# Multi-stream Recognition of Noisy Speech with Performance Monitoring

*Ehsan Variani*[1], *Feipeng Li*[1], *Hynek Hermansky*[12]

Johns Hopkins University
[1]Center for Language and Speech Processing
[2] Human Language Technology Center of Excellence
Baltimore, MD, 21218
variani@jhu.edu, fli12@jhmi.edu, hynek@jhu.edu

10.21437/Interspeech.2013-273

## Abstract

A prototype multi-stream system with a performance monitor for stream selection is proposed to recognize speech in unknown noise. The speech signal is decomposed into seven band-limited streams. Posterior probabilities of phonemes are estimated by a multi-layer perceptron (MLP) in each of these band-limited streams. Estimated posterior vectors of all 127 combinations (processing streams) of the seven band-limited streams form inputs to a second-stage MLP that estimates posterior probabilities of phonemes in each processing stream. A performance monitor is designed to predict the reliability of individual processing streams based on the outputs from these streams. The top $N$ streams that are least affected by noise are selected and their outputs are averaged to yield the final posterior probability vector used in Viterbi search for the best phoneme sequence. Experimental results show that the proposed technique is effective in dealing with noise.

**Index Terms**: Multi-stream speech recognition, Performance monitoring

## 1. Introduction

State-of-the-art automatic speech recognition (ASR) systems are approaching the performance of human beings in many aspects except for their robustness to unknown noises. A major difference between the two is that a typical ASR system has only one processing stream which covers the whole frequency range, whereas the human auditory system might consist of multiple parallel processing streams. Partial corruption of some streams has little impact on speech communication [2]. In realistic environments, it is uncommon to have stationary white noise, which continuously corrupts the whole frequency range. More likely, noise is varying with time and localized in frequency. A robust ASR system should have multiple streams and have the ability to adapt to surrounding environment by switching across the multiple streams, so that it will always listen through the frequency range that is reliable.

This work was supported in parts by the DARPA RATS project D10PC0015, IARPA BABEL project W911NF12-C-0013, and by the Johns Hopkins Center of Excellence in Human Language Technologies. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the DARPA, IARPA or JHU HLTCOE.

## 2. Previous work

A two-stage multi-stream phoneme recognition system, in which the full frequency range is divided into seven band-limited streams was proposed earlier in [7]. The basic idea is to keep information from reliable bands and throw away noisy ones. At the first stage, the seven band-limited streams are classified into 40 phonemes; then a three-layer MLP is trained to integrate each of the 127 possible combinations (processing streams) of seven band-limited streams at the second stage. A similar approach was developed in [10]. Alternative ways of forming multiple independent processing streams were discussed in [9, 12, 13].

Recently, a frequency domain linear prediction (FDLP) speech analysis technique is developed in [1]. It approximates Hilbert envelopes of band-limited signals. Speech features based on FDLP are efficient for automatic speech recognition [6]. Since the FDLP feature extraction decomposes speech signal into multiple band-limited streams, it is suitable for parallel multi-stream processing. In [5] a multi-stream phoneme recognition system is developed using the FDLP features.

A critical research problem of multi-stream ASR is *how to choose the most reliable processing streams*. If the errors of the classifiers were known, it would be trivial to find the best streams. However, in general the errors are not known unless one uses transcribed adaptation data. The performance of a classifier needs to be predicted using some measure that could be described from the performance on unknown data. [12] proposed a selection based on the autocorrelation of posterior probability vectors in training and in test. More recently [8] developed a similar divergence measure which uses both the first and the second order statistics of classifier output. The assumption is that the MLP classifier performs the best on the data used to train it. Any data corruption will cause deviations of the statistic of the classifier output and can be caught by comparing the statistics derived from new data and the ones derived from reference data. Earlier results [12, 8] show that the divergence between the two statistics is highly correlated with the accuracy of recognition.

However, these earlier statistics do not take into consideration the temporal relationship between neighboring speech sounds. Any random permutation of the frames within the integration window will produce the same divergence measure. In this study, we address this issue and investigate an alternative technique for predicting the recognizer performance and use it for scoring the 127 streams for recognizer.

