IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 27, 2021, accepted January 29, 2022, date of publication February 22, 2022, date of current version May 4, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3153469

CITISEN: A Deep Learning-Based Speech Signal-Processing Mobile Application

YU-WEN CHEN, KUO-HSUAN HUNG, YOU-JIN LI, ALEXANDER CHAO-FU KANG, YA-HSIN LAI, KAI-CHUN LIU, SZU-WEI FU, SYU-SIANG WANG, AND YU TSAO, (Senior Member, IEEE)

Research Center for Information Technology Innovation, Academia Sinica, Taipei 11529, Taiwan

Corresponding author: Yu Tsao (yu.tsao@citi.sinica.edu.tw)

This work was supported in part by the Ministry of Science and Technology under Grant 108-2628-E-001-002-MY3, and in part by the Academia Sinica under Grant AS-GC-111-M01.

ABSTRACT This study presents a deep learning-based speech signal-processing mobile application known as CITISEN. The CITISEN can perform three functions: speech enhancement (SE), model adaptation (MA), and background noise conversion (BNC), which allow CITISEN to be used as a platform for utilizing and evaluating SE models and flexibly extend the models to address various noise environments and users. For SE, CITISEN downloads pretrained SE models on the cloud server and then uses these models to effectively reduce noise components from prerecordings or instant recordings provided by users. When it encounters noisy speech signals with unknown speakers or noise types, the MA function allows CITISEN to improve the SE performance effectively. A few audio files of unseen speakers or noise types are recorded and uploaded to the cloud server and then used to adapt the pretrained SE model. Finally, for BNC, CITISEN removes the original background noise using an SE model and then mixes the processed speech signal with new background noise. The novel BNC function can evaluate SE performance under specific conditions, cover people's tracks, and provide entertainment. The experimental results confirmed the effectiveness of SE, MA, and BNC functions. Compared with the noisy speech signals, the enhanced speech signals by SE achieved about 6% and 33% of improvements, respectively, in terms of short-time objective intelligibility (STOI) and perceptual evaluation of speech quality (PESQ). With MA, the STOI and PESQ could be further improved by approximately 6% and 11%, respectively. Note that the SE model and MA method are not limited to the ones described in this study and can be replaced with any SE model and MA method. Finally, the BNC experiment results indicated that the speech signals of original and converted backgrounds have a close scene identification accuracy and similar embeddings in an acoustic scene classification model. Therefore, the proposed BNC can effectively convert the background noise of a speech signal and be a data augmentation method when clean speech signals are unavailable.

INDEX TERMS Speech enhancement, model adaptation, background noise conversion, deep learning, mobile application.

I. INTRODUCTION

In recent years, a wide variety of speech-related applications have been developed. Most of these applications are highly convenient for human–human and human–machine communications. However, the following long-existing and critical issues that may limit the achievable performance of these applications remain to be solved: speech distortions caused by additive/convolutional noises and channel/device effects [1]–[6]. Identifying an effective method of addressing this distortion issue is a critical and challenging task, and numerous approaches have been proposed to this end, among which speech enhancement (SE) is notable.

The goal of SE is to transform noisy speech signals into enhanced speech signals with improved quality and intelligibility [7], [8]. In the past several decades, SE has been widely used as a front-end unit in many voice-based applications, such as automatic speech recognition [9], speaker identification [10], [11], speech coding [12], hearing aids [13], [14], and cochlear implants [15], [16]. The existing SE methods can be divided into two classes. In the first class, SE methods design a filter or function to attenuate noise components. Examples of methods in this class include the Wiener filter and its extensions [17]–[20],

The associate editor coordinating the review of this manuscript and approving it for publication was Cong Pu.

adaptive filtering [21], the minimum mean square error spectral estimator (MMSE) [22]–[24], Karhunen-Loeve transform [25], maximum a posteriori spectral amplitude estimator [26], [27], maximum likelihood spectral amplitude estimator [28], [29], linear prediction models [30], predictive coding [31], orthogonal polynomial-based method [32], super-Gaussian-based methods [33], [34], and the hybrid of orthogonal polynomial and super-Gaussian [35]. Most SE methods of the first class have a common limitation: the inability to effectively contrast non-stationary noise signals in real-world scenarios under unexpected acoustic conditions.

