Noise Suppression in Speech Signals Using a GRU-Based Deep Learning Model

Yahya Sagdic[1], Mustafa Namdar[1], Arif Basgumus[2]

[1]Dept. of Electrical and Electronics Eng., Kutahya Dumlupinar University, Kutahya, Turkiye
yahya.sagdic@ogr.dpu.edu.tr, mustafa.namdar@dpu.edu.tr

[2]Dept. of Electrical and Electronics Eng., Bursa Uludag University, Bursa, Turkiye
basgumus@uludag.edu.tr

Abstract

This paper presents an artificial intelligence-based noise suppression system designed to enhance the intelligibility of speech signals in noisy environments. In the proposed system, a gated recurrent unit (GRU)-based neural network is trained using features extracted from time-frequency analysis. The model generates gain values for each frequency component by leveraging the spectral characteristics of the input signal, thereby suppressing noise while preserving the speech signal. Furthermore, a Gradio-based web interface has been developed to ensure user-friendliness. Experimental results demonstrate that the model substantially improves speech quality under various noise conditions and can be effectively applied in practical scenarios.

1. Introduction

Speech communication plays a critical role in technologies such as teleconferencing, voice assistants, human-computer interaction, and hearing-aid devices. However, background noise in real-world settings often degrades intelligibility and perceptual quality, particularly under low signal-to-noise ratio (SNR) conditions. Conventional noise suppression techniques have been used for decades, including Wiener filtering and spectral subtraction. Although effective in stationary noise environments, they often fail to adapt under dynamic conditions due to their reliance on fixed statistical assumptions [1].

The emergence of deep learning (DL) has led to significant progress in speech enhancement, enabling models to learn complex temporal and spectral patterns directly from data. Recurrent neural networks (RNNs), and particularly their variants such as the Gated Recurrent Unit (GRU), have attracted attention due to their ability to model long-term dependencies while maintaining relatively low computational complexity compared to Long Short-Term Memory (LSTM) networks. Thus, GRU-based architectures are promising candidates for real-time speech enhancement applications, where accuracy and efficiency are critical [2, 3]. Several previous works have highlighted the advantages of deep recurrent models for speech enhancement. Wang *et al.* [2] proposed a fused-feature GRU-based model that significantly improved speech intelligibility by combining deep neural networks with temporal modeling. Saleem *et al.* [3] introduced DeepResGRU, which incorporates residual connections into bidirectional GRU layers for improved denoising and recognition performance. Valin [1] presented a hybrid approach that integrates digital signal processing (DSP) with recurrent neural networks, enabling real-time full-band speech suppression. Cheng *et al.* [4] further enhanced computational efficiency by proposing a dynamic GRU framework that selectively updates hidden units, achieving substantial resource savings. Moreover, Ge *et al.* [5] developed PercepNet+, incorporating SNR and phase-aware GRU mechanisms to improve magnitude and phase spectrum modeling. Westhausen and Meyer [6] proposed the Dual-Signal Transformation LSTM Network (DTLN), demonstrating that recurrent architectures can be effectively applied to Short-Time Fourier Transform (STFT)-based and learned transformation domains. Hao *et al.* [7] introduced UNetGAN, a time-domain GAN-based framework capable of enhancing speech in extremely low SNR conditions. More recently, Li *et al.* [8] investigated causal recurrent models that jointly perform enhancement and recognition, showing the practicality of GRU-like structures in latency-sensitive applications.

Building upon these insights, our study proposes a GRU-based DL [9–15] framework for speech noise suppression. The model is designed to achieve robust performance across diverse noise conditions while maintaining computational efficiency, thus contributing to developing real-time and resource-constrained speech enhancement systems. The aim of this study is to design an effective and low-complexity artificial intelligence (AI)-based noise suppression system that enhances speech intelligibility under noisy conditions. To this end, a GRU-based noise suppression model has been developed to suppress noise across varying conditions, improve speech quality, and provide users with an easily accessible interface [16–19].

