Auditory Image Model features for Automatic Speech Recognition

Mario E. Munich

Evolution Robotics
Pasadena, CA 91103
mariomu@vision.caltech.edu

*Qiguang Lin**

AOL Voice Services
Mountain View, CA 94043
qg_lin@@yahoo.com

Abstract

Conventional speech recognition engines extract Mel Frequency Cepstral Coefficients (MFCC) features from incoming speech. This paper presents a novel approach for feature extraction in which speech is processed according to the Auditory Image Model, a model of human psychoacoustics. We fist describe the proposed front-end, then we present recognition results obtained with the TIMIT database. Comparing with previously published results on the same task, the new approach achieves a 10% improvement in recognition accuracy.

1. Introduction

Research efforts on speech recognition during the last four decades have given birth to a variety of commercial speech recognition systems. Companies like Nuance, SpeechWorks, Philips, IBM, and ScanSoft offer speech recognition products for both desktop and server applications. All these systems use a similar approach in which feature vectors are extracted from an incoming utterance and are fed to a back-end recognizer based on hidden-Markov models (HMMs), where the conventional features used in these recognizers are Mel-Frequency Cepstral Coefficients (MFCC).

Speech recognizers seems to have reached a performance status-quo and relatively very little performance improvement has been achieved in the past few years. It appears that there is a need for a alternative approaches to speech recognition to be able re-invigorate the field. This paper presents a step towards that goal by proposing a novel way of extracting features from utterances. We propose a novel front-end for recognizers that is based on a model of human psychoacoustics, the Auditory Image Model. This approach is similar to the one of Yang et. al. [1] in the sense that it is also biologically inspired.

The paper is organized as follows. Section 2 presents the Auditory Image Model (AIM). Section 3 elaborates on the proposed method for feature extraction. Section 4 describes the experimental set-up for the comparison between conventional MFCC and AIM features. Section 5 discusses the experimental results. Finally, Section 6 draws some concluding remarks.

2. Auditory Image Model (AIM)

The Auditory Image Model was developed in the lab of Roy Patterson at Cambridge University [2] with the goal of modeling human psychoacoustics, rather than develop an ASR front end. In fact, the model correctly represents known psychoacoustic features, such as level dependence, that have an adverse effect on the ASR task.

AIM is a functional model of the human auditory system that pays close attention to biological detail. It consists of three serial processing modules:

- A filter bank of fourth-order gammatone filters spaced according to the ERB scale (similar to the Mel scale).
- A two-dimensional adaptive threshold mechanism.
- A strobed integrator.

These three stages are shown in Figure 1. The filter bank is an ERB-spaced IIR filter bank, whose channel response is unusual in shape; the filter bank output roughly corresponds to the motion of the basilar membrane of the cochlea, and is labeled ”BMM” in Figure 1. The filter bank response can be computed efficiently with a set of gammatone filters [3].

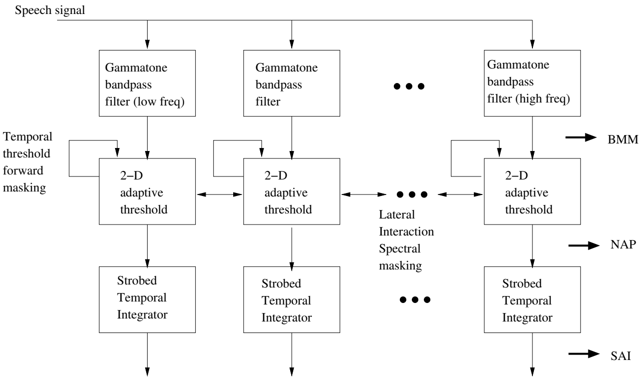
Speech signal
Gammatone bandpass filter (low freq)
Gammatone bandpass filter
Gammatone bandpass filter (high freq)
Temporal threshold forward masking
2–D adaptive threshold
2–D adaptive threshold
2–D adaptive threshold
Lateral Interaction Spectral masking
Strobed Temporal Integrator
Strobed Temporal Integrator
Strobed Temporal Integrator
BMM
NAP
SAI

Figure 1: Auditory Image Model architecture. BMM = Basilar Membrane Motion, NAP = Neural Activity Pattern, SAI = Stabilized Auditory Image.

*The major part of the work was performed while the authors were with VocalPoint, Inc., San Francisco, CA.

