Neuro-Fuzzy voice quality improvement for low-power, half-duplex, communication

Joyraj Chakraborty* *Member, IEEE*, Martin Reed† *Member, IEEE*, Nikolaos Thomos† *Senior Member, IEEE*, Geoff Pratt‡, and Nigel Wilson‡

***Abstract*—Half-duplex mobile radio systems are commonly used for speech communication, often in challenging environments where background noise may prevent speech intelligibility. These systems usually operate with modest battery capacity which strictly limits the available computational power. In this paper, we present a noise suppression approach that uses lightweight machine learning. The machine learning leverages a neuro-fuzzy logic-based neural network to create accurate noise estimation that is used adaptively to create a filter for noise reduction. The system buffers a number of noise samples, triggered by the half-duplex key press, allowing the solution to adapt to recent changes in the background auditory noise. The selection of a neuro-fuzzy logic-based neural network is driven by the necessity for a low-power implementation suitable for mobile, power-constrained terminals. The proposed system is compared with both traditional noise suppression methods as well as with a deep convolutional neural network (DCNN) variant of our system. The results demonstrate that our system significantly outperforms traditional methods in terms of both subjective and objective quality metrics. Further, they show that, although the DCNN variant achieves comparable performance, it has a high computational cost which renders it unsuitable for low-power digital personal radio systems. To validate its practicality, the proposed method is tested in a real-time system, demonstrating its compatibility with constrained devices.**

***Index Terms*—Speech Enhancement, Adaptive Noise Cancellation, Vocoder, Half-duplex communication, ANFIS.**

I. Introduction

MOBILE speech communication is a widely used in everyday life. Half-duplex speech communication is a vital subset of this application space serving important industries such as telecommunications, public safety, and transportation. While such communication systems are effective for long-distance communication, they often rely on a single-channel microphone and are battery powered, which imposes limitations in the opportunities for signal processing. Additionally, mobile voice communication systems rely on low-bit-rate voice codecs due to limited channel capacity, which often results in added distortion. As a result, such systems do not perform well in noisy auditory environments, leading to degraded voice quality and signal distortion. To reduce the distortion introduced by background noise, voice enhancement techniques can be employed to suppress the noise in the content, thus improving speech intelligibility. Existing solutions that can enhance speech by suppressing noise sources from mixed signals include: classical approaches that have been used for decades, such as spectral subtraction schemes [1], Minimum tracking spectral subtraction (MTSS) [2] (building on seminal work by Martin [3]), and adaptive filtering techniques [4], [5]. Other schemes such as the Log-minimum Mean Square Error (LMMSE) algorithm improve speech enhancement by effectively mitigating noise while preserving signal quality [6], but face limitations when dealing with rapid and unpredictable noise changes due to their reliance on short noise frames. Improved performance can be achieved using automatic gain control [7] and Kalman filtering [8]. These hybrid, classical, methods have been the preferred approach to real-time speech enhancement systems as they use modest computational power that suits the limited storage and computational capabilities of the mobile radio devices. However, these techniques struggle to remove non-stationary noise because they over-smooth or even distort the desired signal, especially when the noise characteristics vary significantly over a period of time. Another weakness of these methods is that they assume linearity and stationarity, which is not often the case in real-life noise scenarios. The weaknesses in existing systems motivate the introduction of low-complexity non-linear and adaptive machine learning solutions as proposed in this paper.

A recent review on supervised learning algorithms presented in [9] highlights that machine learning (ML) algorithms are in general more efficient than traditional methods for speech enhancement. For example, automatic noise class detection using artificial neural networks has shown promising results [10]. In [11], non-negative matrix factorization (NMF) and Robust Principal Component Analysis (RPCA) have been shown to be efficient for speech enhancement. However, general ML techniques – such as neural networks – are computationally expensive and require complex weight updating to efficiently reduce noise [9] thus prohibiting their use in low-power systems [12], [13] such as the application area of this paper. The computational complexity of neural networks can be alleviated by combining them with fuzzy logic that reduces the scale of the required neural network. This hybrid solution combines the ability of fuzzy logic to manage uncertainty with the power of neural networks to capture complex relationships between variables. For example, the adaptive neuro-fuzzy inference system (ANFIS) requires significantly fewer computational resources compared to conventional neural networks [12], [14]. Indeed, ANFIS has been used as a simple ML-based filter for speech enhancement [15]. This prior example

J. Chakraborty, is with Department of Engineering Science, University of Oxford, UK. M. Reed and N. Thomos are with the University of Essex, Colchester, UK; G. Pratt and N. Wilson are with CML Microcircuits Ltd, Langford, UK

This work was supported by Innovate UK under Knowledge Transfer Partnership number 012507.

has only been applied to white noise sources and requires subtraction for each frame that can result in the creation of perceptually annoying artifacts in more complex real-world scenarios. However, in stationary, ideal, conditions, this method can achieve a high degree of accuracy and robustness giving rise to the motivation for its use as part of the proposed solution.

In this paper, we propose a novel method to adaptively estimate changing noise in narrow-band half-duplex communication systems. It is worth noting that our method can be incorporated into any mobile communication system with minor adaptation. Our system is substantially evolved from previous work [16], where we introduced a history database model for post-decoder speech denoising. In contrast to other approaches, our denoising system is applied prior to the encoder. Additionally, we utilize the push-to-talk button for noise estimation. The employed push-to-talk mechanism is an exemplar mechanism and can be substituted with voice activity detection [17] or source separation [18], rendering our methodology appropriate to various mobile communication systems. However, to simplify the description, our current work assumes a half-duplex system. We have also extensively evaluated our system and compared its performance with traditional approaches and a deep convolutional neural network (DCNN). Specifically, we employ ANFIS for denoising as it has proven effective in non-stationary and low SNR noisy scenarios, but unlike previous works, we apply it online and, hence, continuously retrain the system. The retraining is based upon small frames of historical, buffered, noise samples to estimate noise characteristics. The training is triggered by the key-press that is made at the start of each "push-to-talk" half-duplex speech segment and draws on the buffered noise samples before the key-press. This can be generalised in future work through use of voice activity detection to replace the need of the key-press event. The estimated noise is used for denoising the speech frames using a combination filter consisting of a Wiener filter and spectral subtraction. This adaptive approach allows us to greatly reduce the amount of noise energy in speech frames while minimizing distortion caused by *over-subtraction*. Furthermore, as we show in the experimental evaluation, it leads to improved communication quality in noisy environments. To summarize, the main contributions of our paper are as follows:

- we propose a novel noise suppression mechanism that uses knowledge from a half-duplex key-press event to sample noise existing before speech starts. However, our system can be generalised, for example to full-duplex systems, with the addition of voice activity detection.
- we employ ANFIS for the first time in half-duplex communication to develop a low-cost system that respects the low-power nature of the considered devices;
- we propose a dictionary-based database approach as an addition mechanism to the key-based noise suppression to further advance the performance of our system in the absence of noise samples. This has insignificant computational cost and only adds a small storage overhead;
- we have extensively evaluated the performance of our system and compared it with other methods, including CNN-based noise reduction. The results demonstrate the superiority of our system in terms of performance compared to traditional systems. Furthermore, they show that while the performance of our approach is comparable to a CNN, it uses significantly lower computation resources than a CNN, indeed a CNN is found to be totally unsuitable for low-power systems;
- we have deployed and tested a real-time implementation of our system to understand the performance considering hardware and power limitations. The results show the feasibility of deploying our approach in real-world battery-powered systems and that the performance is consistent with the simulations.

The rest of the paper is organized as follows. We first discuss related literature in Section II. Then, in Section III, we briefly present the overview of the proposed system before describing our proposed method in detail in Section IV. Next, we present the experimental setup and extensively evaluate the performance of our system in Section V. Finally, we draw conclusions in Section VI.

II. Related work

Traditional signal processing-based noise reduction methods, such as spectral subtraction [1], [19], were built on the assumption of a linear-time domain relation between noise and speech. Spectral subtraction methods subtract the noisy speech frame from the whole speech frame with the estimated noise generated using early speech frames. A major assumption is that there is no speech frame during those periods. Spectral subtraction requires a good estimation of the noise to perform well; however, it can only suppress some artifacts to a certain extent. In order to mitigate this problem, a nonlinear spectral subtraction method was proposed in [20]. Alternatively, multi-band spectral subtraction is used in [21], where spectral subtraction is done separately for each band. Although this approach achieves good performance, it is computationally expensive and introduces additional latency due to its reliance on multi-band spectral estimation. A dual-channel noise reduction algorithm was introduced in [22] for cellular communication. This approach used a combination of a coherence-based algorithm and a Kalman filter for noise reduction. Again, while showing good performance, the additional hardware components to obtain the dual-channel input signals, significantly increase the cost and complexity of the overall system and renders it inappropriate for low-power systems.

The most common approach for speech enhancement is Wiener filtering [23], [24]. It is based on the assumption that speech and noise spectra are Gaussian and uncorrelated. It is considered optimal for noise suppression in communication systems, as it effectively preserves signal quality by adapting to the statistical properties of both the signal and the noise, making it particularly well-suited for linear, stationary systems with known statistics. The main drawback of Wiener filtering is that it requires a significant amount of silent periods to accurately estimate the prevailing noise in the signal. This

requirement poses a challenge in real-life systems, where silent periods are often limited, ultimately affecting sound quality. Typically, Wiener filtering relies on extracting multiple 25 msec of noise chunks, which are then utilized as a reference to subtract from the speech frames, resulting in noise reduction. Voice activity detection (VAD) is adopted in [25] to overcome the problem of detecting suitable periods of noise without speech. This method assumes that noise is present all the time and uses VAD to estimate the (speech) silence periods when the noise can be sampled. Recently, the NMF method has become popular for speech enhancement, where the NMF constructs a noise model during a non-speech frame presuming it follows a Gaussian distribution. In [26], the authors proposed a NMF with an online update of speech and noise bases for speech enhancement. However, the common assumption of Gaussian distribution may not be optimal when working with a limited frame size, as it may fail to capture intricate characteristics of noise data as shown in [27]. An approach to advance Wiener filtering is by employing a codebook[1] [28]. The gain functions of these two methods are weighted to obtain an integrated gain. Codebook-based methods rely on predefined fixed codebooks, which might not be able to adequately represent all types of noise variations encountered in real-life scenarios. The use of codebooks is a costly process that can be partly alleviated as suggested in [29], where generative dictionary learning is proposed for speech enhancement. However, it can fail when noise is non-stationary or SNR is low.