The remainder of this paper is organized as follow. We be-

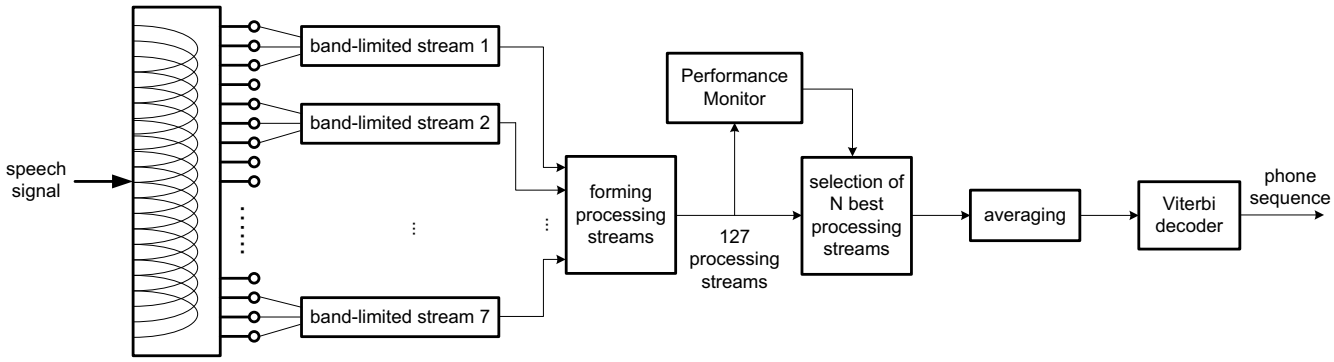


Figure 1: Block diagram of the multi-stream ASR system

gin with an overview of the multi-stream ASR system (§3.1). Next, we describe the proposed performance monitor for stream selection (§3.2). Section three discusses the experimental results, and finally section four summaries the concluding remarks.

# 3. Multi-stream ASR

The proposed multi-stream ASR system takes a three-stage processing scheme (refer to Fig. 1), i.e., stream formation, stream selection, followed by a Viterbi decoder, which estimates the underlying phone sequence. The stream formation block first splits the speech signal into seven independent *band-limited streams*, which are combined to form multiple *processing streams*. The stream selection block then scores each 127 streams and picks the top $N$ best processing streams, and posterior probabilities from these reliable streams are averaged and used to recognize phonemes.

## 3.1. Stream Formation

The speech signal is first decomposed into seven band-limited streams denoted as $S = \{S_1, ..., S_7\}$. Each stream $S_i$ covers about three critical bands along the auditory frequency axis. An independent three-layer MLP is trained to classify the band-limited signal. The band-limited signal, encoded in the FDLP features, which characterize the temporal evolution of signal Hilbert envelope in the sub-band over 200 ms, forms an input to a MLP phoneme classifier. The classifier generates 40 dimensional posterior probability vector $P$. Each item of the posterior vector $P$ represents the posterior probability of a particular phoneme in the band-limited stream. Next, a logarithm is applied to the posterior probabilities to force the data to have a nearly Gaussian distribution, and the dimensionality is reduced from 40 to 25 using the Karhunen-Loeve Transform (KLT).

All nonempty combinations of such outputs from the seven band-limited streams form a total of $2^{|S|} - 1 = 127$ vectors (with non-equal lengths depend on the number of band-limited streams in a particular combination), each representing an input to a second stage MLP classifier, resulting 127 processing streams represent all possible combinations of information from the seven band-limited streams.

$$C_i \subseteq \{S_1, ..., S_7\} \qquad i = 1, ..., 127 \tag{1}$$

The MLPs are three layer MLPs with 1000 hidden node and 40 output classes. The number of input nodes varies depending on the number of streams in the combination.

Depending on the intensity and spectral shape of noise, some processing streams may be more reliable than others. To choose the more reliable processing streams for the final fusion, it is necessary to have a procedure to select the informative processing streams and remove those corrupted by the noise.

