Embedded Knowledge-based Speech Detectors for Real-Time Recognition Tasks

Sabato M. Siniscalchi[1,3], Fulvio Gennaro[1], Salvatore Andolina[1], Salvatore Vitabile[2,4], Antonio Gentile[1], and Filippo Sorbello[1,4]

[1] *Dipartimento di Ingegneria Informatica, Università di Palermo*
V.le delle Scienze (Edif. 6), 90128 Palermo, Italy

[2] *Dipartimento di Biotecnologie Mediche e Medicina Legale, Università di Palermo*
Via del Vespro, 90127 Palermo, Italy

[3] *Center for Signal and Image Processing, School of Electrical and Computer Engineering*
Georgia Institute of Technology, Atlanta, Georgia 30332, USA

[4] *Istituto di CAlcolo e Reti ad alte prestazioni – Consiglio Nazionale delle Ricerche*
V.le delle Scienze (Edif. 11), 90128 Palermo, Italy

marco@ece.gatech.edu, {vitabile, gentile, sorbello}@unipa.it

Abstract

Speech recognition has become common in many application domains, from dictation systems for professional practices to vocal user interfaces for people with disabilities or hands-free system control. However, so far the performance of Automatic Speech Recognition (ASR) systems are comparable to Human Speech Recognition (HSR) only under very strict working conditions, and in general much lower. Incorporating acoustic-phonetic knowledge into ASR design has been proven a viable approach to raise ASR accuracy. Manner of articulation attributes such as vowel, stop, fricative, approximant, nasal, and silence are examples of such knowledge. Neural networks have already been used successfully as detectors for manner of articulation attributes starting from representations of speech signal frames. In this paper the full system implementation is described. The system has a first stage for MFCC extraction followed by a second stage implementing a sinusoidal based multi-layer perceptron for speech event classification. Implementation details over a Celoxica RC203 board are given.

1. Introduction

In [1] the authors proposed a real time implementation of a bank of Multi Layer Perceptron (MLP) with sinusoidal activation function to detect speech attributes, namely fricative, vowel, stop, nasal, approximant, and silence. Inside the speech community, these aforementioned attributes are referred to as manner of articulation events, and they are strongly related to human speech production [2]. Moreover, they show robustness to speech variations [3]. These speech attributes are generated directly from Mel-Frequency Cepstrum Coefficients (MFCCs), and the six detectors actually perform a sort of mapping from the acoustic domain (MFFCs) to the articulatory domain. The term ‘‘mel’’ denotes a measurement of perceived frequency of a tone, which does not vary linearly with the physical frequency of the corresponding tone. A non linear scale is employed since it was found that human auditory system does not perceive pitch in linear manner. The mapping between the real frequency scale (Hz) and the perceived frequency scales (mels) is given in formula (1)

$$F_{mel} = 2595\log(1+\frac{F_{HZ}}{700}) \quad (1)$$

The mapping is approximately linear below 1KHz, and logarithmic at higher frequency, and such an approximation is usually adopted in speech recognition.

In this paper we propose the chip design for the entire system, aimed at embedded applications. Our interest in generating the manner of articulation system is because it is part of the Automatic Speech Attribute Transcription (ASAT) project [4], in which a software

neural network-based architecture for these manner of articulation attributes was already implemented in [5].

The main idea of the ASAT project is that the performance of conventional *knowledge-ignorant* modeling approaches can be improved integrating the knowledge sources available in a large body of speech science literature. In [3] it is showed that the idea of a direct incorporation of acoustic-phonetic knowledge into ASR design raises its accuracy. These "knowledge-based" features (also referred to as **speech attributes** in the same work) are used to augment the front-end module of a conventional ASR system by means of a set of feature detectors able to capture the speech attributes.

The rest of the paper is organized as follows. Section 2 describes the general framework of the event detector module, which we will call *knowledge extraction* to be consistent with the nomenclature used in [1]. In sections 3 and 4 the MFCCs and its digital implementation are given respectively. An overview of the digital implementation of the six MLP detectors is shown in section 5. Section 6 presents the experimental set-up and results with comparison to the baseline architecture. Concluding remarks are given in the last section of the paper to summarize its main contributions.

