INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

# Aligning speech enhancement for improving downstream classification performance

*Yan Xiong, Visar Berisha and Chaitali Chakrabarti*

School of Electrical, Computer and Energy Engineering
Arizona State University, Tempe, Arizona, USA
{yxiong35,visar,chaitali}@asu.edu

## Abstract

Speech-based classification models in the cloud are gaining large-scale adoption. In many applications where post-deployment background noise conditions mismatch those used during model training, fine-tuning the original model on local data would likely improve performance. However, this is not always possible as the local user may not be authorized to modify the cloud-based model or the local user may be unable to share the data and corresponding labels required for fine-tuning. In this paper, we propose a denoiser stored locally on edge devices with an application-specific training scheme. It learns a custom speech enhancement scheme that aligns the local denoiser with the downstream model, without requiring access to the cloud-based weights. We evaluate the denoiser with a common classification task – keyword spotting – and demonstrate using two different architectures that the proposed scheme outperforms common speech enhancement models for different types of background noise.

**Index Terms**: speech enhancement, keyword spotting, capsule network, cloud computing, data privacy

## 1. Introduction

Speech-based classification models in the cloud have gained popularity. Examples include keyword spotting, speaker diarization, audio sentiment classification, among others. However, there are many scenarios where the background noise conditions are different from those used during model training. In such scenarios, fine-tuning the original model on local data would likely improve performance. However, this is not always possible as the local user may not be authorized to modify the cloud-based model, or the local user may not be able to provide labeled data required for fine-tuning. In this setting, we cannot change the classification model in the server with local data, or download a copy of the model for fine-tuning. Furthermore, since multiple users may be using the same cloud-based model, it is impractical for the model parameters to be changed every time. In this paper, we propose a denoiser that is stored locally on the edge device and acts as a pre-processing step for the downstream classification model. Specifically, it learns a custom speech enhancement scheme that aligns the local denoiser with the cloud-based model without changing the model.

We focus on keyword spotting (KWS) as the downstream classification task of interest. KWS plays an important role in speech-based interaction applications such as wake-word detection and voice commands. Under noise-less conditions, deep convolution neural networks with Mel-frequency cepstral coefficients (MFCCs) or mel spectrum inputs have shown superior performance on several keyword spotting tasks [1, 2, 3, 4]. However, performance declines rapidly in noisy conditions [5, 6] This is problematic as devices that use wake-word detection (e.g. Alexa) are deployed in a variety of settings (homes, offices, hospitals, etc.) and confront different types of background noise. Such complex and varying environmental noise conditions pose a challenge for classification, especially under noise conditions that the models have not been trained on [5, 7, 8].

To address the problem of background noise in speech signals, several types of speech enhancement methods have been previously proposed. Conventional speech enhancement models use statistical signal processing [9, 10]. These methods focus on time/frequency domain filtering or statistical estimators to filter out the noise from the noisy speech. Recently, speech enhancement systems based on machine-learning and deep-learning techniques have been proposed in [11, 12, 13, 14]. The aim of existing speech enhancement models is to improve the "perceptual" quality of degraded speech. The improvement in quality metrics such as SNR and PESQ indicate that the enhanced speech is "better" from the listener's perspective. Our objective is different. We aim to develop an enhancement model that improves downstream classification performance and not the perceptual quality.

We propose a KWS-specific training method for speech enhancement that can be tuned locally without requiring a copy of the downstream model. We introduce a new KWS enhancement loss term to match the outputs of our enhancement model with that of a target KWS model. This term allows the denoiser model to learn from the downstream KWS model output and carry out the speech enhancement task in a style preferred by the downstream task. Furthermore, the training is done locally without requiring access to the weights of the large cloud-based model. We evaluate the effect of the proposed KWS enhancement loss term on two denoiser models, ResCap denoiser and Res-FC denoiser, that can be wrapped around any existing KWS model to improve the performance of that model under noisy conditions. The proposed ResCap denoiser model is inspired by the ResCap Network [15], which has demonstrated that capsule networks exhibit superior performance for overlapping keyword spotting tasks. The Res-FC denoiser has the same design as the ResCap denoiser except that the capsule layers are replaced with fully-connected layers. The purpose of evaluating two architectures is to empirically demonstrate that the loss term provides improved performance across different networks. The evaluation results showed that both denoiser models, when trained with the KWS enhancement loss, outperformed the existing methods under various background noise scenarios, including when the testing noise and the training noise are from different sources.


