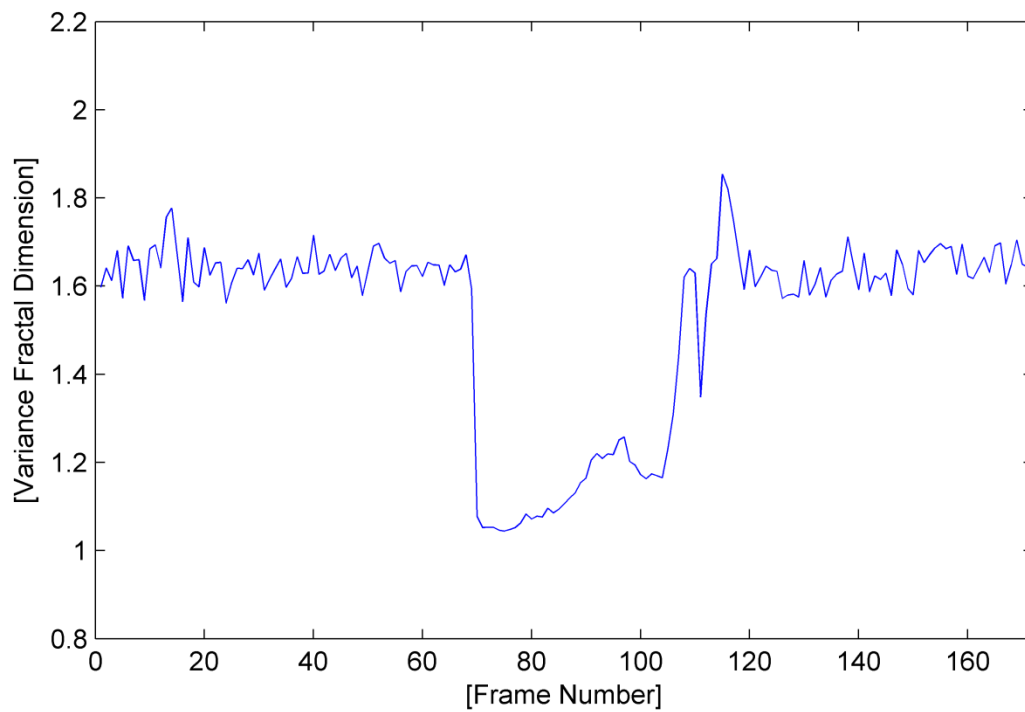


Fig. A.202. The variance fractal dimension trajectory of the utterance “boy”.

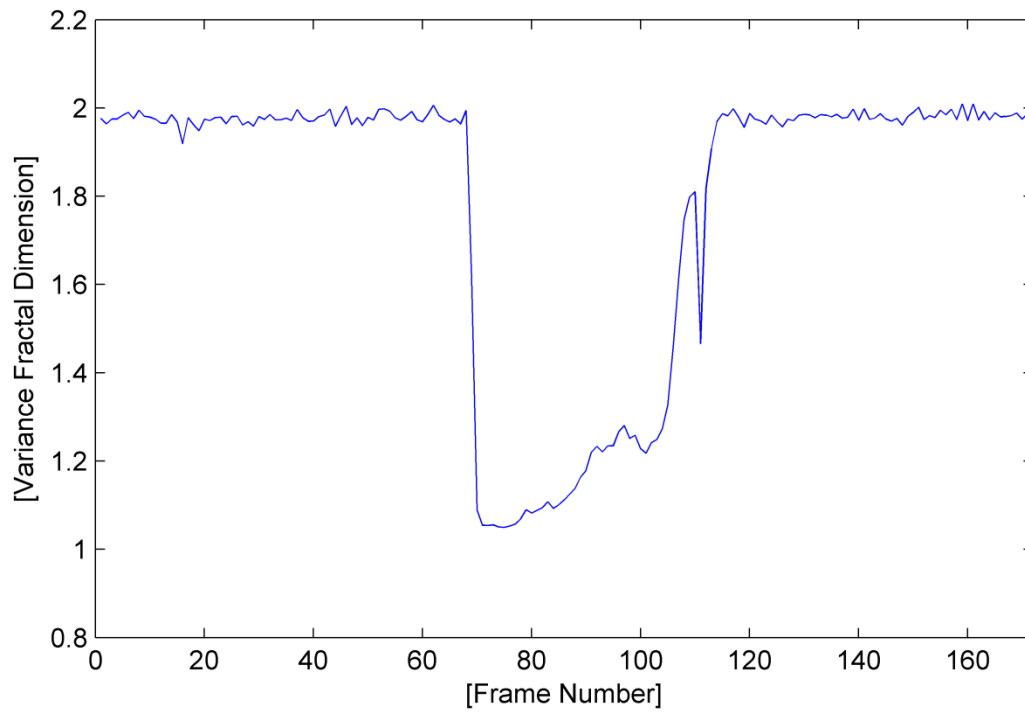


Fig. A.203. The variance fractal dimension trajectory of the utterance “boy” after addition of white noise.

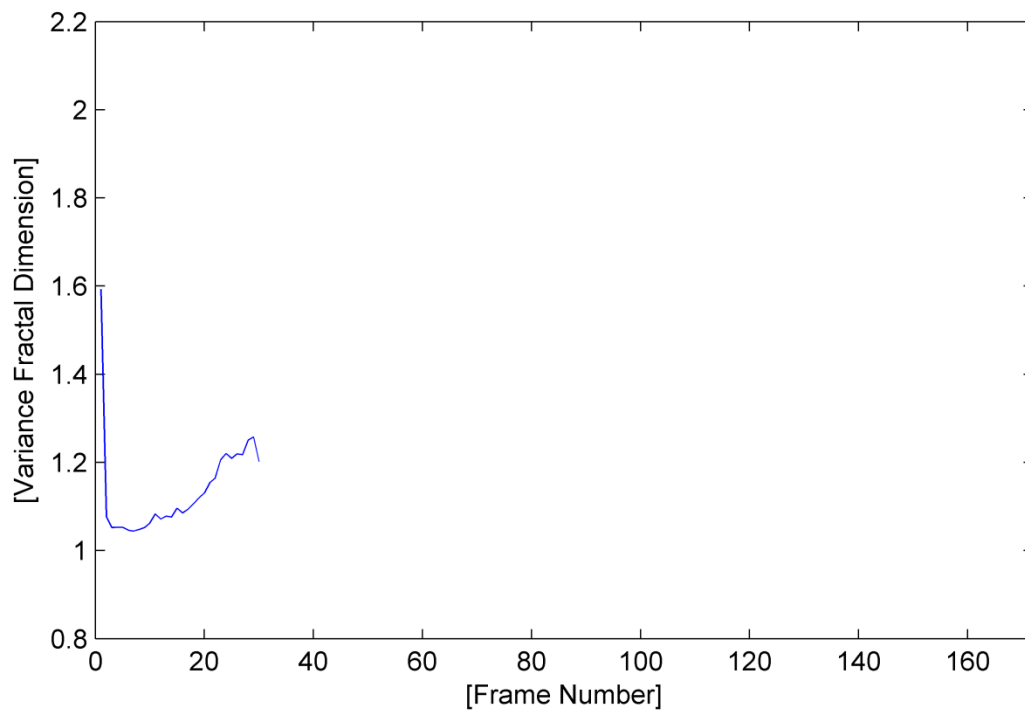


Fig. A.204. The trajectory of the utterance “boy” detected by the voice activity detection algorithm.

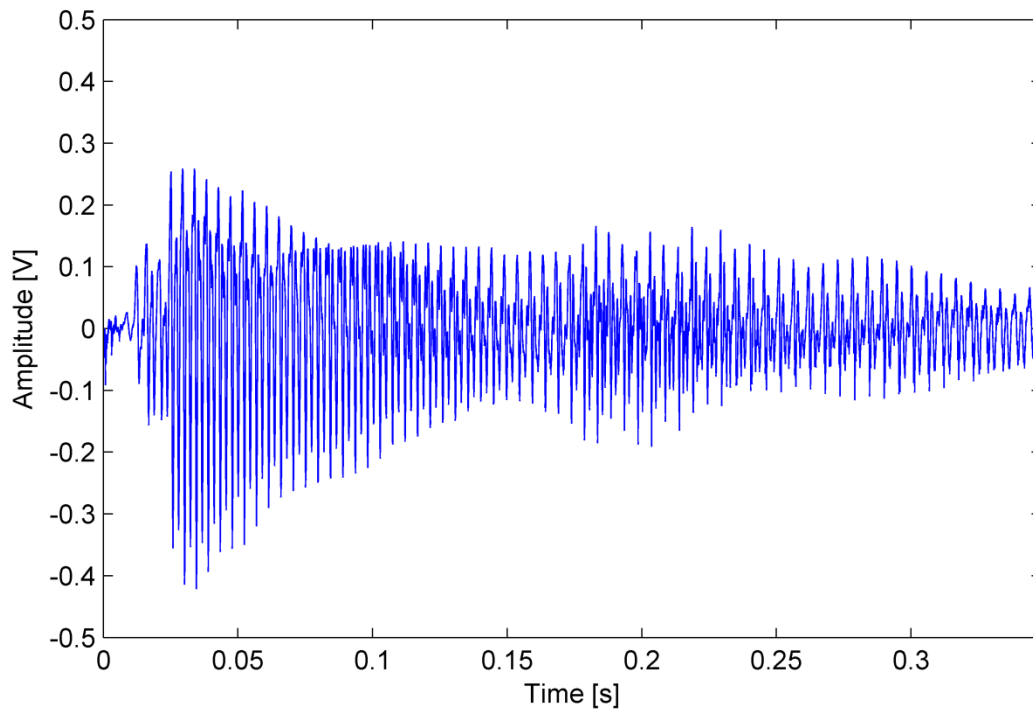


Fig. A.205. The waveform of the utterance "boy" detected by the voice activity detection algorithm.