In [30], the authors proposed a real-time unsupervised method by combining a pre-learned universal non-negative matrix factorization dictionary algorithm with the generalized cross-correlation for speech enhancement, yielding notable improvements in audio source separation and speech intelligibility compared to traditional methods. In tackling varying background noise, artificial neural networks, especially deep learning methods, are a promising solution for improving speech quality [31]. Deep learning approaches have become popular for speech enhancement in the past few years [32]–[34]. These studies show impressive gains but they require significant storage and computational resources. A recent study [35] on a monaural speech enhancement model that uses noise modeling has shown promising results. However, the model's 1.54 million parameters make it too resource-intensive for edge devices. Although cloud or federated learning allows some applications to bypass these resource constraints, this is not possible for radio communication systems which require low latency and have commonly limited communication resources. In the domain of smart speaker [36], prior work has explored 1DCNN-based predictors, demonstrating successful far-field communication; however, the computational demands, exemplified by operation counts of 3.5 M/s, still pose challenges for resource-constrained embedded implementations. In another study [37], the Codec-SUPERB challenge is introduced to evaluate neural audio codecs using a DNN-based model with 73 million parameters. This approach moves away from traditional standards, focusing instead on preserving low-bitrate speech quality. However, the model's high parameter count presents computational challenges for deployment on low-power devices. There has been some work to develop deep learning frameworks suitable for mobile devices [38]–[41] but all these techniques require a mobile GPU or accelerated hardware which is not generally appropriate for low-power systems such as the application area of this paper.

[1]A codebook is a collection of spectral patterns or vectors that represent different speech and noise characteristics

III. System and Problem Overview

In this section, we present the overview of a half-duplex communication system illustrated in Fig. 1 that is used for voice communication in many applications, such as military, industrial and medical, among others.[2] Such half-duplex communication systems, often colloquially termed as a *walkie-talkie*, use a portable two-way, half-duplex radio that allows people to communicate wirelessly with each other. These systems use radio spectrum to transmit and receive audio messages over short distances, typically a few kilometers or less. In a half-duplex communication system, one station termed as the local side (L) can send and receive information only in the forward direction. The remote side (R) uses the same system, and the two communicate over a wireless channel. The terminal is used for half-duplex "push-to-talk" communications, and when the key is pushed, it activates the system to transmit the voice to the receiver's end, i.e., the remote side. A standard half-duplex system comprises a user terminal, an A/D converter module, a memory, a Vocoder (for example, AMBE+ [42][3]), and a transceiver. The system described herein operates as a half-duplex system, aligning with the primary focus of this paper. Nonetheless, the underlying process block remains consistent across most voice communication systems. In alternative communication configurations, such as full-duplex or simplex systems, VAD can effectively replace the key button function of the half-duplex system [25], [43]. VAD plays a crucial role in discerning speech, silence, and noise, while optimizing bandwidth usage. Utilizing VAD, one can extract noise chunks, subsequently employing the methodology outlined in this paper to train a full duplex system effectively, although we have not addressed this in this paper. The novel contribution of our paper lies in the adaptive noise suppression block, as shown in Fig. 1, which is integrated as a singular component within the system, and is placed before the Vocoder block, i.e., prior to the encoder.

A. System overview

Half-duplex communication systems are reliable and effective in many applications, and can be deployed quickly, which makes them ideal choice for emergency and ad-hoc situations. In order to deal with noise, denoising algorithms can be applied either before encoding (local side) or after decoding (remote side). When denoising is performed after decoding, as is common, the noise reduction is applied to decompressed speech signals, which have already been degraded by the

[2]The adaptive suppression block is part of our system and is not present in traditional systems.

[3]Supplied by DVSI https://www.dvsinc.com/products/docs.shtml

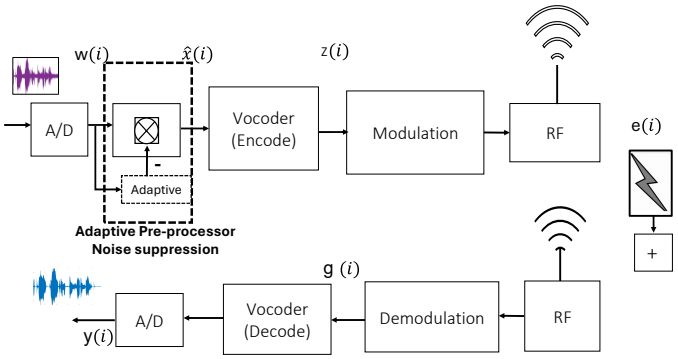

Fig. 1: Simplified block diagram of Half-duplex communication system.

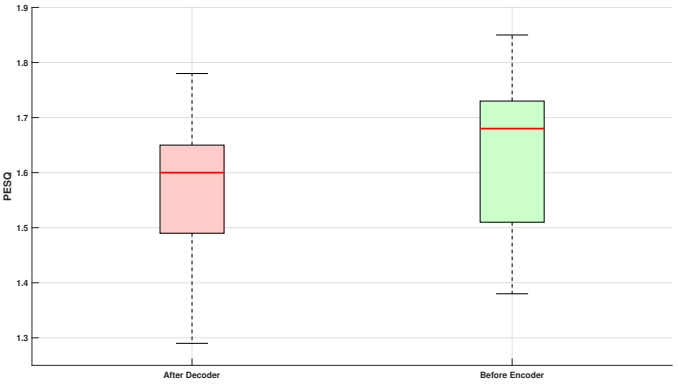

Fig. 2: Achieved PESQ scores when our denoising algorithm is applied before the encoder and after the decoder.

Vocoder. Typically, the encoded speech by a narrowband Vocoder has a perceptual quality measured by means of MOS in the range of $[2.6, 3.4]$, even in the absence of noise. Alternatively, when effective denoising is performed before encoding, the background noise has less effect on the encoding process as the encoders are optimised for speech rather than noise. This can be verified by the comparison shown in Fig. 2, where we show the achieved average PESQ score when applying a simple spectral subtraction method before and after the decoder for twenty noisy sounds. For this comparison, twenty 'clean' male speech clips were mixed with the same street noise. This evaluation motivates us to apply our proposed denoising algorithm *before* the encoder.

B. Problem formulation

In this section, we formally present the considered problem and introduce the related notation. Let us denote the input speech signal by $s(i)$ and the noise signal by $n(i)$. The input signal of the Vocoder, $w(i)$, during active speech (key pressed) is the composite signal at sample index i:

$$w(i) = s(i) + n(i) \tag{1}$$

We will generalize the input signal later to include the non-active period when the key is not pressed.

The Vocoder encodes the input signal $w(i)$ and generates the compressed signal $z(i)$, where $f[]$ represents the encoder. Hence, it is

$$z(i) = f[w(i)]. \tag{2}$$

At the receiver, the recovered signal after the decoder is denoted by $y(i)$. If we now represent the decoder by a function $g[]$, the output signal $y(i)$ can be written as:

$$y(i) = g[f[w(i)]] = s(i) + n(i) + e(i), \tag{3}$$

where the distortion in the speech and noise signals after passing through the encoder and transmission error is represented by $e(i)$.

The noise component $n(i)$ is assumed to be independent of the speech signal. This is the general case as the background noise is not correlated with the original signal, thus the expected cross-correlation R is:

$$R(s(i), n(i)) = 0. \tag{4}$$

IV. Proposed Approach

A. Overview of proposed approach

1) General solution: We begin this section by discussing the fundamental components of our generic solution. Our method performs analysis of the noisy speech in order to characterize the noise. This is an essential part of our system and needs to be done before noise suppression, as it permits to model the noise sources and understand the characteristics of the data. This information is then used to design efficient noise suppression systems. The most common noise characterization method is spectral analysis, which first uses a spectrogram to create a visual representation of the noise and then suppresses it from the sound source. In the context of real-time communication systems, employing this approach may prove inefficient because it relies on the analysis of individual small audio chunks. This poses challenges in effectively distinguishing noise and speech. Hence, it becomes crucial to gain a comprehensive understanding of the noise signal's characteristics before attempting noise reduction.

In order to suppress background noise or other unwanted sounds (e.g., background sounds of others, street sounds, sirens, airport sounds, station sounds, etc.) from a speech frame (during push-to-talk), it is essential to identify the noises before the corresponding speech of interest that has both the noise and the speech mixed together. In one respect, the identification of noise is an independent step from the identification of the noise mixed with the speech signals. The same approach can be applied in full-duplex communication by replacing the key-press events with a voice/event activity detection mechanism [17], [25], [43]. This paper has not considered full-duplex, it would require the voice/event activity to work well in a noisy environment, there have been some recent advancements in this area that could be applied to this work [43], however, it would need further development.