10.21437/Interspeech.2023-1232

## 2. Methodology

### 2.1. Speech enhancement for robust keyword spotting

We assume a gray-box setting where the pre-trained KWS model stays in the cloud computing server. The internal workings of the KWS model cannot be modified, but the activation during the forward pass and gradients during the backward pass can be computed on the server side. Our aim is to train an enhancement model to improve the performance of the gray-box KWS model. To that end, we propose a loss function that consists of a spectrum reconstruction loss and a KWS enhancement loss. The spectrum reconstruction loss is computed locally to guide the auto-encoder to reconstruct the clean spectrum from the noisy input spectrum. It uses a mean-square-error (MSE) loss between the clean spectrum ($S_{clean}$) and the enhanced spectrum ($S_{enhanced}$):

$$L_{Recon} = MSE(S_{clean}, S_{enhanced}) \quad (1)$$

The KWS enhancement loss is calculated as follows. Let $y = G(x)$ denote the target KWS model where the input $x$ is the mel-spectrum of the input speech and the output $y$ is the vector of posterior probabilities generated by the KWS model, one per class. In order for the enhanced spectrum to produce a KWS result similar to the original clean spectrum, we introduce the KWS enhancement loss. This is the MSE loss between the KWS model output of the clean keyword spectrum and the KWS model output of the enhanced keyword spectrum from the speech enhancement model. We choose the MSE loss over other loss functions, such as L1 loss, NLL loss, cross-entropy loss, etc., because it achieves the best performance.

$$L_{KWS} = MSE(G(S_{clean}), G(S_{enhanced})) \quad (2)$$

This term guides the speech enhancement model to enhance the keywords in a way that improves the performance of the target KWS model. The overall loss function is a linear combination of the spectrum reconstruction loss and the KWS enhancement loss given by $Loss = L_{recon} + \mu \cdot L_{KWS}$. We use $\mu = 0.1$ to train our models which is fine-tuned with the development set. Overall, the dataflow of the training scheme is as shown in figure 1. The communication between the server and the local device happens once per batch of training samples. During the forward pass, the enhanced spectrum $S_{enhanced}$ is computed locally and sent to the server along with the reference clean spectrum $S_{clean}$. During the backward pass, the gradient of $S_{enhanced}$ is computed on the server side and sent to the local device. The local device then uses the gradient and local reconstruction loss to complete the backward pass of the local denoiser model and update it. The benefits of the training scheme are: First, the training is carried out in an unsupervised style, and the training samples are not labeled, ensuring data privacy; Second, the server side sends back only the gradient and not the model weights, ensuring model IP protection.

### 2.2. Denoiser models: ResCap and Res-FC

ResCap denoiser uses a capsule network-based auto-encoder structure. As is shown in Figure 2, the encoder consists of 14 2D convolution layers followed by 2 capsule layers. The 2D convolution layers are implemented in residual style, where 2 layers are grouped into a residual block with a bypass. Two capsule layers are used, where the primary capsule layer consists of $486 \times 32$ capsules and the output capsule consists of $1 \times 16$ capsules. The decoder uses a symmetric design to match the encoder as in the SEGAN generator [13]. A three-layer MLP is used to generate a 2D array from the output capsule. The

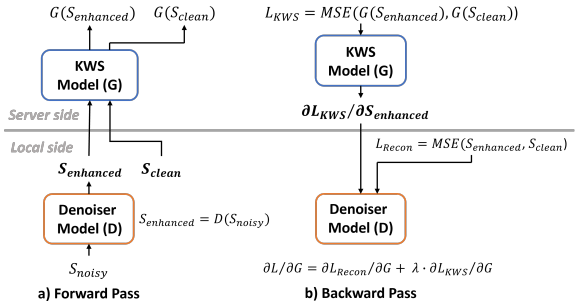


Figure 1: *Data flow of the algorithm-specific training scheme. a) Forward pass, where clean ($S_{clean}$) and enhanced ($S_{enhanced}$) spectra are sent to the server, b) Backward pass, where the gradient of KWS loss is sent to the local device*

output 2D array is then fed into a stack of deconvolution layers. Each deconvolution layer generates the output array with the same size as its corresponding convolution layer in the encoder. The decoder is regularized by using bypasses that pass the intermediate outputs of encoder layers, as shown with gray lines in Figure 2. The decoder outputs the enhanced spectrum which is of the same size as the input spectrum.

