THE UNIVERSITY OF TOKYO

MASTER THESIS

A Comparative Study On Speech Feature Sets For Representing Phonemes

---

音素を表現するための音声特徴に関する比較研究

Author:
Dan RINGWALD

Supervisor:
Dr. Nobuaki MINEMATSU

A thesis submitted in fulfillment of the requirements for the degree of Master of Engineering

in the

Minematsu-Saito laboratory
Department of Electrical Engineering and Information Systems
Graduate School of Engineering

January 29, 2018

Declaration of Authorship

I, Dan RINGWALD, declare that this thesis titled, "A Comparative Study On Speech Feature Sets For Representing Phonemes" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

"To listen is an effort, and just to hear is no merit. A duck hears also."

Igor Stravinsky

THE UNIVERSITY OF TOKYO

Abstract

Graduate School of Engineering
Department of Electrical Engineering and Information Systems

Master of Engineering

A Comparative Study On Speech Feature Sets For Representing Phonemes

by Dan RINGWALD

The present master thesis compares two kinds of feature sets on the ground of their ability to represent phonemes in speech samples.

Namely the Mel-Frequency Cepstral Coefficients (MFCC) alone, and the MFCC set augmented by some Wavelet-Transform Based Coefficients (WTBC) are feeded into a Connectionist Temporal Classification (CTC) Automatic Speech Recognition (ASR) system. When the CTC ASR system decodes the features into a text transcript, a relative drop of 7% in the edit distance to the original transcript is observed when the WTBC set is appended to the MFCC.

When the output of the CTC ASR system is changed to the phoneme transcript of the speech samples, the difference of edit distances to the original phoneme transcripts goes up to 10%

Such results suggest that some of the phonemic information contained in the speech samples is beyond the reach of the MFCC feature set and can be at least partially transcripted by the WTBC set.

Acknowledgements

First and foremost, I address my deepest thanks to the persons without whom I would not have got the opportunity to write this master thesis. Namely, my family for the education it provided to me and the constant support it was in both good and difficult times, and my research supervisor for kindly hosting me in his laboratory and for all the pertinent comments, critics and advices he gave to me.

Then I would like to thank here the Internet, this juxtaposition of passionated invidual persons who contribute, blog post by blog post, comment by comment, to a better sharing of knowledge. I am conscious that I would not have been able to conduct one tenth of the present thesis if it were not for the presence of open source libraries, free access code on github, stackoverflow questions and answers, online tutorials ...

Contents

List of Figures

Chapter 1

Introduction

Before introducing the main hypothesis of the present master thesis (which will be stated in Section 1.3), I would like to present the main challenges and concerns hoovering around the Automatic Speech Recognition (ASR) systems and the feature sets they are using.

1.1 Speech and phoneme recognition

Speech and phoneme recognition are processes which transcript speech audio signals into some corresponding word or phoneme sequences.

Recognising and interpreting phonemes, speech units or full speech occurences is a task bearing a tremendous importance in our daily life. Speech is the backbone of most of the casual interactions between us, humans, and has a dominant role in the shaping of our cognition. Despite the fact that the usage of sonic support for information is so critical in our life, speech and phoneme recognition remains a task that is very difficult to automatize. Furthermore as O. Räsänen highlights it in his state of the art analysis on the computational modeling of phonetic and lexical learning ([17]), the understanding of phonemic and linguistic content is learned by the human children in a mainly unsupervised way, using bidirectional interaction between the infants and their environment, and involves knowledge from a broad range of domains ranging from phonetics or linguistics to signal processing. This naturally raises the question about the feasibility of the integration of all those parameters into a single automatized speech or phoneme recognition process. Indeed, pluridisciplinarity, adaptiveness, unsupervised learning and interactions with the environment are 4 objectives that are among the hardest to achieve for an automated system.

As a direct consequence, even despite a recent boost in the artificial intelligence field caused by a significant progress in artificial neural networks technologies, automatically processing speech data into the associated sequence of phonemes or words is still an active research challenge on which numerous laboratories continue to innovate on a regular basis.

1.2 The importance of the choice of the feature set in automated speech (or phonemes) recognition systems (ASR)

ASR is a process involving many tasks and using possibly different architectures depending on the considered system, but feature extraction is always the first step of the pipeline (Fig 1.1).

In a typical ASR system a digitalized speech audio signal (defined by the amplitude of the inputed sonic wave) is first preprocessed into some features. Such features are then classified into some tokens, characters or more generally some symbols thanks to an acoustic model trained with some sample data. Finally the symbols are desambigued and mapped to a word (or phoneme) sequence using a lexical and/or language model built out of a set of data.

Being the first element in the pipeline, the importance of the feature extraction part is crucial. All the information from the original signal lost during the step will be inaccessible from the following steps. However simply removing the step to transmit the original signal to the core of the ASR system is usually not advisable for three main reasons:

1. First, the primary goal of the feature extraction step is to strategically reduce the extremely high dimensionality of the original signal. Indeed, the audio signal being a sampled, digitalized time series (Fig 1.2), its dimension is equal to the number of samples per second of signal feeded as an input (i.e. 16000 dimensions per second of signal usually). Some classification methodology, like for example neural network technologies, are very resilient to high dimensionality in the input data, however, even in this specific case reducing the dimensionality to sensible bounds is always clearly beneficial to the functioning of the system.

2. In a second place it aims at decorrelating the data using a process external to the core of the ASR system. The signal is a mere sequence of values ordered in time and the sound at each time can't be linked uniquely to the value of the series at that time. When a property considered local (such as phonemic information) depends that much on the values of the signal around the precise instant at which the property is evaluated, direct analysis is very tough and the preprocessing performed by the feature extraction step is valuable.

   It is however worth mentioning that recently some research (like for example [15]) has been led in order to perform some speech recognition directly out of the raw speech signal without any kind of preprocessing. In such research the employed ASR systems usually rely on a Neural Network core, as the high dimensionality of the input has only a limited impact on the functioning of the system. The main idea making the approach feasible is the usage of some convolutional layers directly on

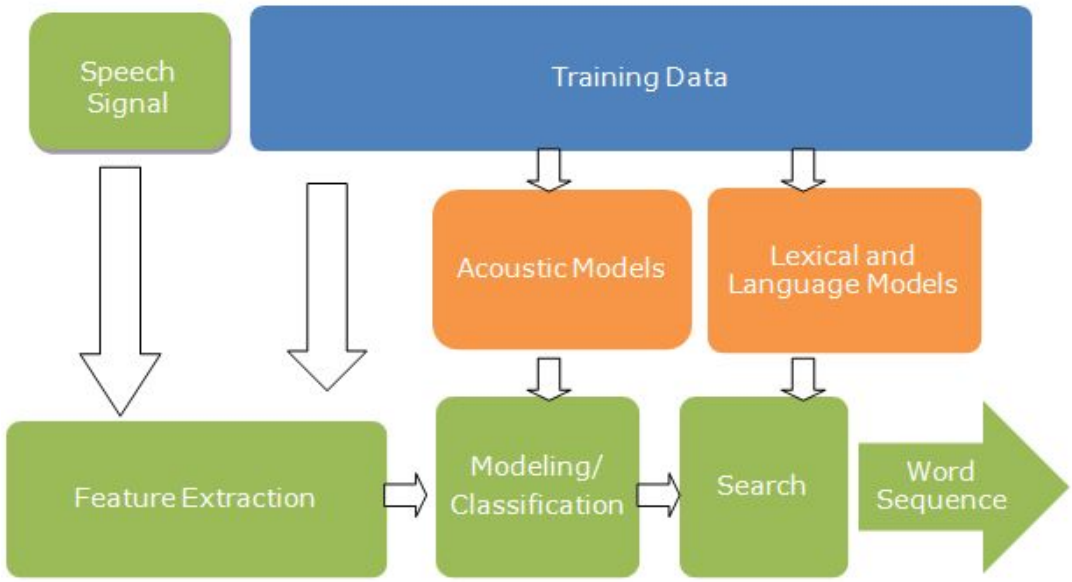

FIGURE 1.1: ASR processing pipeline

top of the neural network input. These layers function more or less like a feature extractor by computing an intermediate representation of the signal based on the convolution of the original signal with some optimized filters.

3. Finally the features should preferentially express the data in a meaningfull and interpretable way, in order to get an easy understanding of the results yielded by all further analysis. For example the Chapters 2 and 3 will highlight how the usage of the Fourier transform helps to get features directly associated with the physical phenomenon producing the vowels.

1.3 A better representation of consonants

I defend in the present master thesis the hypothesis that the feature sets currently used in the state of the art ASR systems can be improved in order to give a better description of consonants.

I therefore set as a goal to prove this claim and to quantify the improvement in recognition rate induced on simple ASR systems by the introduction of more consonant-sensitive features.

Such improvements would have a significant impact from a scientific point of view, expanding the knowledge we have of the properties and parameters which define a consonant. It would in this regard synergize deeply with some existing works (Notably in [14]) about the automatic classification of segmented speech sounds into a finite number of acoustic categories. Also,

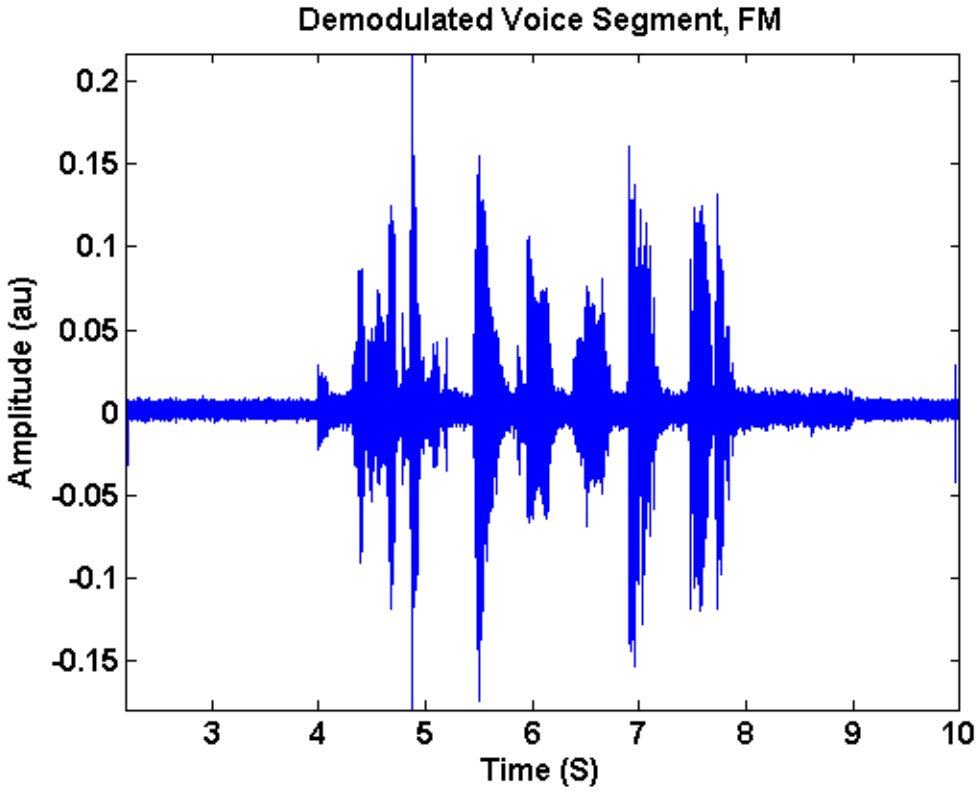

FIGURE 1.2: A speech audio signal example

if variations around a single consonant phoneme corresponding to regional variants of a language (i.e. Scottish and British english to name an example) are considered as distinct categories instead of subcategories of a single phonemic category, the present research may provide better feature sets to the research around the detection, understanding and synthesis of regional accents.

From a point of view closer to the industry world, a better description of consonants can lead to various applications. Today, modifying the aspect of a vowel by changing properties such as its fundamental and harmonics is considered an easy task. However similar easily accessible parameters are harder to find for consonants and a better and more meaningfull representation of consonants can pave the way for voice-morphing technologies (with similar quality to the recent results in image morphing), or finer voice recognition technologies.