There are two main approaches to characterizing noise used in this paper: (a) buffering of noise samples based on key-press events, and (b) use of a dictionary containing historical noise samples. The proposed noise suppression consists of two stages: noise estimation and noise suppression. Both of these stages are carried out online, where noise estimation aims to characterize the noise in the signal $y(i)$ by using the techniques below. The denoising front end includes a noise

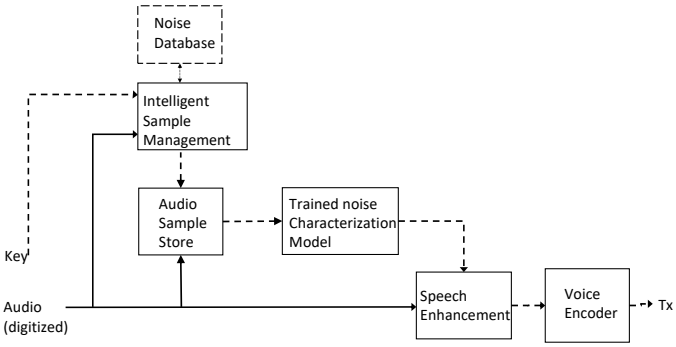

Fig. 3: Proposed adaptive noise suppression mechanism.

estimation module that accurately detects the type of noise in the signal based on buffered noise samples determined from key-press (push-to-talk) events and a dictionary of historical noise samples. These multiple noise chunks are then used for noise estimation using the ML algorithm presented in Section IV-C. In the second stage, the estimated noise is then suppressed in real-time using a combination of a Wiener filter and spectral subtraction to mitigate environmental disturbances. Fig. 3 shows the workflow of our proposed approach, which uses two sources for the noise characterization: either buffering based upon key-press events or a historical noise database; these are each explained below.

Buffering based upon key-press events: In this approach, noise is characterized by analyzing recent data and buffering the data based on key-press events, as shown in Fig. 3. The noise model is derived from input audio sampled before the key is pressed; however, machine learning can take place either before or after the key is pressed. Let $x(t)$ be the input audio signal, where t represents the time frame, it signifies the analog time frame prior to being captured by the microphone. This is the period during which the digital signal samples are stored in the considered half-duplex system in either the active (key pressed) or non-active speech condition (key not pressed), and let $k(t)$ be the key signal that indicates when the person starts talking. We can model the input audio signal as a sum of the noise signal $n(t)$ and the speech signal $s(t)$, such that:

$$x(t) = \begin{cases} n(t), & \text{if } t < t_k \\ w(i) = n(t) + s(t), & \text{if } t \geq t_k \end{cases} \tag{5}$$

where t_k stands for the time the key is pressed and i is the sample corresponding to time t. Later we use t when referring to the analog signal and either i or k to denote the sampled value at a certain time.

To derive the noise model from the input audio, we can use the samples of $x(t)$ before pressing the key. Let T be the duration of the audio input segment that is processed before pressing the key. Then, the noise model $m(t)$ can be derived as:

$$m(t) = \begin{cases} n(t), & \text{if } 0 \leq t < T \\ 0, & \text{if } t \geq T \end{cases} \tag{6}$$

The above equation states that the noise model $m(t)$ is equal to the noise frame $n(t)$ before the key is pressed. Machine learning techniques can be utilized to identify statistical noise using multiple noise frames, as is depicted in Fig. 3. Our system is triggered by the key-press (push-to-talk) event. Noise samples are continuously sampled until the key-press event, at which time the most promising samples are retained; see later in Section IV-B for the method employed to determine the samples to be retained. The implicit assumption is that, prior to the key-press event, the user is not talking to the device. This is often the case, but speech detection might be needed to make sure that this is being observed, and if necessary, samples which contain active speech would need to be discarded. Once the key-press event occurs, the system uses machine learning to characterize the noise as best fits the noise-affected speech signal; later in Section IV-C we present our scheme employing the ANFIS model for this purpose. Then, once the noise characterization step is completed, the system applies denoising through an adaptive combination filter that is explained in more detail later in this section. Buffering based on key-press events can be useful in situations where the noise characteristics are not known *a priori* or where the noise characteristics vary over time. By buffering and analyzing recent data, this approach can adapt to changing noise conditions and provide more accurate estimates of noise properties. It is important to emphasize that the noise chunks are acquired during periods when the device is not transmitting or receiving signals.

Noise database history approach: In some cases, there is not sufficient information at the time of the key press to train the ANFIS model, for example, there is a long period of silence prior to the keypress. In such a case, we can utilise noise characterized by an optimal selection with segments of various types of noises within the offline historical stored data, followed by the construction of a dictionary of noise patterns. In our case, we have opted for this approach when the system fails to obtain an adequate number of noise samples for training. An example of the noise database block of this system is shown in Fig. 3. The dictionary approach contains recent noise sample segments from the offline stored data, which are suitable for similar noise characterization. The dictionary can then be used for noise estimation. The dictionary can be constructed by training the same ANFIS model on a recent noise sample with segments that closely match the noise pattern found in the offline stored data. This is particularly useful if there are not enough buffered noise segments, for example at the start of use, or if the samples were polluted with speech. Consequently, the system can better adjust to changing noise conditions and provide more accurate noise characterization. The addition of this mechanism does not introduce significant processing overhead but does require limited additional storage of historical noise samples.

Let us assume that the first noise frame at time instant k is denoted $\mathbf{n}(k)$ and $\mathbf{N}$ is a matrix in which each column contains samples of different types of noise. The goal is to find the column vector $\mathbf{n}_i$ in $\mathbf{N}$ that best matches the current noise frame (segment) $\mathbf{n}(k)$. One way to achieve this is by computing the Euclidean distance between $\mathbf{n}(k)$ and each column vector in $\mathbf{N}$, $\mathbf{N}_i$, and selecting the column vector with the smallest distance as the best match, which is formally

expressed as:

$$i^* = \arg\min_i \|\mathbf{n}(k) - \mathbf{N}_i\|_2 \,, \tag{7}$$

where i^* is the index of the best-matching column vector in $\mathbf{N}$, and $\|\cdot\|_2$ denotes the Euclidean norm. Once the best-matching noise segment is identified, the subsequent segments can be extracted from it, and an ML algorithm (ANFIS) can be employed for training. Our system uses the ANFIS model presented in Section IV-C to estimate the noise signal based on the historical segment and noise database.

Both the *buffering* and *noise database history* approaches described above are performed *online*, i.e., characterization and learning is carried out with minimal delay. In addition, for both approaches, auto-gain control is an important factor to maintain a consistent signal level in the presence of varying input levels, as in communication applications, the signal amplitude may vary widely over time. An auto-gain control algorithm can thus adjust the gain of the signal to maintain a consistent level, which can improve the overall quality of the processed signal [7]. As noted above using both approaches is useful to allow for situations such as after an initial start condition (the user first turns on the radio) where the *database history* approach is used before the *buffering* approach can be started.

B. Intelligent sample management

The accuracy of these methods is highly dependent on the efficiency of the noise collection process and that the stored noise samples correctly represent the actual noise. Hence, it is crucial to accurately estimate the statistical properties of noise in order to effectively separate it from the speech signal. Additionally, periodic updates of the stored noise samples is essential so that the limited storage contains noise segments that are mutually exclusive i.e., are statistically "different." One common approach to noise estimation and separation is the use of statistical models that capture the statistical properties of signal and noise. The Gamma distribution has been used in [44] and is useful as it generalizes several distributions e.g., Rayleigh, exponential, Weibull, and log-normal [45]. In particular, the work by [44] shows that the amplitude distribution of audio as a Gamma distribution is a useful discriminator. Consequently, we use the Gamma distribution parameters to distinguish the type of noise and distinguish between speech versus noise signals.

C. Proposed low-complexity solution

ANFIS is a hybrid system that combines the advantages of fuzzy logic and neural networks. The ANFIS model consists of a set of fuzzy *if-then* rules, where the input variables are mapped to the output variable through a set of membership functions and a set of adaptive parameters. The ANFIS model can be trained using a set of input-output data, and the parameters can be adjusted using the back-propagation algorithm.

An illustrative example of an ANFIS model with two inputs and two membership functions is shown in Fig. 4. The ANFIS model consists of multiple layers, including the input layer,

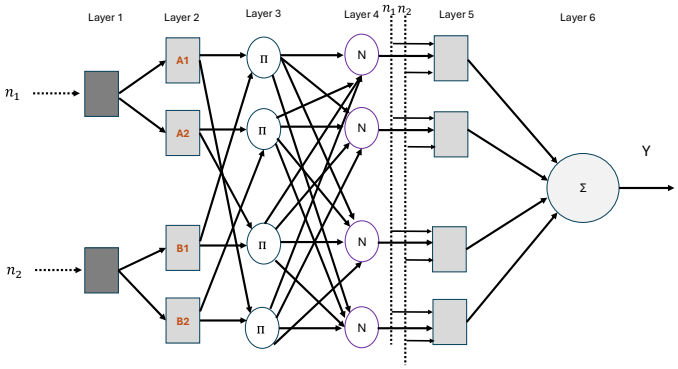

Fig. 4: Example of an ANFIS model with two inputs and membership functions.

a membership function layer, a rule layer, a normalization layer, and an output layer. During training, the model learns the optimal parameters for each layer based on the input data and the desired output. The input layer takes the noisy speech frame and two frames from the noise database as input. The input layer applies appropriate scaling and normalization to the input data to prepare them for processing by the subsequent layers. The second layer is the membership function layer that uses a clustering method (i.e., grid partitioning), which maps the input data to a set of *linguistic variables*, such as "high noise" or "low noise". This layer helps to represent the input data in a way that is more meaningful for the subsequent layers to process. There are many different types of membership function that can be used, but two commonly used functions in ANFIS are the Gaussian and the triangular membership functions. Here, we use the Gaussian membership function, as we have found by experimentation that it can capture more efficiently the statistical properties in our case. The third layer is the rule layer, which applies fuzzy logic rules to the output of the membership function layer to generate a set of inference rules. These rules capture the relationships between the linguistic variables and the desired output (i.e., the estimated noise in the signal). The fourth layer is the normalization layer, which simply scales the output of the rule layer to ensure that they sum to one. The final layer is the output layer, which generates the estimated noise based on the input data and the learned parameters of the model. The model learns to recognize patterns in the data and uses them to estimate the noise in future speech signals.

