Recent improvements in SRI's Keyword Detection System for Noisy Audio

Julien van Hout, Vikramjit Mitra, Yun Lei, Dimitra Vergyri,
Martin Graciarena, Arindam Mandal[1]*, Horacio Franco*

Speech Technology and Research Laboratory, SRI International, Menlo Park, California, USA
julien@speech.sri.com

Abstract

We present improvements to a keyword spotting (KWS) system that operates in highly adverse channel conditions with very low signal-to-noise ratio levels. We employ a system combination approach by combining the outputs of multiple large vocabulary continuous speech recognition (LVCSR) systems. These systems are complementary thanks to different design decisions across all levels of information: three speech activity detections systems; a wide range of front-end signal processing features (standard cepstral and filter-bank features, noise-robust features and multi-layer perceptron features); three statistical acoustic model types (Gaussian mixtures models, deep and convolutional neural networks); two keyword search strategies (word-based and phone-based). We explore the scenario where the keywords are known in advance by adding them to the language model and assigning higher weights to n-grams with keywords in them. The scores of each individual system are fused by a logistic-regression based classifier to produce the final system combination output. We present the performance of our system in the Phase III evaluations of DARPAs Robust Automatic Transcription of Speech (RATS) program for Levantine Arabic and Farsi conversational speech corpora.

Index Terms: keyword spotting, spoken term detection, automatic speech recognition, system combination, noise robustness

1. Introduction

The task of Keyword Spotting (KWS) has gotten increasing attention in the last few years as a tool to index large and possibly noisy audio corpora and retrieve specific spoken terms or short sentences. State-of-the art systems typically employ sequence modeling approaches that use hidden Markov models (HMM) to model keywords and all other words (the *garbage model*). HMM-based approaches can be grouped into four categories: whole-word or acoustic KWS, which models entire keywords and other words (*garbage words*) as HMMs [1]; phonetic KWS, which uses HMMs to model phone-level (or triphone-level) representations of keywords and ergodic HMMs for garbage words [2]; ASR-based KWS, which uses standard HMM-based ASR to produce word-level lattices that are represented as indices for keyword search [3]; and hybrid KWS, which combines phonetic and ASR-based KWS approaches to produce sub-word lattices for generating keyword search indices. A detailed survey of existing KWS techniques can be found in [4, 5].

In this work, we present improvements to our DARPA RATS KWS system targeting channel-degraded conversational speech with signal-to-noise ratios (SNR) ranging from 0-20dB. In order to achieve competitive performance on such degraded audio, we heavily rely on system combination of diverse ASR-based KWS systems. The architecture of our system is shown in Figure 1.

Each KWS sub-system used differed from the rest in at least one of four components: speech activity detection, signal processing feature, statistical acoustic modeling technique and KWS search space representation. For signal processing we used stadard cepstral-based features, such as mel-frequency cepstra coefficients (MFCCs) and perceptual linear prediction (PLP) coefficients, and also noise-robust features based on normalized modulation of cepstra and Gabor/Tandem posteriors, as described in section 3.2. For acoustic modeling we used standard Gaussian mixture models (GMMs), deep neural network models (DNNs) and convolutional neural networks (CNNs). As KWS search space we used either word lattices or phone lattices produced by the LVCSR sub-systems. We considered two scenarios: one where the keywords are known ahead of time and can be used during ASR decoding, and the other where the KWS index is generated without knowledge of the keywords. In the keyword-aware scenario, we add the keywords to the language model as multi-words and assign higher weight to those n-grams.. We fused the individual systems KWS output using a logistic-regression based fusion with two input features: posterior-based and rank-based. In this study, we present analysis of the KWS performance of the above-mentioned system design alternatives that led to our final system configuration.

