AUDITORY MODELS AS FRONT-ENDS FOR SPEECH RECOGNITION IN HIGH NOISE ENVIRONMENTS

M.D. Chau and C. D. Summerfield

Syrinx Speech Systems Pty Ltd

ABSTRACT -- This paper describes a series of experiments conducted by Syrinx to determine performance improvements offered by Auditory Model based speech signal processing front-ends for HMM recognisers. The experiments tested an implementation of the Ghitza Model connected to a HMM recogniser through a number of interface algorithms that reduces the Auditory Model's representation dimensionality to a manageable size. The results show that in high noise environments recognisers incorporating front-ends based on the Ghitza Auditory Model outperform those implemented using traditional Delta Cepstrum speech processing algorithms.

INTRODUCTION

Over the past 5 years there has been continued interest in the applications of Auditory models to increase robustness of speech recognition in high noise environments. Experiments by Seneff (1985) and Ghitza (1987, 1987) have produced some evidence that Auditory models do improve recogniser performance in high noise conditions.

Syrinx was keen to determine the performance improvement offered by using auditory based front-end signal processing algorithms when compared to conventional speech processing algorithms. In a series of experiments, Syrinx compared the performance of a conventional Delta Cepstrum front-end signal processing algorithm with its implementation of the Ghitza Ensemble Interval Histogram (EIH) model.

The Ghitza model consists of three processing elements. Input speech is applied to a filter bank consisting of 40 band pass filters distributed linearly on a frequency scale from 200 Hz to 6707 Hz. Filter outputs are then passed to a threshold crossing detection processor from which 40 individual histograms are constructed for the whole interval of speech, where each histogram is the integration of the period between threshold crossing detected during the processing interval. The final Ensemble Interval Histogram (EIH) output is the composite of the individual interval histogram outputs.

The EIH is a measure of the periods detected between threshold crossing for each bandpass filter in the front-end filter bank. As a consequence, the EIH is effectively a "periodgram" of the input speech signal, where the x-axis corresponds to period (1/f seconds) and the y-axis corresponds to a sum of threshold crossing periods. As the EIH represents a periodgram, its dimensionality needs to be large to both adequately resolve high frequency components and to obtain the necessary frequency coverage. In Syrinx's implementation the EIH has dimensionality of 200. This is well above the dimensionality of conventional speech recogniser front-ends, which are typically 24.

Although the performance gains offered by the EIH could be informally observed through spectrogram representation and by comparisons of LPC fittings, it was difficult to assess the improvements in recogniser performance in noise, if any, offered by the EIH representation. Informal observations have also established that any reductions in algorithm complexity also lead to a concomitant reduction in performance benefits offered by Ghitza Auditory model. The problem

was to develop an interface algorithm that reduced the dimensionality of the EIH to a manageable size (24) whilst retaining the processing benefits offered by the Ghitza Auditory Model approach.

EXPERIMENTAL SET-UP

Cepstral representations have been demonstrated by a number of research groups to provide good recognition performance in quiet conditions and have become a common technique for front-end signal processing in a number of recognition systems.

In developing an interface algorithm to connect the EIH and the recogniser, two processing steps were (generally) required: warping of the EIH period scale to redistribute the period representation to an auditory frequency scale representation, followed by the application of an orthonormal transform to generate an equivalent lower dimensionality (24) Cepstral representation. Five front-end interface algorithms were developed and investigated:

INTERFACE A:- Straight Delta Cepstrum on the speech data (reference interface);

INTERFACE B:- Cosine transform on the Ghitza periodgram;

INTERFACE C:- Logarithm function period scaling, followed by cosine transform;

INTERFACE D:- Weighting (1/N) period scaling, followed by cosine transform; (where N is the number of periods); and

INTERFACE E:- MEL scale warping followed by data reduction, followed by cosine transform. (INTERFACE E attempts to map the period domain axis onto the MEL scale axis.)

Two small speech databases were used to examine the performance of the different interface designs. These comprised a database of sub-word units collected by NAL for cochlea model testing and a whole word database comprising 30 speakers speaking the military alphabet recorded over the Public Switched Telephone Network.

The sub-word unit database was initially used to determine correspondence between the performance of the Delta Cepstral algorithm and the Ghitza Auditory Model plus the various interface designs. Tests were conducted using recogniser trained on both clean and noisy sub-word unit samples.

Two tests were initial conducted to test functionality and performance of the interfaces. A test on clean speech, where both recognition models and test samples were recorded in quiet conditions, followed by a test where both model and test samples where contaminated with Gaussian noise at 15 db signal-to-noise ratio.

The results are summarised in Table 1.