1.4 Structure of the argumentation

Following this short introduction (Chapter 1), I will present:

- The usual modelisation used for analysing speech signal. (Chapter 2)
- The two main feature sets I am commited to compare and how they are linked to the previously detailed modelisation. (Chapter 3)
- I will compare the two main feature sets using a simplistic but visual PCA analysis. (Chapter 4)
- I will introduce the architecture of the ASR system I am using to compare the feature sets in a more integrated context, and disclose the results regarding the performance of the two feature sets. (Chapter 5)
- Finally, I will conclude this master thesis. (Chapter 6)

Chapter 2

The Modelisation

In order to understand the principles guiding the design of the feature sets for speech recognition, let's take a look at how voice is produced and where speech information is located in the voice signal.

The sounds in the human voice can be divided into two main categories when it comes to the physical mechanics used to produce them: voiced sounds and unvoiced sounds. To understand the difference between both let's refer to the anatomic figure 2.1. In the case of voiced sounds, a vibration is initiated by the vocal folds in the larynx. This vibration resonates into the whole vocal tract before going out through the mouth. Changes in the shape of the vocal tract induce different voiced sounds, as the various shapes of the cavities inside our mouth and throat generate different harmonics out of the vibration generated by the vocal folds. The voiced sounds are vowel-like sounds. However some consonants are also associated to some voiced sounds, for example the english ”r” is a voiced sound and the ”g” sound has a vocal component.

In the case of unvoiced sound, there is no vibration of the vocal folds and no resonating phenomenon. The flow of air is simply opposed by the obstruction (partial or complete) of some part of the vocal tract. Most of the consonants have an unvoiced component. The ”s” being completely unvoiced with an obstruction just behind the teeth on the hard palate ([3] on the figure 2.1) and the ”g” still having a voiced component but featuring an obstruction at the back of the tong on the soft palate ([13] on [4] on the figure 2.1).

2.1 Voiced sounds

In the case of voiced sounds the traditional modelisation is quite elementary. Usually the vocal tract is modeled as a sequence of communicating pipes (figure 2.2). This association between pipe shapes and vowels is very old and already evocated in Sir Richard Paget's "theory of the nature of human speech" in the 1920's. Each pipe represents a different cavity in our vocal tract and the vibration produced by the vocal chords is then convoluted with the acoustic filters associated with each one of the pipes. According to this model, to each combination of shapes of the pipes corresponds a different vowel sound, and this is experimentally confirmed.

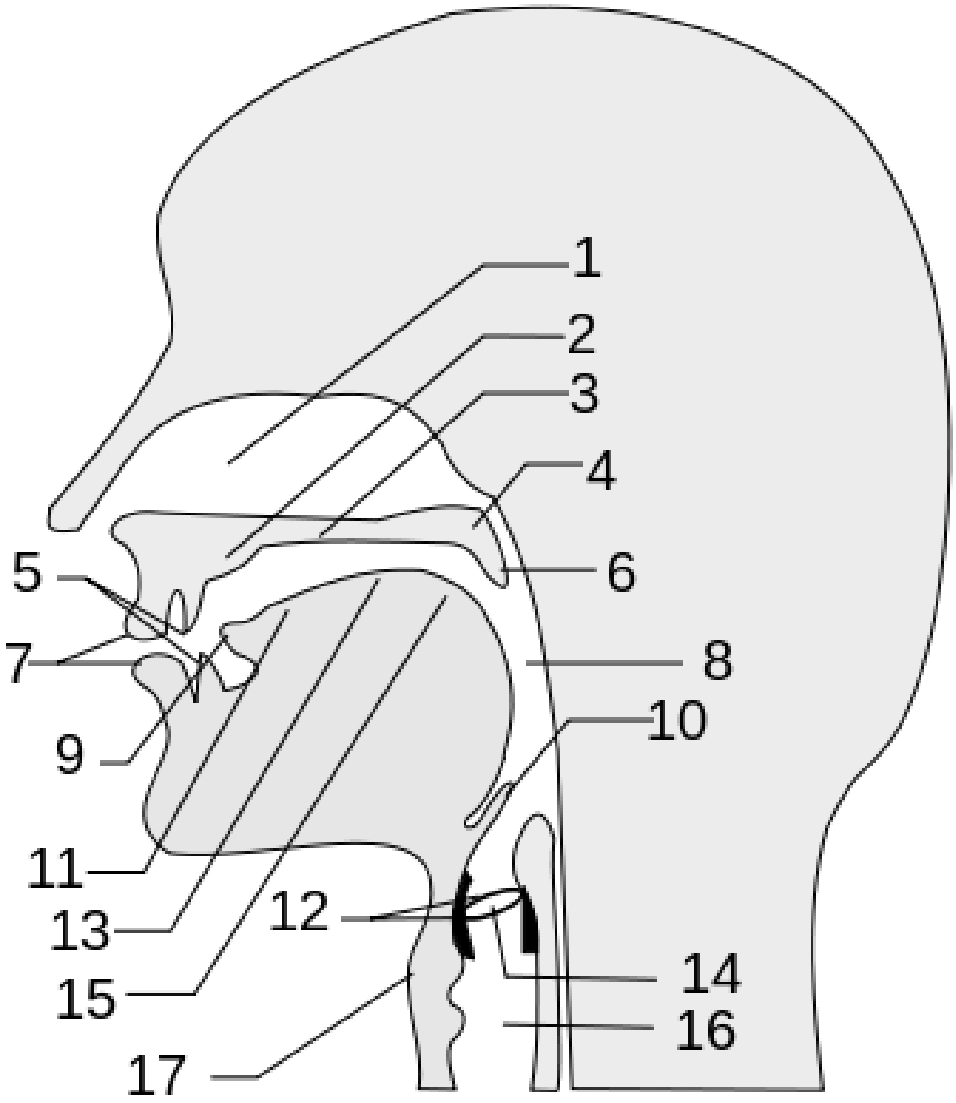

FIGURE 2.1: The vocal tract (image from sound-physics.ius.edu): [1] nasal cavity, [2] oral cavity, [3] hard palate, [4] soft palate, [5] teeth, [6] uvula, [7] lips, [8] pharynx, [9, 11, 13, 15] tongue, [10] epiglottis, [12] vocal cords, [14] glottis, [16] trachea, [17] larynx.

From the point of view of speech recognition, recognizing a voiced sound is equivalent to determining the fundamental and harmonics produced by each one of the pipes.

2.2 Unvoiced sounds

Unvoiced sounds are much more diverse and difficult to model than voiced sounds. They are traditionaly divided into categories, each one following

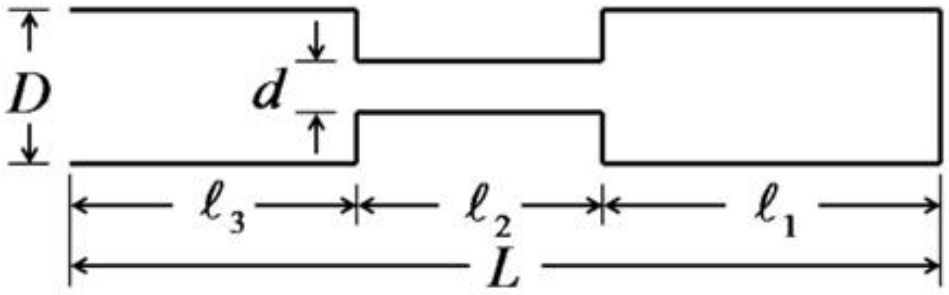

FIGURE 2.2: The pipe modelisation of the vocal tract

a different pronunciation pattern. Naming a few of them, there are "plosives" ("t", "p" etc...) where the flow of air is briefly interrupted before being suddenly released, "fricatives ("s" "ch" etc...) where the flow of air is forced through a nearly totaly obstructed path producing a friction sound ... Usually the only model used is the voiced sound model. The reason why it can still apply to unvoiced sounds is that, mathematically speaking, every periodic signal can be decomposed into a series of harmonic signals (through the Fourier transform). Despite the consonant sounds being far from periodic, they can be considered as locally periodic if the analysis of the sound is performed on only small windows of the signal. Indeed, if only a small window is analysed, the remaining signal can be fictively changed to the mere repetition on the obtained windowed signal. Consonants are then described on a local scale as a the superposition of harmonics that would have been obtain if the analysed portion of signal were periodic. Even if this decomposition has less meaning physically speaking for consonants than for vowels, the results obtained with this modelisation are still good enough to power state of the art Automatic Speech Recognition systems.

Chapter 3

The Feature Sets

3.1 Mel-Frequency Cepstral Coefficents (MFCC): The traditional approach

According to the "pipes" model described in the previous chapter (Chapter 2), the shape of the vocal tract determines the sound coming out. Furthermore, the shape of the vocal tract manifests itself in the envelope of the short time power spectrum (physical property of the linear filtering applied by the vocal tract) and the purpose of the MFCCs is to accurately and succinctly represent this envelope. MFCCs were introduced by Davis and Mermelstein in 1980 [4] and their perfomance lead them to be considered the state-of-the-art for a long time. Previous features before MFCCs included Linear Prediction Coefficients (LPCs) and Lineal Prediction Cepstral Coefficients (LPCCs) but they will be left out in this thesis for the purpose of simplicity.

In this section I explain in details the computing of the Mel-Frequency Cepstral Coefficients, which are used throughout the present thesis. The computing goes through the following steps, which I will explain one by one in a dedicated sub-section:

1. Frame the signal into short frames
2. Compute the power spectrum of each frame
3. Apply the mel filterbank (detailed later) to the power spectra
4. Sum the energy in each channel obtained through the mel filterbank
5. Take the logarithm of all the obtained energies
6. Take the Discreet Cosine Transform (DCT) of the log energy of each channel
7. Keep the coefficients comming from the channels from 2 to 13 and discard the rest
8. Some more feature are also usually produced by taking the derivatives of the MFCCs over time, the first derivatives are called deltas and the second derivatives are called double-deltas or delta-deltas.

3.1.1 Framing

Framing is the action of limiting a signal to a window in time.

There is two reasons for framing the signal before applying the Fourier transform:

- First the signal varies over time. As the Fourier Transform is not localized in time, the signal must be framed to isolate the part we want to analize.
- The other reason is more mathematically justified. The signal is not periodic despite the Fourier transform requiring it to be so. Framing the signal enables the Fourier transform to consider the frame as a period and go around the problem caused by the non-periodicity of voice. This consideration is perticularily pertinent when it comes to unvoiced sounds. Indeed voiced sounds being produced by linear filtering are roughly periodic, however the unvoiced part of the voice being "noise-like" has no periodic behavior whatsoever.

  This second reason justifies the usual frame border smoothing applied on the frames. As the Fourier transform assumes the frame to be a period of a fictive periodic signal, the continuity of the fictive periodic signal imposes the borders of the frames to be nul.

3.1.2 Fourier Transform

The next step is to compute the power spectrum of each frame. This is motivated by the "pipe" model of the production of the speech. According to the model the information of the voice is contained in the filtering of the vibration produced by our vocal folds through our vocal tract, and the Fourier Transform is the best known tool when it comes to analysing linear filtering.

The fact that we compute the power spectrum instead of the regular complex spectrum is justified by the fact that the human ear perceives the power of the audio signal and is completely indifferent to the phase. This is due to the human cochlea (an organ in the ear) which vibrates at different spots according to the frequency of the incoming sound and at different strength according to the power carried by thoses respective frequency.

3.1.3 Mel Filterbank

The power spectrogram still contains a lot of information not required for Automatic Speech Recognition (ASR). In particular, our ear cannot discern the difference between closely spaced frequencies, and this effect increases as the frequency increases (our perception of the pitch is logarithmic). Hence, bands of closely spaced frequencies behave like a single frequency and it would be convenient if each one was separated from the other. That is what the Mel Filterbank does.

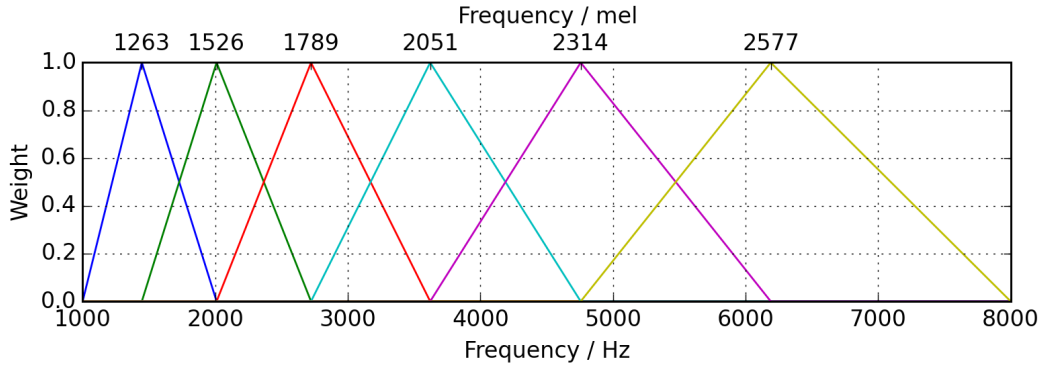

FIGURE 3.1: The Mel Filterbank (from http://siggigue.github.io/pyfilterbank/melbank.html)

Some overlapping triangular filters, logarithmically distributed, filters some bands of frequencies (named channels) out [fig 3.1]. Hereafter each one of the channels will become an independent feature and will be processed independently

3.1.4 Energy aggregation

Up until now, no information was lost. The Fourier transform and the Mel-Frequency filterbank were, mathematically speaking, injective transformation and the information carried by our channels is no less than the information carried by the original signal. However as our ear groups together similar frequencies, the next step will get rid of the distribution of energies inside each channels and reduce each channel to a single number: the sum of the energy contained on each frequencies covered by the channel.