2. Corpora and Task

The speech corpora used in this study were collected under DARPA's RATS program by the Linguistic Data Consortium (LDC), and include speech in noisy or heavily distorted channels in two languages: Levantine Arabic and Farsi. The amount of training data used in acoustic and language modeling, as well as the test sets were unchanged from the 2013 evaluation, see [6] for details. For each language, we used two test sets, each consisting of 10 hours of held-out conversational speech, and referred to as **alv dev-1** and **alv dev-2** for Levantine Arabic and **fas dev-1** and **fas dev-2** for Farsi. A set of 200 keywords were pre-specified for each language. Each keyword was composed of up to three words, was at least three syllables long, and appeared three times on average in the test set.

3. Speech Recognition system

3.1. Speech Activity Detection

We tuned the speech activity detector (SAD) [7] developed under the RATS project for the KWS task specifically. The SAD module was trained in three configurations: using either MFCC, MMeDuSA and PLP features plus 3 voicing features. The 3 voicing features were the COMBO feature [8], the MBCombF0

[1] The author did this work while he was with SRI. He is now with Amazon.

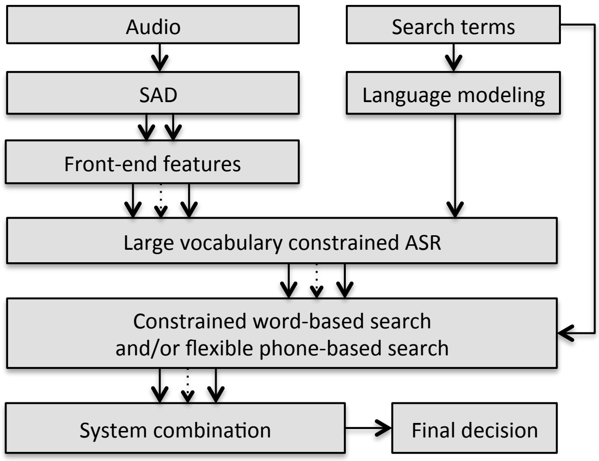
Audio
Search terms
SAD
Language modeling
Front-end features
Large vocabulary constrained ASR
Constrained word-based search
and/or flexible phone-based search
System combination
Final decision

Figure 1: KWS system architecture

feature [9] and the SAcC feature [10]. Each SAD was composed of two AMs, one for speech and one for non-speech, each with a 256 Gaussian mixture full covariance GMM. Long range modeling was performed using a DCT with a multi-frame window. The input feature was 81 dimensional. Log-likelihood ratio (LLR) smoothing was used with a 71 frame window. Padding of 0.1 sec was added to the estimated segment boundaries. There were multiple differences between the current SAD and the one described in [6]. The first difference is in the decoding scheme: the old SAD used a Viterbi decoding based on HMM models, while the current one uses LLR computation with smoothing from GMM models. Another difference is in the long range modeling. The old SAD used deltas, whereas the current one uses DCT. Finally, there are small differences in the voicing features used.

3.2. Noise Robust Features

We explored an array of robust features for our KWS experiments, motivated by human auditory perception and robust signal processing. We have time-contextualized all the features using a context of 7 frames on either side of a given frame, resulting in a concatenation of 15 frames altogether. Interestingly, we observed that for some of the features, vocal tract length normalization (VTLN) helped to improve the KWS performance; hence VTLN was performed for NMC, DOC and GFC features

Damped Oscillator Coefficients (DOC): DOC [11] aims to model the dynamics of hair cell motion within the human ear. The hair cells detect the motion of incoming sound waves and excite the neurons of the auditory nerves. In DOC processing, the incoming speech signal is analyzed by a bank of 40 gammatone filters spaced equally on the equivalent rectangular bandwidth (ERB) scale. The resulting sub-band signals were used as the forcing functions to an array of damped oscillators whose response was compressed using the 15th root and used as an acoustic feature.