INTERFACE	CLEAN SPEECH	15 db SIGNAL-TO-NOISE
Interface A	97.22	88.89
Interface B	16.67	41.67
Interface C	75.93	90.74
Interface D	73.15	94.44
Interface E	76.85	96.30

Table 1 Initial test results using sub-word unit recogniser

The initial test clearly showed that in clean speech conditions, the conventional Delta Cepstrum front-end performed significantly better than the Ghitza Model. In noisy conditions, however, performance of the Ghitza model appeared to increased, with only INTERFACE B (Cosine Transform on Ghitza model periodgram output) giving a poorer result recognition using the Delta Cepstrum response.

Using these results, INTERFACES A, C and E were selected for further experimentation using a whole word recogniser. Because of the inordinate processing time required to train and test the Auditory models only a subset of three words (ALPHA, BRAVO, ZULU) was selected from the speaker independent military alphabet database. The recogniser was trained on ten samples of each word in clean and noisy conditions using the following test criteria:

TEST 1 - Recogniser trained and tested on clean samples of the speech.

TEST 2 -- Recogniser trained on clean samples of speech and tested on 10 db signal-to-noise speech samples: and

TEST 3. Recogniser trained on 10 db signal-to-noise samples and tested 10 db signal-to-noise samples.

The recogniser was then tested on the remaining 20 samples from the speech database. The global recognition results are summarised in Table 2.

INTERFACE	TEST 1	TEST 2	TEST 3
INTERFACE A	98.33	48.48	84.76
INTERFACE C	87.14	64.29	94.29
INTERFACE E	94.29	62.86	95.71

Table 2 Results of Whole Word Recognition

COMMENTS

The results of this short series of experiments shown that the Ghitza auditory model connected through a logarithmic or MEL scales frequency warping algorithms and cosine transforms (INTERFACES C and E) provides superior recognition performance for recognition of speech in noise when compared to the conventional Delta Cepstrum algorithm. What is more, the performance advantage appears to be gained whether the recogniser is initially trained on quiet speech or noisy speech samples.

In comparing the results from TESTS 1 and 2 in Table 2, although the Delta Cepstrum shows around 4% and 11% improved recognition performance in quiet conditions, using the same set of recognition models, the Ghitza front-end with a frequency warping outperforms the traditional Delta Cepstrum response by around 15% at 10 db signal to noise ratio. This would suggest that, although the Delta Cepstrum response may be more representative of the speech features in low noise conditions, the periodgram representation produced by the Ghitza Auditory Model is a more robust for high noise environments.

This results is vindicate by TEST 3, where a 10% improvement recognition performance is achieved using the Ghitza Auditory Model over the Delta Cepstrum for a recogniser trained and tested on the same noise conditions. This would suggest at the EIH representation is more robust in high noise conditions.

Although this series of experiments does demonstrate the performance improvements that can be gained from the use of an Auditory Model algorithm based on the Ghitza approach, it is not an exhaustive investigation. However, the results are encouraging and suggest that further investigations into Auditory Model algorithms as front-end speech processor for high noise applications is warranted.

CONCLUSIONS

This paper has described a short series of experiments undertaken by Syrinx to evaluate the performance benefits offered by an Auditory Model front-end for HMM based speech recognition. The results show performance improvements of around 15% can be gained in 10 db signal to noise over conventional Delta Cepstrum processing.

REFERENCES

Ghitza, O. (1987) *Temporal non-place information in the auditory-nerve firing patterns as a front-end for speech recognition in a noisy environment,* Journal of Phonetics (1988) 16, 109-123.

Ghitza, O. (1987) *Robustness against noise: the role of timing-synchrony analysis,* International Conference on Acoustics, Speech and Signal Processing-ICASSP 1987,4,2372-2375.

Seneff, S. (1985) *Pitch and Spectral Analysis of Speech Based on an Auditory Synchrony Model,* Technical Report 504, MIT Research Laboratory of Electronics Cambridge, Massachusetts 02139.

ACOUSTIC FEATURE EXTRACTION FRAMEWORK FOR AUTOMATIC SPEECH RECOGNITION

A. Samouelian

Speech Technology Research Group
Department of Electrical Engineering,
The University of Sydney

ABSTRACT - This paper presents a feature extraction framework that allows the use of speech knowledge in training a phonetic recognition system. It can train on any combination of features that may be derived from time and/or frequency domains, parametric, acoustic-phonetic and auditory models including speech specific features. The system requires a moderate size, phonetically labeled database. During the training phase, nominated features per frame are automatically extracted and used as a set of attributes to generate a recognition decision tree, using c4.5 Induction Program. During recognition, the feature extraction framework generates the set of attributes, which are then fed through the decision tree, which assigns a phonetic label to each frame. Recognition results on the class of semi_vowels are presented.