2. Knowledge Extraction Module

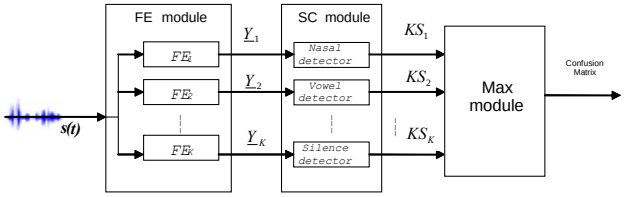
The Knowledge Extraction (KE) module uses a frame-based approach to provide K manner of articulation attributes A_i, where i=1,2, … K, from an input speech signal *s(t)*. In this paper the manner classes were chosen as in [6], and are listed in Table 1.

The KE module, depicted in Figure 1, is composed of two fundamentals blocks: the feature extraction module (FE), and the attribute scoring module (SC). The FE module consists of a bank of K feature extraction blocks FE_i, where i=1,2, … K, and it maps a speech waveform into a sequence of speech parameter vectors $\mathbf{Y}_i$, i=1,2, … K. Actually, each of the FE_i is fed by the same speech waveform *s(t,)* and for each speech-frame it computes a thirteen MFCC feature vector $\mathbf{X}_i$ (12 MFCCs + Energy). The frame length is of 30 msec overlapped by 20 msec. Finally, FE_i produces as output a 117-feature vector $\mathbf{Y}_i$ combining the actual frame with the eight surrounding frames, 4 frames before and after, so that each speech parameter vector represents nine frames.

The SC module is composed of six feed-forward neural networks, and its goal is to attach a score, referred to as *knowledge score* (KS_i), to each vector $\mathbf{Y}_i$. The input of each network is a 9 frames of 12 MFCCs + energy, so that the input layer is of 117 nodes. The output layer has two nodes, one for the desired class, and one for the anti-class. Actually, the value obtained for the desired class for case *i* is defined to be the KS_i.

Manner of articulation attribute definition

Articulation Manner	Class Elements	Anti-Class Elements
Vowel	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX	JH, CH, S, SH, Z, ZH, F, TH, V, DH, B, D, G, P, T, K, DX, M, N, NG, EN, L, R, W, Y, HH, EL, SIL
Fricative	JH, CH, S, SH, Z, ZH, F, TH, V, DH	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX, B, D, G, P, T, K, DX, M, N, NG, EN, L, R, W, Y, HH, EL, SIL
Stop	B, D, G, P, T, K, DX	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX, JH, CH, S, SH, Z, ZH, F, TH, V, DH, M, N, NG, EN, L, R, W, Y, HH, EL, SIL
Nasal	M, N, NG, EN	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX, JH, CH, S, SH, Z, ZH, F, TH, V, DH, B, D, G, P, T, K, DX, L, R, W, Y, HH, EL, SIL
Silence	SIL	IY, IH, EH, EY, AE, AA, AW, AY, AH, AO, OY, OW, UH, UW, ER, AX, IX, JH, CH, S, SH, Z, ZH, F, TH, V, DH, B, D, G, P, T, K, DX, M, N, NG, EN, L, R, W, Y, HH, EL
Approximant (App.)	L R W Y EL	IY IH EH EY AE AA AW AY AH AO OY OW UH UW ER AX HH IX JH CH S SH Z ZH F TH V DH B D G P T K DX M N NG EN SIL



. Knowledge Extraction Module, adapted from[6]. The detectors are based on a MLP neural network.

