Time-Frequency Masking For Large Scale Robust Speech Recognition

Yuxuan Wang[1], *Ananya Misra*[2], *Kean K. Chin*[2]

[1]The Ohio State University, Columbus, OH
[2]Google Inc., Mountain View, CA

wangyuxu@cse.ohio-state.edu, {amisra, kkchin}@google.com

Abstract

Time-frequency mask estimation has shown considerable success recently. In this paper, we demonstrate its utility as a feature enhancement frontend for large vocabulary conversational speech recognition. Additionally, we investigate how masking compares with feature denoising, which directly reconstructs clean features from noisy ones. We train a mask estimator that predicts ideal ratio masks. Experimental results on Google voice search evaluation sets demonstrate that masking is superior to feature denoising, and a lightweight masking frontend produces significant improvements over a strong baseline. We also show that masking improves performance of a multi-condition trained (MTR) acoustic model.

Index Terms: Robust speech recognition, time-frequency masking, deep neural network, feature denoising

10.21437/Interspeech.2015-533

1. Introduction

With the introduction of deep neural networks (DNNs) and increased computing power, the performance of automatic speech recognition (ASR) has increased substantially over the past few years (see e.g. [5]), especially in relatively clean conditions. However, ASR performance in noisy and reverberant environments still lags behind, hindering its applicability in many scenarios (such as in car). Improving the robustness of ASR has therefore become a new focus.

Feature enhancement has been an active research topic for robust ASR due to its simplicity and relatively low computational cost in the test phase. For example, robust representations such as RASTA filtering [4] and power normalized cepstral coefficients [7] have been shown to be useful to deal with channel distortions and/or noise corruptions. Traditional techniques in speech enhancement, such as Wiener filtering and advanced frontend features [8] have also been used. Time-frequency masking is another feature enhancement technique that modulates (multiplies) noisy features by time-frequency masks. Recently, mask estimation has been formulated as a supervised learning problem with considerable success [18, 3, 11]. In its simplest form, a mask estimator is trained as a standard DNN, where the inputs are noisy features and the training targets are ideal binary or ratio masks. The estimated masks are used to clean up noisy features before feeding them to acoustic models for recognition.

Time-frequency masking has been extensively studied in the speech separation community [16, 18, 17], where a major focus is on improving speech intelligibility in noise [3]. It has also been used in robust ASR (see e.g. [14, 15]), helping to achieve the current best results on Aurora-4 and CHiME-2 [11]. Nevertheless, previous studies deal with relatively small to medium vocabulary tasks and use studio quality recordings for training. Therefore, a main goal of this paper is to demonstrate the utility of masking for large vocabulary conversational ASR, where the utterances are real Google voice search queries. We demonstrate that a simple mask estimator can provide significant improvements even when the DNN acoustic model is trained on large amounts of (potentially noisy) transcribed audio data.

An alternative data-driven feature enhancement technique is feature denoising, which aims to directly reconstruct clean speech features. Existing work evaluated its performance for both speech separation [19, 17] and robust ASR [10, 9]. However, a direct comparison between masking and feature denoising specifically for ASR is still missing. Therefore, another goal of this paper is to compare masking with feature denoising on large vocabulary ASR. We show that 1) masking is superior to feature denoising on our task, and 2) masking can alleviate the difficulties of learning clean features.

Robustness of DNN based acoustic models can be significantly improved by training using multi-condition (noisy) data [13]. In fact, it was shown in [13] that traditional enhancement frontends do not improve performance, even after the acoustic model is retrained using enhanced data. So this work focuses instead on comparing different DNN based feature enhancement frontends, and using *independently* trained clean and MTR-style acoustic models. While the approach is simple and has the drawback of not jointly optimizing the frontend and backend, it simplifies the training loop and allows us to easily plug in pretrained frontends into existing acoustic models.

2. Feature Enhancement Frontends

In this section, we describe our mask estimation model along with two other feature denoising based models. The three models are illustrated in Figure 1.