## 3.2. Stream Selection with a Performance Monitor

For any given utterance the top $N$ out of all possible 127 processing streams are selected for speech recognition. It is generally more reliable to select $N$ most reliable processing streams and then fuse them by averaging the posterior probabilities, rather than picking only the best stream [14]. The selection of $N$-best posterior vectors is done by the performance monitoring block.

Figure 2 depicts the principle of a performance monitoring block. It measures the divergence between the description of the classifier output on its training data $P_{ref}$ (typically derived using all training data) and on the test data (during some test interval) ${P_{C_i}}^j$. For each interval of the test data $j$, we calculate the divergence value ${d_{C_i}}^j$, between the test descriptor and reference description,

$${d_{C_i}}^j = div(P_{ref}, {P_{C_i}}^j) \qquad i = 1, ..., 127 \tag{2}$$

In the current work, the test intervals are equal to the lengths of utterances in the TIMIT database. The classifier output description applied in [8] use mean and autocorrelation matrix of posterior vectors from the classifier. Experimental results using performance monitoring presented in [8] shows that the performance monitor works reasonably well but requires a minimum of four second segments of test data to get a stable estimate. Another major problem about this performance monitoring is that it does not include temporal information. In this work we are investigating a new performance monitoring measure which apparently requires shorter test data segments length to stabilize and exploits temporal information. This measure uses a technique, which was earlier used for different purposes in some other studies [11], and evaluates mean cumulative divergence between feature vectors as a function of their proximity in time. It turns out that differences in mean cumulative divergence curves derived on training data and in test predict well performance of the classifier on new test data [4].

For a window of posteriors (containing $N$ consequent frames of test data denoted by $P = (P_i, P_{i+1}, ..., P_{i+N})$), we calculate the divergence between the first frame and the last frame of the window of length $\Delta t$ for any fixed value of $\Delta t$. By moving fixed window of length $\Delta t$ over all $N$ frames, the

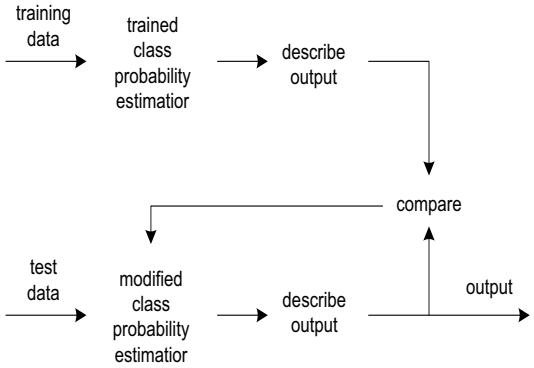


Figure 2: Performance monitoring block: comparing test statistics and training statistics.

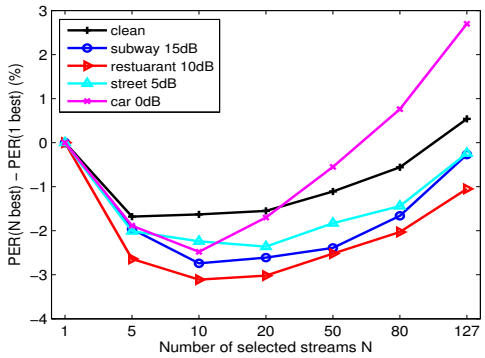


Figure 3: Relative PER vs. different values of $N$.

mean temporal distance will be calculated as follows

$$M(\Delta t) = \frac{\sum_{i=1}^{N-\Delta t} D_{sym}(P_i, P_{i+\Delta t})}{N - \Delta t} \quad (3)$$

where $D_{sym}$ is symmetric Kullback-Leibler distance,

$$D_{sym}(p,q) = \sum_{i=1}^{|p|} p_i \log \frac{p_i}{q_i} + \sum_{i=1}^{|q|} q_i \log \frac{q_i}{p_i}$$

By calculating $M$ for different window lengths, the $M$-curve can be derived for given segment of posteriors, $P$. In all cases, the distance values are increasing for windows of length up to $\Delta t = 200$ ms, and become saturate after that. Another observation is that the $M$-curves for utterance with higher SNR are above the ones with lower SNR.