INTRODUCTION

Recognition systems vary in their approach to speech knowledge. There are systems that use heuristic rules, which are developed from intense knowledge engineering, to develop acoustic to phonetic mapping and to emulate the spectrogram reading capabilities of a trained phonetician [O'Kane, 1983; Zue, 1985]. Feature extraction techniques, which are used to segment and label the speech signal, are developed using acoustic theory of human speech production and the ability of trained phoneticians to identify sounds or phonemes directly from spectrograms. In reality this is a fairly difficult task, since it requires the capturing of all the complex interrelationships in speech sounds. Others use template matching and stochastic modeling systems that generally ignore acoustic features or make no use of speech specific knowledge and instead rely on spectral representation of the speech signal to either create reference templates or develop stochastic models. These systems require a large database to develop good representative models of the speech signal. While others express speech knowledge within a formal framework using well defined mathematical tools, where features and decision strategies are discovered and trained automatically, using a large body of speech database [Zue et al, 1989].

Although knowledge engineering (developing specific rules to interpret the extracted features and provide the mapping to its corresponding phonetic label) is manageable for a small vocabulary and isolated words systems, for large vocabulary systems, which require phonetic recognition, a large body of rules is required (developed from examination of hundreds of speech waveforms and their spectrograms). These rules utilize enormous number of acoustic-phonetic, lexical, syntactic, semantic and prosodic facts and the subtle interaction between them makes this task truly formidable.

To parametize the speech signal, most recognizers use the speech production model, which separates excitation and vocal tract response. For each frame, excitation is typically represented by an overall amplitude or energy term. For the spectral representation 8-14 coefficients are generally used to represent the spectral parameters. These coefficients are usually derived from Linear Predictive Coding (LPC) analysis, Fourier Transform, or bank of bandpass filters. Common parameters are

LPC coefficients, Mel Frequency Cepstral Coefficients (MFCC) and energies in the filterbank. These parameters are classified as features and used to train the recognizer.

A new feature extraction framework is presented that allows any combination of features derived from time and/or frequency domains, parametric, acoustic-phonetic and auditory models including speech specific features to be automatically extracted from input speech signal. These features are used as a set of attributes to train and generate a decision tree, using c4.5 Induction Program (Quinlan, 1983).

INDUCTIVE INFERENCE

Inductive inference has been used to extract classification knowledge from large data bases and collections of examples (Quinlan, 1983; Quinlan et al, 1986). Inductive inference produces decision trees that use attributes that provide the most information about classification are chosen as discriminating attributes. In cases when all or the majority of attributes are numeric, induction can produce a very large and unnecessarily complex decision trees. To simplify the complex tree, branches which do not contribute significantly to the accuracy are pruned off (Quinlan, 1987). The problem with this approach is that a significant branch may be pruned off specially if there are no sufficient examples in the training set. The accuracy of these trees can be greatly increased by using Ripple Down Rules to maintain the tree after induction (Horn, 1991).

Some of the advantages of using inductive learning technique are:

- Examination of database containing many examples allows generalizations.
- A decision tree can be generated using any set of attributes without discriminating between rule based or parametric features.
- Parametric features such as LPC or MFCC coefficients are not well suited for rule based systems, since it is difficult to explicitly associate coefficient values with acoustic or phonetic events. The inductive system can easily examine all of the database and set up appropriate thresholds to generate a decision tree and a set of rules for phonetic classification.
- It allows the true integration of features from existing signalling processing techniques that have proven to produce good results in stochastic modeling, and at the same time allows the incorporation of speech specific knowledge into the decision tree.
- It allows the development of decision trees in planned refinement of the rules if the performance is inadequate. This is achieved through hand modification of the decision tree by changing the features or the combination of features used to classify a specific sound or phoneme class. The planned refinement of the rules and the inductive learning technique should make the task of rule based system manageable and provide a productive tool for evaluating the feature sets, assessing the performance of the recognizer and monitoring the incremental improvement in recognition accuracy as a function of the combination of features.

TRAINING AND RECOGNITION STRATEGY

The feature extraction framework allows the extraction of nominated set of features from the input speech signal and creates the appropriate "data" and "bulk" files for training and testing of the recognition system respectively. The "data" file contains all the attributes per frame with the appropriate phoneme labels appended to the end, while the "bulk" file contains only the attributes per frame.

During the training phase, using c4.5 Induction program, a decision tree is generated from the "data" file. During the recognition phase, the "bulk" file is classified by the decision tree, which involves