The input variables of the ANFIS model are the first noisy speech frame and two chunks of the best-matching historical signals. The output variable is the estimate of the statistical noisy frame at the time instant k. As we mentioned, the ANFIS model can be represented by a set of fuzzy *if-then* rules, such that: if x_1 is A_1 and x_2 is A_2 and $\ldots$ and x_N is A_N, then $y = f(x_1, x_2, \ldots, x_N)$, where $x_1, x_2, ..., x_N$ are the first noisy speech frame and a few chunks from the best-matching column vector, $A_1, A_2, ..., A_N$ are the fuzzy sets defined over the input variables, and $f(x_1, x_2, ..., x_N)$ is a nonlinear function that maps the input variables to the output variable $\mathbf{n}$.

The parameters of the ANFIS model can be adjusted using the back-propagation algorithm together using the gradient

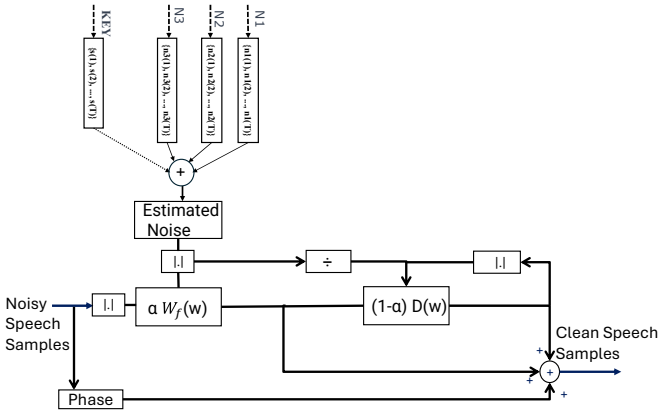

Fig. 5: Block diagram for speech enhancement based on the considered ANFIS model.

descent algorithm, which iteratively updates the parameters of the membership functions and the adaptive parameters [46]. The goal is to minimize the mean squared error between the estimated noise signal and the true noise signal. The block diagram of the speech enhancement process is shown in Fig. 5. The power spectral density (PSD) of the output noise signal generated by ANFIS is the estimate of the noise power in the noisy speech signal, which is then used to calculate the Wiener filter coefficients. The combination of the Wiener filter and spectral subtraction estimates the clean speech signal by using a transfer function that minimizes the mean square error between the noisy signal and the estimated clean signal. This is expressed as

$$\hat{\mathbf{s}}(k,\omega) = \alpha \mathbf{W}_f(\omega)\mathbf{w}(k,\omega) + (1-\alpha)\mathbf{D}(\omega)\mathbf{X}(k,\omega) \quad (8)$$

where $\hat{\mathbf{s}}(k,\omega)$ is the estimated clean speech signal at time instant k and frequency ω, $\mathbf{W_f}(\omega)$ is the frequency response of the Wiener filter, $\mathbf{D}(\omega)$ represents the frequency-dependent attenuation applied in the spectral subtraction path, as it determines the proportion of the noisy signal to be subtracted at each frequency, $\mathbf{w}(k,\omega)$ is the noisy speech signal at time instant k and frequency ω, same as $\mathbf{X}(k,\omega)$ but they are used in different filtering paths (Wiener and spectral subtraction, respectively), and α is a weighting function that controls the contribution of each filter at different frequencies. We have determined by experimentation that the optimal value of α is 0.9. The Wiener filter and spectral subtraction filter are coupled together by the weighting function to achieve better noise suppression performance. The filters are defined as:

$$\mathbf{W}_f(\omega) = \frac{\mathbf{P}_{ww}(\omega)}{\mathbf{P}_{ww}(\omega) + \mathbf{P}_{nn}(\omega)} \quad (9)$$

where $\mathbf{P}_{ww}(\omega)$ is the PSD of the noisy speech signal and $\mathbf{P}_{nn}(\omega)$ is the PSD of the noise signal.

$$\mathbf{D}(\omega) = \frac{\mathbf{P}_{nn}(\omega)}{\mathbf{P}_{ww}(\omega) + \mathbf{P}_{nn}(\omega)} \quad (10)$$

V. Experimental results

A. Experimental Configuration

To evaluate the performance of our proposed solution, we used a half-duplex hardware AMBE2+ Vocoder [42], as it is widely used in Digital-PMR. This device is designed to process audio signals in real-time. It can also introduce signal distortion and channel impairments, such as packet loss or delay, to simulate real-life communication scenarios. The data rate of the Vocoder is 2.4 *kbits/sec*. It applies a forward error correction code (FEC) that results in a total bitrate of 3.6 *kbits/sec*. Our proposed approach provides a background noise suppression technique, which is used as a front end to the Vocoder. In our experiments, each signal was initially sampled at a rate of 8 kHz before the AMBE2+ encode/decode process which uses a native 20 msec frame size mandated by Digital-PMR [47].

In our study we have used 72,000 blocks of speech to test the system. Each block has a durationn of 20 msec. These blocks were obtained from recordings of 10 male and 10 female speakers. Moreover, we incorporated six distinct types of noise conditions: *busy street*, *ambulance*, *exhibition hall*, *busy airport*, *construction*, and *busy station*. Consequently, we combined these clean speech blocks with varying levels of noise to mimic real-world scenarios, encompassing a range of signal-to-noise ratio (SNR) levels spanning from -5 dB to 15 dB. Therefore, in total we have used 720,000 blocks of noisy speech to test our methodology. Given the nuanced nature of speech compared to images and video, our dataset comprised 720,000 blocks encompassing both speech and noise, ensuring comprehensive evaluation of our system's performance. For DNN training, the clean speech signals, we used the widely used TIMIT Acoustic-Phonetic Continuous Speech Corpus [48]. As we have previously discussed, in general, considering the first frame as a representative noise sample works well for some cases. However, in our study, we take into account the worst-case scenario where this is not the case. In doing this, we are improving upon the general approach. By introducing these different types of noise, we aimed to evaluate the effectiveness of our proposed denoising methodology in improving the quality of speech across different types of noise and SNR levels.

B. Evaluation metrics

To evaluate the performance of the proposed approach in reducing the noise level in speech signals while maintaining the quality of the speech, the following metrics were used:

- *Segmental SNR (SSNR)*: This is a metric used to evaluate the quality of speech signals by comparing the quantization noise to the energy in each underlying speech segment. SSNR is calculated by dividing the energy in a speech segment by the energy of the quantization noise in the same segment. A higher SSNR value indicates better speech quality [49].
- *Perceptual Evaluation of Speech Quality (PESQ)*: This is a widely used objective metric that estimates the perceived speech quality of denoised signals. PESQ analyzes the processed speech signal and compares it with the original speech signal to estimate perceived quality. PESQ is based on the human auditory perception model and provides a score on a scale from -0.5 to 4.5, where higher values indicate better speech quality [50].

TABLE I: Mean PESQ and Cbak scores for clean speech and after being encoded by the considered Vocoder.

Speech Condition	μ PESQ score	μ Cbak score
Clean Speech	4.5	5
USB Vocoder	2.67	2.97

- *Composite Measure of Background Noise Distortion (Cbak)*: This is a metric used to evaluate the performance of speech enhancement systems by measuring the predicted rating of background noise distortion in the processed speech signal. Cbak is based on human perception and evaluates the degree of distortion caused by the noise in the background of the speech signal. A higher Cbak value indicates better speech quality and intelligibility in the processed speech signal [49].

Cbak and SSNR are commonly used objective measures of speech quality, while PESQ is a subjective measure that takes into account human perception of speech quality.

We start our investigation by analyzing the clean speech signal after the Vocoder in order to understand the impact of the codec on speech quality. Table I presents the results, indicating that the PESQ and Cbak scores for clean speech decrease from 4.5 to 2.67 and 5 to 2.97, respectively. This suggests that the codec significantly degrades speech quality, making it more distorted and difficult to understand in noisy conditions. It further emphasizes the importance of using noise suppression techniques to improve speech quality. However, it is worth noting that although the codec does not have specific noise suppression, encoding a noisy speech by a Vocoder suppresses, to an extent, the non-speech noise. Therefore, one of the objectives of our evaluation is to determine the extent our active noise suppression technique improves the quality of speech beyond the inherent noise suppression ability of the codec.