Here the informational loss is huge. The dimensionality of the data carried by the channels at each time just got reduced from an innumerable infinity (discretized to the number of samples in our frame, i.e. around 250) to the number of channels (usually around 12)

3.1.5 Log-Energy

The next step is justified by the fact that our perception of the energy contained in a channel is also logarithmic. According to this consideration, we take the logarithm of the aggregated energy in each channel. Furthermore, according to the pipe model of the vocal tract, speech can be considered as a succession of acoustic filters convoluted on a vibration generated by the vocal folds. After applying the Fourier transform the convolutions express themselves as multiplications in the frequency domain. Hence, taking the logarithm of the resulting energy causes the effect of the convoluted filters to behave additively.

3.1.6 Discreet Cosine Transform (DCT)

The final step is to compute the DCT of the log-energies of each channel, and there is two main reasons for this step.

- First the Mel filterbank uses overlapping and the channels' energies are correlated one with each other. The DCT decorrelates the energies (hence decorrelating the features) facilitating the classification step which will use those features.
- Then the highest coefficients in the output of the DCT correspond to fast changes of the channel energies and keeping them actually degrade ASR performance, they are usually discarded.DCT is then a way to filter those "noises" out.

3.1.7 Deltas

Finally, one has to keep in mind that in the first place our model was a static model describing a resonating process. But what about unvoiced sounds, transient sounds and consonants? In the section on modeling we argued that mathematically speaking these sounds would be represented by the evolution in time of the spectrogram. The MFCC coefficients being closely related to the spectral domain this assertion still holds and Furui, a Japanese researcher, introduced in 1986 the concept of "deltas" of the MFCCs [6]. "deltas" and "double-deltas" are the derivative and second derivative of the MFCCs.

These derivatives aim at describing the dynamic behaviour of the voice which was not taken into account in our model. An experiment from Hossan, Memon and Gregory [8] have shown that this differential approach led the success rate of speech recognition from 90% to 96% percent when it comes to the number of correctly understood words (Gaussian mixture Model was used for the classification). It corresponds to reducing the error rate by 60%! Even if the MFCC feature set approach coupled with the delta coefficients performs well in usual applications, it has some lacks, mainly due to the nature of the model it is based upon.

3.1.8 Conclusion on MFCC

The "pipes" model is a static model (the signal convolutions induced by the pipes are performed with functions which are periodic in time), and even if the "delta" approach tries to go around its limitation by performing what can be called a quasi-static approach (analysing the derivatives of a static approach) some articles like [11] point at these lacks. Mainly, the suggested solutions (Notably in the articles [5], [11], [22], [20], [13]) are a quite standard solution to transient and non-linear signal analysis: the Wavelet Analysis.

3.2 The Wavelet Approach

Wavelets are the standard solution to the analysis of transient and noisy signals. The Wavelet Transform has been used an incredibly high number of times in tasks like denoising, classification of noises, noise recognition, noisy signal compression etc...

To get a grasp of the wide reach of the applications of Wavelets I highly recommend to read the papers of S. Mallat and his world famous "A theory for multiresolution signal decomposition: the wavelet representation" ([12]), which had repercussions in each and every research field concerning non-harmonic serial data.

In this section I extensively use the work of gwyddion.net in order to give an introduction to the two main wavelet transforms: The Discrete Wavelet Transform (DWT) and the Continuous Wavelet Transform (CWT).

3.2.1 Introduction

The wavelet transform is similar to the windowed Fourier transform, but with a completely different set of convoluted functions. When the Fourier transform decomposes the signal into sines and cosines, the wavelet transform uses functions that are localized in both the real and Fourier space. Generally, the wavelet transform can be expressed by the following equation:

$$\int_{-\infty}^{\infty} f(x)\psi^*_{(a,b)}(x)dx$$

where the * is the complex conjugate symbol and function ψ is some function. This function can be chosen arbitrarily provided that it obeys certain rules.

The Wavelet transform is hence an infinite set of various transforms, depending on the functions used for convoltion. This is the reason why the term “wavelet transform” is used in very different situations and applications. There are also many ways to sort the types of the wavelet transforms. Here only a distinction based on the wavelet set orthogonality will be considered: orthogonal wavelets for discrete wavelet transform development and non-orthogonal wavelets for continuous wavelet transform development. These two transforms have the following properties:

1. The discrete wavelet transform returns a data vector which length is the same than the input. This corresponds to the fact that it decomposes into a set of wavelets (functions) that are orthogonal to its translations and scalings. Therefore such a signal is decomposed to a same or lower number of wavelet coefficient spectrum than the number of signal data points. Such a wavelet spectrum is very good for signal processing and compression, for example, because there is no redundancy in the output of the transform.

2. The continuous wavelet transform in contrary returns an array one dimension larger than the input data. For a 1D data we obtain an image

of the time-frequency plane. As here is used a non-orthogonal set of wavelets, data in the output is highly correlated and there is a lot of redundancy. The output being a 2D image for a 1D signal, the transform is similar to other kinds of spectra and this helps to see the results in a more human-friendly form.

3.2.2 The Discrete Wavelet Transform (DWT)

The discrete wavelet transform (DWT) is an implementation of the wavelet transform using a discrete set of the wavelet scales and translations obeying some defined rules. In other words, this transform decomposes the signal into mutually orthogonal set of wavelets, which is the main difference from the continuous wavelet transform (CWT), or its implementation for the discrete time series sometimes called discrete-time continuous wavelet transform (DT-CWT).

The wavelet can be constructed from a scaling function which describes its scaling properties. The restriction that the scaling functions must be orthogonal to its discrete translations implies some mathematical conditions on the dilation equation :

$$(x) = \sum_{k=-\infty}^{\infty} a_k \phi(Sx - k)T$$

where S is a scaling factor (usually chosen as 2). Moreover, the area between the function must be normalized and scaling function must be orthogonal to its integer translations, i.e. :

$$\int_{-\infty}^{\infty} \phi(x)\phi(x + l)dx = \delta_{0,l}$$

After introducing some more conditions (as the restrictions above does not produce a unique solution) these equations yield a unique result (i.e. the finite set of coefficients a_k that define the scaling function and also the wavelet). The wavelet is obtained from the scaling function as N where N is an even integer. The set of wavelets then forms an orthonormal basis which we use to decompose the signal. Usually only few of the coefficients a_k are nonzero, which simplifies the calculations.

In the following figure, some wavelet scaling functions and wavelets are plotted. The most known family of orthonormal wavelets is the family of Daubechies. Her wavelets are usually denominated by the number of nonzero coefficients a_k, so we usually talk about Daubechies 4, Daubechies 6, etc. wavelets. In the present thesis, the series Daubechies 6 is used. Roughly said, with the increasing number of wavelet coefficients the functions become smoother. See the comparison of wavelets Daubechies 4 and 20 below. Another mentioned wavelet is the simplest one, the Haar wavelet, which uses a box function as the scaling function.

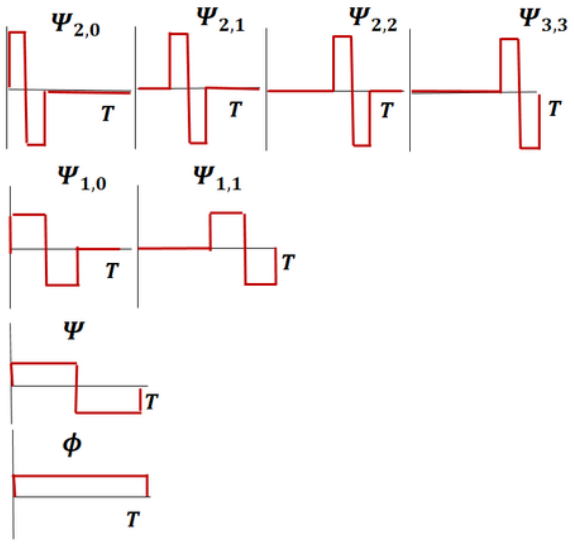

FIGURE 3.2: The Haar scaling function (ϕ), mother wavelet (ψ) and wavelet series ($\psi_{(i,j)}$). (from the publication of M. Chafii [3])

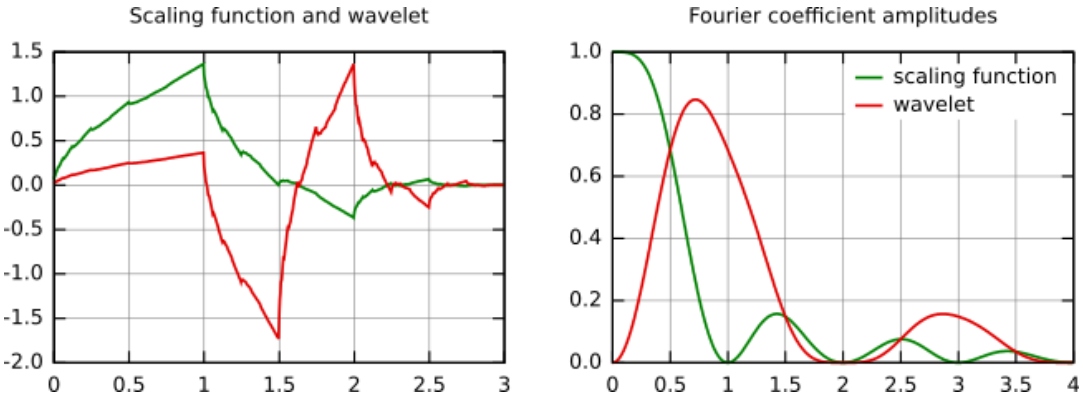

FIGURE 3.3: Daubechies 4 scaling function and wavelet (left) and their frequency content (right).

Discrete wavelet transform can be used for example for easy and fast denoising of a noisy signal. If only a limited number of the highest coefficients are taken out of the discrete wavelet transform spectrum, and an inverse transform is performed (with the same wavelet basis), The reconstituted signal is more or less denoised.

3.2.3 Continous Wavelet transform

Continuous wavelet transform (CWT) is an implementation of the wavelet transform using arbitrary scales and almost arbitrary wavelets. The wavelets

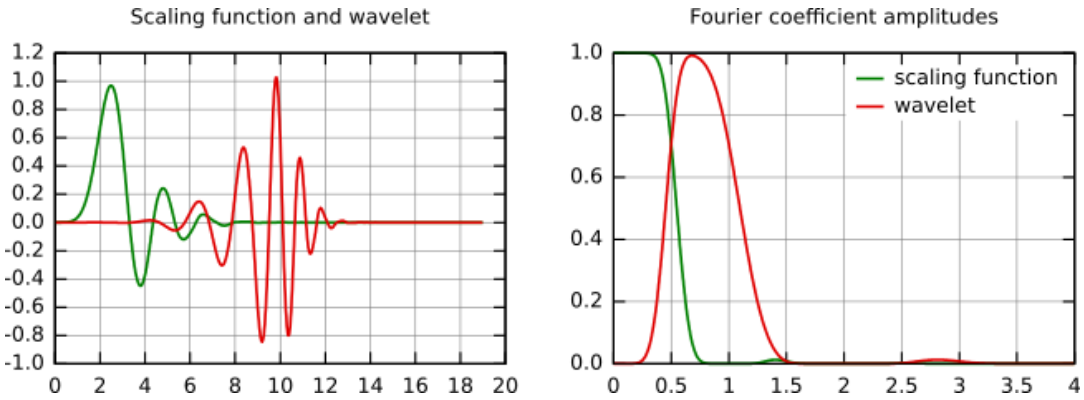

FIGURE 3.4: Daubechies 20 scaling function and wavelet (left) and their frequency content (right).

used are not orthogonal and the data obtained by this transform is highly correlated. It can be used for discrete time series as well, but with the limitation that the smallest wavelet translations must be equal to the data sampling. This is sometimes called Discrete Time Continuous Wavelet Transform (DT-CWT) and it is the most used way of computing CWT in real applications.