Normalized Modulation Coefficients (NMC): NMCs [12] are motivated by the fact that amplitude modulation (AM) of subband speech signals plays an important role in human speech perception and recognition [13]. The speech signal was analyzed using a time-domain gammatone filterbank with 40 channels spaced equally on the ERB scale. The subband signals from the gammatone filterbanks were then processed using the Discrete Energy Separation algorithm (DESA) [14], which produced instantaneous estimates of AM signals. The powers of the AM signals (NMC) were then root compressed using the 15th root (NMC+pn), and resulted in 40 dimensional features

Modulation of Medium Duration Speech Amplitudes (MMeDuSA): Like the NMCs, MMeDuSA features also track the subband AM signals of speech, but they use a medium duration analysis window of 51 ms and also track the overall summary modulation. The summary modulation plays an important role in tracking speech activity as well as in locating events such as vowel prominence/stress, etc. [15, 16]. A time-domain gammatone filterbank with 40 channels was used to compute the band-limited signals on which we applied the Teager energy operator [17] to extract the AM, before applying 15th root compression. Additionally, the AM signals from the subband channels were bandpass-filtered to retain the modulation information within the range 5 to 200 Hz, which was then summed across the frequency scale to produce a summary modulation signal. The 15th root-compressed power signal of the modulation summary yields 11 dimensional features, which were concatenated with 40 dimensional features from the previous step to produce 51 dimensional MMeDuSA features.

Gammatone Filter Coefficients (GFCs): Gammatone filters are a linear approximation of the auditory filterbank performed in the human ear. The power of the bandlimited time signals within an analysis window of 26 ms was computed at a frame rate of 10 ms. Subband powers from 40 filters were then root compressed using the 15th root

Log-spectrally Enhanced Power Normalized Cepstral Coefficients (LSEN-PNCC): The LSEN feature was initially introduced in [18] for enhancement of the mel-spectrum with applications to noise-robust speech recognition. It was adapted in [19] to enhance the gammatone power-normalized spectra of noisy speech obtained with the PNCC features pipeline [20], and renamed LSEN-PNCC.

Gabor-MFCC: The Gabor/Tandem posterior [21] features use a mel-spectrogram convolved with spectro-temporal Gabor filters at different frequency channels. A multi-layer perceptron (MLP) is used to predict monophone class posteriors of each frame, which were then Kurhunen-Loeve transformed to 22 dimensions and appended with standard 39-dimensional MFCCs to yield 64 dimensional features.

3.3. Acoustic Modeling

We pooled training data from all eight noisy channels to train multi-channel acoustic models of three types, all of which used three-state left-to-right HMMs to model crossword triphones. The models differed in the way they model the HMM state output probability: one used standard GMMs, and the others used DNNs or CNNs. The training corpus was clustered into pseudo-speaker clusters using unsupervised agglomerative clustering. The features were also transformed using heteroscedastic linear discriminant analysis (HLDA). For GMMs, we trained speaker-adaptive maximum likelihood (ML) models which were then speaker-adapted using maximum likelihood linear regression (MLLR). We performed cross-adaptation of systems by exchanging the MLLR reference hypotheses between the CNN, DNN and GMM systems, each of which used a different front-end feature. For the GMM system, we used SRI International's Decipher™ ASR engine [22]. The CNNs were trained using cross entropy on the alignments from the HMM-GMM. Five hidden layers of size 2048 were used for the DNNs. For CNNs, the additional convolutional layer included two hundred convolutional filters of size 8, where the pooling size was set to three without overlap.

3.4. Language modeling

We developed two types of language models: keyword-aware and keyword-agnostic. The keyword-aware scenario assumed that the keywords were known at decoding time, in which case we used an approach identical to the one described in [6]. In the keyword-agnostic scenario, knowledge of the keywords was not used to alter the LM training procedure.

4. Keyword Search

This section describes how we produced KWS search indices and keyword detections from the ASR lattices for each individual system, and how we fused those outputs.

4.1. Word-level Search

In the keyword-aware scenario, the keywords can be hypothesized in the ASR lattices, either as sequences of words or directly as a multiword-keyword. The approach to obtain those keyword detections was previously described in [6].