The remainder of this paper is organized as follows: Section 2 introduces the proposed methods, including the preparation of training data, feature extraction, and the GRU-based model architecture. Section 3 presents the experimental results that evaluate the performance of the proposed system. Finally, Section 4 concludes the paper with a discussion of potential application areas.

2. Proposed Methods

The AI-based noise suppression system developed in this study, whose overall architecture is depicted in Figure 1 initiates by constructing paired training data, in which clean speech recordings are mixed with diverse noise signals at multiple SNR levels. Spectral features are then extracted from the resulting audio, and frame-wise extended feature vectors are formed. These features are provided to a GRU-based neural network, which performs suppression via spectral masking. At inference, the model outputs are applied to real-time audio; the spectrally enhanced signal is transformed back to the time domain using the inverse Fourier transform (FT) and delivered to the user.

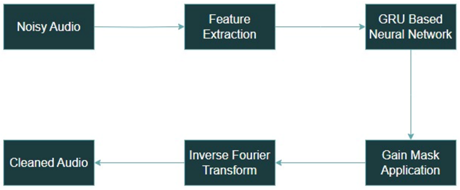

Figure 1. A block diagram summarizing the overall structure of the proposed system.

2.1. Training Data Generation

In the proposed system, a supervised learning approach is adopted for model training, as it is widely used in recent studies [20–22]. In this setting, the network learns to map each noisy input to its corresponding clean target. Because obtaining a clean reference that exactly matches a noisy speech signal is rarely feasible in practice, paired data are synthesized by mixing clean speech with environmental noise at multiple SNR levels. For this purpose, the McGill TSP speech dataset [23] and the DEMAND noise dataset [24] are employed. The training data generation pipeline begins with normalization of the audio signals.

Both the clean signal $c[n]$ and the noise signal $w[n]$ are amplitude-normalized so that their sample values lie within the interval $[-1, 1]$. The noisy input signal is demonstared by $x[n]$ and defined in he following expression as,

$$x_{\text{norm}}[n] = \frac{x[n]}{\max\left(x[n]\right) + \varepsilon} \qquad \varepsilon > 0. \tag{1}$$

The objective is to remove biases introduced by variability in recording levels across audio files and to map all signals onto a common, comparable scale. After normalization, the average power of each signal is computed as the root mean-squared value:

$$P_c = \frac{1}{N}\sum_{n=1}^{N} c[n]^2, \qquad P_w = \frac{1}{N}\sum_{n=1}^{N} w[n]^2. \tag{2}$$

Here, P_c denotes the power of the clean signal and P_w denotes the power of the noise. These quantities are used to determine the noise power required for a desired SNR. For a target SNR (in dB), the corresponding noise power is

$$P_{w,\text{target}} = \frac{P_c}{10^{\text{SNR}/10}}. \tag{3}$$

The noise signal is then scaled to attain this power level:

$$w'[n] = w[n]\sqrt{\frac{P_{w,\text{target}}}{P_w + \varepsilon}}, \qquad \varepsilon > 0, \tag{4}$$

where ε is a small constant to avoid division by zero. The noisy mixture is obtained by summing the clean and scaled noise, $x[n] = c[n] + w'[n]$. Because mixing can drive amplitudes outside the interval $[-1, 1]$, the resulting signal is normalized once more. Finally, each clean utterance and its corresponding noisy mixture are saved in **wav** format as `clean_XXXXX.wav` and `noisy_XXXXX.wav`, respectively; these pairs serve as target and input examples during model training.