By averaging $M(\Delta t)$ over the interval of length $T$ ms, we drive the $\overline{M_P}$ for the window of posteriors $P$.

$$\overline{M_P} = \frac{\sum_{\Delta t=\Delta t_{low}}^{\Delta t_{low}+T} M(\Delta t)}{T} \quad (4)$$

We used the difference between this value and similar value derived using training data as a measure for performance monitoring.

$$div(P_{ref}, P) = \overline{M_{ref}} - \overline{M_P} \quad (5)$$

$\Delta t_{low}$ is chosen to be 200 ms and $T$ to be 600 ms. Our experiments show that the proposed performance monitoring measure is highly correlated with recognition accuracy [4].

This measure predicts how close (or far) the performance of classifier on a given test utterance performance is from the performance expected on the training data. The 127 combinations are ranked according to their divergence value and the top $N$ combinations with lowest divergence value will be selected for recognition. Finally, we compute the average of the selected $N$ posteriors to provide final posterior vector for recognition. It is worth to mention that authors try some other fusion like product or probability model based fusion, but simple arithmetic average outperforms other methods.

## 4. Experiments

The experiments are conducted to test the proposed multi-stream ASR system with performance monitoring on clean and different noisy conditions. TIMIT data base is used as benchmark with 3696 utterances for training and 1344 utterances for testing. The experiments are done on 4 SNR level ranging from 0 dB to 15 dB SNR level with 5 dB increment. The additive noises in the experiments were added using FaNT tool [16].

To evaluate the effectiveness of performance monitor, the results of the proposed system are compared to a multi-stream system without stream selection (baseline system) which is built by training one MLP on posteriors of stream that contains all 7 sub-bands. There is also the oracle multi-stream system using manually picked the best stream which represents the theoretical upper limit of the performance monitor. For comparison, the conventional single-stream full-band system is also tested under the same conditions.

Experimental results (refer to Tab. 1) indicate that the proposed multi-stream phoneme recognition system is effective in dealing with noise. Using performance monitoring with 10 best selected streams lead to relative improvement ranging from 3% on exhall noise at 5 dB SNR to 17% relative improvements on car noise at 0 dB SNR. The system also shows relative improvement of 4% on clean condition. For some noises like subway, restaurant, and car noise the error reduction is more than others, which could be explained by the fact that these noises are band-limited, so they are more suitable for multi-stream system.

Figure. 3 investigates the effect of $N$ on PER. Selecting $N = 10$ streams results to the most reduction in phoneme recognition error.

Compared to the oracle multi-stream system, the phoneme recognition error of the performance monitor multi-stream system is about 7% absolute above the multi-stream system using manually picked the best stream, suggesting that there might be still room for the proposed performance monitor to improve.

## 5. Discussion

The performance monitor is effective in rejecting noisy streams and enhancing reliable ones. It is interesting to see that stream selection by performance monitor also helps improve the performance under clean condition. It is worth to mention that, the proposed multi-stream system outperforms the system using the earlier version of performance monitor described at [8].

The complexity of the system is on the side of training MLPs. Once the networks are trained, the evaluation of the signals could be done in parallel for each stream, making it feasible for online ASR. Although the final performance is less than the state-of-the-art on this dataset [15], the proposed method carries

| Conditions | full-band | baseline | multi-stream-PM 10 best | multi-stream-hand 1 best |
|---|---|---|---|---|
| clean | 31.35 | 31.27 | 29.98 | 23.78 |
| babble noise 15 at dB SNR | 57.10 | 52.80 | 49.68 | 42.85 |
| subway noise at 15 dB SNR | 46.62 | 45.15 | 40.79 | 34.11 |
| factory1 noise at 10 dB SNR | 68.1 | 69.87 | 67.10 | 59.91 |
| restaurant noise at 10 dB SNR | 63.14 | 65.03 | 61.61 | 55.18 |
| street noise at 5 dB SNR | 67.26 | 68.47 | 65.27 | 58.08 |
| exhall noise at 5 dB SNR | 70.67 | 71.16 | 68.67 | 61.85 |
| car noise at 0 dB SNR | 54.32 | 48.76 | 40.24 | 34.30 |