SE methods in the second class are based on machine-learning algorithms; these methods typically prepare a model for noisy-to-clean transformation in a data-driven manner. Notable SE methods belonging to this class include hidden Markov models [36], non-negative matrix factorization [37]–[39], compressive sensing [40], and robust principal component analysis [41]. In addition, artificial neural networks (ANNs), as a successful machine-learning model, have been used for SE because of their powerful nonlinear transformation capability. In [42]–[45], a shallow ANN was used to map noisy speech signals to clean ones. More recently, various types of ANNs with deep structures have been used for SE (e.g., deep neural networks (DNNs) [46]–[49], deep recurrent neural networks and long-short term memory (LSTM) networks [50], [51], convolutional neural networks (CNNs) [52], [53], and convolutional recurrent neural networks (CRNNs) [54]). Also, [55] proposed a hybrid architecture of CNN and a tensor-train layer and compared the performance between DNN and CNN.

To improve the performance of these ANN-related approaches, several SE studies have applied a generative adversarial network (GAN) model [56]–[59]. The GAN model is used to generate enhanced samples for a discriminator to determine whether the input follows the distribution of a real clean speech signal. In addition, some researchers applied a transformer technique to perform SE, in which the attention mechanism was utilized to capture long-term temporal correlations to extract clean components from noisy input [60]–[62]. Moreover, instead of using a large amount of training data to perform SE, a transfer learning technique has been commonly used to enhance the generalization of models in unseen environments. For example, [63] fine-tuned the generator in a pretrained GAN-based SE model with small amounts of data and confirmed the efficiency of transfer learning. In [64], the authors proposed the use of a teacher-student learning strategy to adapt an SE model to unlabeled noisy speech signals. Furthermore, the FA-MK-MMD approach was proposed in [65] to train a neural network model from the labeled source domain to extract the shared representation to enhance the unlabeled input. Although the effectiveness of these SE approaches has been verified, their performance in mobile applications is yet to be confirmed.

In this study, we present a speech signal processing mobile application called CITISEN.[1] CITISEN is a standardized SE software with a user interface that can be used as a platform for utilizing and evaluating newly performed deep-learning-SE models by simply replacing the default settings with the associated model. Based on SE, two extended functions—model adaptation (MA) and background noise conversion (BNC)—were also implemented in CITISEN. The MA function was built to further improve the SE performance for a specific user or under certain noise environments. The adaptation data were prepared by the users to meet their requirements, thus making the framework a customized tool. The BNC function converts the original background noise to another one. BNC can be used to evaluate SE performance under practical conditions. In this condition, the residual noises in an enhanced source speech signal are combined with different background interference and affect the quality and intelligibility of a target speech signal. In addition, the BNC can be used to cover people's tracks by converting the original environment noises to noises from other places when a positioning system is unavailable or not being used because of limited access to the technology or the lack of intention. Furthermore, the BNC can also be used for entertainment purposes, such as adding background music or sound effects.

The contribution of this study is summarized as follows:

- To the best of our knowledge, the proposed CITISEN is the first to integrate BNC and MA functions with SE in a mobile application.
- CITISEN has a user interface for performing SE on a prerecording or instant recording. The experimental results confirmed the SE function of improving short-time objective intelligibility (STOI) [66] and perceptual evaluation of speech quality (PESQ) [67] scores.
- CITISEN has an MA function that allows users to adapt the SE models to unseen background noises or speakers. The MA function is proven to provide notable STOI and PESQ improvements compared to the results without MA.
- CITISEN provides a novel BNC function that can evaluate SE performance under specific conditions, cover people's tracks, and provide entertainment. The listening test results indicated that the BNC function could convert the background noise while maintaining the clarity and intelligibility of the converted speech signals.
- An acoustic scene classification (ASC) model was used to evaluate the BNC performance. The results showed that new background noise could be successfully recognized. Moreover, the ASC embeddings suggested that the conversation results from a silent background were close to a noisy background. Therefore, the BNC function can potentially serve as a data augmentation

[1] CITISEN GitHub Page: https://github.com/yuwchen/CITISEN If there is any problem with testing pretrained SE models on CITISEN, please contact the corresponding author.

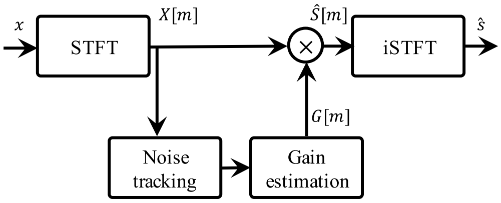


FIGURE 1. Traditional filter-based SE architecture. STFT and iSTFT denote the short-time Fourier transform and inverse STFT, respectively.

method for the ASC model when clean speech signals are unavailable.