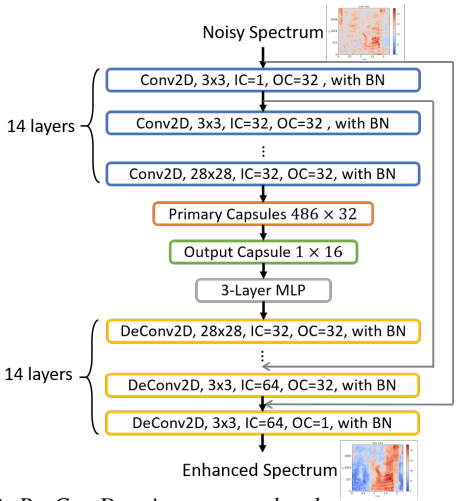


Figure 2: *ResCap Denoiser: a speech enhancement model with ResCap generator and deconvolution-based decoder*

The Res-FC denoiser maintains the 2D convolution and deconvolution stack and the by-pass design of the ResCap denoiser and replaces the primary capsule layer and output capsule layer with a fully-connected layer of the same size. The 3-layer MLP is also kept to restore the 2D array from the output of the fully-connected layer. The deconvolution stack generates the enhanced spectrum, as in the ResCap denoiser.

## 3. Evaluation

### 3.1. Experimental Setup

We use keywords from Google's Speech Commands Dataset [16] to train and evaluate the KWS model and the speech enhancement model. We select the same ten classes of keywords as [15]: "backward", "bed", "follow", "forward", "marvin", "nine", "sheila", "six", "visual" and "wow". To build a noisy keyword dataset, we use noise samples from Microsoft Scalable Noisy Speech Dataset (MS-SNSD) [17] and mix the noise source and keyword samples with an SNR randomly selected (uniformly distributed) in the range of 0dB to 10dB. The noisy keyword samples are generated by adding a noise segment that

is of the same length as the keyword with an SNR randomly selected in the range of 0 dB to 10 dB. 90,000 samples are used for training, 13,000 for developing, and 13,000 for testing.

We use the mel spectrogram as the input of both the KWS and existing enhancement models. The mel-spectrum settings are as follows: the window length and FFT length are 1024, the hop length between STFT windows is 256, the frequency ranges from 20 to 8000 (at a sampling rate of 16kHz), and the number of mel filters is 80. For the KWS model, we use the ResNet model proposed in [18]. The model is trained with clean keywords and fine-tuned to achieve the highest KWS accuracy. The trained model is used as the target KWS in all experiments. The KWS accuracy is measured by computing the percentage of correctly predicted samples.

### 3.2. Evaluation for different loss functions

We consider three models in this section: ResCap from [15], proposed ResCap and Res-FC denoisers. We build the training set using a mix of seven types of noise sources: babble, airport announcement, air conditioner, printer, neighbor speaking, and vacuum cleaner. Three different loss functions are used: Recon-only loss, KWS-only loss, and the combined loss of reconstruction loss and KWS loss. The model is evaluated with three representative types of noise sources: babble, printer, and mixed noise, each representing speech-like noise environment, mechanical noise environment, and the complex environment that a KWS system may face. Table1 shows the KWS accuracy using denoising models trained with different loss functions for different noise scenarios.

Table 1: *KWS accuracy using denoising models trained with different loss functions*

| Model | Babble | Printer | Mixed |
|---|---|---|---|
| Clean Keywords | 98.14% | 98.14% | 98.14% |
| Noisy Keywords | 83.44% | 88.30% | 89.17% |
| Recon-only Loss | | | |
| ResCap [15] | 64.98% | 67.04% | 61.17% |
| Res-FC denoiser | 83.08% | 90.72% | 83.74% |
| ResCap denoiser | 85.92% | 95.06% | 88.09% |
| KWS-only Loss | | | |
| ResCap [15] | 60.33% | 62.48% | 61.05% |
| Res-FC denoiser | 80.13% | 86.28% | 81.73% |
| ResCap denoiser | 85.50% | 90.78% | 86.42% |
| Recon+KWS Loss | | | |
| **Res-FC denoiser** | **91.50%** | **96.95%** | **93.50%** |
| **ResCap denoiser** | **91.83%** | **96.94%** | **93.64%** |

In the presence of different noise sources, the KWS accuracy significantly declined by 8.9% to 14.7%. When trained with reconstruction-only loss (eqn.1), both the Res-FC denoiser and the ResCap denoiser outperform the ResCap model on KWS accuracy by $> 15\%$. When trained with KWS-only loss (eqn.2), while the KWS accuracy decreases compared to using the reconstruction-only loss, both Res-FC and ResCap denoiser still have significant performance improvement compared to the ResCap model. When trained with Recon+KWS loss, the accuracy performance of the Res-FC denoiser and ResCap denoiser significantly increase by around 7% on average, compared to models trained with reconstruction-only loss.