First, we train a DNN based mask estimator. The input to the network is 26 frames of 40-dimensional log mel filterbank energies, consisting of the current frame concatenated with 20 previous and 5 future frames. The training target, i.e. the network output, is an important choice. The ideal ratio mask (IRM) has been shown to be suitable for ASR [10]. In training, we assume that we have clean speech and the corresponding noisy mixture, where the latter can be obtained through re-recording or artificial corruption. Therefore, a natural definition of the IRM is:

$$IRM(t,f) = \frac{S_{mel}(t,f)}{Y_{mel}(t,f)}, \quad (1)$$

where $S_{mel}(t,f)$ and $Y_{mel}(t,f)$ denote the clean and noisy mel filterbank energy at time t and frequency f, respectively. We use mel filterbank here because the acoustic models are trained using log mel filterbank energies. Clearly, the perfect estimation

The first author performed this work as an intern at Google.

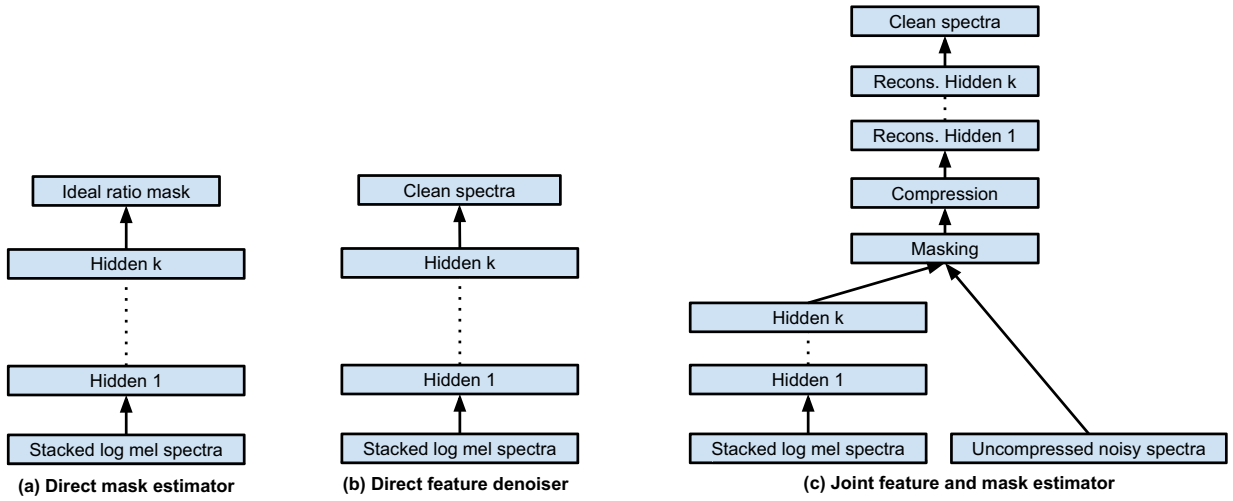

Figure 1: Illustration of the three feature enhancement frontends. Left: direct mask estimation. Middle: direct feature denoising. Right: joint feature and mask estimation.

of Eq. (1) leads to perfect clean feature reconstruction. However, in practice the IRM defined this way is unbounded since noisy mixture energy could be smaller than clean energy. Therefore, we need to bound Eq. (1) to make it a suitable training target. We found that simply capping (hard thresholding) the IRM yields reasonably good performance. The DNN uses rectified linear units (ReLU) as hidden activations and is trained by the adaptive gradient descent algorithm [2]. We use the standard mean squared error as the loss function. The training is carried out using the DistBelief distributed neural network training framework [1] on a large CPU cluster. These settings are the same for the feature denoising models described below. At test time, the enhanced features are obtained by multiplying estimated masks with uncompressed noisy energies, which are then converted to the log domain.