3. Mel-Frequency Cepstrum Coefficients Extractor

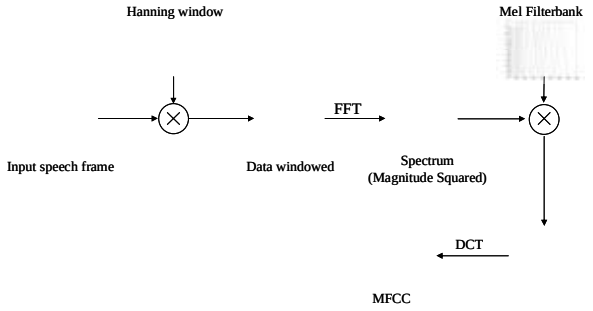
In the feature extraction phase a set of useful parameters termed as Mel-frequency cepstrum coefficients (MFCC) are extracted directly from the speech waveforms. To compute the MFCCs, the speech

waveform of the input utterance is partitioned into sequence of consecutive frames using windowing analysis. For each frame, the vector of mel frequency cepstrum coefficients are extracted from the frame samples. The resulting sequence of feature vectors represents the input utterance.

The general form of this filter bank is illustrated in Figure 2. As can be seen the filters used are triangular and they are not equally spaced along the mel-scale but which is defined by equation (1).

. Triangular weighted functions in frequency domain.

The block diagram of the entire process is depicted in Fig. 3.



. Block diagram of the entire MFCC extraction module.

A description of each individual step is given below.

Step 1: Frame Blocking

In this step the continuous speech signal is blocked into frames of N samples, with adjacent frames being separated by M (M < N). This process continues until all the speech is accounted for within one or more frames. Typical values for N and M are N = 256 (which is equivalent to ~ 30 msec windowing and facilitate the fast radix-2 FFT) and M = 100.

Step 2: Windowing

The next step in the processing is to window each individual frame so as to minimize the signal discontinuities at the beginning and end of each frame. If we define the window as *w(n) with 0 n N-1*, where N is the number of samples in each frame, then the result of windowing is the signal

$$y_l(n) = x_l(n)w(n), \quad 0 \le n \le N-1 \qquad (2)$$

Step 3: Fast Fourier Transform (FFT)

The next processing step is the Fast Fourier Transform, which converts each frame of N samples from the time domain into the frequency domain. The FFT is a fast algorithm to implement the Discrete Fourier Transform (DFT) which is defined on the set of N samples {xn}, as follow:

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N}, \qquad n = 0,1,2,...,N-1 \qquad (3)$$

Step 4: Mel-frequency Wrapping

An approach to simulate the human being auditory system is to process the spectrum S(ω) of Xn by a filter bank spaced uniformly on the mel scale (see Figure 2). That filter bank has a triangular bandpass frequency response, and the spacing as well as the bandwidth is determined by a constant mel frequency interval. The number of mel spectrum coefficients, K, is typically 20.

Step 5: Cepstrum

In this final step, we convert the log mel spectrum back to time. The result is called the mel frequency cepstrum coefficients (MFCC). Because the mel spectrum coefficients (and so their logarithm) are real numbers, we can convert them to the time domain using the Discrete Cosine Transform (DCT). Therefore if we denote those mel power spectrum coefficients that are the result of the last step $\tilde{S}_k$, $k = 1,2,...,K$, we can calculate the MFCC's ($\tilde{c}_n$) as

$$\tilde{c}_n = \sum_{k=1}^{K} (\log \tilde{S}_k) \cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K} \right], \qquad n = 1,2,...,K \qquad (4)$$

3.1. Implementation on FPGA

The front-end has been implemented and prototyped onto a Celoxica RC203 board equipped with a Virtex II XC2V3000-4 donated by Xilinx. The extractor is

described using a C-like hardware description language, Handel-C, developed by the *Oxford Hardware Compilation Group* at the University of Oxford (UK).

The actual prototype runs at a rather low operating frequency of 12.5 MHz, operating on 10 ms windows and requiring 2.1 ms per input frame.

4. Feed Forward Neural Network digital design

In [7] an efficient MLP digital implementation for road signs recognition and high energy physics experiments classification has been proposed. This initial design has been adapted and optimized for automatic speech classification and is presented in this section.