Table 1: PER (%) of the proposed multi-stream system with performance monitor and several other systems under clean and different noisy conditions. Noise pairs in different SNR values are selected in random.

a novel idea which could be adapted and generalized to other ASR.

## 6. Conclusions

In this work, we create a multi-stream ASR system by decomposing a speech signal into seven band-limited sub-bands, which are combined to form 127 processing streams for speech recognition. A performance monitoring block, which utilizes temporal evolution of speech, is proposed to select the most reliable streams for speech recognition. The proposed technique of stream selection is effective in rejecting corrupted processing streams and yields substantial improvements in accuracy of recognition of both clean and noisy speech.

## 7. Acknowledgement

We would like to thank Hesam Sagha from EPFL (Switzerland) for his comments and effort on editing this paper.

## 8. References

[1] Athineos, M., Ellis, D. P. W. "Autoregressive modelling of temporal envelopes.", IEEE Trans. Signal Processing. 55(11):5237–5245, 2007.

[2] Allen, J. B. (1994), *How do humans process and recognize speech?* IEEE Trans. Speech and Audio **2**(4), 567–577.

[3] Bourlard, H. and Morgan, N. "Connectionist speech recognition: a hybrid approach", Springer 1994

[4] Hermansky, H., Variani, E., Peddinti, V. "Mean temporal distance: predicting ASR error from temporal properties of speech signal", IEEE ICASSP 2013.

[5] Li, F., Mallidi, H., and Hermansky, H. "Phone recognition in critical bands using sub-band temporal modulations," in *Proceedings Interspeech*, 2012, p. P7.C06.

[6] Ganapathy, S., Thomas, S., and Hermansky, H., "Temporal envelope compensation for robust phoneme recognition using modulation spectrum.", J. Acoust. Soc. Amer. 128(6):3769–3780, 2010.

[7] Sharma, S., "Multi-stream approach to robust speech recognition.", Ph.D. Thesis, Oregon Graduate Institute of Science and Technology, Portland. 1999.

[8] Variani, E. and Hermansky, H. "Estimating Classifier Performance in Unknown Noise," in *Proceedings Interspeech*, 2012.

[9] Badiezadegan, S. and Rose R. C., "A Performance Monitoring Approach to Fusing Enhanced Spectrogram Channels in Robust Speech Recognition," in *Proceedings Interspeech*, 2011: 477–480.

[10] Hagen, A., "Robust speech recognition based on multi-stream processing,", Ph.D. Thesis, Ecole Polytechnique Fdrale de Lausanne, 2001.

[11] H. Hermansky and J. Cohen, "Report from 1996 JHU Workshop", Center for Language and Speech Processing, the Johns Hopkins University, 1996.

[12] N. Mesgarani and S. Thomas and H. Hermansky, "Adaptive Stream Fusion in Multistream Recognition of Speech.", in *Proceedings Interspeech*, 2011.

[13] N. Mesgarani, S. Thomas, and H. Hermansky, "Towards optimizing stream fusion, Express Letters of the Acoustical Society of America, vol. 139, no. 1, pp. 1418, 2011.

[14] T. Ogawa. "Personal Communicatons, JHU summer workshop," Aug., 2012.

[15] Sivaram, G. S. V. S, Hermansky, H., "Sparse Multilayer Perceptron for Phoneme Recognition", IEEE Transactions on Audio, Speech, and Language Processing, 20(1), 23-29, 2012.

[16] Hirsch, H.G. "FaNT: Filtering and Noise Adding Tool", http://dnt.kr.hsnr.de/download.html (date last viewed 04/01/12).