4.2. Phone-level Search

We also converted the word lattices of each system to phone lattices which were used to build an index using weighted finite state transducers (WFST), as described in [23]. Each lattice is represented as a factor transducer, in which each path represents a sub-sequence of phones in the original lattice, a structure which allows for very efficient search for queries. The query terms were represented as weighted finite state acceptors (WFSA), obtained by composing the WFSA obtained from the exact term pronunciation with a WFST encoding phone confusions. The search can be performed by composing the index and query automata and sorting the paths through the resulting transducer with the shortest-path algorithm. All the FST manipulations were realized using the OpenFST library [24]. Allowing for phone confusion during search can recover some of the ASR errors by hypothesizing keywords through phone sequences not present in the lattice. A phone-deletion penalty of 0.01 significantly improved KWS performance, but we did not find helpful to allow for insertions and substitutions.

4.3. System Normalization and Combination

Score-level system combination of KWS search indices was performed using the logistic regression framework described in [25, 6]. Binary side-information was appended to the feature vector to account for individual system's missing scores.

We also experimented with a secondary feature, a non-linear version of the keyword rank. First used in [26] as a replacement for the posterior score, the rank r of a keyword in a given corpus is defined by the number of detections of that keyword with a higher posterior in that corpus. While the relation between posterior and rank is monotonic, it is keyword-dependent and provides a feature that can be used to offset an ASR system's inherent tendency to over- or under-hypothesize certain keywords. For each keyword, a non-linear mapping from posterior to rank was trained on the dev1 set using one of the CNN systems, and was used in all fusion experiments. Because of the different dynamic range of logit-posteriors and ranks, we found it beneficial to use a non-linear mapping of the rank such as $r \rightarrow log(4 + r)$. While [26] uses the rank as a replacement for the posterior, we use it in combination with the keyword posterior in our logistic regression framework.

5. Results & Discussion

In this section, we present our results in terms of the P_{miss} and P_{fa} KWS metrics used under the RATS program. Evaluation was performed on the **alv dev2** set for Levantine Arabic and on the **fas dev2** set for Farsi with 200 keywords. In the keyword-agnostic condition, the fusion was trained on **dev1** with 1000 extra keywords that we selected for each language. In the keyword-aware condition, 200 of these 1000 keywords were selected at random to train a boosted LM that was used to run KWS on **dev1**, and only these 200 terms were used for fusion training. The goal was to perform keyword-boosting with a similar number of search terms in training and testing.

SAD	m@0.1%fa	fa@30%m	fa@30-60%m
baseline	27.2	0.0654	0.0099
MFCC+4feats	24.6	0.0367	0.0070
MMeDuSA+4feats	29.1	0.0896	0.0140
PLP+4feats	24.8	0.0394	0.0068

Table 1: Influence of SAD on KWS for a Levantine Arabic CNN-based MMeDuSA keyword-aware word system. In this table and all others, "m" stands for P_{miss}, "fa" stands for P_{fa} and all reported numbers are in %.

First, we found that the SAD introduced in 3.1, whose parameters (smoothing, padding, thresholding) were tuned for KWS, performed significantly better than the baseline SAD we used in our 2013 system. Table 1 shows the performance of a CNN system trained with MMeDuSA features, decoded with our 3 SAD configurations: using MFCC, PLP and MMeDuSA features, respectively. We found that our MFCC and PLP-based SADs outperformed the baseline by a significant amount at P_{miss} rates less than 40%. At P_{fa} rates of 0.01% or more, we obtained a gain in P_{miss} of 3% to 4% absolute. We also found that fusion of the KWS output of systems using the same acoustic and language models but different SADs can bring supplemental gains. For instance, the fusion of the PLP-SAD and MFCC-SAD systems outperformed either system at almost all operating points, as the average P_{fa} between 30% and 60% P_{miss} was 0.0068% for the PLP-SAD, 0.0070% for the MFCC-SAD and 0.0064% for the fusion.