A single MLP digital architecture is used to implement each of the detectors described in Figure 1. As depicted in Figure 4, this architectural design aims to satisfy high design modularity, high density of neurons on device, high recognition rate and speed. As a result, (a) data input acts in a serial way; (b) data processing acts in parallel among the neurons and serially within each neuron; (c) second layer processing is pipelined with first layer processing. The Winners Takes All (WTA) circuit selects, among a set of *m* numbers, the greatest activation level units.

. Functional block diagram of the MLP architecture

The basic digital neural network elements, as multipliers and accumulators, are designed following the standard solutions. The output activation function is a linear function, whilst sinusoidal activation function is employed as activation function of the hidden layer. Fixed point arithmetic with two's complement representation is used for the chip implementation of the MLP. Principal constrains of this project are the compromise between the neural network accuracy and the bit depth for input and weight data, and the compromise between the neural network accuracy and the bit depth for the pre-synaptic value and the post-synaptic value of the hidden activation function.

5. Experiments and results

The evaluation of the proposed Manner of Articulation Extraction module was performed on the TIMIT Acoustic-Phonetic Continuous Speech Corpus database [8], which is a well-known speech corpus in the speech recognition field. This database is composed of a total of 6300 sentences; it has a one-channel, 16-bit linear sampling format, and it was sampled at 16000 samples/sec. The MLP detectors were trained on 3504 randomly selected utterances, and to be consistent with [3] and [9] the four phones "cl", "vcl", "epi", and "sil" were treated as a single class, thus reducing the TIMIT phone set to a set of 45 context-independent (CI) phones. The front-end module is in the process of being implemented following the guidelines given in [10]. Instead the max module is a simple comparator circuit. The MLP module is the focus of this work, and a detailed description is given in what follows.

Each of the six detectors is a three-layer network the input of which is a window of nine frames, that is, 117 parameters. The nodes of hidden layers are 100. The output layer contains two units, and a simple linear activation function is used. Finally, the max module applies a max function to the KS_i outputs in order to compute the overall confusion matrix.

As previously stated, the detectors work in a frame-based paradigm, so that their performance was evaluated in term of frame error rate. Each frame was classified according to the neural network with the largest value.

Hardware phoneme percentages accuracies for the manner of articulation attributes using sinusoidal activation function

%	Vowel	Fricative	Stop	Nasal	App.	Silence
Vow.	**89,85**	1,38	1,53	1,26	4,64	0,19
Fric.	3,16	**87,02**	5,53	1,02	0,89	1,24
Stop	6,32	7,41	**79,89**	1,71	1,57	1,96
Nas.	9,65	2,44	3,25	**81,04**	2,20	0,90
App.	30,82	2,88	3,26	2,74	**58,07**	1,19
Sil.	1,10	1,09	1,88	0,61	0,58	**94,21**

The global confusion matrix for the manner of articulation attributes is given in Table 2. The (p, q)-th

element of the confusion matrix measures the rate of the p-th attribute being classified into the q-th class.

The digital version Knowledge-based Automatic Speech Classifier is implemented on Celoxica RC203 board [11] equipped with a Xilinx VirtexII XC2V3000-4 FPGA. Neural architectures were described using the VHDL language and were synthesized using the Xilinx ISE 6.3 tools.

According with the results reached in [7], the number of hidden virtual neurons for each of the MLPs has been fixed to 10, representing the best trade-off between execution time and allocated resource. The above MLP digital implementation requires 1187 cycles and, consequently, 0,0236ms for its execution. Combined with the 2 ms execution of the front-end, the execution time clearly allows for real-time execution. Table 3 illustrates the synthesis report for the MFCC Extractor Module, for the entire scoring module and the total allocated resources required by the entire system. It is easy to see that the chosen configuration for each MLP allows the implementation of the 6 detectors in a single FPGA.