2.2. Feature Extraction

To extract speech-specific information from noisy audio, processing in the time-frequency (TF) domain is adopted instead of direct waveform analysis. This choice reflects that time-domain samples alone are often insufficient to distinguish speech from noise components. TF representations encode both spectral and temporal characteristics of the signal, enabling the model to learn speech-specific patterns more effectively. In the developed system, the feature-extraction pipeline is designed to capture both the spectral content and the temporal dynamics of the speech signal. This stage is crucial for producing more accurate gain masks and comprises several fundamental steps. The signal is first transformed into the frequency domain using the STFT:

$$X(m, k) = \sum_{n=0}^{N-1} x[n]w[n - mH]e^{-j\frac{2\pi kn}{N}}. \tag{6}$$

In this context, m denotes the frame index, k represents the frequency bin index, H refers to the hop length, N indicates the window length (also corresponding to the FFT size), and $w[n]$ stands for the Hann window function. This transformation divides the audio signal into short segments, enabling spectral analysis. The spectrum obtained from the short-time Fourier transform (STFT), denoted as $|X(m, k)|^2$, is subsequently mapped onto the Mel scale, yielding $S_{\text{mel}} = \text{Mel}\left(|X(m, k)|^2\right)$. The Mel scale is designed to approximate the perceptual characteristics of the human ear, providing finer resolution at lower frequencies and coarser resolution at higher frequencies. This approach is particularly effective in accurately representing the meaningful frequency components of speech signals.

Initially, a logarithmic transformation is applied to the signal in order to compress its dynamic range. Subsequently, the Mel spectrogram is transformed by the Discrete Cosine Transform (DCT) to extract the Mel-Frequency Cepstral Coefficients (MFCCs) as follows:

$$S_{\text{log}}[k] = \log_{10}(S_{\text{mel}}[k] + \epsilon), \tag{5}$$

$$\text{MFCC}[m] = \sum_{k=0}^{K-1} S_{\text{log}}[k]\cos\left(\frac{\pi m(k + 0.5)}{K}\right). \tag{6}$$

In this formulation, m denotes the index of the cepstral coefficient ranging from 0 to 12, K represents the total number of Mel filter banks (typically set to 40) and the DCT is employed to decorrelate the resulting coefficients, thereby enhancing their suitability for machine learning applications. MFCCs represent the short-time power spectrum of an audio signal on the Mel scale. They are particularly effective in capturing the distinctive structure of speech, making them widely used in speech recognition applications. The DCT decorrelates the coefficients, thereby facilitating model learning. To capture temporal variations in speech, the first-and second-order derivatives of the MFCC coefficients are also extracted as

$$\Delta = \delta(\text{MFCC}), \quad \Delta^2 = \delta(\text{MFCC, order} = 2). \tag{9}$$

The first derivative, Δ, indicates the rate of change in speech features over time (i.e., "velocity"), while the second derivative, Δ^2, represents the rate of change of this velocity (i.e., "acceleration"). The symbol δ represents a function that computes derivatives using a finite difference approach. These features are crucial for distinguishing between stationary and transient speech segments. For each frame, a 60-dimensional feature vector is constructed by concatenating MFCC, Δ, and Δ^2:

$$\text{features}_t = [\text{MFCC}_t,\, \Delta_t,\, \Delta_t^2] \in \mathbb{R}^{60}. \tag{10}$$

However, speech is not a sequence of independent frames; a single phoneme typically extends across multiple consecutive frames. Therefore, to enable the model to learn temporal context, information from both preceding and succeeding frames is incorporated. A contextual window of five frames (± 2 frames) is used for each frame, resulting in a 300-dimensional vector. Each of these vectors corresponds to one frame and is provided as input to the model.

2.3. Model Architecture and Training

The model architecture employed in this study is designed to capture the time-frequency dependencies of speech signals. The model sequentially processes the 300-dimensional input feature vectors through a fully connected layer (FC1), a two-layer GRU network, and an output layer (FC2), ultimately producing a gain mask in the frequency domain for each frame. The first layer (FC1) applies a linear transformation to the input feature vector, followed by a ReLU activation function:

$$h_1 = \mathrm{ReLU}(W_1 x + b_1). \tag{11}$$

This layer transforms the mixed and high-dimensional input into a more meaningful representation, thereby enabling the model to better distinguish relevant features. In the first fully connected layer, W_1 represents the trainable weight matrix, while b_1 corresponds to the bias vector, both of which are updated during the training process. The ReLU activation sets negative values to zero, which facilitates learning and reduces irrelevant information. In the second stage, the learned representations are passed to the GRU network, a recurrent neural network (RNN) variant designed to model sequential relationships in time-series data. The two-layer GRU captures dependencies between consecutive frames over time:

$$h_2 = \mathrm{GRU}(h_1). \tag{12}$$

The internal gating mechanisms of the GRU allow it to retain past information, making it particularly effective in modeling contextual dependencies in temporal data such as speech [25]. The output layer then processes the GRU output and generates a value between 0 and 1 for each frequency, indicating which frequency components should be preserved and which should be attenuated:

$$y = \sigma(W_2 h_2 + b_2). \tag{13}$$

Here, σ denotes the sigmoid activation function, which constrains the output values to the $[0, 1]$ range. This enables the model to assign values close to 0 to noise (suppressing it), and values close to 1 to important speech components (preserving them). In the output layer, W_2 and b_2 correspond to the trainable weights and biases, which are optimized during the training process. The gain vector predicted by the model is compared with the target gain, and the loss function is computed using the Mean Squared Error (MSE):

$$L = \frac{1}{N} \sum_{i=1}^{N} \left| Y_i - \hat{Y}_i \right|^2. \tag{14}$$

This loss quantifies the discrepancy between the predicted gain mask $\hat{Y}_i$, and the target gain mask Y_i, across all N frames in the batch. The objective is to minimize this difference, enabling the model to generate more accurate spectral masks. Training was conducted over 30 epochs, with optimization performed using mini-batches of 256 samples randomly drawn from different data batches in each epoch. The Adam optimization algorithm was employed, and the learning rate was reduced by half every 10 epochs to ensure stable convergence of the model.

2.4. Noise Reduction and Reconstruction Process

During the application phase, the effect of the gain masks learned by the model on the audio signal is computed, and a noise-suppressed signal is generated using these masks. This process consists of four main steps: transformation to the frequency domain, feature extraction and gain mask estimation, mask application, and inverse transformation with normalization.

First, the noisy time-domain signal is transformed into the frequency domain using the FT. Various spectral features are then extracted from the resulting signal and grouped into 300-dimensional vectors, which serve as the model's input. For each frame, the model produces a gain value $G_{m,k} \in [0, 1]$ for each frequency component. These values determine whether the corresponding component should be suppressed, depending on the model's prediction of whether it contains noise. Noise reduction is performed by multiplying the amplitude spectrum with the gain mask obtained from the model:

$$|X_{m,k}^{\text{denoised}}| = |X_{m,k}^{\text{noisy}}| G_{m,k}. \tag{15}$$

In this process, frequency components predicted to contain speech (high $G_{m,k}$) are preserved, while those predicted to contain noise (low $G_{m,k}$) are suppressed. The inverse Short-Time Fourier Transform (ISTFT) is subsequently used to convert the masked spectrum back to the time domain. The reconstructed audio signal is normalised to control the output level and prevent excessive amplitudes. Through this process, the model dynamically suppresses noise by applying learned frequency-dependent decisions and resynthesizes the output signal both temporally and spectrally. This approach provides a more flexible and data-driven enhancement compared to classical filtering methods. Alternatively, generative adversarial networks (GAN)-based models can also be employed to further improve the performance of speech recognition systems [26].

To enable interactive experimentation with the developed system, a practical user interface was implemented as a web-based application using the Gradio library [27]. Through this interface, users can upload an audio file and listen to its noise-suppressed version within seconds.

3. Experimental Results

The AI-based noise suppression system developed in this study was subjected to a comprehensive evaluation process. The analyses were designed not only to demonstrate the theoretical accuracy of the model but also to emphasize its practical effectiveness. In this section, the performed analyses are presented together with their corresponding results.

3.1. SNR Comparison

In Figure 2(a), the SNR values of a sample audio file used during training are compared before and after applying the model to assess its noise suppression performance. The low SNR of the noisy input signal indicates a substantial loss of information, whereas the higher SNR obtained after applying the model demonstrates that the noise has been effectively suppressed and the speech signal has become more intelligible.