Our second feature enhancement frontend is a DNN based direct feature denoiser, which is identical to the mask estimator except for the training target. The model directly learns to reconstruct clean speech features. In this work, we predict clean log mel filterbank energies (without additional normalization), which are the features used by the underlying acoustic models. In test time, the network outputs directly serve as the enhanced features.

The last model unifies the previous two by joint feature and mask estimation, as illustrated in Figure 1. This is still a feature denoising model, as the training targets here are clean features. The key difference is that we treat the activations of a hidden layer as mask values and introduce a masking layer. This masking layer multiplies the hidden masks and uncompressed noisy energies (i.e. without the log compression), outputting masked features that can further be transformed to reconstruct clean features. The entire pipeline can be jointly trained via backpropagation. One motivation for introducing this model is to investigate whether the explicit modeling of masking can help in reconstructing clean features, which might be too hard to learn from scratch.

We describe their training and test settings in detail in the next section.

3. Experiments

3.1. Data and Experimental Settings

Supervised feature enhancement models typically need stereo noisy and clean training data, where the clean utterances are used either directly as training targets (feature denoising) or to create ideal masks (masking). To create these data, we use the WADA signal-to-noise ratio (SNR) estimator [6] to extract one million high-SNR utterances (from 25 to 35 dB) from Google voice search logs. These utterances (about 700 hours) are disjoint from both acoustic model training and evaluation data. We point out that these utterances still contain significant audible noise due to SNR estimation errors and real environments, resulting in noisy (imperfect) training labels. Nevertheless, this represents real use cases because large-scale studio quality data are hard to obtain in practice. These one million utterances are artificially corrupted to create the corresponding noisy training utterances.

Our main evaluation sets are Google voice search query (US English) data sets. These are artificially corrupted by adding various degrees of noise and reverberation. The reverberation time varies between 0 and 0.4 seconds, with a distribution biased toward smaller reverberation time. The SNR is distributed between 5 and 25 dB, biased toward higher SNRs. The distance between the target speaker and the microphone varies between 0.1 and 2m, biased toward shorter distances. The noise sources are from YouTube and daily life noisy environment recordings. The YouTube noises consist of audio tracks such as music, which are obtained from various YouTube video clips. Additionally, we consider an evaluation set containing real in-car recordings, as described in Section 3.3. All the data (both speech and noise) used in our experiments are anonymized.

The underlying acoustic model uses a 8-hidden layer ReLU DNN with 2560 hidden units per layer, which is independently trained on 2000+ hours of human-transcribed (potentially noisy) audio data. The input to the network also consists of 26 frames of 40-D log mel filterbank energies. We highlight that this is a competitive acoustic model, which already possesses some degree of robustness.

Table 1: Comparisons between different frontends

Frontend	WER (del/ins/sub)
None	30.1 (13.1/3.3/13.7)
ERM-CAP1	25.8 (8.0/3.7/14.0)
Denoiser	31.2 (9.2/4.3/17.7)
JFM-scratch-log	27.8 (8.4/3.9/15.5)
JFM-init-log	26.8 (7.8/4.0/15.0)
JFM-init-linear	26.1 (7.5/3.9/14.7)

3.2. Comparing Feature Enhancement Models

Before extensive formal evaluations, we compare the three feature enhancement models to identify the promising ones. We compare the following settings:

- **ERM-CAP1**: a direct mask estimator that predicts IRM capped at 1.0. The DNN uses 5 hidden layers and 2048 units per layer (abbreviated as $5 \times 2k$).
- **Denoiser**: a $5 \times 2k$ direct feature denoiser that predicts clean log mel filterbank energies.
- **JFM-scratch-log**: a joint feature and mask estimator, where the target is log compressed filterbank energies. The model is trained from scratch.
- **JFM-init-log**: Same as above, except that all layers below the masking layer are initialized from ERM-CAP1.
- **JFM-init-linear**: Same as JFM-init-log, except that the network predicts uncompressed filterbank energies.