### 3.3. Comparison against baseline enhancement methods

To show the advantage of our proposed algorithm-specific denoiser, we consider two state-of-the-art speech enhancement models, that increase the perceptual quality of speech, as baselines: SEGAN [13] and DEMUCS [19]. In this experiment, both the training set and testing set use noise from the same source. For a fair comparison, we fine-tune each enhancement model on our noisy speech training dataset. Then we denoise the noisy test samples before feeding them into the KWS model.

Table 2 shows the speech enhancement performance of the proposed model and the baseline methods. While all speech enhancement methods increase the KWS accuracy under all noise cases, both the Res-FC denoiser and the ResCap denoiser outperform the baseline models. Among the two denoiser models, the ResCap denoiser achieves the highest performance. It improves the KWS accuracy by 9.36% for babble noise, 6.78% for printer noise, and 4.64% for mixed noise. The average accuracy gain is 4.93% over SEGAN and 3.01% over DEMUCS.

Table 2: *KWS accuracy when the model is trained and tested with the same noise source*

| Model | Babble | Printer | Mixed |
|---|---|---|---|
| Noisy Keywords | 83.44% | 88.30% | 89.17% |
| SEGAN[13] | 86.74% | 89.87% | 90.29% |
| DEMUCS[19] | 89.46% | 90.86% | 92.34% |
| **Res-FC denoiser** | **92.58%** | **94.97%** | **93.40%** |
| **ResCap denoiser** | **92.80%** | **95.08%** | **93.81%** |

### 3.4. Evaluation on out-of-domain noise

The differences in the background noise can also be challenging to a local denoiser model which has not encountered those noisy scenarios during training. To evaluate the robustness of the proposed speech enhancement method, we evaluate its performance on out-of-domain noise. First, to validate the robustness across different corpora, we train all models using the mixed-noise training set and test them on three new noise sources. These include "Restaurant" noise from the MUSAN [20] noise dataset and WHAM! [21] noise dataset. Table 3 results show that overall these noise sources cause less KWS accuracy degradation compared to babble noise, printer noise, and mixed noise (shown in Table 2). Under these noise sources, SEGAN and DEMUCS models exhibit very little increase in KWS accuracy relative to noisy keywords. Under MUSAN noise, SEGAN-enhanced keywords have almost the same accuracy as that of noisy keywords, and DEMUCS-enhanced keywords achieve an accuracy improvement of only 0.29%. In comparison, the ResCap denoiser maintains high speech enhancement performance for all three evaluations and outperforms SEGAN by 1.66%-2.42% and DEMUCS by 1.32%-1.79%.

Table 3: *KWS accuracy of cross-corpora evaluation*

| Model | MUSAN | WHAM! |
|---|---|---|
| Noisy Keywords | 93.73% | 92.52% |
| SEGAN [13] | 93.68% | 92.64% |
| DEMUCS [19] | 94.02% | 93.27% |
| **Res-FC denoiser** | **94.95%** | **94.77%** |
| **ResCap denoiser** | **95.34%** | **95.06%** |

In a second out-of-domain evaluation experiment, we show the performance of the ResCap denoiser when there is a training-testing mismatch across the different noise sources within the MS-SNSD dataset. Table 4 shows the KWS accuracy results for all training-testing noise source combinations. As expected, we see that the highest accuracy is achieved when the training and testing set consists of the same type of noise; however, the ResCap denoiser consistently improves the KWS performance when the training and testing noises mismatch. The average row shows that the highest overall accuracy is achieved when the training set uses babble noise. We also find that end-

Table 4: *KWS accuracy when the ResCap denoiser model is trained using one noise type and tested with different noise types*

| | | Training noise | | | | | | Baseline |
|---|---|---|---|---|---|---|---|---|
| | | **Babble** | **Airport** | **A. C.** | **Printer** | **Neighbor** | **Cleaner** | **Noisy** |
| **Testing Noise** | **Babble** | 92.8 | 91.73 | 88.66 | 89.87 | 91.75 | 90.21 | 83.44 |
| | **Airport** | 94.36 | 95.45 | 93.23 | 93.81 | 93.99 | 93.84 | 89.77 |
| | **A. C.** | 96.55 | 96.93 | 97.34 | 96.7 | 96.26 | 96.88 | 95.41 |
| | **Printer** | 92.85 | 9175 | 91.85 | 95.08 | 92.48 | 94.23 | 88.3 |
| | **Neighbor** | 92 | 91.38 | 87.19 | 89.18 | 93 | 89.59 | 84.04 |
| | **Cleaner** | 91.34 | 90.36 | 89.16 | 93.36 | 91.93 | 94.95 | 85.64 |
| **Average** | | **93.32** | **92.93** | **91.24** | **92.83** | **93.24** | **93.28** | **87.77** |