In principle the continuous wavelet transform works by using directly the definition of the wavelet transform, the output being a convolution of the signal with the scaled wavelet. For each scale an array of the same length N as the signal's length is obtained. By using M arbitrarily chosen scales a field NxM that represents the time-frequency plane directly is yielded. The algorithm used for this computation can be based on a direct convolution or on a convolution by means of multiplication in Fourier space (this is sometimes called Fast Wavelet Transform).

The choice of the wavelet that is used for time-frequency decomposition is the most important thing. By this choice we can influence the time and frequency resolution of the result. We cannot change the main features of WT by this way (low frequencies have good frequency and bad time resolution; high frequencies have good time and bad frequency resolution), but we can somehow increase the total frequency of total time resolution. This is directly proportional to the width of the used wavelet in real and Fourier space. If we use the Morlet wavelet for example (real part – damped cosine function) we can expect high frequency resolution as such a wavelet is very well localized in frequencies. In contrary, using Derivative of Gaussian (DOG) wavelet will result in good time localization, but poor one in frequencies.

3.2.4 A brief theoretical comparison of the Wavelet Transform with the Fourier Transform

Comparing the Wavelet Transform to the Fourier Transform yields the following considerations:

- Wavelets, unlike cosine and sine, are localized in time. They are therefore well suited to describe transient signals (typically a 'p' or 't' sound) and at a lack when describing periodic signals (typically a 'a'). This property led them to be used for example in the analysis of the noise of industrial machines (such analysis can be used to predict the odds of the machine breaking in the near futur).

- Wavelets take as parameters time shift and scale. The scale parameter make them well suited to describes noises which are generally random processes caracterized by their statistical distribution at each pitch scale. This property led them to be used in the compression of noisy signals (like for example as an alternative to mp3 compression on drum recordings)

- The Wavelet Transform belongs to the so-called constant-Q transformation family, which is caracterized by a scaling of the frequency resolution of the transform which is similar to the way our ears perceive sounds. As an example, the log-scaling Mel frequency filterbank is equally constant-Q. This property is very interesting in speech analysis, as it enables the compression of the information in a way that the lost information is very similar to the information lost by our own ears in the hearing process

Such properties make the wavelets an interesting candidate to be used for describing the unvoiced and transient parts of voice signals, namely the 'consonant' parts.

3.2.5 Wavelet Transform for feature extraction in speech signals: the state of the art.

Despite the number of papers on the usage of wavelets for denoising ([9], [2], [19], [10], [18]) or speech segmentation (differentiating types of speech or detecting the presence of speech) being astonishingly high ([21], [1], [24], [23]), papers discussing its usages in feature extraction are scarce. Futhermore, they are far from being major papers and the experimentations they conduct present some biases. For example, like in [20], experimentations are often conducted on a task implying the classification of full words, which is a biased task. Indeed, as the strong point of wavelets lays in the recognition of transients and noises, and the strong point of the MFCC feature set lays in the recognition of resonances, and as almost every words contains both kind of sounds the experimentation cannot bring decisive conclusion on the performance of the wavelet analysis on transients and noisy sounds. Even if wavelets performed better on unvoiced segments of the word, if MFCC performs better on voiced segments the comparison struggles to make some sense. Approaches like [5], classifying isolated phonemes (elements of speech), are better. However each unvoiced phoneme depends a lot on the surrounding voiced part of the speech, introducing a bias favorizing MFCC approach. However despite this bias, wavelet based features show a slightly

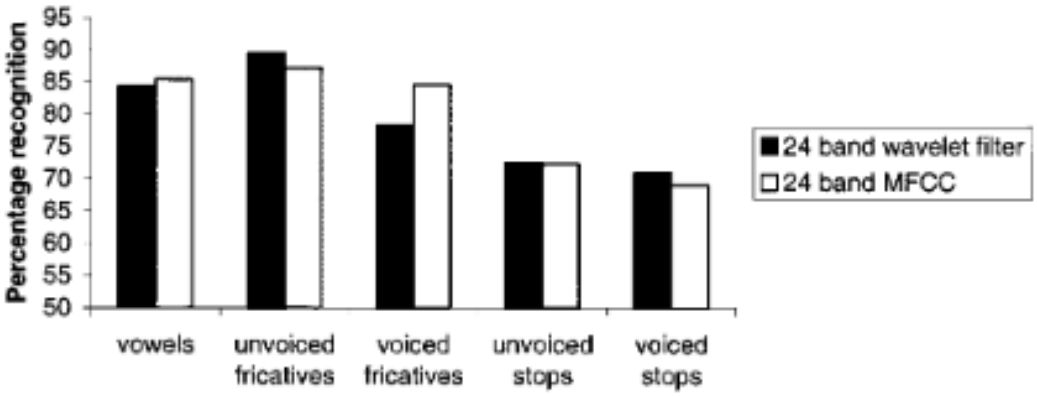

FIGURE 3.5: Per phoneme comparison of the recognition rates induced by wavelet based and MFCC based preprocessing on a neural network based classification, from the publication of O Farooq and S Datta [[5]].

better performance in the recognition of voiced stops ("p" for example), unvoiced stops ("t" for example) and fricatives ("s" for example). (cf fig 3.5).

[22] doesn't have the same bias as it compares a hybrid MFCC/wavelet approach with the MFCC approach. However they apply the wavelet transform on the Mel-scale filter bank (when some of the noisy and transient information is already lost), and they conduct their experiments on full words and there is no way to isolate the results for the unvoiced part of the words. Despite that bias they however find better results for the hybrid approach. There is only a 1% gain in the successful recognition rate on clean speech signals but it reaches a 10% gain on noisy signals. [13] has the same result (only on clean data however), with the same bias. Indeed, one of the most recognized usage of the wavelet transform is for denoising signals, and getting an increased resilience to noise when adding wavelet based coefficients to the feature set should not be a surprise. Finally within this few number of papers about wavelet based feature sets, no paper entirely satisfied me and I think some open problems remain.

Maybe the most relevant experiment to conduct on the difference between the Wavelet based features and the Fourier based feature would be to compare the recognition rate of words which differ only by one phoneme. However for such an experiment to be conclusive, one need a dataset with only such words, and in significant amounts. As this is a very exigent requirement, I didn't had such a dataset at my disposal and I settled on two other experiments with lower requirements:

- In Chapter 4 I conduct a qualitative experiment through a Principal Component Analysis (PCA) of the feature sets applied to some basic phonemes. PCA being a very visual analysis, it enables me to present in a clear way the differences between the two feature sets

- Then, in Chapter 5, I quantify the difference in performance that the Wavelet based feature sets can bring to ASR systems by comparing the performance of the MFCC feature set with a MFCC/Wavelet hybrid feature set on two different systems. As the MFCC/Wavelet hybrid feature set is a superset of the MFCC feature set, any improvement in the recognition rate of the model using the hybrid feature set compared to the traditional one can be directly linked to the usage of the wavelet-based features.

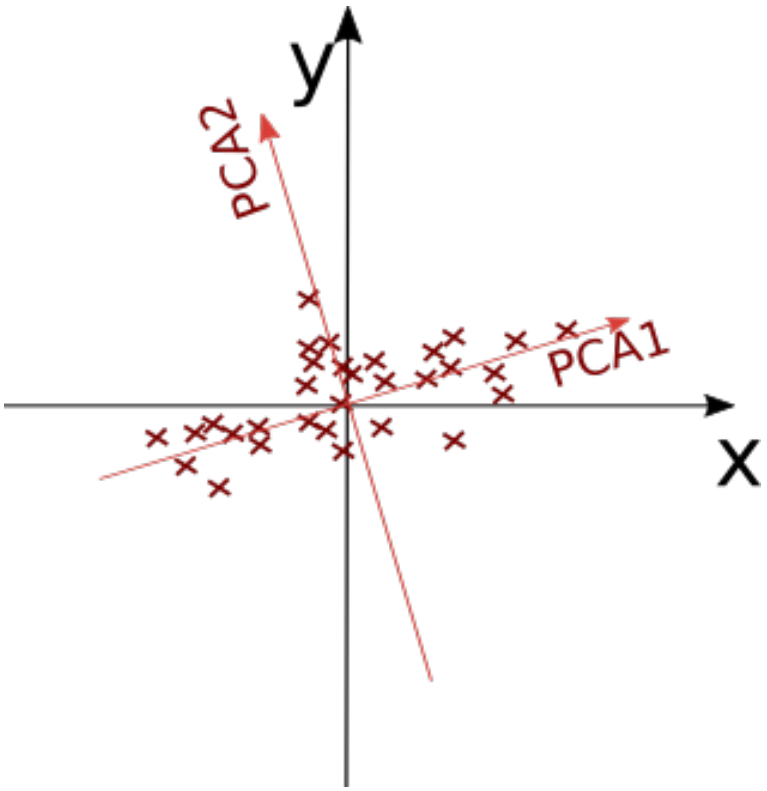

FIGURE 4.1: An example of pca analysis

Chapter 4

A Preliminary Comparison

In this chapter, I present a preliminary comparison of the MFCC and a Wavelet-based feature sets by using a very simple PCA based approach in order to show the strengths and weaknesses of both approches. As the PCA analysis gives some visual representations of the data, it is a good tool to get a grasp on the pros and cons of the feature sets.

PCA works by determining a basis of vectors in the feature space describing the dataset the more succinctly possible. Its goal is to condense the variance of the dataset along as few axis as possible. To take an example let's look at the figure 4.1: In the dataset presented it is obvious that most of the information is carried along an axis which is a combination of the two axis of the original feature space. PCA will find this axis and complement it with its orthogonal axis to return a new base in the feature space explaining better the distribution of the dataset.

4.1 The data set

I first computed, simplified through PCA, and vizualized the feature set for some samples of the sounds 'apa' and 'apha' (the 'h' stands for a stronger p) in order to confirm if the wavelet approach was more interesting than MFCC approach when it comes to the analysis of 'non-vocal' sounds. Then, I performed some more data vizualisation on the 's' and 'th' sounds, to enrich my analysis with another comparison.

4.1.1 The processing of the MFCC features

I computed on one side the MFCC features as a reference feature set, exactly the way explained in the previous chapter. Then in order to perform some visualisation I applied a PCA transformation to find the two most relevant combination of features. It is important to know that in order to have a more meaningfull comparison the PCA transformation is the same for both of the studied sounds. The plots of the sounds 'apa' and 'apha' are represented in figures 4.2 and 4.3. In those plots color is representing time and the points are describing a trajectory from blue to red to green as the sound is pronounced. One can notice that the left cluster of points aggregating the points at the begining and end of the sample corresponds to the 'a' sound, and that the right cluster of point, quickly sliding from bottom to top, corresponds to the p sound. On the plot of the 'apha' sound with a strong and very plosive 'p', a new cluster at the top of the plot appears, corresponding to the exhalation occuring at the end of the 'p'.

It is satisfying to observe the emergence of a cluster at the top corresponding to the exhalation on the 'apha' sound. Furthermore, the horizontal axis seems to translate to vocal part of the observed sounds: The purely vocal 'a' being located on the far left, the half vocal exhalation 'h' in the middle, and the purely non-vocal 'p' stop on the far right of the plot. However one can notice that the same axis (the vertical one) bears the information for both the 'p' stop' and the 'h' exhalation, which hinders any possibility to associate a physical meaning to the observed vertical feature.

In order to give a more exhaustive image of my work I present in figure 4.4 a plot of 10 'apa' sounds (represented all in white, with no notion of time) and 10 'apha' sounds (represented all in red, with no notion of time) superposed. The same pattern can still be observed even when the number of sample goes up.