. Synthesis report for the MFCC Extractor Module, for the entire scoring module and the total allocated resources required by the entire system

Available Resources	Slices	FFs	LUTs	RAMs
	14336	28672	28672	96
MFCC Extractor	6439	1319	11205	3
	44,9%	4,6%	39,1%	3,1%
MLP scoring module	4830	4058	8234	60
	33,7%	14,2%	28,7%	62,5%
Total Resources	11269	5377	19439	63
	78,6%	18,8%	67,8%	65,6%

Implementation results on FPGA show that use of sinusoidal activation functions decrease hardware resource usage of more than 50% for slices, FFs, LUTs and of more than 35% for FPGA RAM when compared with the standard sigmoid-based neuron implementation. Furthermore, neuron virtualization allows for a significant decrease of concurrent memory access, resulting in improved performance for the entire attribute scoring module [7].

6. Summary

The performance of Automatic Speech Recognition (ASR) systems are comparable to Human Speech Recognition (HSR) only under very strict working conditions, and in general far lower. Incorporating acoustic-phonetic knowledge into ASR design has been proven a viable approach to raise ASR accuracy. Manner of articulation attributes such as vowel, stop, fricative, approximant, nasal, and silence are examples of such knowledge. Neural net-works have already been used successfully as detectors for manner of articulation attributes starting from representations of speech signal frames.

The preliminary experimental results offer good evidence of the real-time capability of the system. and they demonstrates its implementation on embedded devices as part of full speech recognition systems.

In this paper an embedded knowledge-based speech detectors for real-time execution is described. The system has a first stage for MFCC extraction followed by a second stage implementing a sinusoidal based multi-layer Perceptron for speech event classification. Implementation details over a Celoxica RC203 board have been given.

Execution time for the entire system is slightly above 2 ms per frame and allows for real-time speech event classification on embedded devices.

Currently research works underway to incorporate the other stages for full large dictionary speech recognition embedded IP engine.

7. References

[1] S.M. Siniscalchi, F. Gennaro, S. Vitabile, A. Gentile, F. Sorbello (2005). “Efficient FPGA Implementation of a Knowledge-based Automatic Speech Classifier”, L.T. Yang et al. (Eds.): ICESS 2005, Lecture Notes in Computer Science, vol. 3820, pp. 198-209, Springer-Verlag.

[2] K. Kirchhoff. "Combining Articulatory and Acoustic Information for Speech Recognition in Noisy and Reverberant Environments", Proc. of the International Conference on Spoken Language Processing, Sydney, Australia, pp. 891-894

[3] J. Li, Y. Tsao and C.-H. Lee, “A Study on Knowledge source integration for candidate rescoring in automatic speech recognition,” Proc. of ICASSP05.

[4] Lee, C.-H., “From knowledge-ignorant to knowledge-rich modeling: a new speech research paradigm for next generation automatic speech recognition,” Proc. ICSLP, 2004.

[5] S. M. Siniscalchi, J. Li, G. Pilato, G. Vassallo, M. A. Clements, A. Gentile, F. Sorbello, “Application of E-αNets to Feature Recognition of Manner of Articulation in Knowledge-based Automatic Speech Recognition,” Proceeding of the Italian Workshop on Neural Nets (WIRN 2005), Springer-Verlag.

[6] M. Porrmann, U. Witkowski, H. Kalte, U. Ruckert, “Implementation of Artificial Neural Hardware Accelerator”, 10th Euromicro Workshop on

Parallel, Distributed and Network-based Processing, pp.243-250, January 9-11, 2002, Spain.

[7] S. Vitabile, V. Conti, F. Gennaro, F. Sorbello (2005). “Efficient MLP Digital Implementation on FPGA”, 8° EUROMICRO Conference on Digital System Design (DSD 2005), pp. 218-222, IEEE Computer Society Press.

[8] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, “DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus,” U.S. Dept. of Commerce, NIST, Gaithersburg, MD, February 1993.

[9] K. F. Lee and H. W. Hon, “Speaker-independent phone recognition using hidden Markov models”, IEEE Trans. On Acoust., Speech and Signal Process., Vol. 37, No. 11, pp. 1641-1648, 1989.

[10] J.-C. Wang et al, Chipdesign of MFCC extraction for speech recognition, INTEGRATION, the VLSI journal 32 (2002) 111–131)

[11] RC203 Software Manual http://www.celoxica.com/support/documentation