C. Neuro-Fuzzy parameters tuning

Recall that, the ANFIS model is used to estimate the noise present in speech signals. The parameterization of the ANFIS model is important as it affects the quality of the noise characterization. Therefore, in order to determine the best parameters for our ANFIS model, we have extensively tested it. The model consists of 21 nodes that represent the input and output variables of the system. The input variables include the first chunk of the signal after pressing the key and a history of three chunks of noise signal, while the output variable is the estimated noise. The model has a total of 24 parameters, 12 of which are linear and 12 nonlinear. During the training process, the ANFIS model adjusts its parameters to minimize the error between the estimated noise level and the actual noise. The model did not have any checking data pairs, as it was not necessary to assess its performance on a distinct dataset in this situation. This is because it is designed for online training, where data pairs might not be accessible. The ANFIS model used in the paper has eight fuzzy rules, which are derived from the input-output data pairs. These fuzzy rules are used to model the relationships between the input variables and the output variable. The rules can be expressed

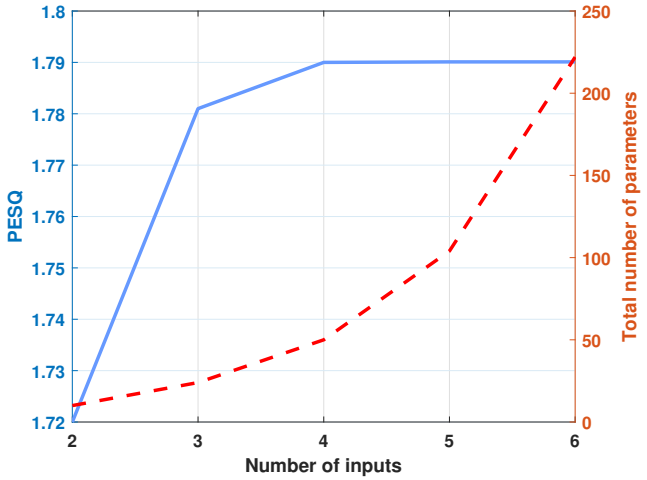

Fig. 6: Relationship between the number of inputs of ANFIS, PESQ Score, and the number of parameters of the ANFIS model.

in equivalent linguistic terms, such as "*if the speech signal is high and the noise signal is low, then the estimated noise level is low*". The ANFIS model was trained with a two-step process, involving a forward pass using fuzzy inference rules and a backward pass employing the least squares method to minimize error. We found that 10 epochs are sufficient. However, an epoch number between 3 to 10 can be suitable for different noise scenarios. In addition to tuning the best parameters for the ANFIS model, we have also examined the impact of the number of inputs to the performance. We have evaluated the performance of the ANFIS model by examining the PESQ values after denoising the speech, as more accurate noise estimation results in better PESQ scores. Our results show that using two inputs achieves an average PESQ score of 1.72 for 5 dB noise scenario. However, using three and four inputs the performance of the noise estimation is improved and the average PESQ score increases to 1.78 for three inputs and 1.79 for four inputs. We notice that beyond four inputs, the performance gains are rather marginal. We should highlight that a higher number of inputs is associated with an increase in the number of parameters in the ANFIS model, making it more computationally expensive. The tradeoff between quality performance, in terms of PESQ score, and complexity (number of parameters of ANFIS) is shown in Fig. 6.

D. Buffering based upon key-press events

As we have already mentioned, the proposed buffering-based method utilizes three consecutive 20 msec frames as input for noisy frames. To select the three frames, the system samples 60 msec segments for speech and noise frame analysis. For each 20 msec frame, the envelope is calculated using the Hilbert transform, and its amplitude distribution is fitted to a Gamma distribution as shown in Fig. 8. After computing the CDF of the Gamma distribution, a threshold value is calculated on its basis. If the amplitude of the envelope is below the threshold value, it is considered as speech; differently, it is considered as noise, and the sample is stored as a noise sample. This process is repeated for three noisy frames, and if the noise

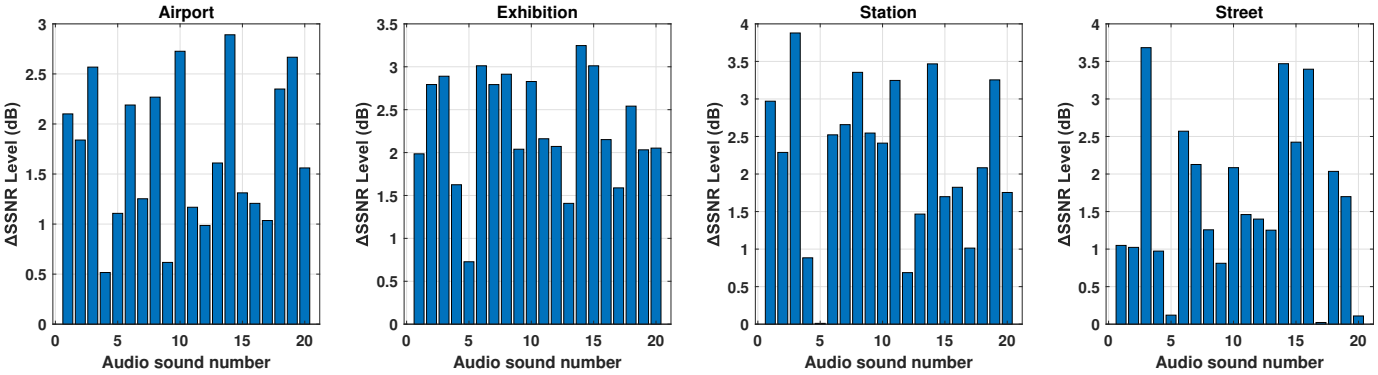

Fig. 7: SSNR level improvement for all audio sounds and different types of noise sources.

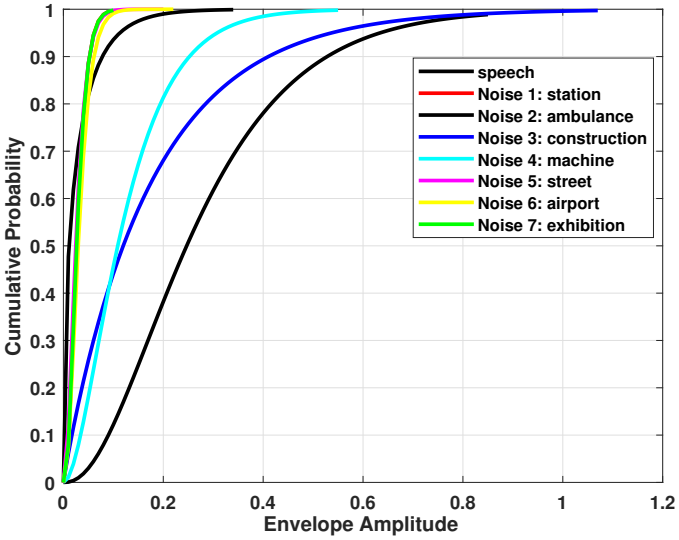

Fig. 8: Different types of noise sources fitted using Gamma cumulative distribution function.

samples meet certain criteria, they are used to update the noise model. If the criteria are not met, the process starts again with the next set of three noisy frames. In our work, we used the amplitude of the audio as a criterion to discriminate between the noise samples to be chosen, i.e., the noise samples with the largest deviation from typical speech were chosen. After the user presses the key and starts transmission, the system takes the first 20 msec chunk of audio and trains it with the previously stored noise samples to obtain the most reliable noise statistics using ANFIS training. Every two sec, the system updates the noise statistics by selecting new noise chunks based on the CDF, and the ANFIS algorithm is able to generate noise statistics based on this new data. In order to evaluate the performance of the proposed method, the objective evaluation used SSNR while (emulated)- subjective evaluation was performed using the speech quality metric, PESQ and Cbak. First, we evaluate the example noisy speech by analyzing its spectrogram. The results show that our adaptive approach outperforms in handling non-stationary noise sources. It effectively reduces background noise, enabling the prominent frequency components of speech to become more discernible. Additionally, we assess the SSNR improvement achieved by our adaptive method compared to the noisy speech. The results are presented in Fig. 7. They demonstrate that our adaptive method consistently achieves an average SSNR gain of 2 dB in different noisy scenarios. This highlights the strong noise suppression capability of our adaptive method.

E. Comparison of the proposed approach with existing techniques

Next, we evaluate our approach using emulated subjective scores and compare it with other conventional methods, namely combination Wiener filter [24], MTSS [3], and LMMSE [6]. The results are shown in Fig. 9 for all comparison methods. From the results, we can observe that the proposed adaptive method consistently shows superior performance compared to its counterparts across various noisy scenarios, including exhibition, street, station, airport, construction, and ambulance noisy situations, and for different SNR values ranging from -5 dB to 15 dB. PESQ and Cbak comparisons indicate significant gains when using the proposed adaptive method, with an average PESQ score gain of 0.2 to 0.4 in all noisy situations. These gains are attributed to our superior noise estimation technique. This is particularly evident when we compare with the combination filter using a Wiener filter, where the key difference lies in our enhanced noise estimation process. Furthermore, the proposed adaptive method exhibits consistent improvements compared to the MTSS method. However, we note that, in some specific scenarios, such as the ambulance scenario, our proposed method and the LMMSE method show similar gains in terms of PESQ which is attributed to the specific noise characteristics and signal conditions. Nonetheless, when considering the Cbak level (see Fig. 10), our adaptive method outperforms the LMMSE method for the same noisy scenario.

We continue the evaluation by examining the quality of the output of our proposed enhancement method when the denoised speech is encoded and then decoded using the USB Vocoder. We compare with the case the noisy speech is not denoised but only processed (i.e., encoded and decoded) by the USB Vocoder. The aim of this comparison is to examine potential gains with respect to native suppression of the codec. Fig. 11 presents the results, which demonstrate that the enhanced speech by the proposed method consistently achieves PESQ gains ranging from 0.1 to 0.2 compared to the native