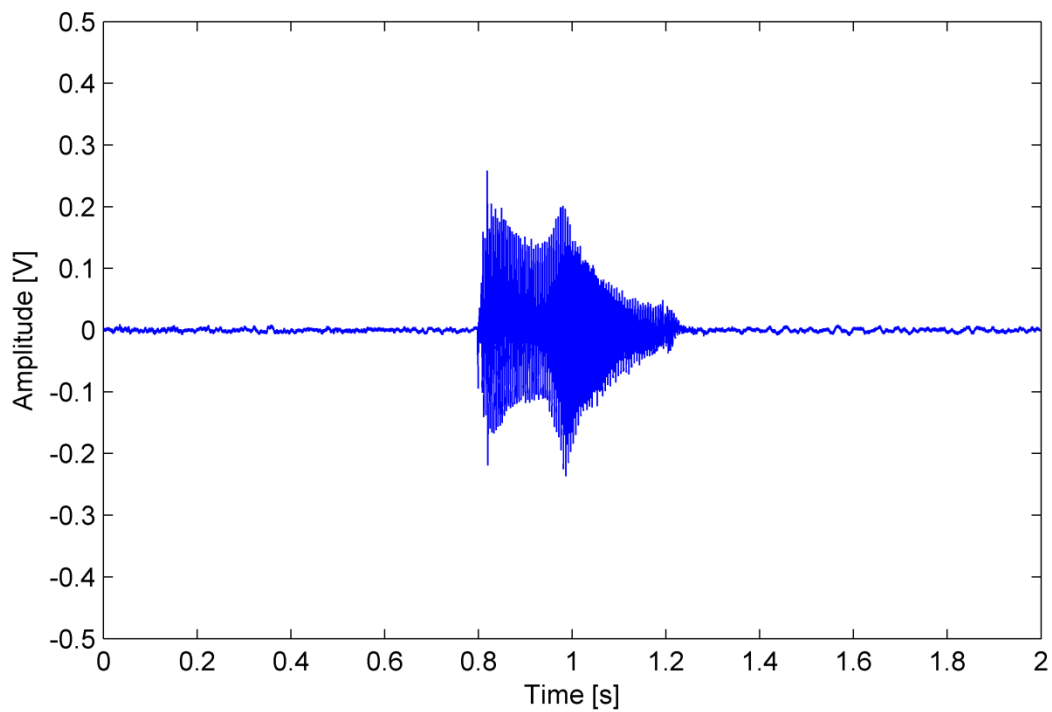


Fig. A.206. The waveform of the utterance "beer".

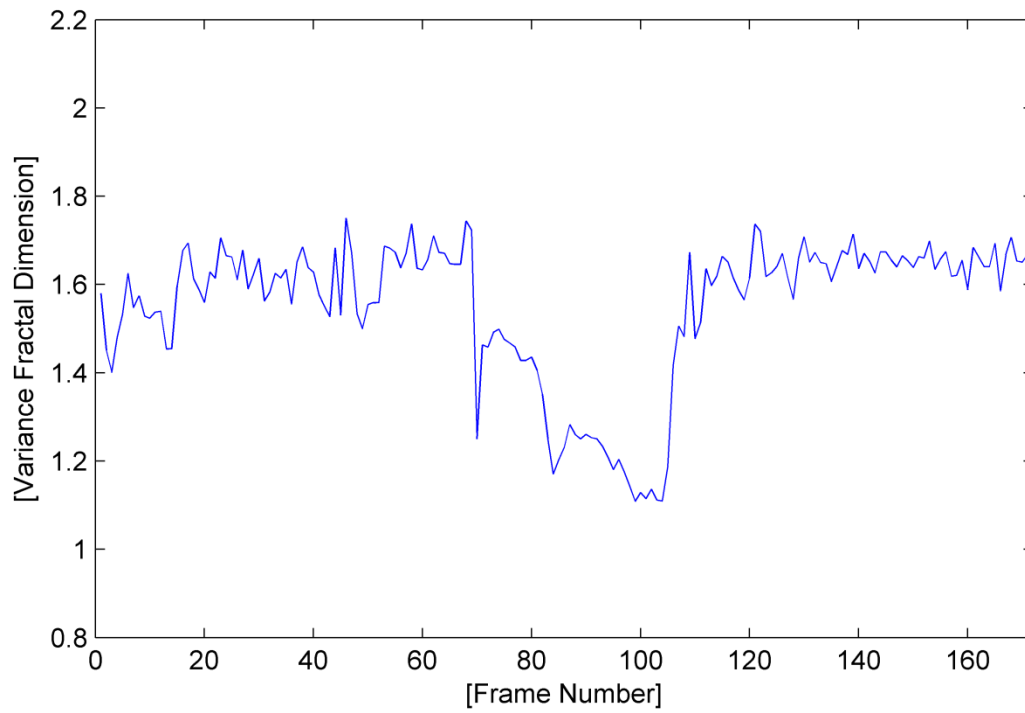


Fig. A.207. The variance fractal dimension trajectory of the utterance "beer".

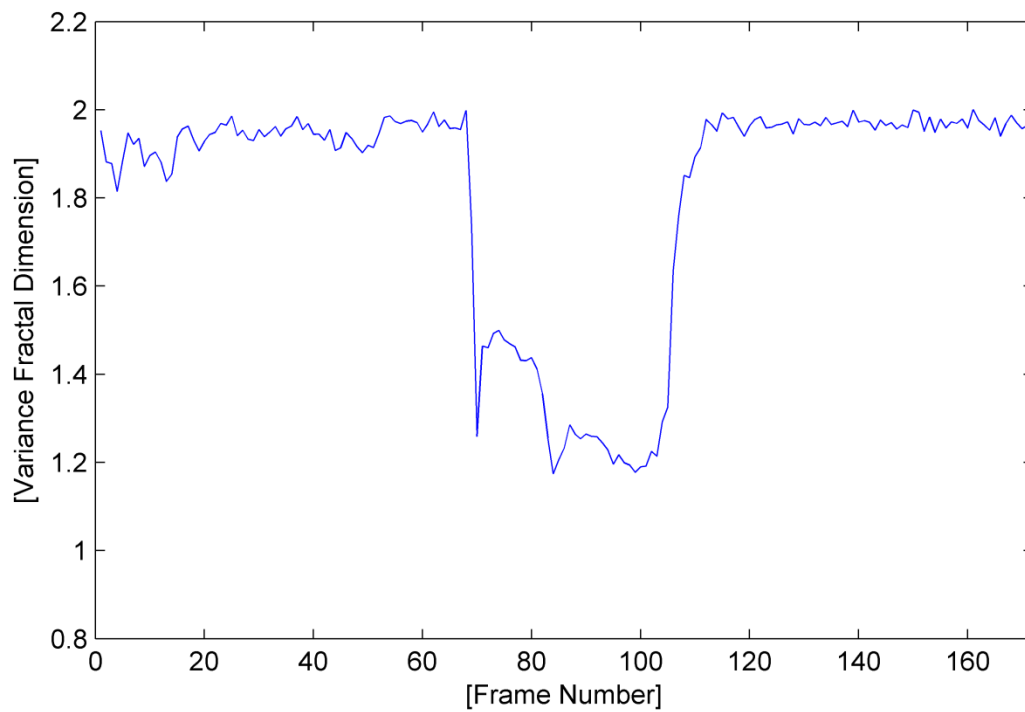


Fig. A.208. The variance fractal dimension trajectory of the utterance "beer" after addition of white noise.