The second stage is where AIM begins to differ from traditional models. First, the BMM signal is half-wave rectified and passed through a compressive, logarithmic non-linearity. Then, adaptive thresholding is applied in two dimensions: time and frequency. The temporal threshold includes a short-term memory of past activity: if recent activity was low, then the system is sensitized and more of the BMM will be *gated through* the threshold. This behavior accounts for *forward masking*, which is a property of the human auditory system as well as of AIM, but which is not a property of traditional speech processors. Forward masking removes speech features that are irrelevant to perception. The spectral threshold is based on interactions between neighboring frequency channels: strong activity in a channel will partially suppress activity in less strongly stimulated neighbors. This interaction (indicated with horizontal arrows in Figure 1) underlies frequency masking or instantaneous masking, which is also a property unique to AIM and to the human cochlea, and which sharpens spectral features, removing irrelevant or noisy features. The output of this second stage mimics the neural activity pattern (NAP in Figure 1) of the auditory nerve, which connects the cochlea to brain stem nuclei.

Further stages apply a strobed integration of the NAP representation, which synchronizes to the period between maxima of the NAP (for voiced speech, this corresponds to the pitch period). This has the effect of regularizing the NAP image relative to the pitch period.

3. AIM-based cepstrum coefficients

The auditory image model described in section 2 was the primary influence on the design of the alternative front end. The usual approach to extracting features from a speech signal which are adapted to ASR focuses attention on the short-time spectral envelope of the speech signal.

The output of certain stages of the AIM model provides useful representations of the short time speech spectrum, but attention must be paid to the special feature requirements of an HMM-based ASR system. The BMM and NAP auditory representations shown above contain a great deal of interesting information, but for an ASR system they are over-detailed in time. The fine temporal information which is generated by the adaptive thresholding of the NAP, for example, generates short-time features on time scales of just a few milliseconds, particularly in the high-frequency bands, and it contains a great deal of information about the fundamental frequency (when the speech is voiced). As a result, it is not particularly well suited to be an ASR feature extractor, because these short-time features are very speaker- and situation-dependent, and because it is computationally expensive. It is well known that the envelope of the spectral-envelope of speech is well represented by a windowing operation with inter-frame period not much smaller than 5 ms or so, and a window size of roughly at least twice this duration. That is, the bulk of the information content of the spectral contour changes on a timescale above 5 ms or so (or put differently, changes occurring faster than this are primarily due to the voicing of the speech), and a window which is much smaller than 10 ms risks losing low frequency information. Further, a much longer window will introduce too much of the fundamental frequency of the speech signal (or its harmonics). We found it useful, in any case, to extract from AIM a representation with coarser time information than is illustrated above.

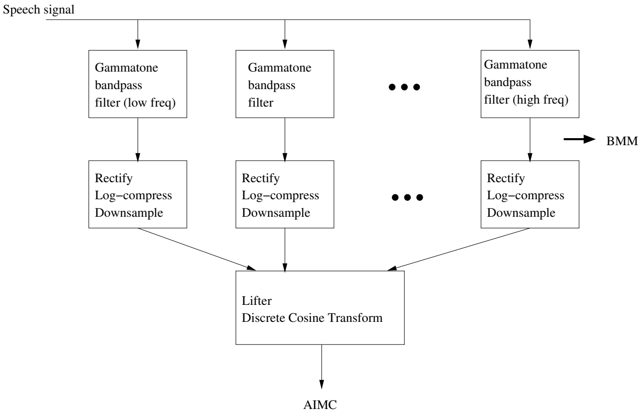
Speech signal
Gammatone bandpass filter (low freq)
Gammatone bandpass filter
•••
Gammatone bandpass filter (high freq)
BMM
Rectify Log−compress Downsample
Rectify Log−compress Downsample
•••
Rectify Log−compress Downsample
Lifter Discrete Cosine Transform
AIMC

Figure 2: AIM Cepstrum (AIMC) computation flowchart.

In the further interest of efficiency, it is important to have a representation with good information-packing properties, and which is well adapted to the statistical model, which is being used. Since we have focused on an ASR system based on HMMs, a natural choice is the Mel cepstrum (MFCC), representing the discrete cosine transform of the log-spectral envelope of the speech signal. The architecture of the AIM-based front end, which was used for the experiments, is illustrated in Figure 2. We took as a starting point the BMM representation, which is offered by AIM. Following some preemphasis of the channel outputs, a local downsampling in time was performed, based on a measurement of the signal in each channel for each of a set of successive overlapping frames. The frame duration is 25 milliseconds, and the inter-frame period is 10 milliseconds. The downsampling measurement is determined by a choice of norm for the framed speech signal. The conventional cepstrum uses the L_1 norm on the frequency domain of the windowed speech. Several norms were tested in our experiments. The best results were obtained with the L_∞ norm and the L_2 norm on the time domain. The L_∞ norm on the time domain is just the maximum of the speech signal on the window domain. (There are typically a number of peaks in each window of the signal, and the magnitude of the largest of these is retained in downsampling.) This produces relatively smoother feature trajectories than either the frequency domain norms or the L_k norms in the time domain for other values of k. After downsampling, the features in each channel are logarithmically compressed, and the first twelve coefficients of the discrete cosine transform across the filter bank are extracted. After some liftering and cepstral mean subtraction, a conventional log-energy term and time-derivative features are appended.