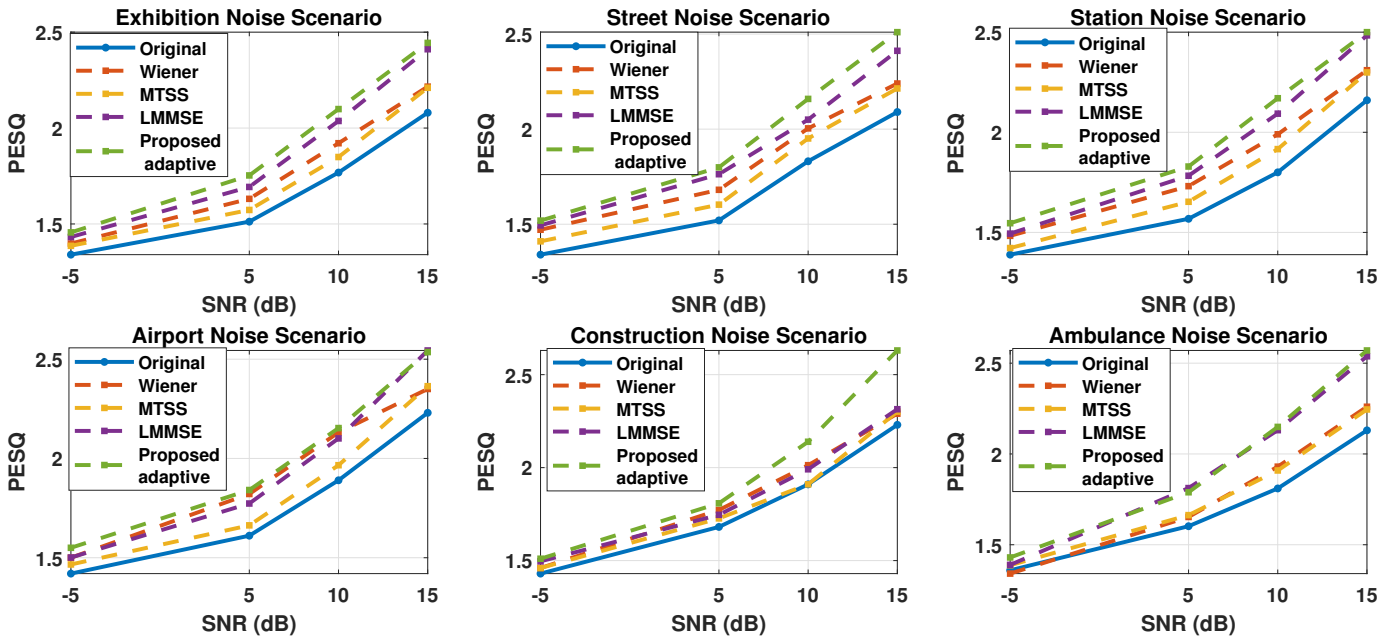

Fig. 9: PESQ comparison of the proposed adaptive method with the schemes under comparison for various SNRs and different noise sources.

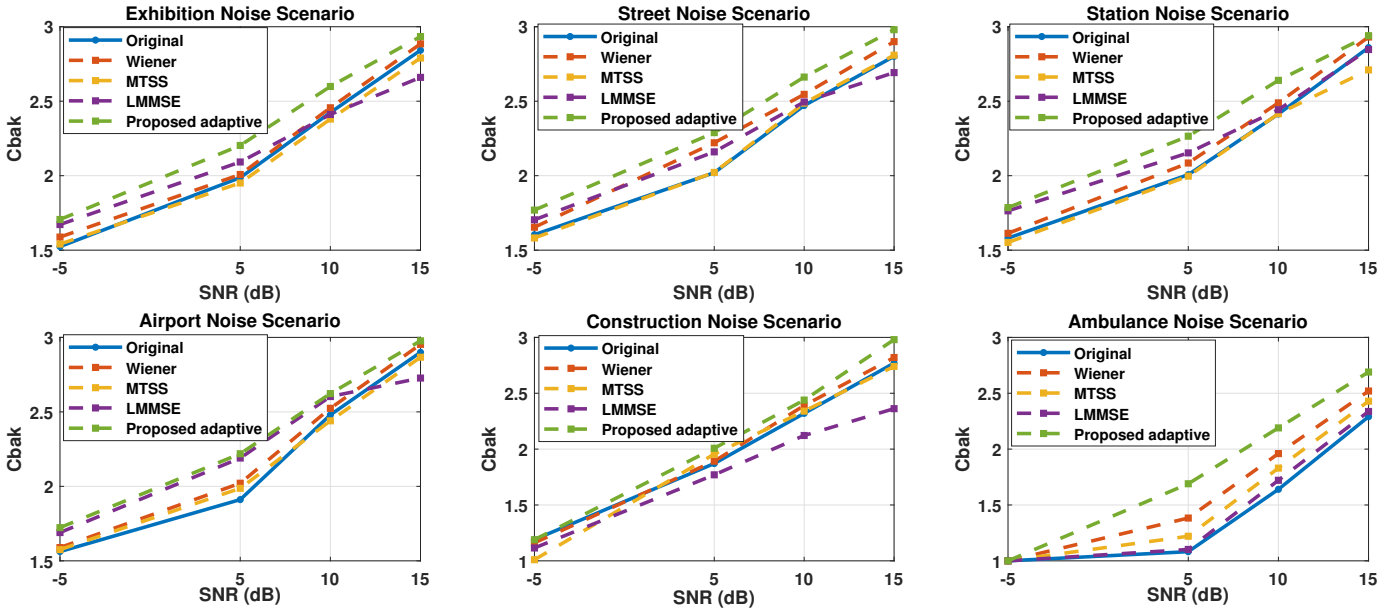

Fig. 10: Cbak comparison of the proposed adaptive method with the schemes under comparison for various SNRs and different noise sources.

suppression of the codec. Furthermore, our results align with a recent CNN-based enhancement study [33], although that study focused on a different codec. Similar improvements are observed with respect to Cbak as presented in Table II, further supporting the efficacy of our proposed method.

F. Noise database history approach

To assess the effectiveness of utilizing various types of noisy historical sound databases, we conducted an evaluation using a noise sound database. For example, when the system captures a noise sample just before a key-press event, followed by the user pressing the key for transmission, it utilizes this noise sample segment to search for the closest match within the database. Upon finding the closest match in the database, it proceeds to extract the next two segments for training purposes. Then, the same ANFIS model and parameters are employed to train both the initial noisy frame and the best-matching frames from the database. Subsequently, the estimated noise is utilized in the combination filter to enhance the speech. The results for SNR equal to 5 dB are presented in Table III, where the noise database approach is compared with the native noise suppression achieved by only using the codec. The noise database history approach exhibits comparable performance to the buffering-based method when the noise is close in nature to the noise stored in the database. We have considered a worst-case scenario, such as a noisy street

TABLE II: Cbak comparison of the proposed adaptive method with Vocoder's native noise suppression for various SNRs and different noise sources. The Cbak scores are measured in the receiver end.

	SNR = -5dB		SNR = 5dB		SNR = 10dB		SNR = 15dB	
	Native	**Adaptive**	**Native**	**Adaptive**	**Native**	**Adaptive**	**Native**	**Adaptive**
Exhibition	1.33	**1.45**	1.68	**1.77**	1.97	**1.99**	2.11	**2.13**
Station Noise	1.30	**1.48**	1.64	**1.82**	1.98	**2.06**	2.14	**2.16**
Street Noise	1.29	**1.52**	1.59	**1.91**	1.88	**2.08**	2.04	**2.23**
Airport Noise	1.33	**1.36**	1.65	**1.73**	1.94	**1.97**	2.15	**2.16**
Ambulance Noise	1	**1**	1.53	**1.69**	1.77	**1.90**	2.03	**2.09**
Construction Noise	1.10	**1.17**	1.65	**1.72**	1.82	**1.94**	2.03	**2.09**
Bird Noise	1.27	**1.36**	1.58	**1.64**	1.81	**1.91**	2.09	**2.25**
Siren Noise	1	**1**	1.56	**1.67**	1.79	**1.91**	2.01	**2.10**
Const and Street Noise	1.26	**1.36**	1.61	**1.75**	1.79	**1.87**	1.98	**2.08**

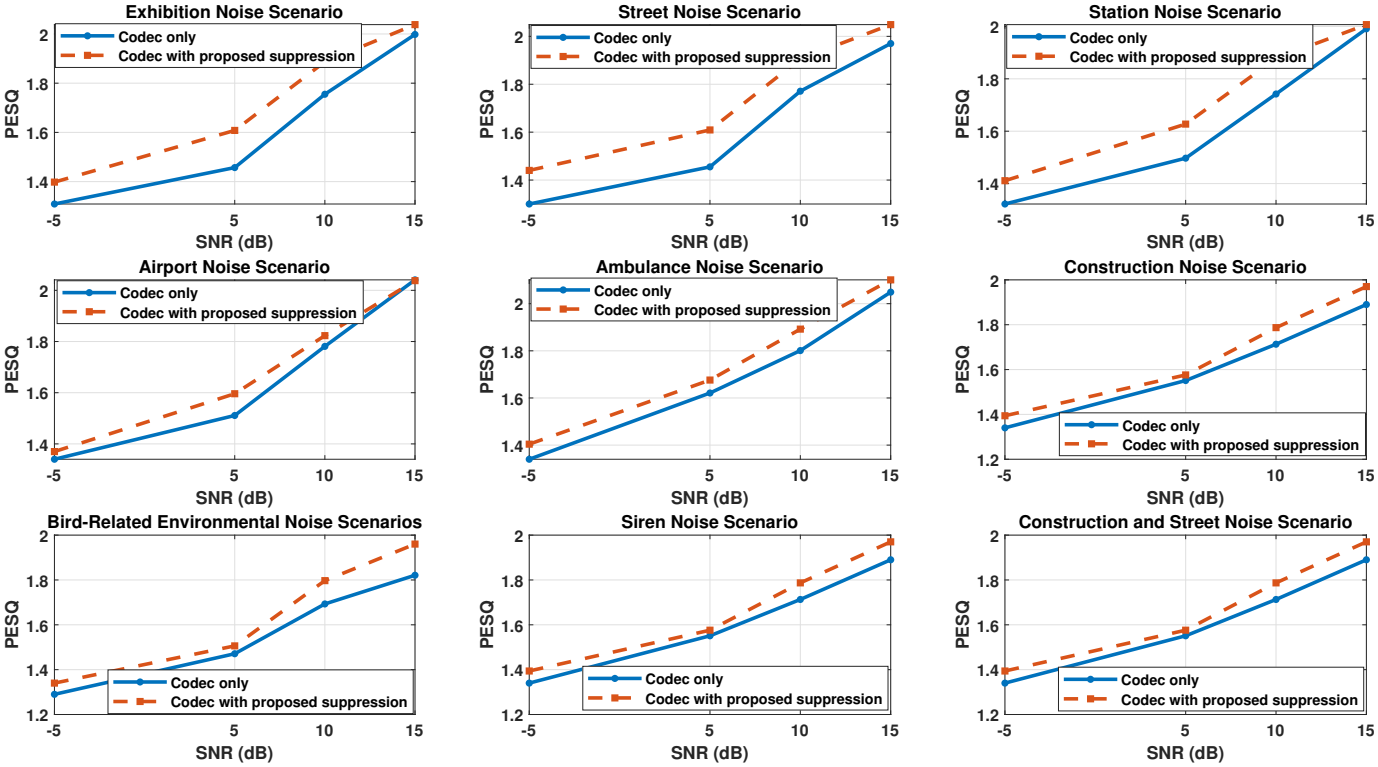

Fig. 11: PESQ comparison of the proposed adaptive method with Vocoder's native noise suppression for various SNRs and different noise sources. The PESQ scores are measured in the receiver end.