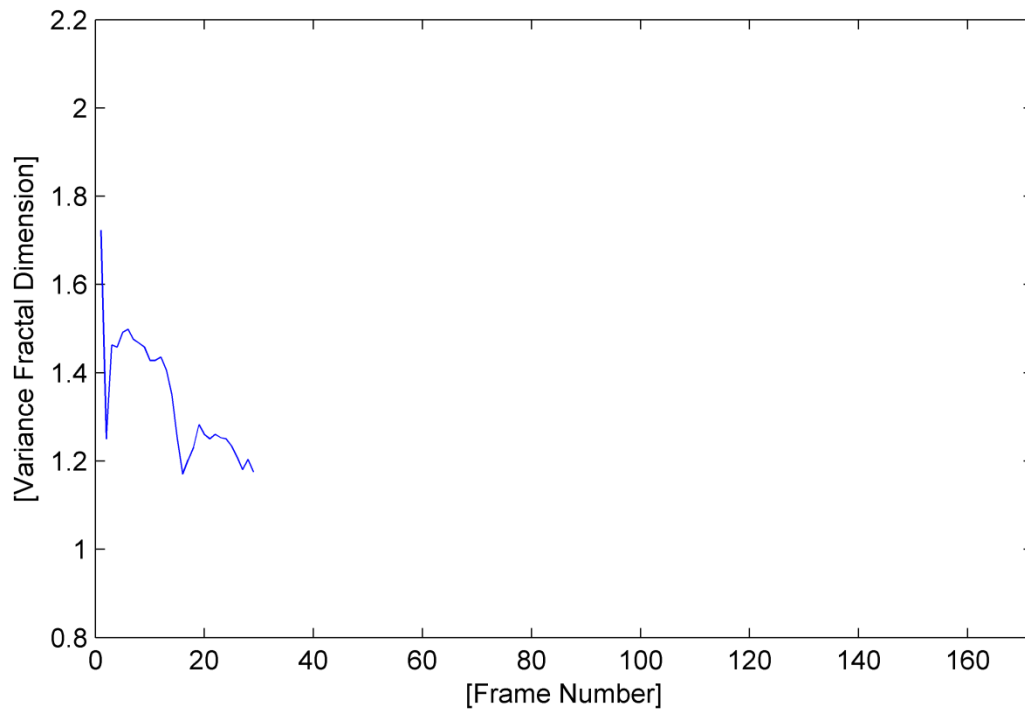


Fig. A.209. The trajectory of the utterance “beer” detected by the voice activity detection algorithm.

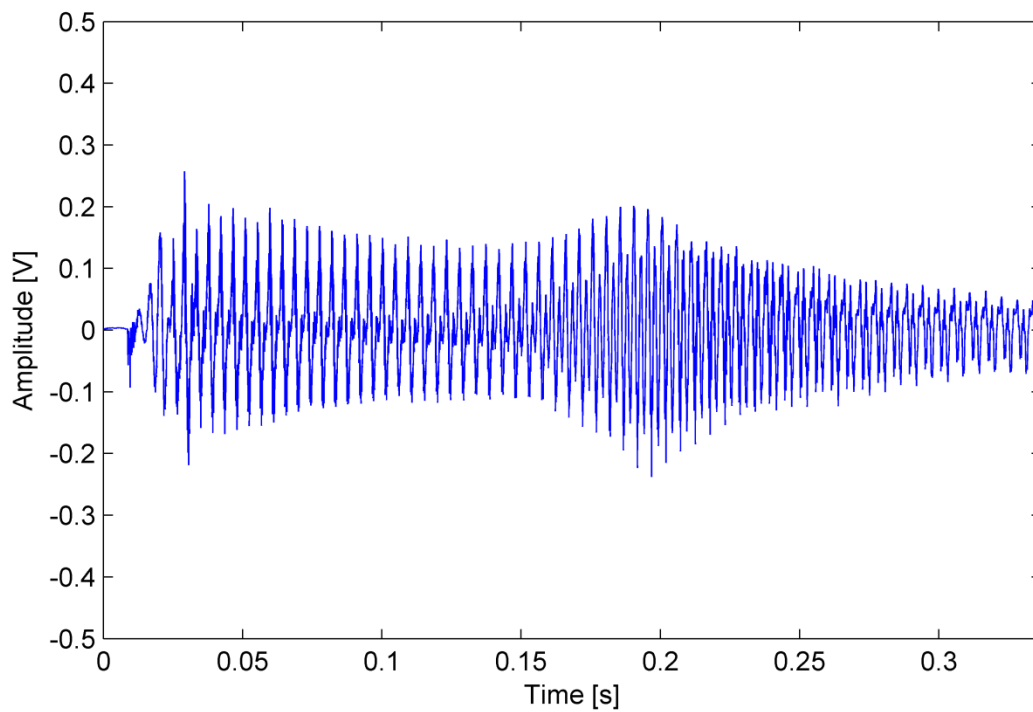


Fig. A.210. The waveform of the utterance “beer” detected by the voice activity detection algorithm.

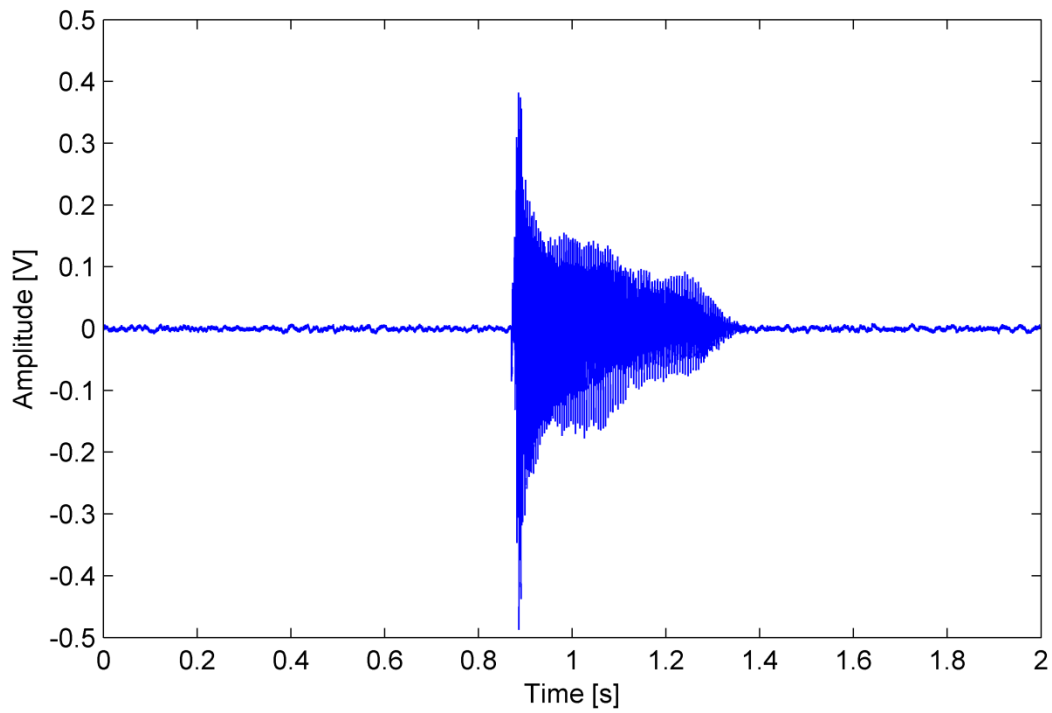


Fig. A.211. The waveform of the utterance "bear".