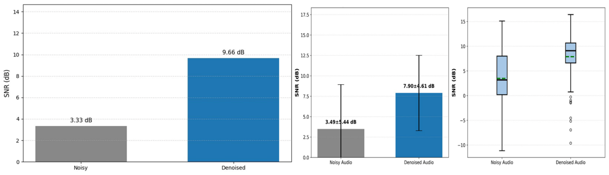

Figure 2. (a) Comparison of the SNR variations observed in a representative audio sample, comparing the noisy input with the denoised output generated by the proposed model. (b) performance evaluation under a Monte Carlo simulation framework comprising 100 independent trials.

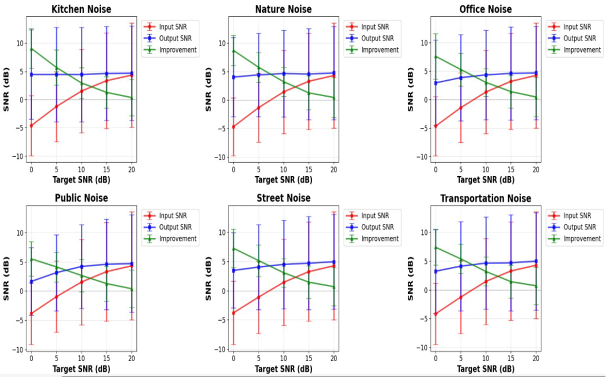

Figure 3. Distributions of SNR across different noise categories, along with the corresponding improvement profiles.

Furthermore, the Monte Carlo simulation results presented in Figure 2(b) demonstrate consistent performance gains across 100 randomly generated test mixtures, yielding an average improvement of approximately +4–6 dB while exhibiting lower variance relative to the noisy input.

To assess the model's robustness under diverse real-world noise conditions, it was evaluated across six representative acoustic environments: Kitchen, Nature, Office, Public, Street, and Transportation. As illustrated in Figure 3, the processed audio exhibited notable clarity improvements. The model consistently enhanced perceptual sound quality in all scenarios, yielding an average gain of 4–6 dB, even in the presence of dynamically varying background noise.

Figure 4(a) provides a detailed analysis of the model's performance across varying input SNR levels ranging from 0 to 20 dB. The denoising algorithm demonstrates its highest effectiveness in low-SNR scenarios (0–5 dB), yielding substantial relative improvements, while preserving consistent output quality at higher SNR values. Figure 4(b) presents the distribution of SNR enhancements across all non-stationary noise categories. Each boxplot reflects the range of SNR gains (in dB) obtained from multiple Monte Carlo simulations. The majority of samples are positioned well above the zero-improvement threshold (indicated by the dashed red line), affirming the model's consistent ability to improve signal clarity.

3.2. Spectrogram Analysis

Spectral characteristics were examined using both linear and Mel-scale spectrogram representations, as shown in Figures 5(a) and 5(b). The noisy spectrogram reveals prominent high-frequency noise components, whereas the denoised counterpart exhibits substantial attenuation in these regions, thereby uncovering harmonic patterns that closely resemble the clean reference signal. The Mel-scale spectrograms further emphasize the model's ability to recover perceptually important frequency bands, particularly within the 500 Hz to 4 kHz range.

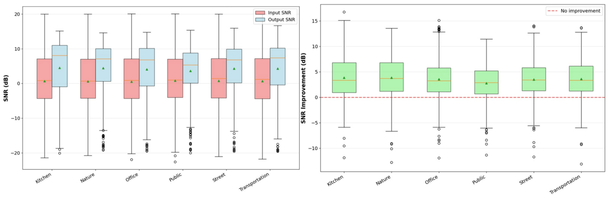

Figure 4. (a) Evaluation of the model's performance across distinct noise environments. (b) Statistical distribution of SNR gains achieved by the model when exposed to dynamic, real-world noise environments.