System	fa@15-50%m		fa@15%m	
	alv dev-2	fas dev-2	alv dev-2	fas dev-2
Mel-filterbank	n/a	0.060	n/a	0.576
NMC	0.124	0.103	**0.761**	1.057
NMC+vtln	0.147	0.057	1.132	0.474
NMC+vtln+pn	0.139	0.057	1.124	0.421
MME2	0.133	0.063	1.308	0.432
MME2+vtln	0.141	0.057	1.124	**0.389**
DOC	0.117	0.067	1.182	0.436
DOC+vtln	0.115	**0.054**	1.080	0.413
SYD	0.184	0.069	1.458	0.485
LSEN-PNCC	0.210	0.077	1.442	0.678
GFC	**0.107**	0.071	0.905	0.587

Table 2: Influence of front-end features and VTLN on keyword-agnostic phonetic system performance using CNN modeling. The best performing feature is highlighted for each condition.

Second, we analyzed the influence of front-end features and VTLN on our CNN systems. Table 2 shows the CNN keyword-agnostic phone systems that were decoded with the PLP-based SAD for Levantine Arabic and Farsi. The performance in terms of average P_{fa} between 15% and 50% P_{miss} made GFC the best feature for Levantine Arabic and DOC+vtln the best feature for Farsi. In Farsi, four of the features were more competitive than mel-filterbanks for CNN modeling. In Levantine, we did not train a mel-filterbank CNN system because preliminary experiments with smaller model sizes showed performance inferior to the other features. Besides the good individual performance of these front-end features in KWS, they extract different information and therefore provide complementarity that can be exploited by performing system fusion.

Search	fa@30%m	fa@45%m	fa@12-18%m
Word-based	0.0160	0.0015	n/a
Phone-based	0.0222	0.0028	0.642
Fusion	0.0208	0.0022	0.443

Table 3: Performance of various search strategies for a Levantine Arabic LSEN-PNCC CNN keyword-aware system.

Third, we found that our word-based systems and phone-based systems had different characteristics. While our word systems generally exhibited a lower P_{fa} at a given P_{miss}, their lowest achievable P_{miss} did not reach below 15% to 20% for either language, because of the limited thickness of word lattices. The WFST-based phonetic search is more flexible because it does not enforce word-boundary conditions on phones and allows for phone deletion. Therefore, it can achieve a much lower P_{miss}, but is subject to increased false-alarms as compared to word systems. In the keyword-aware condition, we had the option to use both search techniques on the same lattices, and found that the two approaches can be complementary. Table 3 shows our Levantine Arabic LSEN-PNCC CNN system with MFCC-SAD with the word and phone search, as well as the fusion of the two. While the word system performed the best at P_{miss} above 30%, the fusion of word and phone systems outperformed both approaches at P_{miss} below 20%.

Keyword-agnostic – Levantine Arabic					
Front-end	SAD	Model	Search	fa@15%m	fa@25%m
DOC+vtl	plp	CNN	Phone	1.080	0.086
GFC	plp	CNN	Phone	0.905	0.093
LSEN-PNCC	mfcc	CNN	Phone	0.881	0.121
MFCC	plp	GMM	Phone	2.751	0.582
fusion				0.385	0.062

Keyword-aware – Levantine Arabic					
Front-end	SAD	Model	Search	fa@15%m	fa@25%m
DOC+vtl	plp	CNN	Phone	0.847	0.051
GFC	plp	CNN	Phone	0.880	0.074
LSEN-PNCC	mfcc	CNN	Word	–	0.056
MFCC	plp	GMM	Phone	2.814	0.535
fusion				0.195	0.035

Table 4: Levantine Arabic selected systems and 4-way fusion.

Finally, we confirmed that system fusion can bring very significant gains to KWS systems. For each condition and each language, we tried a large number of combinations (more than 1000) with the goal of minimizing the average P_{fa} for $12\% < P_{\mathrm{miss}} < 18\%$. In all case, four systems provided the best combination that incorporated different SADs, front-ends and/or search strategies. Tables 4 and 5 show the KW-agnostic and KW-aware single systems along with their fusion for each language. At the target P_{miss} of 15%, the fusion brought a relative gain in P_{fa} of 56% and 45% for Levantine Arabic and Farsi respectively for the agnostic submission, and gains of 78% and 38% for the aware submission. Prior knowledge of the keyword was very beneficial to KWS performance. Indeed, our keyword-aware submission produced about half the false-alarms of our keyword-agnostic submission, at a P_{miss} of 15% and 25%.