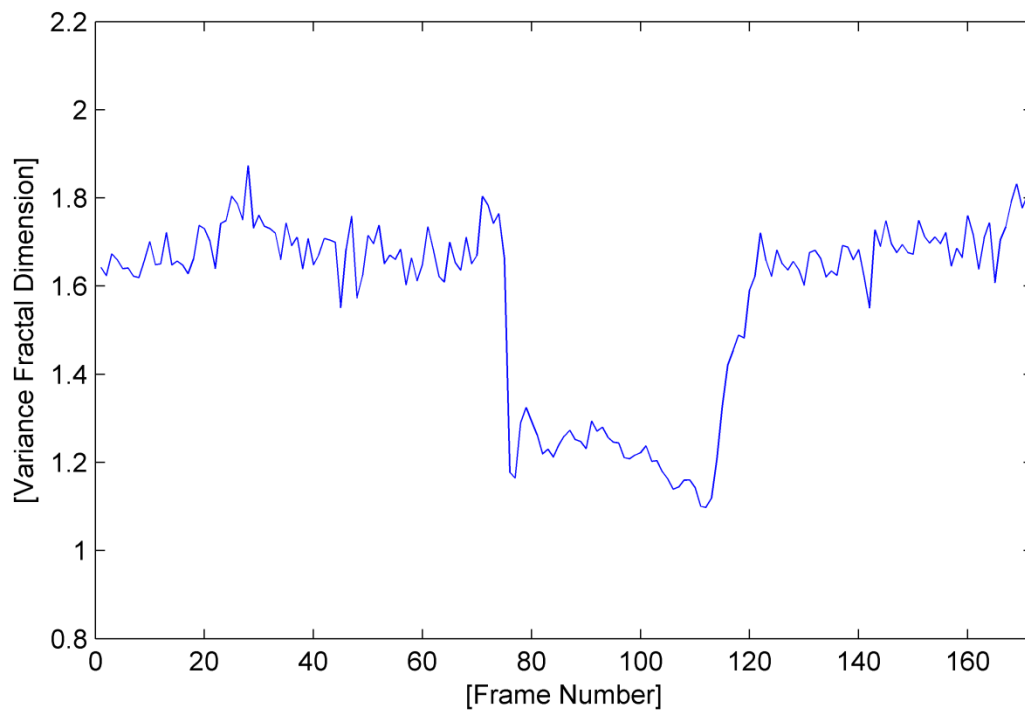


Fig. A.212. The variance fractal dimension trajectory of the utterance "bear".

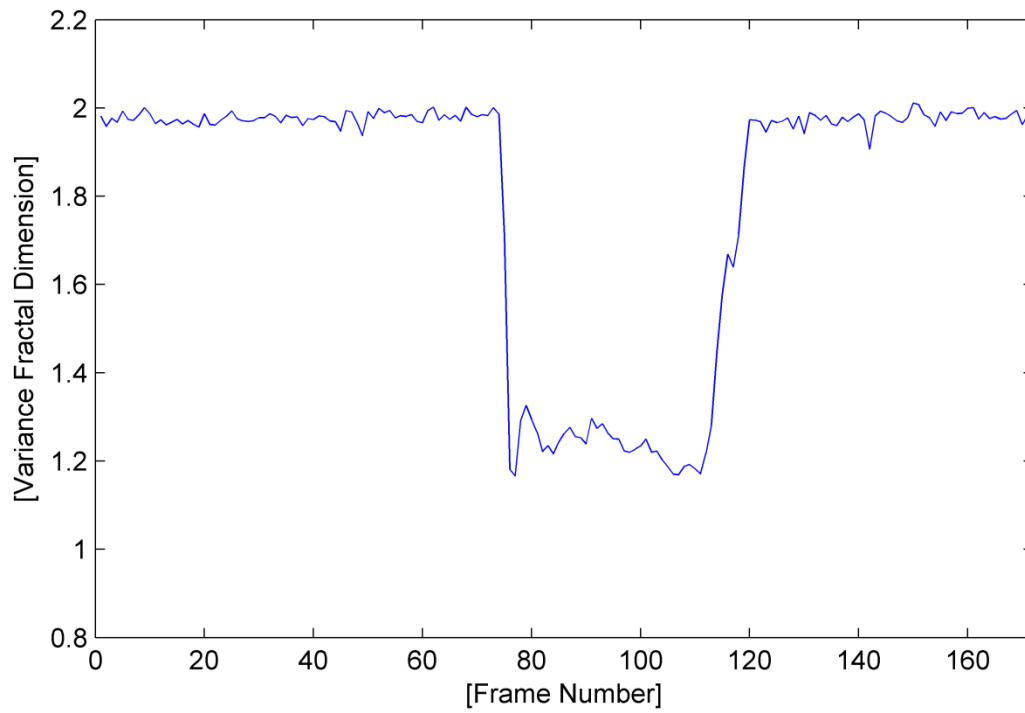


Fig. A.213. The variance fractal dimension trajectory of the utterance "bear" after addition of white noise.

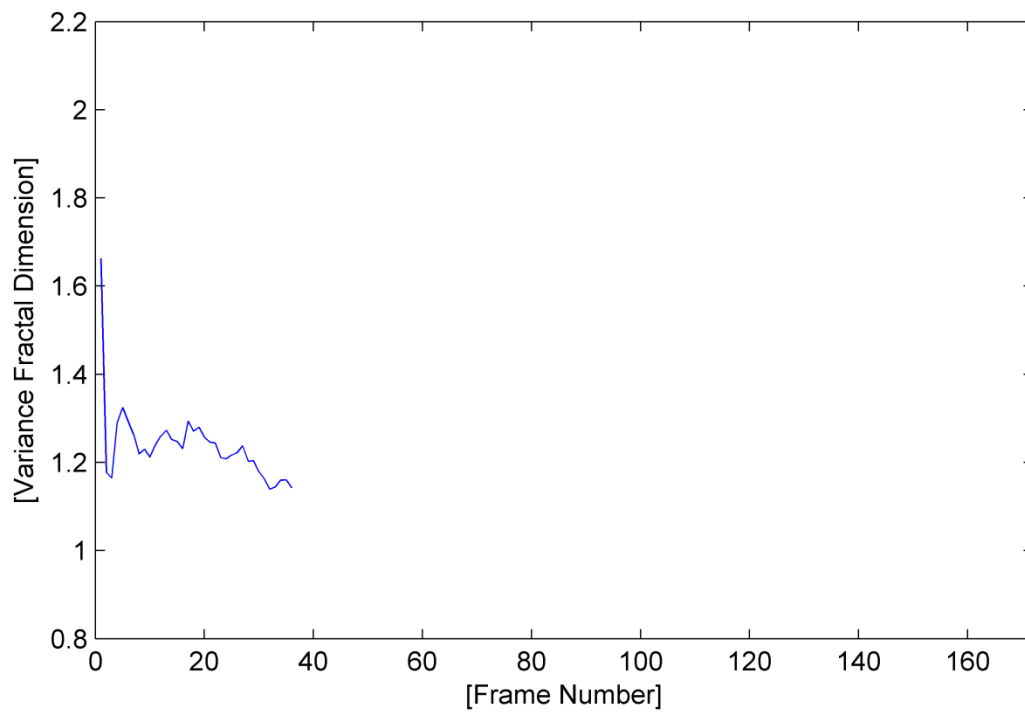


Fig. A.214. The trajectory of the utterance "bear" detected by the voice activity detection algorithm.

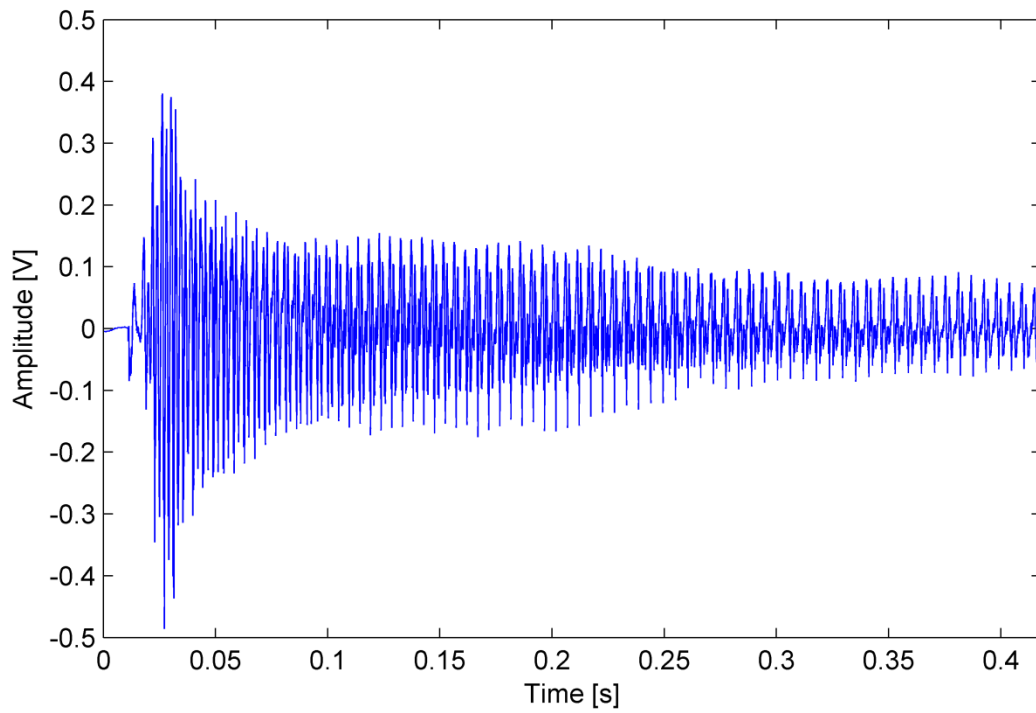


Fig. A.215. The waveform of the utterance "bear" detected by the voice activity detection algorithm.