4.2 The processing of the wavelet features

In a second time I computed the wavelet features of the same sounds. I have used the Daubechies 6 wavelet series over 4 scales, with a discrete wavelet transform (More detail about the transform and the wavelet series can be found in the previous Chapter). However the information yielded by the

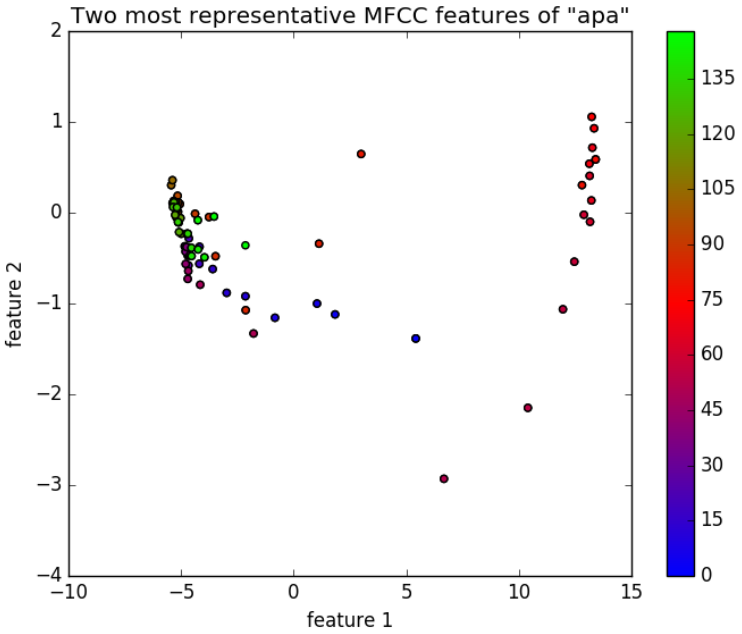

FIGURE 4.2: The 2 main components of the 'apa' sound represented through MFCC features. Color represents time, running from blue to red to green.

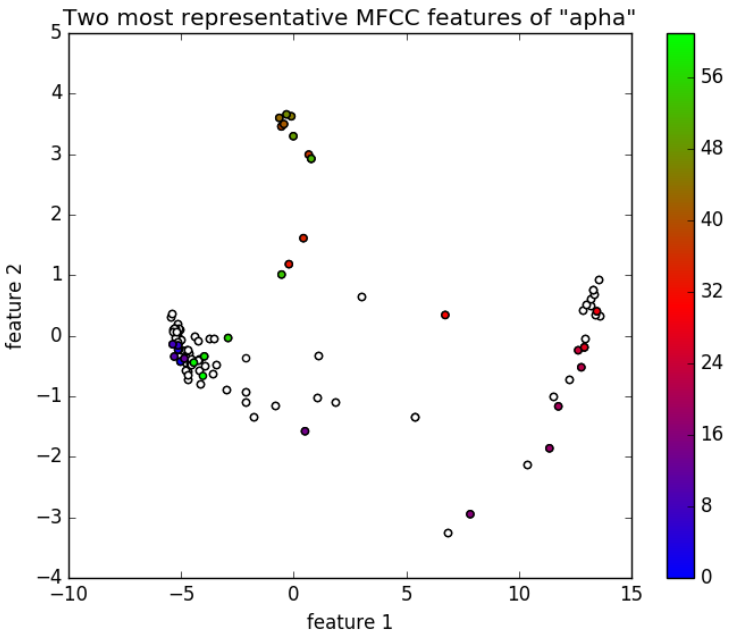

FIGURE 4.3: The 2 main components of the 'apha' sound represented through MFCC features. Color represents time, running from blue to red to green.

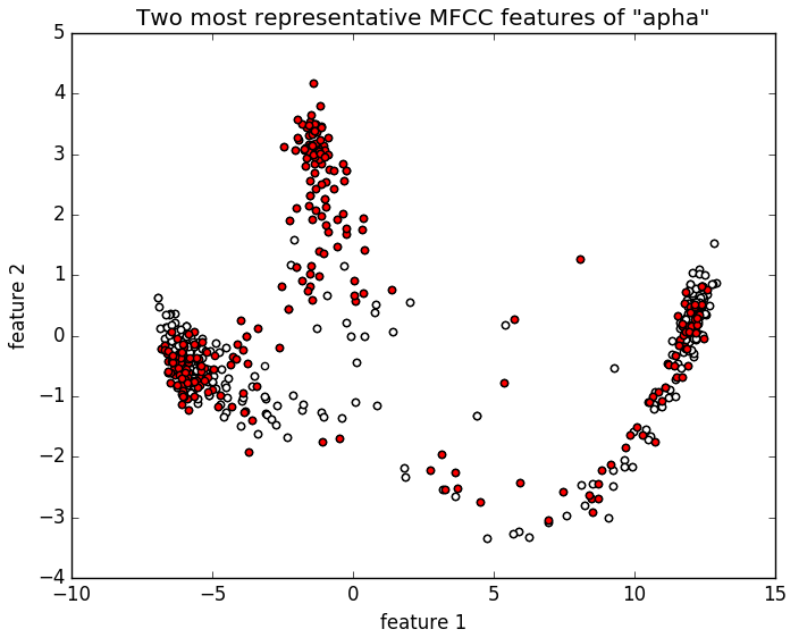

FIGURE 4.4: Superposed 'apa' and 'apha' sounds. 'apa' sound samples are all in white, 'apha' sound samples are in color with time running from blue to red to green.

wavelet transform being less compact than the one offered by MFCC features, I performed some more preprocessing before getting into the PCA dimensionality reduction. Wavelet coefficients describing the 'texture' of the sound, I extracted some statistical aggregators of the coefficients at each scale (mean, variance, skewness, kurtosis). After some unsuccessful tries using the mean, I set out to use the variance, letting investigations on possible usages of the skewness and the kurtosis to future work. I have then computed the 2 main components through PCA transformation (The transform being the same for the two sounds) and plotted the speech samples in the same way than the MFCC-based analysis. The 'apa' plot corresponds to the figure 4.5, the 'apha' sound corresponds to the figure 4.6, and a superposition of 10 'apa' sounds and 10 'apha' sounds (with the 'apa' sounds in white) can be found in figure 4.7.

This time there is no clear cluster corresponding to the vocal 'a' sound. However in the first plot ('apa' sound), the trajectory of the plot points to the left when the 'p' stop occurs. Comparing this with the 'apha' plot, one can notice that the plosive friction resulting from the strong 'p' induces a spike on the left side of the plot. What is truely remarquable is that when it comes to the 'p' and 'ph' sounds, the first feature (abscissa) corresponds to the p stop and the second feature (orthogonal, ordinate) corresponds to the friction at the end of the consonant sound). It is interesting to notice that, even if the classification between vowel and consonant is way better in the MFCC representation, the consonant representation has much more meaning

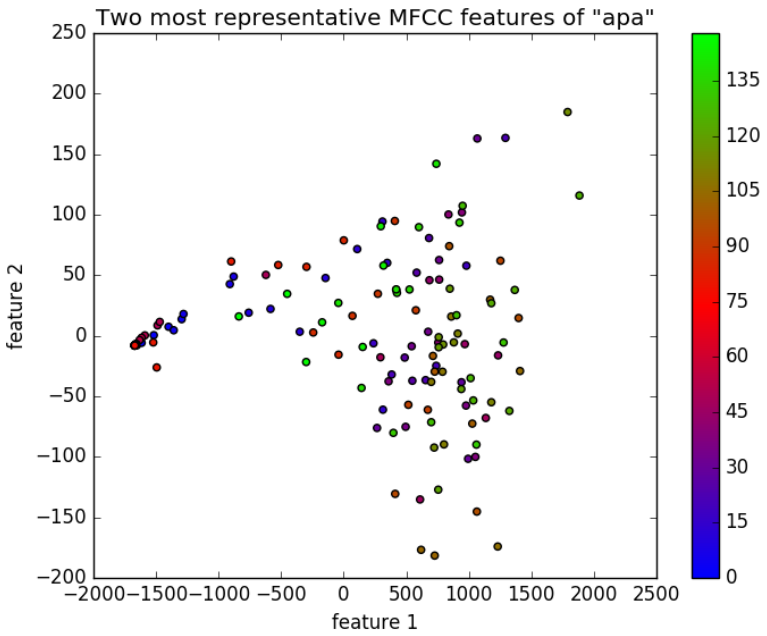

FIGURE 4.5: The 2 main components of the 'apa' sound represented through wavelet features. Color represents time, running from blue to red to green.

in the wavelet analysis. Also, one more time, the plot perfomed on some more numerous samples confirm the results obtained on a smaller number of signals.

In order to confirm the result on some more non-harmonic phonemes, I repeated the experimentation on the 's' and english 'th' sounds, which are completely unvoiced fricatives. The results are presented in the figures 4.8 and 4.9, with the warm colors corresponding to the 'th' sound and the cold colors corresponding to the 's' sound. The time is represented by the evolution of the colors from light to dark: from yellow to dark red for 'th' and from pale blue-green to dark navy blue for 's'.

As one can see, the 's' and 'th' sounds are way easier to discriminate using the wavelet transform than the Fourier transform.

The PCA analysis of the feature sets suggesting that the two feature sets are effectively complementary, I will in the next chapter compare them in a more integrated context.

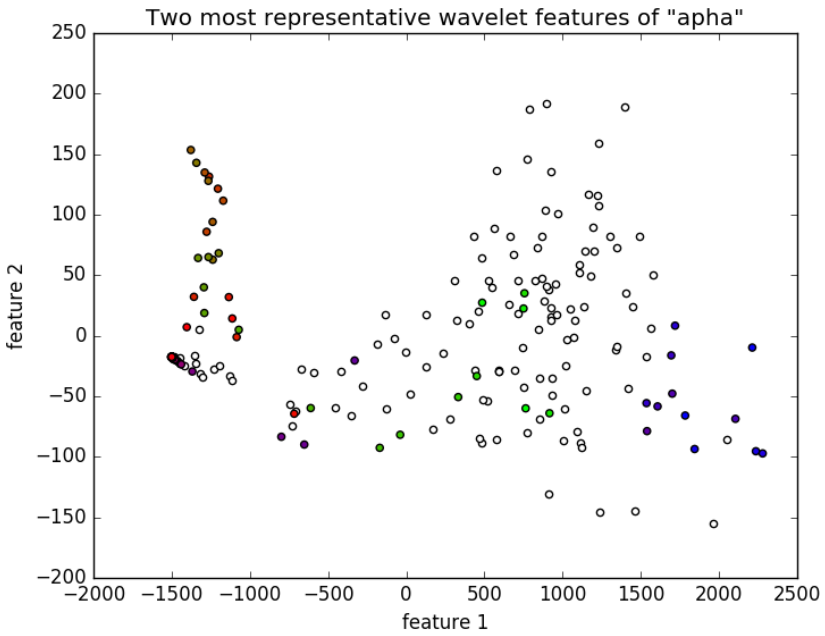

FIGURE 4.6: The 2 main components of the 'apha' sound represented through wavelet features. Color represents time, running from blue to red to green.

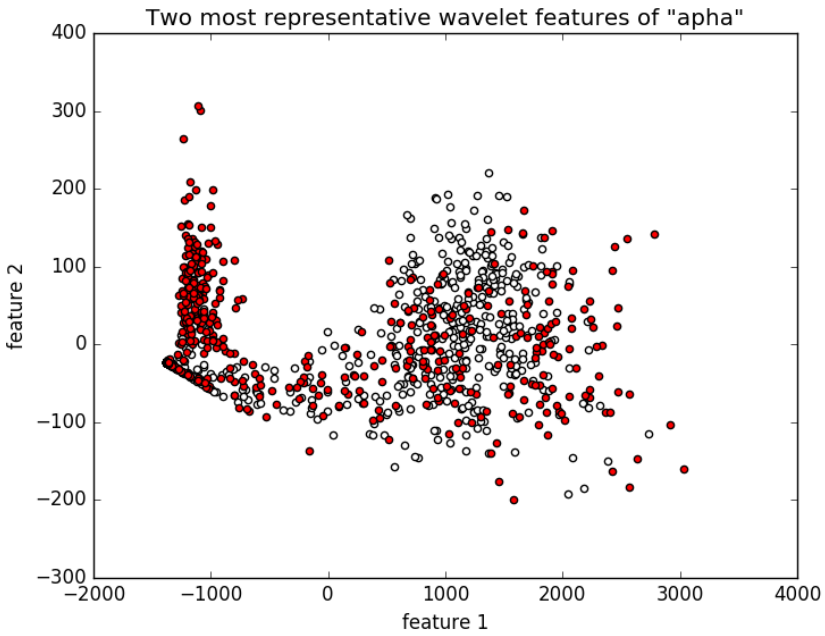

FIGURE 4.7: Superposed 'apa' and 'apha' sounds. 'apa' sound samples are all in white, 'apha' sound samples are in color with time running from blue to red to green.