For this experiment, we decide to use the matched training and test setting for feature enhancement frontends, i.e., the training utterances are corrupted using the same YouTube and daily life noise sources used for creating the evaluation set. Their SNR and reverberation time also follow the same distribution as the evaluation set. This helps us eliminate a lot of unnecessary comparisons, as a model that does not perform well in the matched setting is unlikely to perform well in the mismatched setting.

We list word error rates (WER) in Table 1. As a reference, the WER for the uncorrupted evaluation set is 17.6%. Although our acoustic model is trained on large-scale labeled data, it is clear that the degradation is substantial after corruption (17.6% versus 30.1%). A simple masking frontend reduces WER to 25.8%, where the deletion errors are significantly reduced without impacting insertions and substitutions much. While Denoiser also reduces deletions, it significantly increases insertions and substitutions, resulting in a WER that is even worse than the baseline. Joint feature and mask estimation clearly outperforms direct feature denoising, indicated by the difference between Denoiser and JFM-scratch-log. Bootstrapping JFM from ERM-CAP1 produces 1% absolute gain. Interestingly, by comparing JFM-init-linear with JFM-init-log, we can see that reconstructing uncompressed energy (which is converted to log domain later) is better than reconstructing log energy directly. An explanation is that the same amount of regression error made in log domain may have bigger impact than in linear domain. Although the best result of JFM is slightly worse than direct masking (26.1% versus 25.8%), the comparisons between Denoiser and JFMs show that regularizing the denoising task with masking does help to reduce the difficulties of learning clean features.

Figure 2 shows the enhanced features for three utterances (concatenated together). Looking at the first 350 frames, we

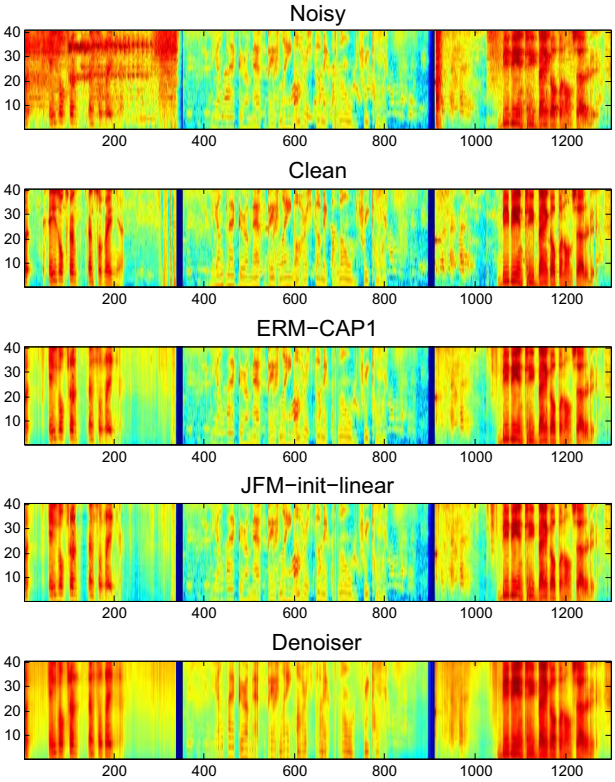

Figure 2: Enhanced log mel filterbank energy features from different feature enhancement frontends.

can see that the noise energies are effectively removed by both masking and feature denoising, which is consistent with the reduction in deletions. However, an issue with the Denoiser is that the reconstruction deforms the original features (e.g. frame 400 to 800), which is detrimental to pretrained acoustic models. This partly explains the significantly increased insertions and substitutions in Table 1. Directly learning a mapping to clean features is difficult, especially for large-vocabulary and speaker-independent tasks, where spectro-temporal patterns vary a lot. The Denoiser tries to learn many patterns (such as formant structure and harmonic spacing) that are already present in the noisy observations. In contrast, masking *modifies* the noisy observation by spectro-temporal weighting, which better preserves existing patterns. In addition, IRM seems to be easier to learn, as its structure is more stable. We note that recent work in supervised speech separation [17] also shows that masking works better than spectral envelope estimation even with studio quality recordings.