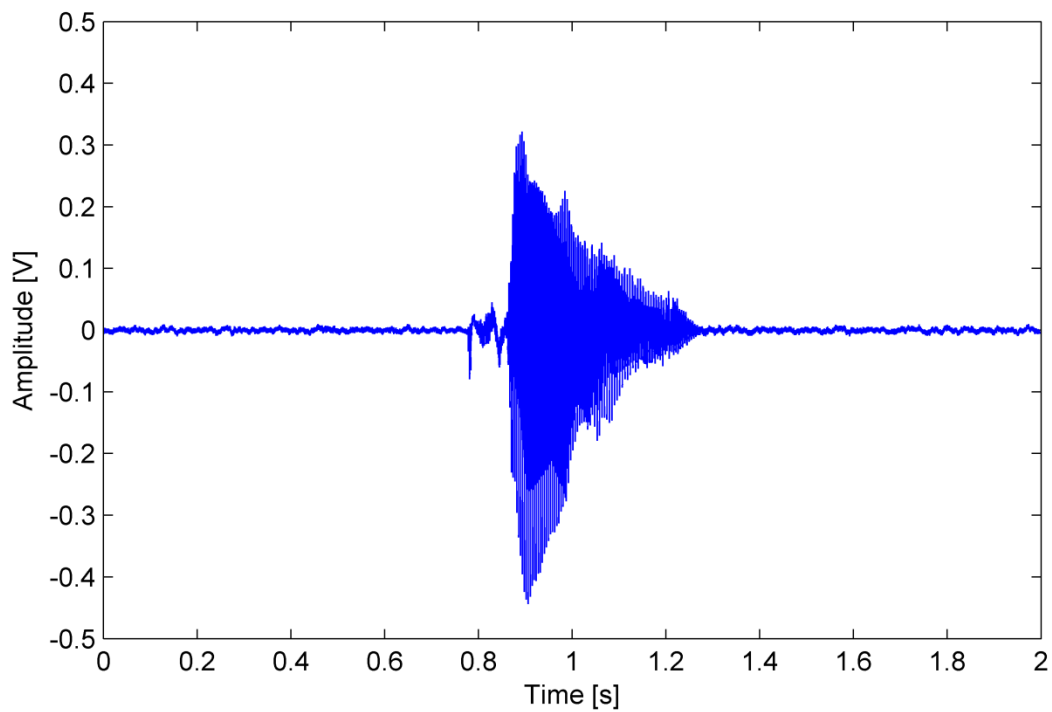


Fig. A.216. The waveform of the utterance "poor".

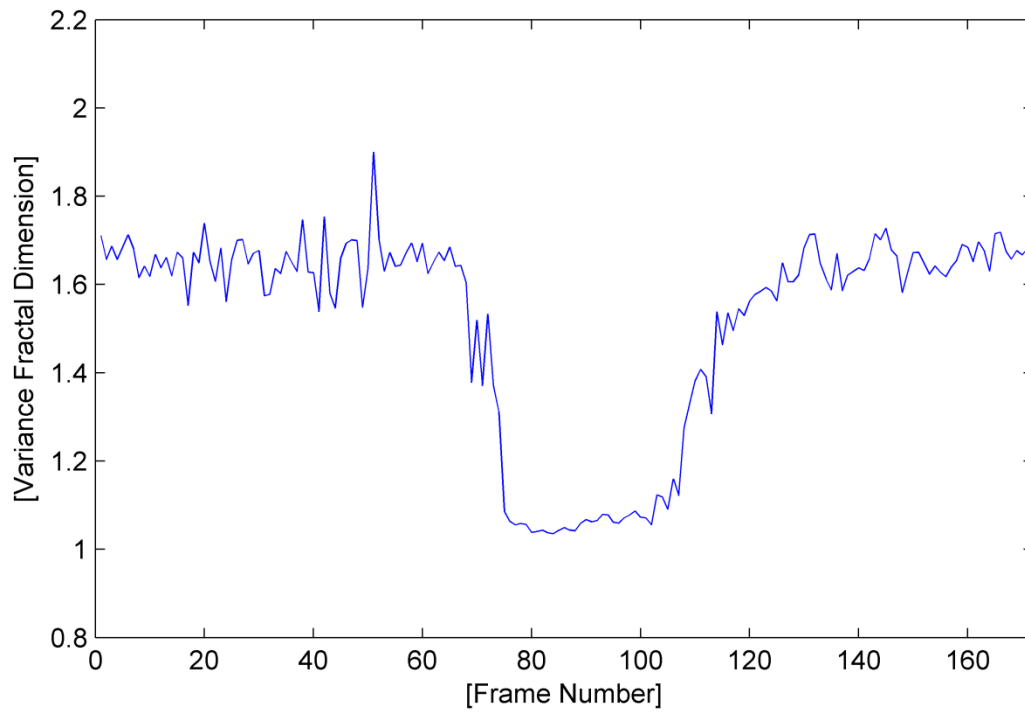


Fig. A.217. The variance fractal dimension trajectory of the utterance “poor”.

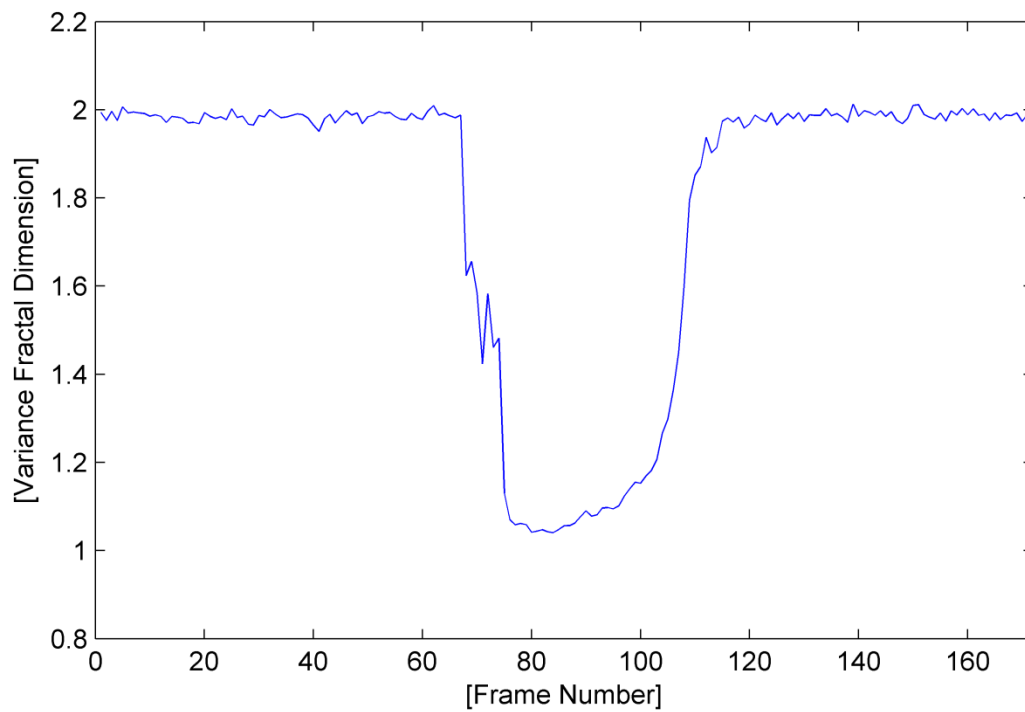


Fig. A.218. The variance fractal dimension trajectory of the utterance “poor” after addition of white noise.