Keyword-agnostic – Farsi					
Front-end	SAD	Model	Search	fa@15%m	fa@25%m
NMC+vtl+pn	plp	CNN	Phone	0.421	0.054
MME2	mfcc	CNN	Phone	0.360	0.060
Gabor-MFCC	mfcc	DNN	Phone	0.721	0.088
MFCC	plp	GMM	Phone	1.393	0.168
fusion				0.206	0.049

Keyword-aware – Farsi					
Front-end	SAD	Model	Search	fa@15%m	fa@25%m
NMC+vtl+pn	plp	CNN	Word	0.161	0.039
MME2	mfcc	CNN	Phone	0.257	0.046
Gabor-MFCC	mfcc	DNN	Word	–	0.084
MFCC	plp	GMM	Word	0.411	0.103
fusion				0.099	0.034

Table 5: Farsi selected systems and 4-way fusion.

System	fa@12-18%m			fa@60%m		
	p	r	p+r	p	r	p+r
alv-aware	0.344	**0.213**	0.234	**3e-4**	9e-4	5e-4
alv-agnostic	0.980	0.417	**0.412**	**9e-4**	1.2e-3	1.0e-3
fas-aware	0.152	**0.099**	**0.099**	**2.0e-3**	2.4e-3	**2.0e-3**
fas-agnostic	0.625	0.214	**0.207**	3.2e-3	**1.8e-3**	**1.8e-3**

Table 6: Fusion performance using the posterior feature p, the rank feature r and both features in combination. The best feature is highlighted for each system and each metric.

We experimented with different score features in the LLR fusion: the logit-posterior p, the rank-based feature r, and both features. Results are reported in Table 6 for all languages and conditions. We found that in the range of P_{miss} of interest to RATS (around 15%), the rank and the combination of posterior and rank performed equally well, and outperformed use of only the posterior. This shows the benefits of using rank-normalization to avoid high false-alarm rates due to a few frequent keywords. At higher P_{miss} rates (e.g. 60%), use of the rank alone degraded performance in all but one case because it makes an assumption of equal appearance of each keyword and no longer represents the true likelihood of each detection being correct. Interestingly, the use of posterior and rank in combination provided performance very close or better to using the posterior alone. We believe that by adding both features in the LLR fusion, we allow our system to be better calibrated in both the high and low false alarm regions.

Acknowledgements This material is based upon work supported by DARPA under Contract No. D10PC20024. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the DARPA or its Contracting Agent, the U.S. Department of the Interior, National Business Center, Acquisition & Property Management Division, Southwest Branch. Approved for Public Release, Distribution Unlimited.

6. References

[1] J. R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden Markov modeling for speaker-independent word spotting," in *International Conference on Acoustics, Speech, and Signal Processing, 1989*. IEEE, 1989, pp. 627–630.

[2] H. Bourlard, B. D'hoore, and J.-M. Boite, "Optimizing recognition and rejection performance in wordspotting systems," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1994*, vol. 1. IEEE, 1994, pp. I–373.

[3] M. Weintraub, "LVCSR log-likelihood ratio scoring for keyword spotting," in *International Conference on Acoustics, Speech, and Signal Processing, 1995*, vol. 1. IEEE, 1995, pp. 297–300.

[4] J. Keshet, D. Grangier, and S. Bengio, "Discriminative keyword spotting," *Speech Communication*, vol. 51, no. 4, pp. 317–329, 2009.

[5] I. Szoke, P. Schwarz, P. Matejka, L. Burget, M. Karafiát, M. Fapso, and J. Cernocky, "Comparison of keyword spotting approaches for informal continuous speech," in *Proc. of Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, 2005.

[6] A. Mandal, J. van Hout, Y.-C. Tam, V. Mitra, Y. Lei, J. Zheng, D. Vergyri, L. Ferrer, M. Graciarena, A. Kathol, and H. Franco, "Strategies for high accuracy keyword detection in noisy channels," in *Proc. of Interspeech*, 2013.