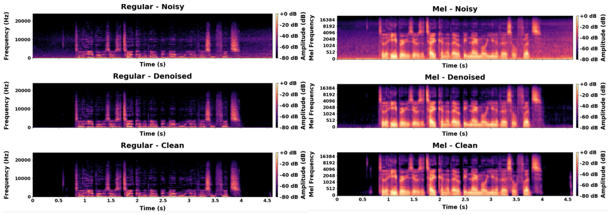

Figure 5. (a) Linear spectrogram representations of the noisy, denoised, and clean signals. (b) Mel-scale spectrograms illustrating the same signal conditions for perceptual analysis.

3.3. Waveform Comparison

Figure 6 presents a visual comparison of the model's noise suppression performance in the time domain. The graph displays the waveforms of the same audio sample, where the noisy signal at the top illustrates the presence of background noise throughout the entire duration, substantially reducing the clarity of the speech segments. In contrast, the cleaned signal at the bottom demonstrates that the model effectively suppresses the noise and enhances the intelligibility of the speech.

3.4. Triple Spectrogram Comparison

To evaluate the model's performance not only in terms of noise reduction but also in terms of its ability to reproduce the target clean signal, a training sample containing both clean and noisy versions was selected. The spectrograms of the noisy, cleaned, and clean versions of this sample are compared in Figure 7. This comparison demonstrates how accurately the model has learned to reconstruct the clean signal from the training data.

4. Conclusions

In this study, an effective noise suppression system was developed using a GRU-based neural network model. Feature extraction based on MFCCs and their derivatives enabled the modeling of contextual information, leading to high-accuracy spectral masking. The trained model exhibited robustness against various types of noise and enhanced speech quality across multiple evaluation metrics. With further improvements to the system architecture, the proposed approach could be effectively applied in hearing aids, video conferencing, defense industry applications, and voice assistant systems, highlighting its strong potential for widespread real-world deployment.

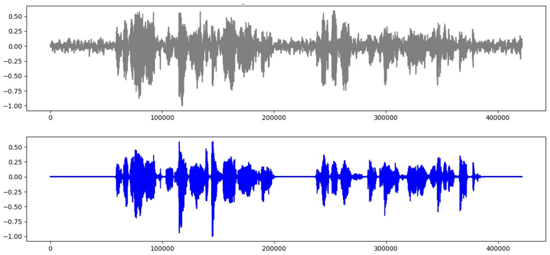

Figure 6. Time-domain comparison of the model's noise suppression performance ((a) noisy audio waveform versus (b) denoised audio waveform).

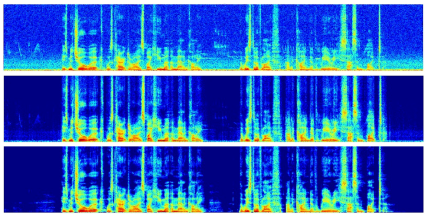

Figure 7. Spectrogram comparison of a training sample in (a) noisy, (b) cleaned, and (c) clean versions, illustrating the model's performance in noise reduction and accurate reproduction of the target clean signal.

5. References

[1] J. M. Valin, "A hybrid DSP/deep learning approach to real-time full-band speech enhancement," in *Proc. IEEE Workshop on Multimedia Sig. Proc. (MMSP)*, 2018.

[2] Y. Wang, *et al.* "Speech enhancement from fused features based on deep neural networks and gated recurrent unit network," *EURASIP J. on Adv. in Sig. Proc.*, vol. 2021, no. 31, pp. 1–15, 2021.

[3] N. Saleem, *et al.* "DeepResGRU: Residual gated recurrent neural network-based model for speech enhancement and recognition," *Knowledge-Based Sys.*, vol. 238, 2021.

[4] L. Cheng, *et al.* "Dynamic gated recurrent neural network for compute-efficient speech enhancement," in *Proc. Interspeech*, 2024, pp. 4264–4268.

[5] X. Ge, et al., "PercepNet+: A phase and SNR aware PercepNet for real-time speech enhancement," in *Proc. Interspeech*, pp. 916–920, 2022.