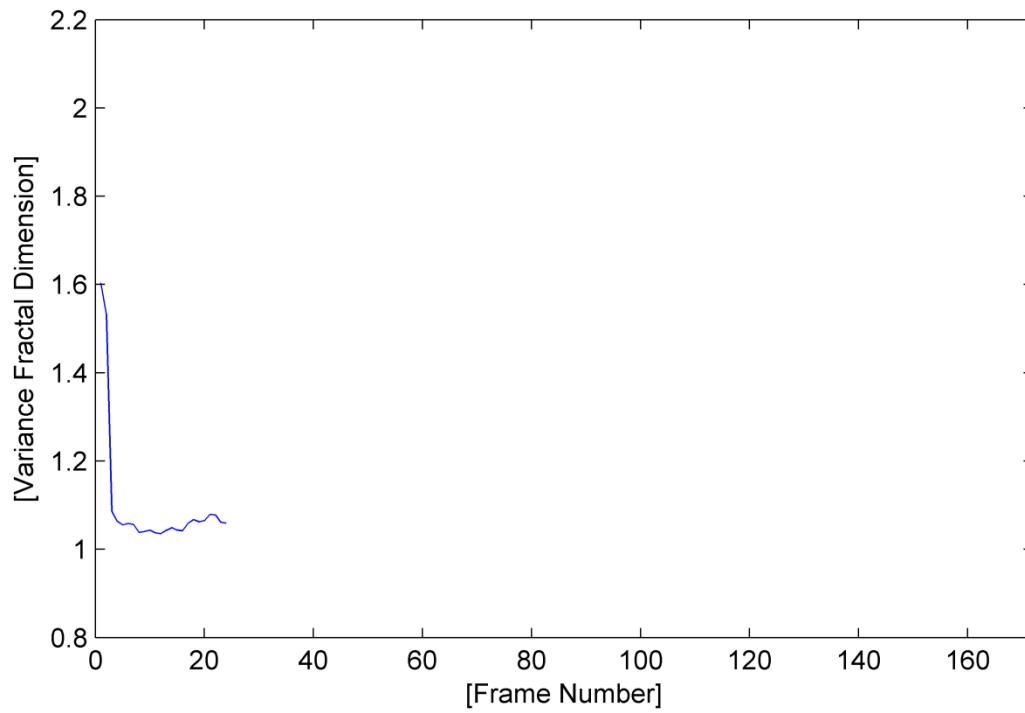


Fig. A.219. The trajectory of the utterance “poor” detected by the voice activity detection algorithm.

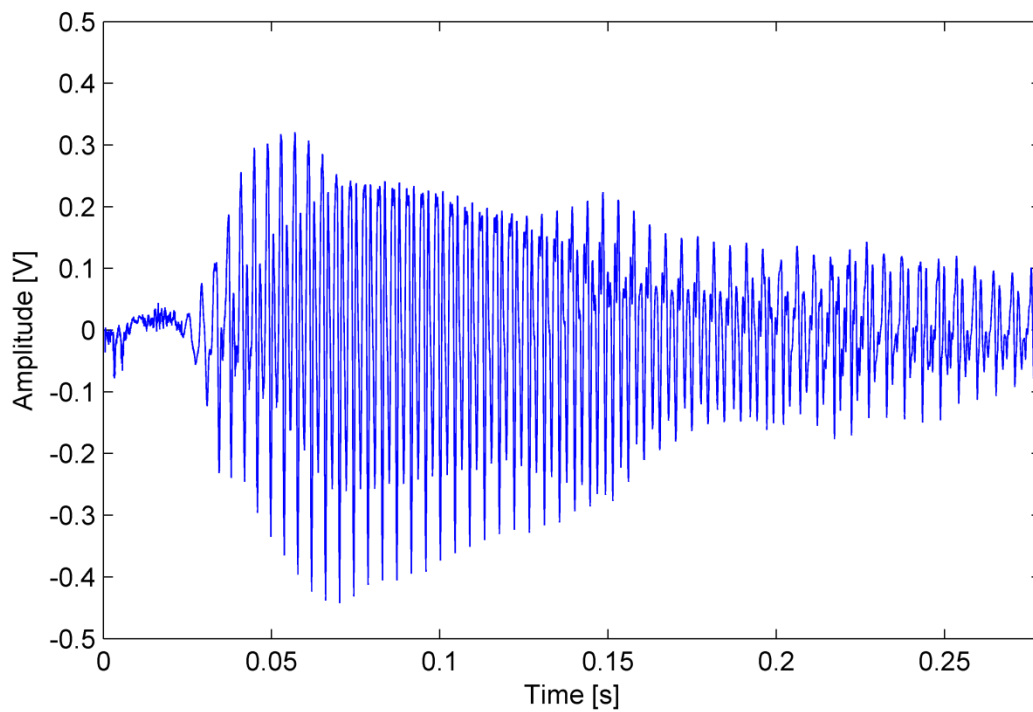


Fig. A.220. The waveform of the utterance “poor” detected by the voice activity detection algorithm.

APPENDIX B

Software Flowcharts

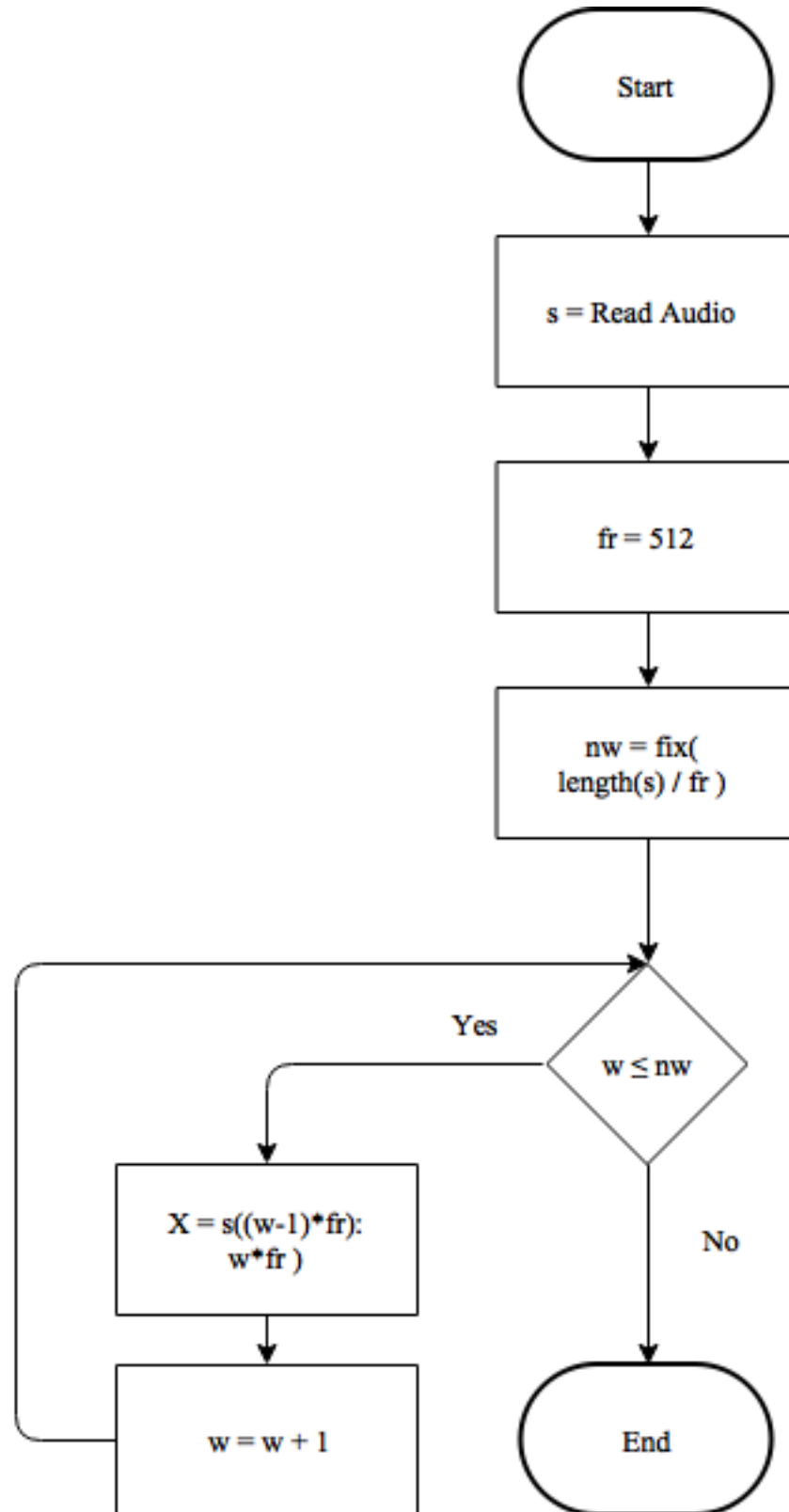


Fig. B.1. The flowchart for the framing of speech program.