[7] M. Graciarena, A. Alwan, D. Ellis, H. Franco, L. Ferrer, J. H. Hansen, A. Janin, B.-S. Lee, Y. Lei, V. Mitra, N. Morgan, O. Omid Sadjadi, T. Tsai, N. Scheffer, L. Ngee Tan, and B. Williams, "All for one: Feature combination for highly channel-degraded speech activity detection," in *Proc. Interspeech*, 2013.

[8] S. O. Sadjadi and J. H. Hansen, "Unsupervised speech activity detection using voicing measures and perceptual spectral flux," *Signal Processing Letters, IEEE*, vol. 20, no. 3, pp. 197–200, 2013.

[9] L. N. Tan and A. Alwan, "Multi-band summary correlogram-based pitch detection for noisy speech," *Speech Communication*, vol. 55, no. 7, pp. 841–856, 2013.

[10] B. S. Lee, "Noise robust pitch tracking by subband autocorrelation classification," Ph.D. dissertation, Columbia University, 2012.

[11] V. Mitra, H. Franco, and M. Graciarena, "Damped oscillator cepstral coefficients for robust speech recognition," in *Proc. of Interspeech*, 2013.

[12] V. Mitra, H. Franco, M. Graciarena, and A. Mandal, "Normalized amplitude modulation features for large vocabulary noise-robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4117–4120.

[13] R. Drullman, J. M. Festen, and R. Plomp, "Effect of reducing slow temporal modulations on speech reception," *The Journal of the Acoustical Society of America*, vol. 95, no. 5, pp. 2670–2680, 1994.

[14] P. Maragos, J. F. Kaiser, and T. F. Quatieri, "Energy separation in signal modulations with application to speech analysis," *Signal Processing, IEEE Transactions on*, vol. 41, no. 10, pp. 3024–3051, 1993.

[15] V. Mitra, H. Franco, M. Graciarena, and D. Vergyri, "Medium duration modulation cepstral feature for robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), to appear*. IEEE, 2014.

[16] V. Mitra, M. McLaren, H. Franco, M. Graciarena, and N. Scheffer, "Modulation features for noise robust speaker identification," in *Proc. of Interspeech*, 2013.

[17] H. Teager, "Some observations on oral air flow during phonation," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 5, pp. 599–601, 1980.

[18] J. van Hout and A. Alwan, "A novel approach to softmask estimation and log-spectral enhancement for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, March 2012, pp. 4105–4108.

[19] J. van Hout, "Low Complexity Spectral Imputation for Noise Robust Speech Recognition," Master's thesis, University of California, Los Angeles, USA, 2012.

[20] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (pncc) for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4101–4104.

[21] B. T. Meyer, S. V. Ravuri, M. R. Schädler, and N. Morgan, "Comparing different flavors of spectro-temporal features for ASR," in *Proc. of Interspeech*, 2011, pp. 1269–1272.

[22] A. Stolcke, H. Bratt, J. Butzberger, H. Franco, V. R. Gadde, M. Plauché, C. Richey, E. Shriberg, K. Sönmez, F. Weng, and J. Zheng, "The SRI March 2000 Hub-5 conversational speech transcription system," in *Proc. NIST Speech Transcription Workshop*, 2000.

[23] D. Can and M. Saraclar, "Lattice indexing for spoken term detection," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 8, pp. 2338–2347, 2011.

[24] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "OpenFST: A general and efficient weighted finite-state transducer library," in *Implementation and Application of Automata*. Springer, 2007, pp. 11–23.

[25] M. Akbacak, L. Burget, W. Wang, and J. van Hout, "Rich system combination for keyword spotting in noisy and acoustically heterogeneous audio streams," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013.

[26] B. Zhang, R. Schwartz, S. Tsakalidis, L. Nguyen, and S. Matsoukas, "White listing and score normalization for keyword spotting of noisy speech," in *Proceedings of Interspeech 2012*, 2012.