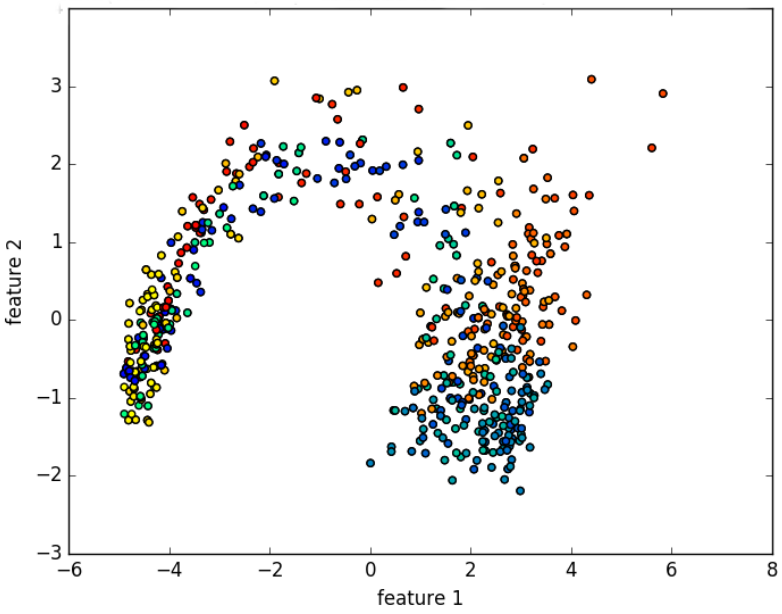

FIGURE 4.8: 'asa' sound in cold colors, 'atha' sound in warm colors

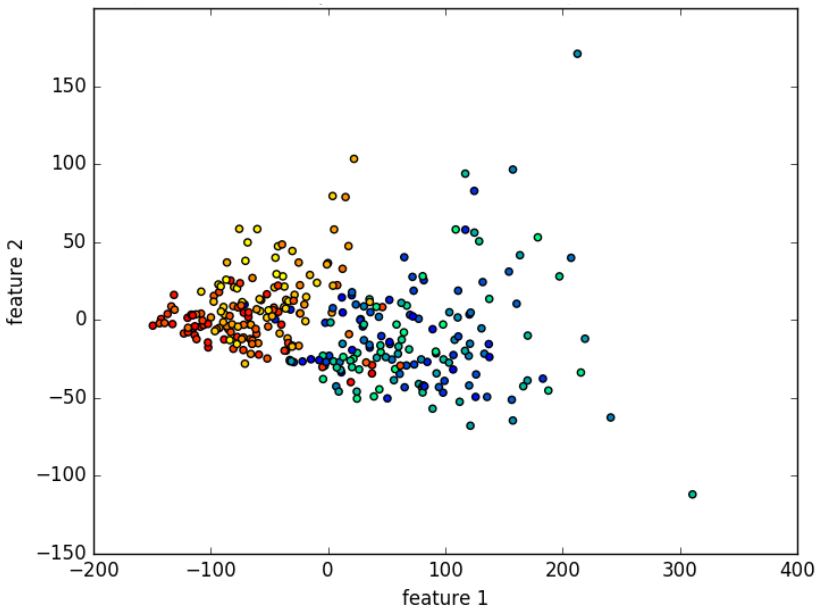

FIGURE 4.9: 's' sound in cold colors, 'th' sound in warm colors

Chapter 5

The ASR system

In the two previous chapters it emerged that on paper and in basic experiments the wavelet based coefficients contain some information which is not accessible from the MFCC feature set. However it is still important to assess the gain in performance in actual ASR systems this new feature set can bring. In this chapter I detail the nature and results of the experiments I lead in order to quantify the improvement on a simple ASR system brought by appending a wavelet-based feature set to the traditional MFCC feature set.

5.1 The model

In this section I explain the category of model I have used as an ASR system to perform an in situ comparison of the two feature set.

I first chose to train an end-to-end Connectionist Temporal Classification model converting features directly into english written sentences. This kind of models is very convenient as data with (speech, transcript) couples is easier to obtain in high quantity than phoneme-transcripted data. My model is composed of a Bi-directional Recurrent Neural Network (BiRNN) suffixed by a character beam decoder. The structure of the system can be consulted on fig. 5.1.

Then I repeated the same experiment on data made of (speech, phoneme transcript) couples. The beam decoder decoding then the output of the neural network into a sequence of phonemes.

Specifically, I have used the architecture proposed by Graves et. al. ([7]) of which I set up below to explain the fundamentals. I would like to draw the attention to the fact that I have used extensively the great tutorial presented by Andrew Gibiansky on Connectionist Temporal Classification (CTC) for Automatic Speech Recognition systems (on the following url: http://andrew.gibiansky.com/blog/mac learning/speech-recognition-neural-networks/) in order to provide the following introduction to CTC ASR systems. If the reader is not familiar with the basic notions about Bidirectionnal Recurrent Neural Networks I invite him to read the work of Mr. Gibiansky as it contains some further details I had to cut to leave more room to the presentation of my own work.

5.1.1 The BiRNN architecture

The architecture ultimately proposed by Graves et. al. in their paper utilizes both BiRNNs and LSTM units. However, in addition, they extend the architecture by adding more hidden layers at each timestep (Which I have not done in the present thesis for the sake of the speed of training of the model). Instead of only having one hidden layer between the input and the output, the BiRNN has N hidden layers.

By combining a BiRNN with LSTM units, the model is very effective at reaching information from both far ahead and far behind each time step. Furthermore, the usage of multiple layers is usefull to mix both the information ahead with the information behind in order to get an accurate prediction. However a simple BiRNN using LSTM is not enough to perform proper speech recognition, as the following subsection will explain.

5.1.2 The acoustic model

The first goal for speech recognition is to build a classifier which can convert from a sequence of sounds into a sequence of letters or phonemes.

Let's suppose that we have an input sequence x (sound data) and a desired output sequence y (phonemes). Even if the output sequence is short (for example two spoken words, maybe ten or twenty sounds), the input sequence will be much longer, as each sound will stretch over many samples on the inputed sampled signal. Thus, x and y will be of different lengths, which poses a problem for a standard RNN architecture (in which predicts one output for one input).

There are several options for correcting this problem. The first option is to align the output sequence y with the input sequence, each element y_i of the output sequence is placed on some corresponding element x_i. Then, the network is trained to output y_i at timestep i (with input x_i) and output a "blank" element on timesteps for which there is no output. These sequences are said to be "aligned", since we've placed each output element y_i in its proper temporal position.

Sadly, aligning the sequences is an onerous requirement. While unaligned data may be easy to come by (simply record sound and ask speakers to transcribe it), aligned data may be much harder to acquire; it may require careful aligning as well as understanding of the sounds being produced (and a sound understanding of phonology).

Instead of requiring aligned data, however, the network can be trained directly on unaligned data. This requires some clever tricks, objective functions, and output decoding algorithms. Collectively, this method is known as Connectionist Temporal Classification.

5.1.3 Connectionist Temporal Classification

For the purposes of Connectionist Temporal Classification (CTC), let's consider the entire neural network to be simply a function that takes in some input sequence x (of length T) and outputs some output sequence y (also of

length T). As long as there is an objective function on the output sequence y, the network can be trained to produce the desired output.

Let's suppose that for each input sequence x (sound data) there is a label l. The label is a sequence of letters from some alphabet L, which is potentially shorter than the input sequence x; let U be the length of the label. The key idea behind CTC is that instead of somehow generating the label as output from the neural network, the output is designed to be a probability distribution at every timestep (from $t = 1$ to $t = T$). We can then decode this probability distribution into a maximum likelihood label. Finally, the network is trained by creating an objective function that coerces the maximum likelihood for a given sequence x to correspond to our desired label l.

Such a process requires a derivable function to match the sequence of probability distribution to the most probable sequence of characters from the alphabet L. This by done using a Beam Search algorithm. Beam Search algorithms come in many different fashions, differing in the way they explore and rate the different possibilies offered by the sequence of probability distribution. The strategy employed by Graves et. al. is a Prefix Search, which employ heuristics to guide the search.

5.2 The data

In order to train the model for the first experiment (the speech to english text experiment) I have computed the two feature sets (around 32000 speech samples) over the LibriSpeech corpus ([16]) which contains some (speech, transcript) pairs extracted from read public-domain books. Each pair represents around 3 secondes of speech and is pronounced with a clean academic pronounciation and without noises.

For the second experiment, I have used the standard TIMIT dataset as it is one of the rare dataset which presents phoneme transcripts (which the LibriSpeech corpus doesn't have). Here again the samples (I have used around 6000 of them) are short sentences read with a clean pronounciation, without noises. However the feature set encompasses a broad range of different english accents.

5.3 The feature sets

I have computed the two following feature sets from the dataset:

- First a hybrid feature set, concatenating the MFCC features and some wavelet-based features. As wavelet-based features, I have chosen to take local means and maximums of the log-power of the coefficients of the Discrete Wavelet Transform based on the Daubechies 6 mother wavelet over 4 scales. The process is represented in fig.5.2. I chose to compute the maximums and mean over thirds of the signal window as each window is 3.5 ms wide and unvoiced phonemes have a scale of

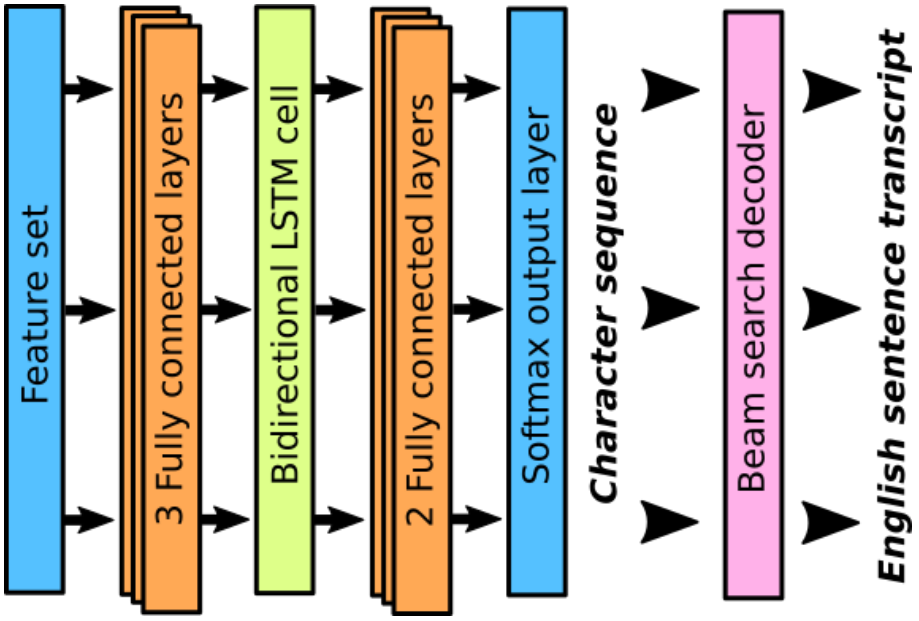

FIGURE 5.1: The structure of the end-to-end model

roughly 1 ms. My choice was motivated by the fact that the wavelet coefficent purpose is to describe the energy present in some noise signal bands described by the wavelet functions. The means and maximums of the energy give a very compact and representative image of the evolution of the energy in each band. The MFCCs feature set having 13 coefficients and the wavelet-based features amounting to 24 coefficients, this hybrid feature set counts 37 features

- Then I computed the MFCC (and only MFCC) feature set. However, having only 13 MFCCs, I chose to append some copies of the MFCCs to the feature set in order to get two feature sets of the same size for a more equal comparison.

5.4 The Bidirectional Recurrent Neural Network (BiRNN)

The BiRNN architecture I have used follows the architecture of figure 5.1. It has the advantage of being recommended by [7] and being readily implemented on Github (https://github.com/philipperemy/tensorflow-ctc-speech-recognition, which is itself largely inspired by https://github.com/mozilla/DeepSpeech), even if I had to retouch it a lot (Introducing some L2 normalisation, changing the normalisation of the features (which was badly implemented), introducing support for wavelet based features, tuning the sizes of the hidden layers ...). The final results show good enough performances to consider the experiment as a comparison for the feature sets in real-life ASR system conditions.