TABLE III: Comparison between the *native* suppression of the codec and our *adaptive* noise database history approach for three different noisy scenarios (5dB) in terms of PESQ and Cbak values.

Noisy Scenario (5dB)	Method	PESQ	Cbak
Street Sound	Native	1.44	1.59
	Adaptive	**1.52**	**1.89**
Ambulance Sound	Native	1.61	1.53
	Adaptive	**1.69**	**1.81**
Exhibition Sound	Native	1.45	1.68
	Adaptive	**1.53**	**1.77**

environment in the real world. However, the database lacks the pedestrian sound component in the stored street noise, which is present in the real scenario. Although the training process may adapt to these noise characteristics, it could potentially lead to a performance decrease when compared to buffering based on key-press events in real noisy situations. If the noise characteristics in the database match those of the real noisy environment, the improvement would be similar. Hence, it becomes evident that in scenarios where the system fails to retain recent noise frames (for instance, when a user switches off the radio device and later turns it on to engage in a conversation), the historical noise database approach can be employed effectively to mitigate noise and improve sound quality.

G. Deep neural network approach

For the sake of understanding the impact of the employed machine learning framework, we designed a variant of our

proposed scheme where we replace our model with a 1-dimensional CNN (1DCNN). The 1DCNN model comprises two 1-dimensional convolutional layers (Conv1D) with a filter width of 20 and 50 filters each. These layers apply convolution operations to the input data, resulting in a fixed output shape. ReLU activation layers follow each Conv1D layer to introduce non-linearity and preserve the output shape. To prevent overfitting and improve generalization, dropout layers are inserted after each ReLU activation. These dropout layers randomly set a fraction of input units to 0 during training, making the model more robust. After the dropout layers, 1-dimensional max pooling layers (MaxPooling1D) are utilized with a pool size of 5 and strides of 2, enabling downsampling and dimensionality reduction. A flatten layer is introduced to convert the 3D tensor output into a 1D vector, preparing the data for the fully connected layers. The flattened representation is then passed through two dense layers. The first dense layer consists of 1006 nodes and employs the ReLU activation function, enabling the model to learn complex patterns and representations. The second dense layer comprises 513 nodes and also utilizes the ReLU activation function. This layer serves as the final output layer of the model. We use the same set of speech samples as described in Section V-A. The proposed 1DCNN model undergoes training on 80% of the dataset and is subsequently tested on the remaining 20% of unseen samples. We use MSE between the enhanced speech predicted by the model and the corresponding clean speech as the loss function. MSE is computed in the STFT domain. Specifically, it is calculated between the spectral magnitudes of the enhanced speech predicted by the model and the corresponding clean speech. This choice of domain for MSE computation allows for a direct comparison between the predicted and clean speech signals in the frequency domain, providing a measure of the error in the spectral magnitudes.

We compare the 1DCNN approach with the proposed method using ANFIS. The results are presented in Table IV. From this table, we can observe that the 1DCNN performs better than our proposed method for the 5 dB ambulance noisy scenario (ambulance noise is distinct from babble noise and challenging to suppress due to its unique characteristics). Specifically, the 1DCNN achieves 0.09 PESQ score and 0.2 Cbak score improvement compared to our proposed adaptive method. We also compare the performance of both schemes for speech samples with Chinese speakers affected by street noise (V), where the noise was included during the training process. From this comparison, it is evident that our proposed adaptive method significantly outperforms the 1DCNN approach in terms of both PESQ and Cbak. These findings serve as evidence that although DNN (offline training) can effectively improve speech quality in a few situations, it may not be as effective in real-time systems where the noise conditions and speech characteristics vary across different situations. This is because DNNs rely on sets of clean and noisy speech during their training phase. Consequently, in real-life environments where noise is unpredictable and varies significantly, the 1DCNN approach might not effectively reduce noise and enhance speech quality. The model's reliance on specific training data could limit its generalization capabilities in handling various real-world noise scenarios, making it less suitable for real-time applications where robust noise reduction is crucial.

TABLE IV: Achieved PESQ and Cbak score by the normal (native suppression of the codec), adaptive approach, and 1DCNN approaches for the Ambulance noisy scenario with English speaker when SNR is 5 dB.

Method	**PESQ**	**Cbak**
Native	1.61	1.53
Adaptive	**1.68**	**1.69**
1DCNN	1.76	1.89

TABLE V: Achieved PESQ and Cbak score by the normal (native suppression of the codec), adaptive approach, and 1DCNN approaches for the Street noisy scenario with Chinese speaker when SNR is 5 dB.

Method	**PESQ**	**Cbak**
Native	1.41	1.39
Adaptive	**1.53**	**1.47**
1DCNN	1.26	1.19

H. Comparing computational complexity of Deep neural network approach and our adaptive approach

We also assess the computational efficiency of the 1DCNN model and our proposed ANFIS-based model, considering its inference time and resource requirements. Our proposed ANFIS-based method demonstrates a significant improvement over 1DCNN in terms of parameter efficiency. While 1DCNN requires a massive 6,906,497 parameters, our approach only requires 50 parameters. Additionally, the 1DCNN approach heavily relies on offline training, which typically consumes around two hours of processing time using a GPU for training data that spans several hours. In contrast, our proposed ANFIS-based method enables real-time training, eliminating the need for offline training. Moreover, DNNs necessitate separate offline training for different speech and noise variations, such as Chinese speech. On the contrary, our method does not require separate training for different languages or noise types.

In terms of run-time performance, i.e., after all training is carried out, we have calculated the multiply-accumulate operations (MACs) for both approaches for a 20 ms block of audio. For the 1DCNN-based denoising system, the total MACs are 15,384,358 while for the proposed ANFIS-based system it is only 22,720 MACs. Our method outperforms 1DCNN with key advantages: efficient parameters, real-time training, language/noise independence, and enhanced low-power device performance.

I. Deployment efficiency

We have successfully implemented our algorithm in an open-source 32-bit microprocessor architecture, incorporating optional custom instructions in a realistic test system. During our assessment, we made certain assumptions regarding power consumption that follow typical use cases e.g. 10 hour device use with typical talk/receive duty cycles. Our evaluations were based on a battery capacity of 2000 mAh and a voltage of 7.2 V, providing a battery power capacity of 14400 mWh. In the deployment phase, we evaluated the power consumption

of various system components. We found the RF accounted for 70% of the battery power, audio processing as a whole consumed 4% and remainder for general device operation such as small user interface. The denoise functionality contributed minimally to power consumption, accounting for only 0.012% of the battery's capacity. ANFIS execution consumed only 0.011% of battery capacity, matching real-time sound performance, consistent with results in Table II. The de-noising system introduces an additional 24ms of latency to the system which is relatively small, even for a real-time speech system. It should be noted that the ANFIS model update does not introduce any additional latency. Thus, confirming that the contribution is extremely lightweight and reasonable for real-time system.

VI. Conclusion

Our study introduces a novel denoising methodology that utilizes ANFIS for noise segments collected historically before key-press events as well as a noise database. This approach yields substantial improvements in speech quality across varying noise conditions and signal-to-noise ratio levels. Notably, our method possesses a distinctive capability to estimate noise through training on recent noise segments and seamlessly integrating this knowledge into the combination filter, setting it apart as an adaptive, low-latency solution. The system continually updates noise segments over time, ensuring its adaptability. Subsequently, it undergoes training using these updated noise segments upon pressing the push-to-talk key, followed by the combination filter. The evaluation shows that our method is highly efficient and adaptive, as the system continuously updates noise segments and retrains on them, in contrast to conventional denoising techniques and deep convolutional networks. Our method offers a practical solution for speech enhancement, especially in settings with limited resources. Furthermore, our approach outperforms the native noise suppression ability of the Vocoder, showcasing its effectiveness in real-life communication scenarios. The successful deployment on a low-power microprocessor further validates its real-life applicability, emphasizing its potential to enhance speech quality while conserving energy. Future work aims to refine our method to show greater improvement across all noise types and expand its application to wider systems such as full duplex by utilising, recently, improved voice activity detection.

References

[1] R. Miyazaki, H. Saruwatari, T. Inoue, Y. Takahashi, K. Shikano, and K. Kondo, "Musical-noise-free speech enhancement based on optimized iterative spectral subtraction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 7, pp. 2080–2094, 2012.

[2] P. D. Paikrao, A. Mukherjee, U. Ghosh, P. Goswami, M. Novak, D. K. Jain, M. S. Al-Numay, and P. Narwade, "Data driven neural speech enhancement for smart healthcare in consumer electronics applications," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 2, pp. 4828–4838, 2024.

[3] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, Jul 2001.

[4] R. Yao, Z. Zeng, and P. Zhu, "A priori snr estimation and noise estimation for speech enhancement," *EURASIP journal on advances in signal processing*, vol. 2016, no. 1, p. 101, 2016.