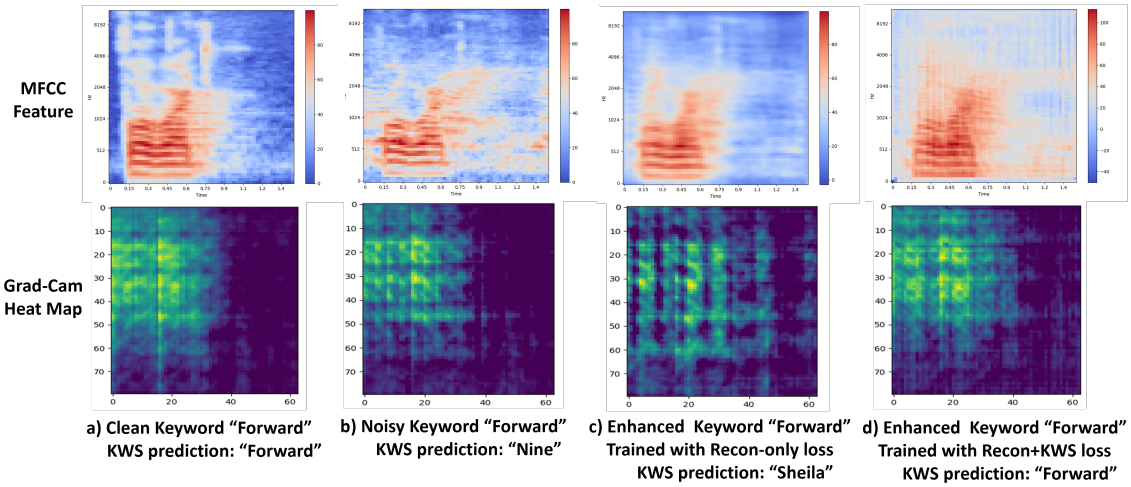


Figure 3: *Spectrum and Grad-CAM heat map of clean, noisy, and enhanced keywords when using the ResCap denoiser*

to-end fine-tuning risks overfitting to the noise type on which it is trained, while our denoiser model has shown good generalizability under noise mismatch conditions.

### 3.5. Discussion

To better understand the effects of the KWS loss term, we use Grad-CAM [22] as a visualization tool to explain how the KWS model performs differently on different enhanced spectra. GradCAM provides class-specific visualization of a model by producing a coarse localization map highlighting important regions in the image for predicting the class. Figure 3 shows a sample case of the keyword "forward". As shown in a), the clean keyword is correctly classified, and the heat map indicates that the middle-high frequency part has the highest importance in the prediction. Figure 3b) shows that the middle-high frequency features of the keyword are blurred by the background noise, which leads to incorrect classification while the heat map shows that the attention of the KWS model is still in the right region. Figure 3c) and d) show the enhanced spectra from the ResCap models trained with the two loss functions. The heat maps show that when the keyword is enhanced with a model trained with reconstruction-only loss, the KWS model fails to focus on the correct region when making the decision, leading to the incorrect recognition of the keyword. When the model is trained with reconstruction+KWS loss, the heat map shows that the KWS model focuses on the correct location. The enhanced spectrum also maintains keyword features that contribute the most to the classification task. As a result, the spectrum is correctly classified.

The Grad-CAM outputs indicate that the enhanced keyword must fulfill two conditions to successfully classify a keyword. First, the attention of the KWS model should be driven to the area corresponding to the keyword. Second, the enhanced spectrum must maintain the features of the keyword in the area that the downstream KWS algorithm focuses on. The spectrum and heatmap are shown in figure 3c) and d), indicate that the KWS term guides the denoiser model to focus on locations that contribute the most to the decision of the downstream model.

While this paper is focused on the utility of the denoiser models in a KWS setting, the method is generally useful as a pre-processor before any speech classification model where the algorithm designer does not have access to the internal workings of the classifier. The denoiser model only needs access to the output posteriors of the model during training for the denoiser to match the enhancement process.