[6] N. L. Westhausen and B. T. Meyer, "Dual-signal transformation LSTM network for real-time noise suppression," in *Proc. ICASSP*, 2020, pp. 2477–2481.

[7] X. Hao, et al., "UNetGAN: A robust speech enhancement approach in time domain for extremely low SNR conditions," in *Proc. Interspeech, pp. 1786–1790*, 2019.

[8] Z. Li, et al., "Deep causal speech enhancement and recognition using efficient long-short term memory recurrent neural network," *Plos One*, vol. 19, no. 1, 2024.

[9] M. Namdar, et al., "Ergodic capacity estimation with artificial neural networks in NOMA-based cognitive radio systems," *Arabian J. for Science and Eng.*, vol. 49, no. 5, pp. 6459–6468, 2024.

[10] E. Can, et al., "Deep learning based target tracking and diving algorithm in kamikaze UAVs," in *Proc. 2024 Innovations in Int. Sys. and App. Conf.*, Oct. 2024, pp. 1–6.

[11] E. Bayhan, et al., "Unimpeded walking with deep learning," in *Proc. 2022 Sig. Proc. and Comm. App. Conf.*, May 2022, pp. 1–4.

[12] E. Bayhan, et al., "Deep learning based object detection and recognition of unmanned aerial vehicles," in *Proc. 2021 Int. Congr. Human-Computer Interaction, Opt. and Robotic App. (HORA)*, Jun. 2021, pp. 1–5.

[13] Y. E. Kar, et al., "Machine learning assisted autonomous vehicle design and control," in *Proc. Int. Symp. Multidisciplinary Studies and Innov. Tech.*, Oct. 2021, pp. 462–466.

[14] M. Taser, et al., "The effect of interference limited area method on the total data rate in device-to-device communication," in *Proc. Int. Symp. Multidisciplinary Studies and Innov. Tech.*, Oct. 2020, pp. 1–5.

[15] M. Namdar, et al., "NOMA tabanlı bilişsel radyo sistemlerinde sinir ağı yöntemleri ile ergodik kapasite tahmini ve başarım analizi," *Uludag Universitesi Muhendislik Fakultesi Dergisi*, vol. 28, no. 1, pp. 253–272, 2023.

[16] Y. Xu, et al., "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. on Audio, Speech, and Lang. Proc.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[17] X. Hao, et al., "An attention-based neural network approach for single channel speech enhancement," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc.*, Brighton, UK, 2019, pp. 6895–6899.

[18] J. Abdulbaqi, et al., "Residual recurrent neural network for speech enhancement," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc.*, Barcelona, Spain, 2020, pp. 6659–6663.

[19] M. Liu, et al., "Speech enhancement method based on LSTM neural network for speech recognition," in *Proc. IEEE Int. Conf. Sig. Proc.*, Beijing, China, 2018, pp. 245–249.

[20] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Transactions on Audio, Speech, and Lang. Proc.*, vol. 26, no. 10, pp. 1702–1726, Oct. 2018.

[21] Y. Li, et al., "Joint learning with shared latent space for self-supervised monaural speech enhancement," in *Proc. Sensor Sig. Proc. for Defence Conf.*, Edinburgh, UK, 2023, pp. 1–5.

[22] Z. Cui, et al., "Semi-supervised speech enhancement based on speech purity," in *Proc. IEEE Int. Conf. Acoustics, Speech and Sig. Proc.*, Rhodes Island, Greece, 2023, pp. 1–5.

[23] McGill University, "Audio and speech datasets," [Online]. Available: `https://www-mmsp.ece.mcgill.ca/Documents/Data/`

[24] Zenodo, "DEMAND: Diverse Environments Multichannel Acoustic Noise Database," [Online]. Available: `https://zenodo.org/records/1227121`

[25] H. Zhao, et al., "Convolutional-recurrent neural networks for speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 2401–2405.

[26] C. Donahue, et al., "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 5024–5028.

[27] Gradio, "Gradio: Build and share machine learning demos," [Online]. Available: `https://www.gradio.app`