[5] D. Bismor and K. Bartnicki, "Real-time advanced speech enhancement using low-power, fixed-point hardware," in *Proc. of the IEEE Int. Conf. on Signal Processing: Algorithms, Architectures, Arrangements, and Applications'2016*, Sep. 2016.

[6] F. Ykhlef and H. Ykhlef, "A smoothed minimum mean-square error log-spectral amplitude estimator for speech enhancement," in *Proc. of the IEEE Int. Conf. on Multimedia Computing and Systems, ICMCS'14*, Apr 2014.

[7] P. N. Petkov and Y. Stylianou, "Adaptive gain control for enhanced speech intelligibility under reverberation," *IEEE signal processing letters*, vol. 23, no. 10, pp. 1434–1438, 2016.

[8] P. Meyer, S. Elshamy, and T. Fingscheidt, "A multichannel kalman-based wiener filter approach for speaker interference reduction in meetings," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing, ICASSP'20*, May 2020.

[9] N. Saleem and M. I. Khattak, "A review of supervised learning algorithms for single channel speech enhancement," *International Journal of Speech Technology*, vol. 22, no. 4, pp. 1051–1075, Oct 2019.

[10] R. Jaiswal, "Automatic noise class detection for improving speech quality using artificial neural networks," in *Proc. of the 7th IEEE Int. Conf. on Communication and Electronics Systems, ICCES'22*, Jun 2022.

[11] Y. Hu, X. Zhang, X. Zou, M. Sun, Y. Zheng, and G. Min, "Semi-supervised speech enhancement combining nonnegative matrix factorization and robust principal component analysis," *IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E100.A, no. 8, pp. 1714–1719, 2017.

[12] M. N. M. Salleh, N. Talpur, and K. Hussain, "Adaptive neuro-fuzzy inference system: Overview, strengths, limitations, and solutions," in *International conference on data mining and big data*. Springer, 2017, pp. 527–535.

[13] G. Karimi, S. B. Sedaghat, and R. Banitalebi, "Designing and modeling of ultra low voltage and ultra low power LNA using ANN and ANFIS for bluetooth applications," *Neurocomputing*, vol. 120, pp. 504–508, Nov 2013.

[14] J. Chakraborty, J. C. Varma, and M. Erman, "Anfis based opportunistic power control for cognitive radio in spectrum sharing," in *2013 International Conference on Electrical Information and Communication Technology (EICT)*. IEEE, 2014, pp. 1–6.

[15] R. Martinek, M. Kelnar, J. Vanus, P. Koudelka, P. Bilik, J. Koziorek, and J. Zidek, "Adaptive noise suppression in voice communication using a neuro-fuzzy inference system," in *2015 38th International conference on telecommunications and signal processing (TSP)*. IEEE, 2015, pp. 382–386.

[16] J. Chakraborty, M. Reed, N. Thomos, G. Pratt, and N. Wilson, "Machine learning based noise suppression in narrow-band speech communication systems," in *Proc. of the 24th IEEE Int. Conf. on International Conference on Digital Signal Processing'23*, Jun 2023.

[17] S. Yadav, P. A. D. Legaspi, M. S. O. Alink, A. B. J. Kokkeler, and B. Nauta, "Hardware implementations for voice activity detection: Trends, challenges and outlook," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 70, no. 3, pp. 1083–1096, 2023.

[18] S. Soni, R. N. Yadav, and L. Gupta, "State-of-the-art analysis of deep learning-based monaural speech source separation techniques," *IEEE Access*, vol. 11, pp. 4242–4269, 2023.

[19] C. Li, T. Jiang, and S. Wu, "Single-channel speech enhancement based on improved frame-iterative spectral subtraction in the modulation domain," *China Communications*, vol. 18, no. 9, pp. 100–115, Sep 2021.

[20] P. G., I. J., and K. N., "Tamil speech enhancement using non-linear spectral subtraction," in *Proc. of the IEEE Int. Conf. on Communication and Signal Processing, ICCSP'14*, Apr 2014.

[21] N. Upadhyay and A. Karmakar, "An improved multi-band spectral subtraction algorithm for enhancing speech in various noise environments," *Procedia Engineering*, vol. 64, pp. 312–321, 2013.

[22] W. Nabi, N. Aloui, and A. Cherif, "A dual-channel speech enhancement method for cellular communication," in *Proc. of the 4th IEEE Int. Conf. on Advanced Technologies for Signal and Image Processing, ATSIP'18*, Mar 2018.

[23] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, Dec 1984.

[24] R. K. Jaiswal, S. R. Yeduri, and L. R. Cenkeramaddi, "Single-channel speech enhancement using implicit wiener filter for high-quality speech communication," *International Journal of Speech Technology*, vol. 25, no. 3, Aug 2022.

[25] M. Hoffman, Z. Li, and D. Khataniar, "GSC-based spatial voice activity detection for enhanced speech coding in the presence of competing

speech," *IEEE Trans. on Speech and Audio Processing*, vol. 9, no. 2, pp. 175–178, Dec 2001.

[26] K. Kwon, J. W. Shin, and N. S. Kim, "NMF-based speech enhancement using bases update," *IEEE Signal Processing Letters*, vol. 22, no. 4, pp. 450–454, Apr 2015.

[27] S. Park, E. Serpedin, and K. Qaraqe, "Gaussian assumption: The least favorable but the most useful [lecture notes]," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 183–186, May 2013.

[28] D.-M. Zhang, C.-C. Bao, and F. Deng, "Integrating codebook and wiener filtering for speech enhancement," in *Proc. of the IEEE Int. Conf. on Signal Processing, Communications and Computing, ICSPCC'15*, Sep 2015.

[29] C. D. Sigg, T. Dikk, and J. M. Buhmann, "Speech enhancement using generative dictionary learning," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1698–1712, Aug 2012.

[30] S. U. N. Wood and J. Rouat, "Unsupervised low latency speech enhancement with RT-GCC-NMF," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 332–346, May 2019.

[31] C.-T. Lin, "Single-channel speech enhancement in variable noise-level environment," *IEEE Trans. on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 33, no. 1, pp. 137–144, Jan 2003.

[32] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, Jan 2014.

[33] Z. Zhao, H. Liu, and T. Fingscheidt, "Convolutional neural networks to enhance coded speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 4, pp. 663–678, 2019.

[34] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement," in *Proc. of Interspeech 18*, Sep 2018.

[35] Y. Hu, Q. Yang, W. Wei, L. Lin, L. He, Z. Ou, and W. Yang, "Mn-net: Speech enhancement network via modeling the noise," *IEEE Transactions on Audio, Speech and Language Processing*, pp. 1–13, 2025.

[36] V. N. Mitnala, M. J. Reed, J. Bicknell, and J. Chakraborty, "Intelligent speech handover for smart speakers through deep learning: a custom loss function approach," *IEEE Transactions on Consumer Electronics*, pp. 1–1, 2025.

[37] H. Wu, X. Chen, Y.-C. Lin, K. Chang, J. Du, K.-H. Lu, A. H. Liu, H.-L. Chung, Y.-K. Wu, D. Yang, S. Liu, Y.-C. Wu, X. Tan, J. Glass, S. Watanabe, and H.-Y. Lee, "Codec-superb @ slt 2024: A lightweight benchmark for neural audio codec models," in *2024 IEEE Spoken Language Technology Workshop (SLT)*, 2024, pp. 570–577.

[38] L. N. Huynh, Y. Lee, and R. K. Balan, "DeepMon:Mobile GPU-based Deep Learning Framework for Continuous Vision Applications ," in *Proc. of the 15th ACM Int. Annual Int. Conf. on Mobile Systems, Applications, and Services, MobiSys'17*, Jun 2017.

[39] Y. Geng, Y. Yang, and G. Cao, "Energy-efficient computation offloading for multicore-based mobile devices," in *Proc. of the IEEE Int. Conf. on Computer Communications, INFOCOM'18*, Apr 2018.

[40] M. Xu, M. Zhu, Y. Liu, F. X. Lin, and X. Liu, "DeepCache: Principled cache for mobile deep vision," in *Proc. of 24th ACM Int. Annual Int. Conf. on Mobile Computing and Networking, MobiCom'18*, Oct 2018.

[41] Y.-C. Lin, C. Yu, Y.-T. Hsu, S.-W. Fu, Y. Tsao, and T.-W. Kuo, "Seofp-net:compression and acceleration of deep neural networks for speech enhancement using sign-exponent-only floating-points," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1016–1031, 2022.

[42] Digital Voice Systems, Inc. (DVSI), "AMBE+2 vocoder: Next generation voice compression," Whitepaper, January 2019. [Online]. Available: https://www.dvsinc.com/wp-content/uploads/2019/01/AMBE2-Whitepaper.pdf

[43] R. Çolak and R. Akdeniz, "A novel voice activity detection for multi-channel noise reduction," *IEEE Access*, vol. 9, pp. 91 017–91 026, 2021.

[44] J. W. Shin, J.-H. Chang, S. Barbara, H. S. Yun, and N. S. Kim, "Voice activity detection based on generalized gamma distribution," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, ICASSP'05*, Mar 2005.

[45] S. Intajag and S. Chitwong, "Speckle noise estimation with generalized gamma distribution," in *Proc. of the IEEE Int. Joint Conf. on SICE-ICASE*, Oct 2006.

[46] D. Karaboga and E. Kaya, "Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey," *Artificial Intelligence Review*, vol. 52, no. 4, pp. 2263–2293, Jan 2018.

[47] *Electromagnetic compatibility and Radio spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) protocol*, ETSI, TS 102 361-1.

[48] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus [speech database]," National Institute of Standards and Technology, 1993, available at: https://catalog.ldc.upenn.edu/LDC93S1.

[49] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, Jan 2008.

[50] "ITU-T Rec. P.862," 2000, objective quality measurement of speech codecs. [Online]. Available: https://www.itu.int/rec/T-REC-P.862/en