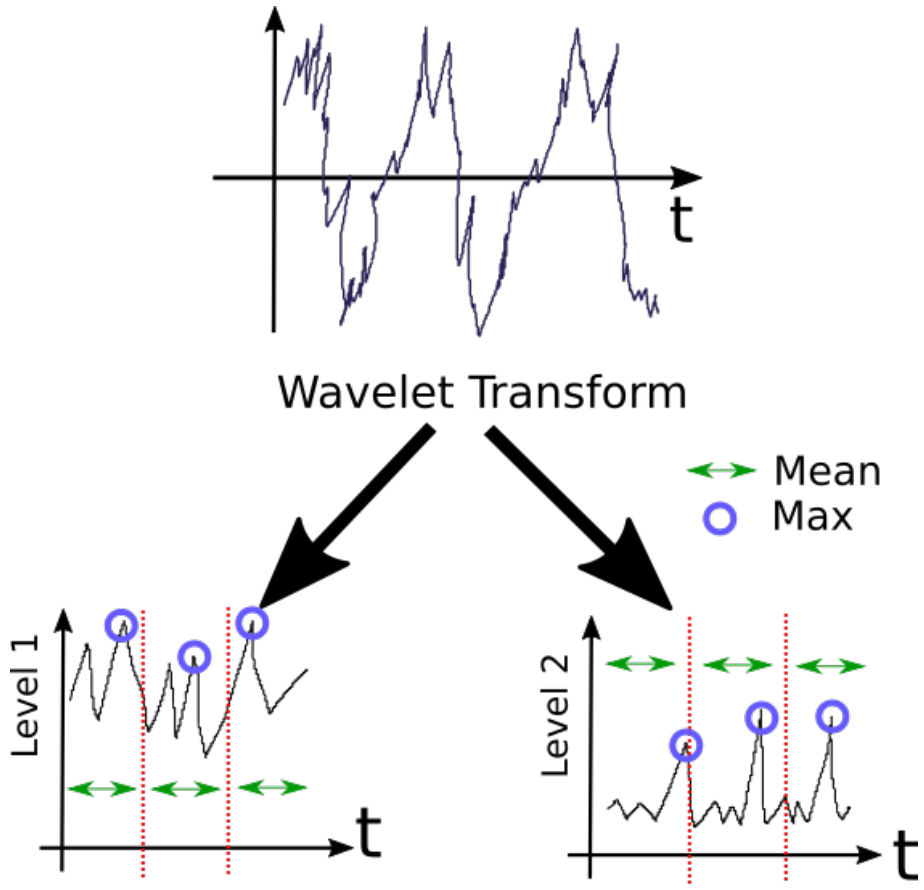

FIGURE 5.2: The extraction of the wavelet based features

For the (speech, text) experience, The BiRNN outputs a 29-cells wide softmax layer describing the most probable character at each time. For the (speech, phoneme) one, the output is 60-cells wide.

5.5 The beam decoder

The beam decoder relies on a character-based (resp. phoneme-based) model and maps the character stream outputted by the neural network to english sentences (resp. phoneme sequences). I have used an already made and standard beam decoder, as it was not my point of focus (The one provided by python's tensorflow framework). More details about the beam decoder and Connectionist Temporal Classification (CTC) ASR systems has been given earlier in this Chapter.

5.6 The results

The training of the (speech, text) model was performed over with 15 epochs of 32768 samples, reporting the performance of the intermediary model each epoch. The training of the (speech, phoneme) model was performed over

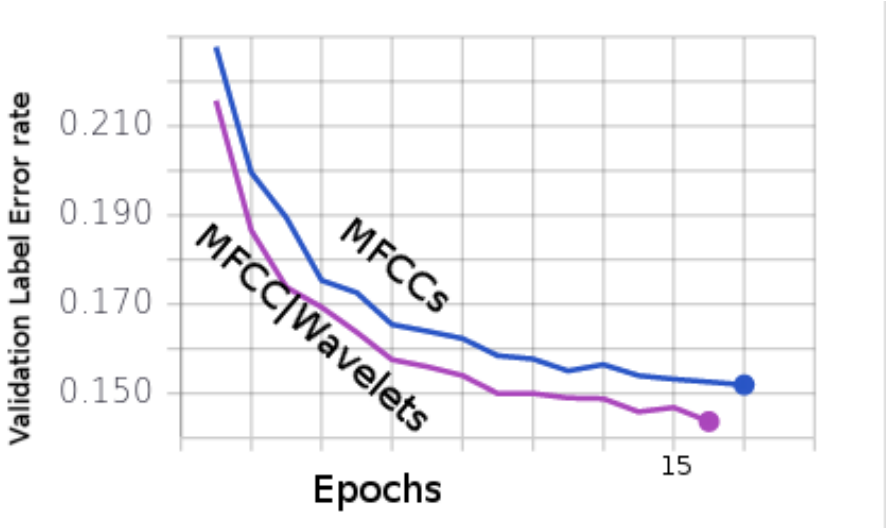

FIGURE 5.3: Validation label error rates of the two feature sets over time for a speech to text translation

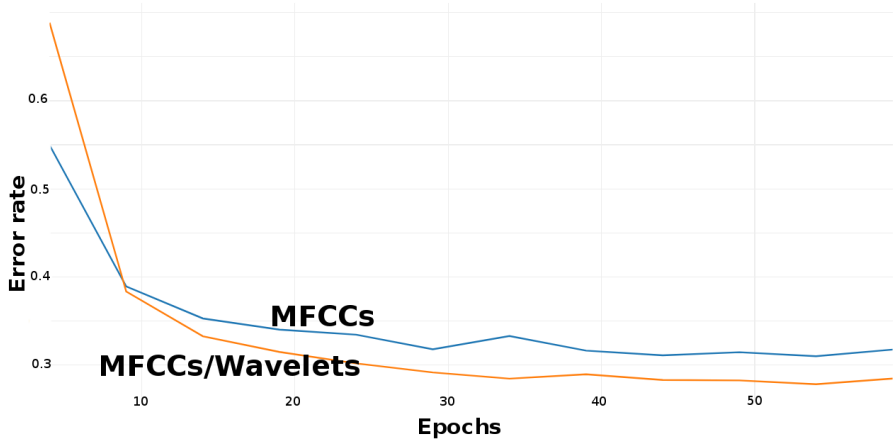

FIGURE 5.4: Validation label error rates of the two feature sets over time for a speech to phoneme translation

with 60 epochs of just under 5000 samples, reporting the performance of the intermediary model each 5 epochs. The performance is measured by the Validation Label Error Rate, which is the rate of "edit distance" error over a validation dataset over which the network has never been trained.

The results for the first experiment (fig.5.3) show a significant advantage (7% of relative improvement) for the (speech, text) model trained over the hybrid feature set. When it comes to the second experiment, the results (fig. 5.4) show an even sharper improvement with an advantage of 10% for the (speech, phoneme) model trained over the hybrid feature set.

Due to the various possible natures of the errors in the text to phoneme translation, analysing the errors yielded by ASR system is really challenging. However, for the sake of comprehensiveness I list below (and provide a brief

analyse) some samples of data run through the BiRNN. A nomenclature of the pairing between mono/bi/tri-grams and the associated phonemes is provided in Appendix A. The Appendix B lists some more samples outside of the main body of the thesis for legibility reasons.

5.6.1 Example sentence 1

Original sentence:
Mom strongly dislikes appetizers.

Human transcription:
pau m aa m s tcl t r ao ng l iy dcl d ix s pau l ay kcl k s q ae pcl p ix tcl t ay z axr z pau

MFCC only model transcription:
pau q aa n z tcl t aa n l iy dcl jh ix s l ay kcl s q ae bcl b ax tcl t ay z er z pau

Hybrid model transcription:
pau m ao n s tcl t r ao l iy dcl d ix s pau l ay kcl k s q ae pcl p ix tcl s ay z er z pau

The MFCC only model presents the following alterations:

- Omissions:
  r, k (after a kcl), pau
- Tranforms:
  m -> q, m -> n, s -> z, ao -> aa, ng -> n, d -> jh (after dcl), [bcl, b] -> [pcl p], ix -> ax, axr -> er

The hybrid model presents the following alterations:

- Omissions:
  ng
- Tranforms:
  m -> n, t -> s (after tcl), axr -> er

For this example, which presents many consecutive consonants, fricatives and plosives, the hybrid feature set clearly presents a strong lead compared to the MFCC feature set.

However as one can see in the following examples this is not a systematic behaviour.

5.6.2 Example sentence 2

Original sentence:
She had your dark suit in greasy wash water all year

Human transcription:
pau sh iy hv ae dcl d y axr dcl d aa r kcl k s ux tcl t q ix n gcl g r iy s iy w ao sh pau w ao dx axr q ao l y ih axr pau

MFCC only model transcription:
pau sh iy hv ae dcl d y axr dcl d aa r kcl k s ux tcl t ix n gcl g r iy s iy w aa sh pau w ao dx axr q ao l y ih axr pau

Hybrid model transcription:
pau sh iy hv ae dcl d axr dcl d aa r kcl k s ux tcl t q ix n gcl g r iy s iy wh aa sh epi wh ao dx axr q ao l y ih axr pau

The MFCC only model presents the following alterations:

- Tranforms:
  w -> wh

The hybrid model presents the following alterations:

- Omissions:
  y
- Tranforms:
  w -> wh (2 times), ao -> aa

For this example, which still presents some fricatives and plosives, the hybrid feature set's performance lags behind the MFCC only model.

5.6.3 Example sentence 3

Original sentence:
Put the butcher block table in the garage

Human transcription:
pau p uh tcl dh ax bcl b uh tcl ch axr bcl b l aa kcl t ey bcl b el ax n dh ix gcl g er aa sh pau

MFCC only model transcription:
pau p uh tcl dh ax bcl b ah tcl ch axr bcl b l aa kcl t ey bcl b el ix n ix gcl g er aa sh pau

Hybrid model transcription:
pau pcl p ah tcl dh ax bcl b ah tcl ch axr bcl b aa tcl t ey bcl b n dh ax gcl g er ao sh pau

The MFCC only model presents the following alterations:

- Omissions:
  dh

- Tranforms:
  uh -> ah, ax -> ix

The hybrid model presents the following alterations:

- Insertion:
  pcl (before p)

- Omissions:
  l

- Tranforms:
  uh -> ah (2 times), kcl -> tcl, en -> [n dh], aa -> ao, ix -> ax

For this example yet again, despite the presence of a lot of plosive consonants, the hybrid feature set's performance looses to the MFCC only model's one.

As a conclusion, even if on average the hybrid model performs better and trains faster than the MFCC model, it is by no mean a systematic behaviour and it is very difficult to quantify a per phoneme category improvement. The main obstacle to that being the broad range of possible errors in the transcripts.

Chapter 6

Conclusion

By comparing the classic MFCC feature set with a hybrid superset encompassing some wavelet-based features, I have successfully shown that the usage of wavelet-based coefficients can improve significantly the performance of an ASR system. However for now I can only conjecture that this improvement is due to a better performance on unvoiced parts of the speech, thanks to the qualitative PCA analysis I have led in Chapter 4. As suggested at the end of Chapter 3, in order to prove that conjecture I am planning as a futur work to run the models I have computed over samples of sounds featuring mainy times the same two words, differing only by one (or two consecutive) consonant or vowel phoneme (for example 'screen' and 'spleen', or 'cat' and 'bat'). In that way it will be much easier to build a phoneme category based comparison of the two feature sets. However, such an experiment will require the construction of a custom dataset and may require a lot of time.

Also, as a matter of scientific rigor, I am planning to try to train model based on a different architecture with the two same feature sets in order to confirm that the improvement in performance observed with the hybrid feature set is in no way linked to a specific model.