4. Continuous speech recognition: A comparison of MFCC and AIMC

Two continuous speech recognition systems were developed based on the HTK Toolkit [4]. The two systems were identical with the exception that they used different front-end feature sets. The first system employed the conventional mel-frequency cepstrum coefficients (MFCC) and the second one employed the AIM-based cepstrum coefficients (AIMC). Recognition experiments were conducted using the TIMIT [5] database.

4.1. ASR development with HTK

As a continuous speech recognition development platform, we used the state-of-the-art simulation package HMM Tool Kit (HTK, v. 3.0), made available through the Cambridge University Engineering Department. To train a CDHMM recognizer using the HTK toolkit from scratch, one needs to follow these procedures:

- Initial monophone models (43 monophone plus 2 silence phones, sil and sp) were first formed by linearly segmenting the training utterances
- New alignments of the reference transcriptions were obtained using the Viterbi algorithm, word level transcriptions, and the dictionary.
- Triphone models were cloned from the monophone models, based on the set of word-internal triphones appearing in the dictionary. The transition matrices of triphones within each phone class were tied.
- The center states of the short-pause and silence model were tied, and forward and backward skip transitions are added to these models.
- Three rounds of embedded Expectation-Maximum (EM) reestimation were applied.
- A decision tree partitioning triphones according to center phone and left- and right-context classes was used to tie those single Gaussian PDF's sharing a center phone and both right and left context classes. The classes used were: stops, fricatives, nasals, liquids/glides, front vowels, back vowels, center vowels, and silence.
- One round of embedded reestimation was applied to these models.
- The number of Gaussian mixtures was then increased from one to three and then to six.
- Additional one round of embedded EM reestimation was applied to obtain the final recognition system.

This training procedure produces a continuous density, triphone HMM models, with mixture tying to overcome sparse training data for some of the triphones. Each HMM state has 6 Gaussian mixtures. Note that the use of AIMC or MFCC had no effect on the training procedure. However, if a recognizer has been trained using AIMC, it should be tested using AIMC, too. Similarly, if a recognizer has been trained using MFCC, it should also be tested using MFCC.

4.2. The TIMIT Database

The TIMIT database is a phonetically balanced database containing read speech from 630 speakers [5]. It is the first publicly available speech corpus of its size and its fine analysis. The TIMIT database has been commonly used for bootstrapping speech recognition systems.

The TIMIT database is divided into the *training* set and the *testing* set. All the sentences in the training set were used to train the present systems. However, only the SX-sentences in the testing set were utilized in our experiments. SX stands for phonetically-compact sentences. They are designed to provide a good coverage of pairs of phones, with extra occurrences of phonetic contexts thought to be either difficult or of particular interest. Each speaker read 5 of these SX sentences, and each text was spoken by 7 different speakers. In other words, our experiments did not include the testing sentences from phonetically-diverse sentences (the SI sentences) or from the so-called dialect sentences. The dialect distribution of speakers in the used SX test set is given in Table 1. As shown, there are twice as many male speakers as female speakers. Table 1 also shows the number of testing sentences from each dialect region (males + females).

Dialect Region	# male	# female	# sentences
New England	7	4	21
Northern	18	8	59
North Midland	23	3	59
South Midland	16	16	74
Southern	17	11	60
New York City	8	3	23
Western	15	8	51
Army Brat (moved around)	8	3	24
Total	112	56	371

Table 1: Dialect distribution of speakers in the used SX test set.

Furthermore, because our primary purpose was to address the relative advantages between the AIMC feature sets and the MFCC feature sets, we decided to use no language models so that all recognition results would completely rely on the acoustic representations. For this reason, we constructed a null grammar of all the words in the test set. There were a total of 852 unique testing words. As a result of a null grammar, each of these 852 words were equally likely in terms of language modeling scores (namely, 1/852). Only the acoustic modeling scores now contributed to the decoding and hence enable us to more directly assess the superiority between MFCC and AIMC. However, it should be noted that the penalty parameter between the insertion error and the deletion error was tuned to obtain the best performance.