3.3. Full Evaluations

Based on previous results, we focus on more thorough evaluations for the direct mask estimator. We use the same one million utterances as clean targets; these are corrupted by YouTube, cafeteria, and daily environment noises to create the stereo training data for training the feature enhancement frontends. Unlike the previous experiment, these noises are completely disjoint from those in the evaluation set (i.e. test noise segments are never seen during training), thereby creating a mismatched training and test setting. We also use a more optimized evaluation pipeline and tune recognition system parameters (such as

Table 2: WER variation with different network architectures

	#Weights	WER (del/ins/sub)	Rel. Gain
None	n/a	25.0 (9.9/2.5/12.5)	0%
$5 \times 2k$	18.9M	21.8 (6.8/2.7/12.2)	12.8%
$5 \times 1k$	5.3M	21.7 (7.0/2.6/12.1)	13.2%
$4 \times 1k$	4.2M	21.7 (7.0/2.6/12.1)	13.2%
$3 \times 1k$	3.2M	21.8 (7.2/2.6/12.0)	12.8%
3×512	1.0M	22.7 (8.0/2.5/12.2)	9.2%
ERM-CAP2-$4\times1k$	4.2M	21.4 (6.4/2.7/12.4)	14.4%

Table 3: WER comparisons with MTR-AM. The WER of the pretrained AM on the full set is 25.0 (see Table 2)

	Overall	SNR $\leq$ 10 dB	SNR $>$ 10 dB
MTR-AM	20.4	26.2	18.3
MASK + MTR-AM	20.3	25.7	18.3

language model weights) in order to see if the performance gain can be subsumed by other components. Therefore, the results in this section are not directly comparable to those in Table 1.

3.3.1. Effects of Model Size

First, we vary the number of hidden layers and hidden units for ERM-CAP1. The motivation is that mask estimation in relatively high SNR conditions might be an easy learning task and a $5 \times 2k$ network might be overkill. As can be seen in Table 2, this is indeed the case. Using a $4 \times 1k$ network that has one fourth of the original size produces even slightly better results. In fact, even a small network (3×512) with only 1M weights can produce a 9.2% relative gain. This suggests that masking is potentially useful for quick acoustic environment adaptation and also suitable for applications requiring low resources, such as small footprint keyword spotting.

We made a few attempts to improve the performance of the direct mask estimator. We found that predicting IRM capped at 2.0 coupled with mask rescaling (by raising the mask to a power) works well. Rescaling estimated masks can help reduce speech distortions at the expense of less noise reduction [17, 11]. The final model (ERM-CAP2-$4\times1k$) produces 21.4% WER – a 14.4% relative gain. Predicting IRM capped at higher values did not give better results, likely because the training target becomes more difficult to learn.

3.3.2. Comparisons with Multi-Condition Acoustic Model

Previous results are based on an acoustic model pretrained on (relatively) clean speech utterances. Now we compare a multi-condition acoustic model (MTR-AM) trained on 2000+ hours of speech corrupted with the same noises used for training the mask estimator, following the same SNR, reverberation and distance distributions. The MTR-AM is trained on alignments generated on clean data. We point out that MTR-AM is a standard robust ASR technique, and represents a very strong baseline especially when DNN based acoustic models are used along with large amounts of noisy data [13, 10]. We use ERM-CAP2-$4\times1k$ as the masking frontend, where the enhanced features are evaluated using the MTR-AM (MASK+MTR-AM). Table 3 shows the overall results and results grouped by evaluation utterance SNR. Clearly, multi-condition training significantly improves performance over the pretrained AM (20.4% v.s. 25.0%, see Table 2). In low SNR conditions ($\leq$10dB), masking further improves MTR-AM by 0.5% absolute, even though the training data for both had comparable SNR distributions. The gain diminishes for high SNR ($>$10 dB) utterances. We note that the mask estimator is trained on significantly less data (about one third) than MTR-AM. It is likely that our masking frontend will also benefit from a larger training set, which will be considered in future studies.