Appendix A

Nomenclature of the annotation of the phonemes

In this appendix I establish the correspondance between the (mono/bi/tri)gram notation of the phonemes used in the TIMIT dataset and the international phonetic alphabet (IPA)

(mono/bi/tri)gram	**IPA**	**example word**
	Stops	
b	[b]	Bee
d	[d]	Day
g	[ɡ]	Gay
p	[p]	Pea
t	[t]	Tea
dx	[ɾ]	muDDy, DirTy
q	[ʔ]	baT
	Affricates	
jh	[dʒ]	Joke
ch	[tʃ]	CHoke
b	[b]	Bee
	Fricatives	
s	[s]	Sea
sh	[ʃ]	SHe
z	[z]	Zone
zh	[ʒ]	aZure
f	[f]	Fin
th	[θ]	Thin
v	[v]	Van
dh	[ð]	Then
	Nasals	
m	[m]	MoM
n	[n]	NooN
ng	[ŋ]	siNG
em	[m̩]	bottOM
en	[n̩]	buttON
eng	similar to ng	washINGton
nx	[ɾ̃]	wiNNer

	Semivowels and Glides	
l	[l]	Lay
r	[ɹ]	Ray
w	[w]	Way
y	[j]	Yacht
hh	[h]	Hay
hv	[ɦ]	aHead
el	[l̩]	bottLE
	Vowels	
iy	[i]	bEEt
ih	[ɪ]	bIt
eh	[ɛ]	bEt
ey	[ɛɪ]	bAIt
ae	[æ]	bAt
aa	[ɑ]	bOtt
aw	[aʊ]	bOUt
ay	[aɪ]	bIte
ah	[ʌ]	bUt
ao	[ɔ]	bOUght
oy	[ɔɪ]	bOY
ow	[oʊ]	bOAt
uh	[ʊ]	bOOk
uw	[u]	bOOt
ux	[ʉ]	tOOt
er	[ɜ] (retroflex)	bIrd
ax	[ə]	About
ix	[ɨ]	debIt
axr	[ə] (retroflex)	buttER
ax-h	[ə]	sUspect
	Others	
pau	[]	(pause, silence)

Also, a plosive followed by cl (pcl, kcl etc...) corresponds to the implosive version of the associated consonant. These sounds occur during the closure of the mouth preceding the associated plosive.

Appendix B

Phoneme transcription of speech samples

This Appendix lists some more samples run through the models described in Chapter 5, at the end of which the first 3 samples are presented.

B.1 Example sentence 4

Original sentence:
Elderly people are often excluded.

Human transcription:
pau q eh l dcl d axr l iy pcl p iy pcl p el aa r ao f ax nx ih kcl k s kcl k l uw dx ix dcl d pau

MFCC only model transcription:
pau ow dcl g l iy pcl p iy p el aa f ix m iy s kcl w ih dcl pau

Hybrid model transcription:
pau aw l tcl d l iy pcl p iy pcl wh ao r ao f axr m ey s kcl k wh ix pau

B.2 Example sentence 5

Original sentence:
Aim to balance your employee benefit package.

Human transcription:
pau q ey m tcl t ux bcl b ae l ah n s y er ix m pcl p l ow iy bcl b eh nx ax f ix tcl t pcl p ae kcl k ix dcl jh pau

MFCC only model transcription:
pau q ey n jh axr bcl b aw n hh er ix kcl p ao ux bcl b eh m f ix tcl pcl p ae kcl k ix jh pau

Hybrid model transcription:
pau q ey tcl ix bcl g eh ow n z axr ix pcl p ao iy y ux bcl b eh dx axr f ih tcl ch pcl p ay kcl k ix dcl jh pau

B.3 Example sentence 6

Original sentence:
Catastrophic economic cutbacks neglect the poor.

Human transcription:
pau k ae dx ah s tcl t r aa f ih kcl k eh kcl k ax n aa m ix kcl k ah tcl b ae kcl s pau n ix gcl g l eh kcl dh ax pcl p ao r pau

MFCC only model transcription:
pau k ae v ax s kcl k r aa f ix kcl ah kcl k ax n ax nx ix kcl k aa pcl p ae kcl k s pau m ix gcl g l ae kcl dh ax pcl p ao pau

Hybrid model transcription:
pau t ae dx ax s tcl k r ao ix kcl l ay kcl k ax n q ao nx ix kcl k aa pcl b ae kcl k s pau n ix gcl g l ae kcl dh ax pcl p ao r pau

B.4 Example sentence 7

Original sentence:
Bob papered over the living room murals.

Human transcription:
pau b aa bcl p ey pcl p axr dcl d ow v axr dh el l ih v ix ng r uw m y er r ax l s pau

MFCC only model transcription:
pau b ao pcl p ey pcl p axr dcl ah pcl v axr gcl el ih dh ix ng axr ix ng y ih ow l z z pau

Hybrid model transcription:
pau b ao pcl p ey pcl p axr dx ow v ax v ax l ih v ix ng r ix m y axr r el z pau

B.5 Example sentence 8

Original sentence:
Beg that guard for one gallon of gas

Human transcription:
pau b ey gcl g dh eh tcl g aa r dcl f axr w ah n gcl g eh l ax n ix v gcl g ae s pau

MFCC only model transcription:
pau b ih gcl n ae gcl g ao r dcl d f r axr n gcl g eh l n ax v gcl g eh s pau

Hybrid model transcription:
pau b ey ng dh ey gcl g aa r dcl d f r ao r n gcl d l ax n ax v kcl p ae s pau

B.6 Example sentence 9

Original sentence:
A chosen few will become Generals.

Human transcription:
pau q ah tcl ch ow z ih n f y ux w el bcl b iy kcl k ah m jh eh nx axr el s pau

MFCC only model transcription:
pau q ah dcl ch ow z ax ix n ix v dcl jh iy l bcl b iy gcl g ah m jh ix n ow l z pau

Hybrid model transcription:
pau q ah tcl ch ow z ax n ax f y ux wh el bcl b ix kcl k ah m z eh nx axr el z pau

B.7 Example sentence 10

Original sentence:
She is thinner than I am.

Human transcription:
pau sh iy ih z th ih nx er dh eh n ay ae m pau

MFCC only model transcription:
pau sh iy s t ih m er dh ih n ow iy m pau

Hybrid model transcription:
pau sh iy ih s pcl p ih nx axr dh eh n ow hv ax m pau

B.8 Example sentence 11

Original sentence:
Drop five forms in the box before you go out.

Human transcription:
pau d r aa pcl f ay f ao m z en dh ax bcl b aa kcl k s bcl b ax f ao y ux gcl g ow aw tcl pau hv pau

MFCC only model transcription:
pau d r aa pcl f ay f ao m z en dh ax bcl b aa kcl k s bcl b ax f ao y ux gcl g ow aw tcl pau hv pau

Hybrid model transcription:
pau r aa f ay f ao r n z en dh ax bcl b ao s en f ao r iy gcl g uw l aw tcl pau

B.9 Example sentence 12

Original sentence:
Those were especially the ones that all other grownups laughed at loudest.

Human transcription:
pau dh ow z w axr ix s pcl p eh sh pau el iy dh ax w ah n z eh tcl t q ao l ah dh ix r gcl g r ow n ah pcl s pau l ae f tcl t eh tcl t l aw dx ix s tcl t pau

MFCC only model transcription:
pau d ow z r er s tcl p r eh sh l iy dh ax w n z eh dx ao l ax dh ix gcl g r ah n dh ah v s l eh f bcl ae dcl l aw dx ix s tcl t pau

Hybrid model transcription:
pau dh ow z wh er s pcl p ah sh pau l iy dh ax wh ah n z eh dx ao l ax dh axr gcl g r ah nx el pcl p s pau l ae f tcl t ae tcl l aw dx ix s tcl t pau

B.10 Example sentence 13

Original sentence:
Did you buy any corduroy overalls?

Human transcription:
pau jh ux bcl b ay nx iy kcl k ao r dx axr oy ow v axr ao l z pau

MFCC only model transcription:
pau sh bcl b ay dx ix kcl k ao dx axr ow v r ow l z pau

Hybrid model transcription:
pau sh iy bcl b ay ix ng kcl k ao r dx r oy ow v r ao z pau

B.11 Example sentence 14

Original sentence:
Each untimely income loss coincided with the breakdown of a heating system part.

Human transcription:
pau q iy tcl ch ah n tcl t ay m l iy ih n kcl k ah m l ao s kcl k ow ix n s ay dx ix dcl w ix th ax bcl b r ey kcl d aw nx ax v ix hv iy dx iy ng s ih s tcl t em pcl p aa r tcl t pau

MFCC only model transcription:
pau y ux tcl ch ih n tcl t ay m iy ix m kcl k eh m ao s kcl k ow n s ay dx axr w ax tcl th ax bcl b r ey d aa dx ax dx iy dx ix ng s ih s tcl t en pcl p aa r tcl t pau

Hybrid model transcription:
pau y iy tcl ch eh n tcl t ay m iy ix n kcl k ah m el z kcl k ah n s ay
dx ix wh ax th ax bcl b r ey kcl d aa nx ax m iy q iy dx ix ng k s ix
s tcl t en m pcl p aa r tcl t pau

Bibliography

[1] Ahmed Alani and Mohamed Deriche. A novel approach to speech segmentation using the wavelet transform. In *Signal Processing and Its Applications, 1999. ISSPA'99. Proceedings of the Fifth International Symposium on*, volume 1, pages 127–130. IEEE, 1999.

[2] Mohammed Bahoura and Jean Rouat. Wavelet speech enhancement based on the teager energy operator. *IEEE Signal Processing Letters*, 8(1):10–12, 2001.

[3] Marwa Chafii, Jacques Palicot, Rémi Gribonval, and Faouzi Bader. A necessary condition for waveforms with better papr than ofdm. *IEEE Transactions on Communications*, 64(8):3395–3405, 2016.

[4] Steven Davis and Paul Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing*, 28(4):357–366, 1980.

[5] O Farooq and S Datta. Mel filter-like admissible wavelet packet structure for speech recognition. *IEEE Signal Processing Letters*, 8(7):196–198, 2001.

[6] Sadaoki Furui. Speaker-independent isolated word recognition based on emphasized spectral dynamics. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'86.*, volume 11, pages 1991–1994. IEEE, 1986.

[7] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1764–1772, 2014.

[8] Md Afzal Hossan, Sheeraz Memon, and Mark A Gregory. A novel approach for mfcc feature extraction. In *Signal Processing and Communication Systems (ICSPCS), 2010 4th International Conference on*, pages 1–5. IEEE, 2010.

[9] Yi Hu and Philipos C Loizou. Speech enhancement based on wavelet thresholding the multitaper spectrum. *IEEE transactions on Speech and Audio processing*, 12(1):59–67, 2004.

[10] Yu LI, Feng-qin YU, and Shu-kai FAN. Speech enhancement based on improved noise variance estimation in wavelet domain [j]. *Audio Engineering*, 3:022, 2008.

[11] Christopher John Long and Sekharajit Datta. Wavelet based feature extraction for phoneme recognition. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 1, pages 264–267. IEEE, 1996.

[12] Stephane G Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7):674–693, 1989.

[13] Robert Modic, Børge Lindberg, and Bojan Petek. Comparative wavelet and mfcc speech recognition experiments on the slovenian and english speechdat2. In *ISCA tutorial and research workshop on non-linear speech processing*, 2003.

[14] Steven J Nowlan. Maximum likelihood competitive learning. In *Advances in neural information processing systems*, pages 574–582, 1990.

[15] Dimitri Palaz, Ronan Collobert, et al. Analysis of cnn-based speech recognition system using raw speech as input. Technical report, Idiap, 2015.

[16] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5206–5210. IEEE, 2015.

[17] Okko Räsänen. Computational modeling of phonetic and lexical learning in early language acquisition: Existing models and future directions. *Speech Communication*, 54(9):975–997, 2012.

[18] Jong Won Seok and Keun Sung Bae. Speech enhancement with reduction of noise components in the wavelet domain. In *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, volume 2, pages 1323–1326. IEEE, 1997.

[19] Yann Soon, Soo Ngee Koh, and Chai Kiat Yeo. Wavelet for speech denoising. In *TENCON'97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE*, volume 2, pages 479–482. IEEE, 1997.

[20] Beng T Tan, Minyue Fu, Andrew Spray, and Phillip Dermody. The use of wavelet transforms in phoneme recognition. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 4, pages 2431–2434. IEEE, 1996.

[21] Beng Tiong Tan, Robert Lang, Heiko Schroder, Andrew Spray, and Phillip Dermody. Applying wavelet analysis to speech segmentation and classification. In *Wavelet Applications*, volume 2242, pages 750–762. International Society for Optics and Photonics, 1994.

[22] Z Tufekci and JN Gowdy. Feature extraction using discrete wavelet transform for speech recognition. In *Southeastcon 2000. Proceedings of the IEEE*, pages 116–123. IEEE, 2000.

[23] Christopher Wendt and Athina P Petropulu. Pitch determination and speech segmentation using the discrete wavelet transform. In *Circuits and Systems, 1996. ISCAS'96., Connecting the World., 1996 IEEE International Symposium on*, volume 2, pages 45–48. IEEE, 1996.

[24] Bartosz Ziółko, Suresh Manandhar, Richard C Wilson, and Mariusz Ziółko. Wavelet method of speech segmentation. In *Signal Processing Conference, 2006 14th European*, pages 1–5. IEEE, 2006.