5. Experimental Results

Both for the MFCC-based and AIMC-based systems, the energy term and the first 12 cepstrum coefficients were utilized. In addition, their first and second derivatives were also used. Therefore, the feature vector was composed of 39 elements per frame. Mean normalization of the feature vectors were also performed.

	MFCC	AIMC_∞	AIMC_2
Accuracy (%)	61.4	66.1	67.7
Correct (%)	64.5	67.4	70.2

Table 2: Recognition results, in terms of both word accuracy and word correct (or hit rate).

The experimental results are presented in Table 2. As is seen, MFCC yields an accuracy rate of 61.4% for the task. There are two columns for the AIMC, depending on the norm used to compute AIMC within a frame. AIMC_2 denotes the use of the L_2-norm in computing AIMC, AIMC_∞? denotes the use of the L_∞-norm.

Table 2 indicates that, at least for this experiment, AIMC_2 is more effective than AIMC_∞, which in turn is more effective than MFCC. The recognition accuracy is elevated from 61.4% for MFCC to 67.7% for AIMC_2. That is an absolute 6.3% improvement (or 10% relative improvement).

A number of papers [6, 7, 8, 9] presented word recognition accuracy results for similar experiments on the TIMIT database. Most other papers only gave the phone recognition accuracy. The best system publicly published is the one from Zhao [6]. In her system, MFCC was employed. A great deal of improvements were, however, made on the back end decoder including novel algorithms for combining acoustically similar dictionary words, for context-dependent grammar design, and for the HMM training. Fortunately, Zhao also performed a null-grammar experiment for the SX set of the testing sentences. (She labelled them as SX3-200 test sentences.) The recognition results for this particular experiment are described in Table 3. For comparison purpose, the results of AIMC_2 from Table 2 are also given.

From Tables 2 and 3, it is clear that the results of Zhao is better than the present study when MFCC was used. However, it is also clear that when AIMC was utilized, our results are much better than those of Zhao. And we believe that if we could incorporate those improvements Zhao introduced to her system, we would be able to achieve an even better system.

	Zhao	AIMC_2
Accuracy (%)	62.9	67.7
Correct (%)	66.3	70.2

Table 3: Recognition results from Zhao [6] and from the present study, in terms of both word accuracy and word correct (or hit rate).

6. Conclusions

This paper has presented a novel approach to feature extraction that is inspired by the Auditory Image Model of human psychoacoustics. Testing the front-end on a set of phonetically-compact sentences from the TIMIT dataset shows a relative improvement of 10% in word recognition accuracy. The experimental results were compared with the ones present in the literature (same exact experiment on the same database) and showed an 8% improvement (even though our experiment does not include any of the back-end optimizations that were described in the literature).

7. References

[1] X. Yang, K. Wang, and S. Shamma, “Auditory representations of acoustic signals,” *IEEE Trans. Information Theory*, vol. 38, pp. 824–839, 1992.

[2] R.D. Patterson, M.H. Allerhand, and C.D. Giguere, “Time-domain modelling of peripheral auditory processing: A modular architecture and a software platform,” *Journal of the Acoustical Society of America*, vol. 98, no. 4, pp. 1890–1894, 1995.

[3] R.D. Patterson, M.H. Allerhand, and J. Holdsworth, “Auditory representation of speech sounds,” in *Visual Representation of Speech Signals*, pp. 307–314. John Wiley, 1993.

[4] P.C. Woodland and S.J. Young, “The htk tied-state continuous speech recognizer,” in *Proceedings of EuroSpeech*, 1993.

[5] V. Zue, S. Seneff, , and J. Glass, “Speech database development at mit: Timit and beyond,” *Speech Communication*, vol. 9, pp. 351–356, 1990.

[6] Y. Zhao, “A speaker-independent continuous speech recognition system using continuous mixture gaussian density hmm of the phoneme-sized units,” *IEEE Trans Speech Audio Processing*, vol. 1, no. 2, pp. 345–361, 1993.

[7] Y. Zhao, H. Wakita, and X. Zhuang, “An hmm based speaker-independent continuous speech recognition system with experiments on the timit database,” in *Proc. of Int. Conf. Acoust. Speech and Signal Processing ICASSP92*, 1992, pp. 333–336.

[8] X. Wang, L. Ten Bosch, and L. Pols, “Integration of context-dependent durational knowledge into hmm-based speech recognition,” in *Proc. ICSLP’96*, 1996, pp. 1073–1076.

[9] L. Pols, X. Wang, and L. Ten Bosch, “Modeling of phone duration (using the timit database) and its potential benefit for asr,” *Speech Communication*, vol. 19, pp. 161–176, 1996.