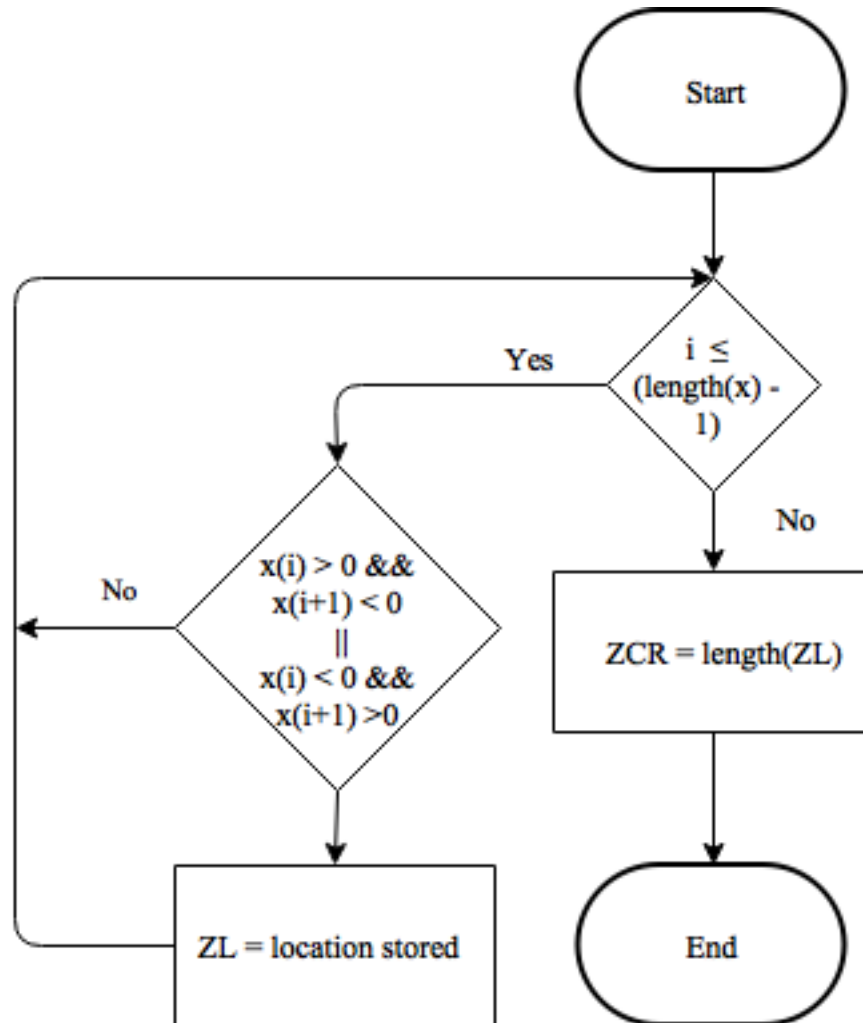


Fig. B.2. The flowchart for the zero crossing rate algorithm.

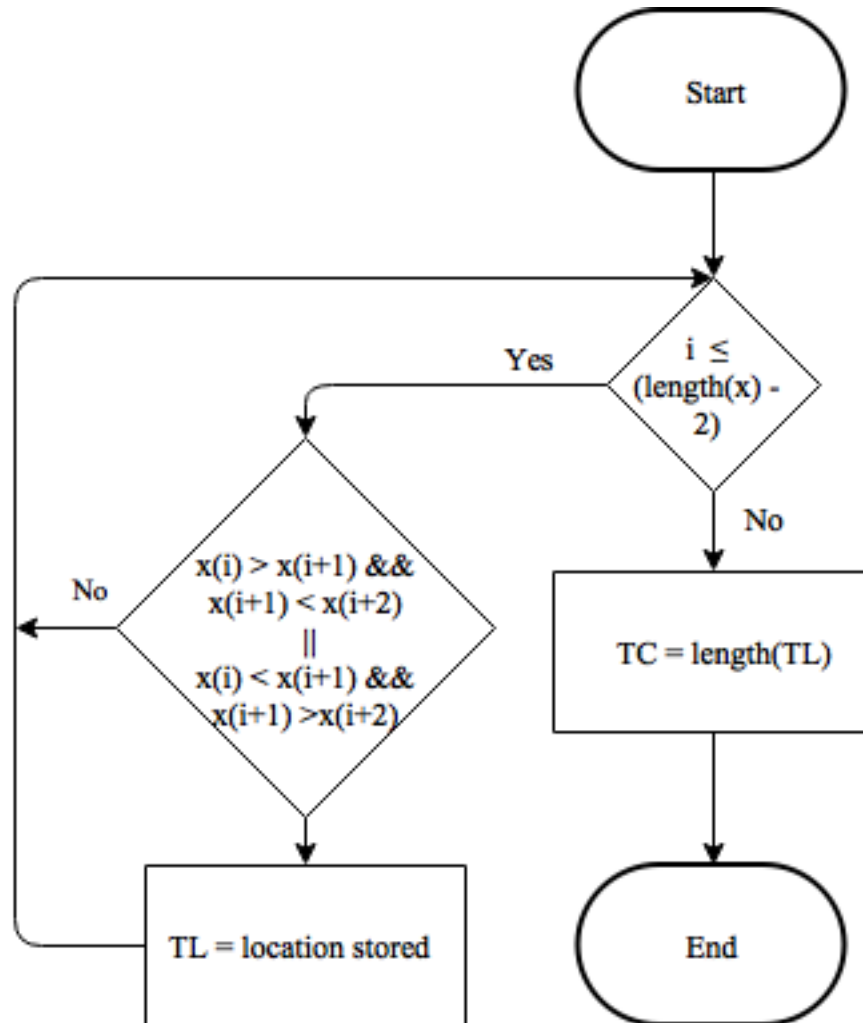


Fig. B.3. The flow chart for the turns count algorithm.

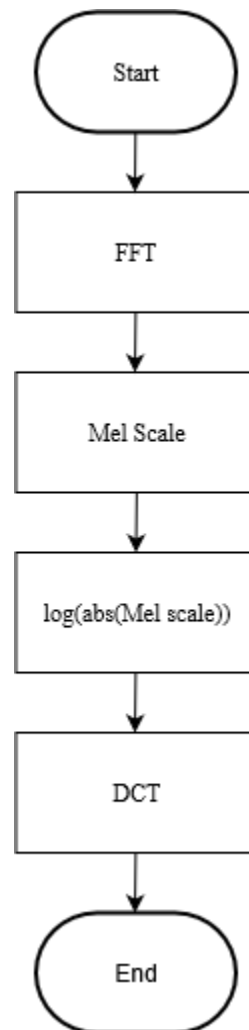


Fig. B.4. The flowchart for the Mel-frequency cepstral coefficients algorithm.

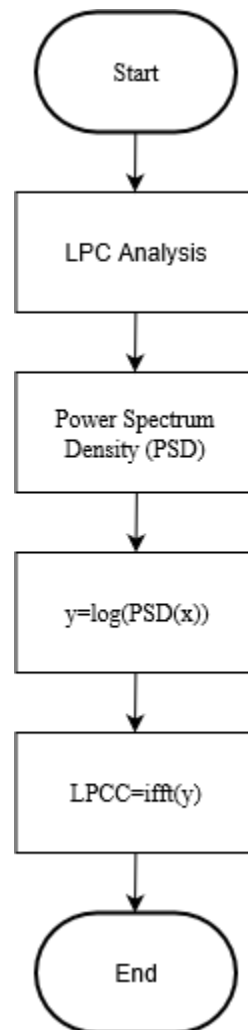


Fig. B.5. The flowchart for the linear prediction cepstral coefficients algorithm.

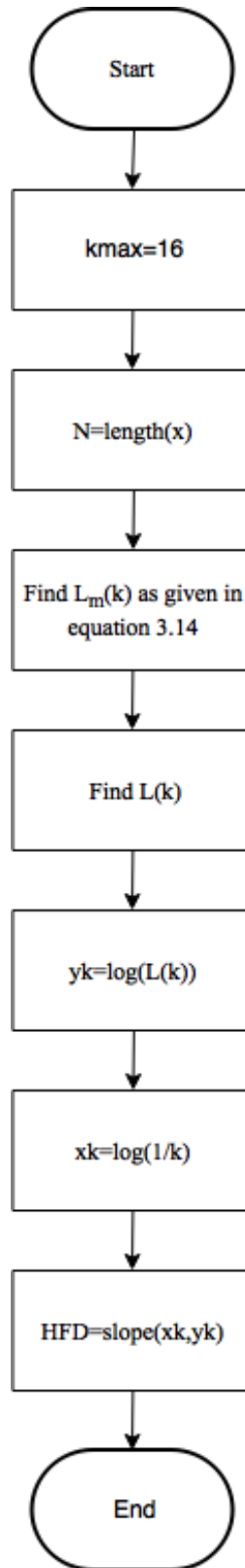


Fig. B.6. The flowchart for the Higuchi fractal dimension algorithm.