Table 4: WER on handsfree in-car utterances

	WER on noisy	WER on clean
AM	15.9	13.8
MTR-AM	15.9	14.1
MASK + AM	15.6	13.7

3.3.3. Evaluations on In-Car Recordings

In addition to the main evaluation set which is artificially corrupted, our last experiment uses an evaluation set containing handsfree in-car noisy utterances. These noisy utterances are created by re-recording (i.e. playing back clean utterances) in a moving car. Therefore, we have a clean test set and the corresponding noisy test set. The car noise is an unseen noise type to both the mask estimator and MTR-AM. From Table 4, we can see that MTR-AM does not improve performance, whereas masking coupled with the pretrained clean AM (MASK+AM) does give a small improvement. This calls for more comparisons between masking and MTR-AM in future, especially in more mismatched training and test conditions.

4. Concluding Remarks

We have presented our preliminary investigations of time-frequency masking for large scale robust ASR, which shows promising results. First, we showed that masking is a better choice than feature denoising as a feature enhancement frontend. Then, we demonstrated that a simple and lightweight mask estimator produces large gains on top of a fairly strong AM, and that it also helps a multi-condition trained AM.

We considered a very simple masking frontend in this work. Going forward, mask estimation using Long-Short Term Memory (LSTM) recurrent nets [12] and joint frontend and backend training [11] are promising techniques to explore. Future work will also study the effectiveness of model based techniques which have not been explored in this work, e.g., using the noise estimates from the estimated mask as additional features to train the acoustic model [13, 11].

5. Acknowledgements

The authors thank Arun Narayanan for very useful discussions and for training the MTR-AM.

6. References

[1] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. Le, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Ng, "Large scale distributed deep networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1223–1231.

[2] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, pp. 2121–2159, 2011.

[3] E. Healy, S. Yoho, Y. Wang, and D. Wang, "An algorithm to improve speech recognition in noise for hearing-impaired listeners,"

Journal of the Acoustical Society of America, pp. 3029–3038, 2013.

[4] H. Hermansky and N. Morgan, "RASTA processing of speech," pp. 578–589, 1994.

[5] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, pp. 82–97, 2012.

[6] C. Kim and R. Stern, "Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis." in *INTERSPEECH*, 2008, pp. 2598–2601.

[7] ——, "Power-normalized cepstral coefficients (PNCC) for robust speech recognition," in *Proc. ICASSP*, 2012, pp. 4101–4104.

[8] P. C. Loizou, *Speech enhancement: theory and practice*. CRC press, 2007.

[9] A. Maas, Q. Le, T. O'Neil, O. Vinyals, P. Nguyen, and A. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *INTERSPEECH*, 2012.

[10] A. Narayanan and D. Wang, "Investigation of speech separation as a front-end for noise robust speech recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, pp. 826–835, 2014.

[11] ——, "Joint noise adaptive training for robust automatic speech recognition," in *Proc. ICASSP*, 2014, pp. 2523–2527.

[12] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *INTERSPEECH*, 2014.

[13] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*, 2013, pp. 7398–7402.

[14] M. Seltzer, B. Raj, and R. Stern, "A Bayesian classifier for spectrographic mask estimation for missing feature speech recognition," *Speech Communication*, vol. 43, no. 4, pp. 379–393, 2004.

[15] S. Srinivasan, N. Roman, and D. Wang, "Binary and ratio time-frequency masks for robust speech recognition," *Speech Communication*, vol. 48, no. 11, pp. 1486–1501, 2006.

[16] D. Wang and G. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*. Hoboken, NJ: Wiley-IEEE Press, 2006.

[17] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, pp. 1849–1858, 2014.

[18] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, pp. 1381–1390, 2013.

[19] Y. Xu, J. Du, L. Dai, and C. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, pp. 66–68, 2014.