## 4. Conclusions

This work proposed a denoiser model with an application-specific training scheme that carries out speech enhancement locally on the edge device for the speech processing model housed in the cloud server. The denoiser model can be fine-tuned locally without requiring labeled data from the user side and access to the speech processing model from the server side. Specifically, we use keyword spotting to showcase and evaluate the performance of the proposed loss term using two denoiser models, namely, ResCap and ResFC. The proposed denoiser models are evaluated under various noise environments and the results show that both denoiser models achieve higher KWS accuracy under various background noise conditions and outperform state-of-the-art general-purpose speech enhancement algorithms.

## 5. Acknowledgements

This work is funded in part by NIH - NIDCD R01 DC006859.

# 6. References


[1] W. Shan, M. Yang, J. Xu, Y. Lu, S. Zhang, T. Wang, J. Yang, L. Shi, and M. Seok, "14.1 a 510nw 0.41 v low-memory low-computation keyword-spotting chip using serial fft-based mfcc and binarized depthwise separable convolutional neural network in 28nm cmos," in *2020 IEEE International Solid-State Circuits Conference-(ISSCC)*. IEEE, 2020, pp. 230–232.

[2] J. K. Rout and G. Pradhan, "Data-adaptive single-pole filtering of magnitude spectra for robust keyword spotting," *Circuits, Systems, and Signal Processing*, pp. 1–17, 2022.

[3] E. van der Westhuizen, H. Kamper, R. Menon, J. Quinn, and T. Niesler, "Feature learning for efficient asr-free keyword spotting in low-resource languages," *Computer Speech & Language*, vol. 71, p. 101275, 2022.

[4] D. Peter, W. Roth, and F. Pernkopf, "End-to-end keyword spotting using neural architecture search and quantization," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 3423–3427.

[5] Z. Zhang, J. Geiger, J. Pohjalainen, A. E.-D. Mousa, W. Jin, and B. Schuller, "Deep learning for environmentally robust speech recognition: An overview of recent developments," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 9, no. 5, pp. 1–28, 2018.

[6] I. López-Espejo, Z.-H. Tan, J. H. Hansen, and J. Jensen, "Deep spoken keyword spotting: An overview," *IEEE Access*, vol. 10, pp. 4169–4199, 2021.

[7] A. Prasad, P. Jyothi, and R. Velmurugan, "An investigation of end-to-end models for robust speech recognition," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6893–6897.

[8] C. Cioflan, L. Cavigelli, M. Rusci, M. De Prado, and L. Benini, "Towards on-device domain adaptation for noise-robust keyword spotting," in *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 2022, pp. 82–85.

[9] Y. Ephraim and H. L. Van Trees, "A signal subspace approach for speech enhancement," *IEEE Transactions on speech and audio processing*, vol. 3, no. 4, pp. 251–266, 1995.

[10] Y. Lu and P. C. Loizou, "Speech enhancement by combining statistical estimators of speech and noise," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2010, pp. 4754–4757.

[11] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *Interspeech*, vol. 2013, 2013, pp. 436–440.

[12] P. Pertilä and J. Nikunen, "Microphone array post-filtering using supervised machine learning for speech enhancement." in *Interspeech*, 2014, pp. 2675–2679.

[13] S. Pascual, A. Bonafonte, and J. Serrà, "Segan: Speech enhancement generative adversarial network," *Interspeech 2017*, pp. 3642–3646, 2017.

[14] M. Nikzad, A. Nicolson, Y. Gao, J. Zhou, K. K. Paliwal, and F. Shang, "Deep residual-dense lattice network for speech enhancement," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 8552–8559.

[15] Y. Xiong, V. Berisha, and C. Chakrabarti, "Residual+ capsule networks (rescap) for simultaneous single-channel overlapped keyword recognition." in *Interspeech*, 2019, pp. 3337–3341.

[16] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[17] C. K. Reddy, E. Beyrami, J. Pool, R. Cutler, S. Srinivasan, and J. Gehrke, "A scalable noisy speech dataset and online subjective test framework," *Proc. Interspeech 2019*, pp. 1816–1820, 2019.

[18] R. Tang and J. Lin, "Deep residual learning for small-footprint keyword spotting," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5484–5488.

[19] A. Defossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," in *Interspeech*, 2020.

[20] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[21] G. Wichern, J. Antognini, M. Flynn, L. R. Zhu, E. McQuinn, D. Crow, E. Manilow, and J. L. Roux, "Wham!: Extending speech separation to noisy environments," *arXiv preprint arXiv:1907.01160*, 2019.

[22] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.