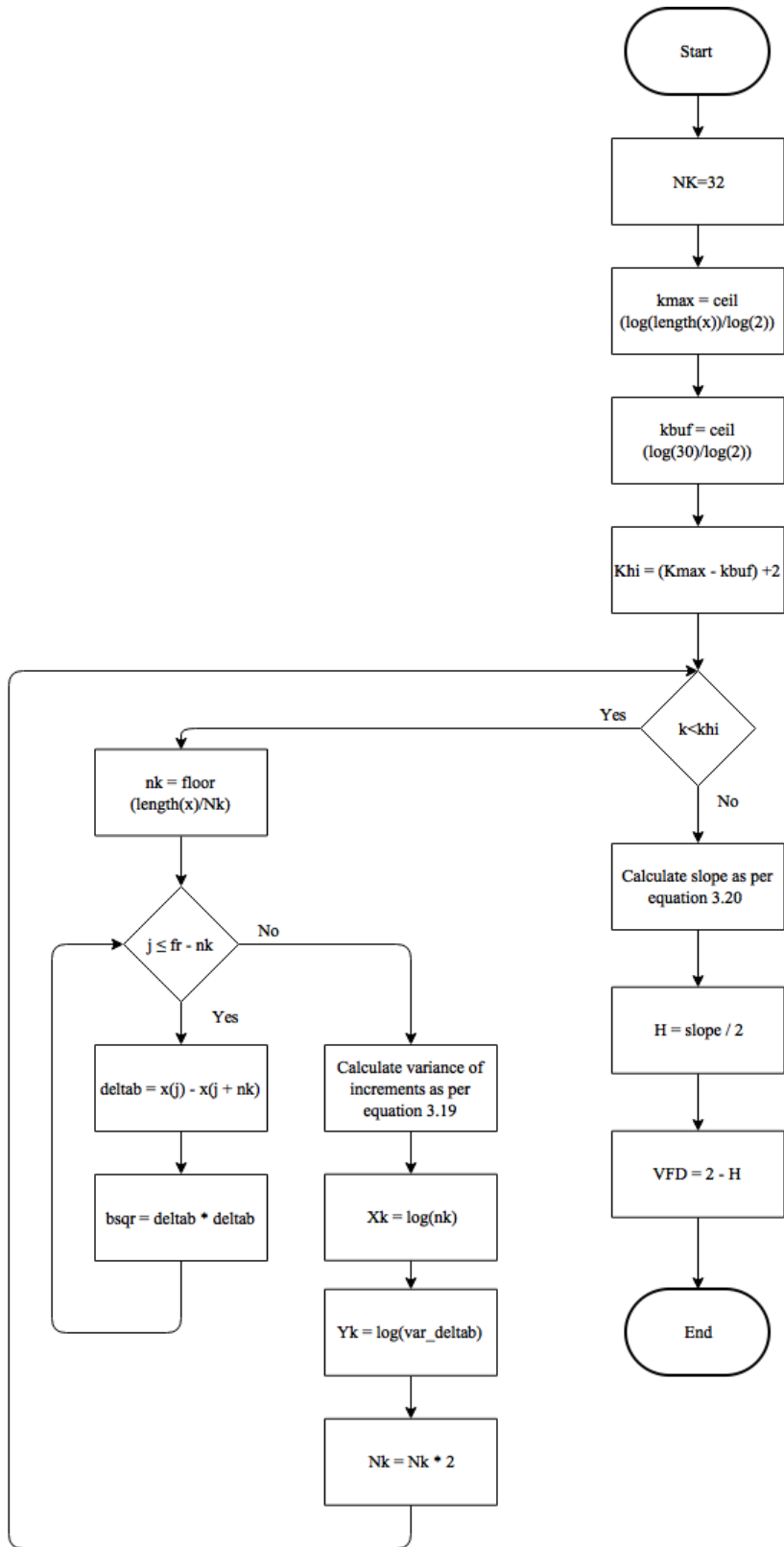


Fig. B.7. The flowchart for the variance fractal dimension algorithm.

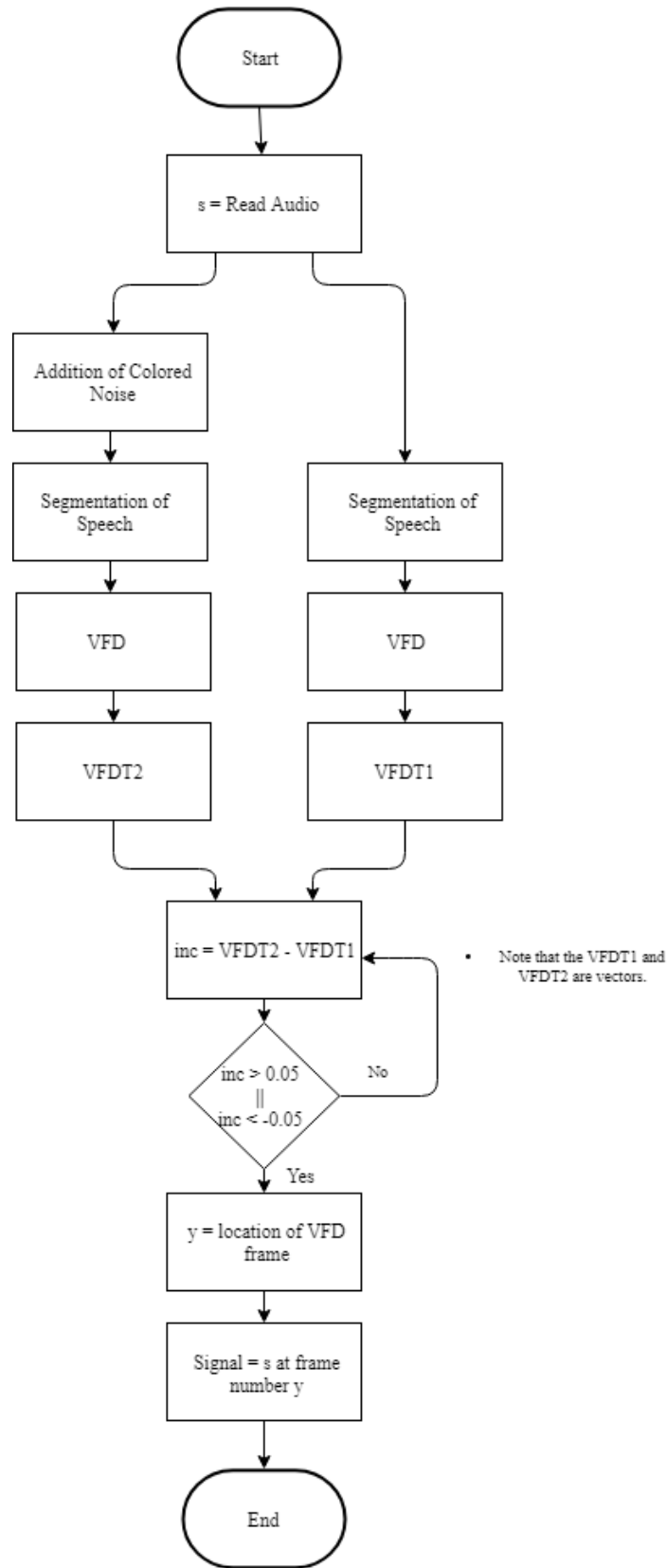


Fig. B.8. The flowchart for the voice activity detection algorithm.

APPENDIX C

Experimental Codes and Data

The recorded data for this thesis which is used for the analysis is attached to the disc. Furthermore, the experimental codes written for the algorithms discussed in chapter 3 and the experiments conducted in chapter 5 are provided on the attached disc. This includes the source code:

- To add colored noise to the signal (Noise_add.m).
- To find the zero crossing rate (ZCR.m).
- To find the turns count (TC.m).
- To find the mel-frequency cepstral coefficients (MFCC.m).

- To find the linear prediction cepstral coefficients (LPCC.m).
- To find the variance fractal dimension (VFDT.m).
- To find the Higuchi fractal dimension (HFDT.m).
- To generate the Weierstrass function (wsc.m).
- To generate the fractional Brownian motion (fBm1.m).
- The voice activity detection algorithm that utilizes the variance fractal dimension (VAD_VFD.m).
- To extract features using the voice activity detection that utilizes the variance fractal dimension (feature_extraction_VAD_VFD.m).
- To extract features using the voice activity detection that utilizes the Higuchi fractal dimension (feature_extraction_VAD_HFD.m).
- To extract features using the voice activity detection that utilizes the amplitude threshold scheme (feature_extraction_VAD_Amplitude.m).
- To extract features using the voice activity detection that utilizes the energy of the signal (feature_extraction_VAD_energy.m).
- To read the training and testing features and input to the support vector machine to build a model and test the training data. (SVM_classification.m).

APPENDIX D

Colophon

This thesis is a typeset in Microsoft Office 2007. The body is written in 12 point Times New Roman with figure and table captions displayed in 10 point Arial.

The figures published in this thesis are generated using Matlab version 2014a and Adobe Photoshop CC 2016. The flowcharts in appendix B were generated using the website www.draw.io. All the images were saved as *portable network graphics* (PNG) file.

All the work was performed using Windows 10 with a 1.6 GHZ intel Core i5 processor and 4 GB of DDR3 memory.