- By simply replacing the settings with the associated model, CITISEN can utilize and evaluate other deep learning-based SE models not described in this study. Therefore, CITISEN can effectively reduce the development interval for converting SE models to industrial applications.

The remainder of this paper is organized as follows. Section II reviews related works. Section III elaborates the functions and user interface of CITISEN. Section IV presents the experimental setup and results. Finally, Section V provides some concluding remarks regarding this research.

II. RELATED WORKS

In this section, we first review one traditional filter-based SE method and four neural-network-based SE models used for comparison in the experiments. Then, we introduce the concept of MA.

A. TRADITIONAL GAIN FUNCTION-BASED SE METHOD

In the SE task, we generally assume that the noisy speech signal x contains a clean speech signal s and a noise signal v.

$$x = s + v \tag{1}$$

For the MMSE SE [22], [68] approach, the time-domain signal, x, is first converted to a spectral feature, X, using the short-time Fourier transform (STFT). After the STFT, Eq. 1 can be expressed as:

$$X[m] = S[m] + V[m] \tag{2}$$

where m denotes the mth frequency bin in the entire set of spectral features. By estimating the a priori and a posteriori signal-to-noise ratio (SNR) statistics based on a noise-estimation approach, we can estimate a function $G[m]$. The enhanced speech signal, $\hat{S}[m]$, is obtained by filtering $X[m]$ through $G[m]$. Finally, an inverse STFT (iSTFT) is applied to convert the spectral features $\hat{S}$ to the time-domain signal $\hat{s}$, as shown in Fig.1.

B. NEURAL-NETWORK-BASED SE METHOD

In this work, we used one waveform-based SE model, fully convolutional network (FCN) [52], and three spectral-based SE models, namely, deep denoising autoencoder (DDAE) [48], LSTM [51], and CRNN [54]. Table. 1 summarizes the NN models used in this study. Similar to traditional SE methods, the goal of the neural-network-based SE is to find the enhanced speech signal $\hat{s}$ that is close to the clean speech signal s.

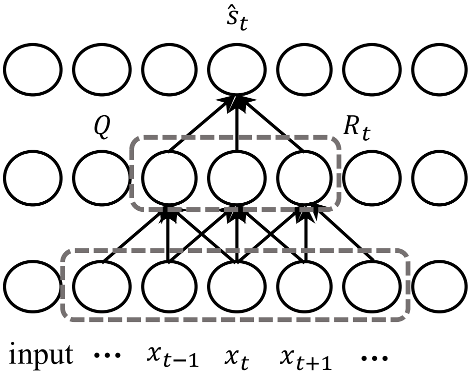


FIGURE 2. FCN-based SE architecture.

1) FCN-BASED SE MODEL

Fig. 2 shows an FCN model, which is similar to a conventional CNN, but all the fully connected layers are removed. As reported in [53], the FCN model can address the high-and low-frequency components of the raw waveform simultaneously. The relation between the output sample $\widehat{s_t}$ and the connected hidden nodes R_t can be represented by:

$$\widehat{s_t} = Q^T R_t \tag{3}$$

where Q denotes one of the learned filters and subscript t indexes the time step. Then, the objective function of the FCN-based SE model is defined as:

$$\mathcal{L}(\theta_F) = ||\widehat{s} - s||^2 \tag{4}$$

where θ_F denotes the model parameters of FCN.

2) DDAE-BASED SE MODEL

During the training of DDAE, noisy-clean speech signal pairs were used to compute the mapping function from noisy to clean spectral (logarithm amplitude in this study) features. The DDAE model aims to transform the noisy speech signal to a clean speech signal by minimizing the reconstruction error between the predicted $\widehat{S}$ and the reference clean spectral features S such that:

$$\theta_D^* = \underset{\theta_D}{\arg\min} \mathcal{L}(\theta_D) + \rho C(\theta_D), \tag{5}$$

with

$$\mathcal{L}(\theta_D) = \|\widehat{S} - S\|^2, \tag{6}$$

where θ_D denotes the model parameters of DDAE. ρ is a constant that controls the trade-off between the reconstruction accuracy and regularization term $C(\theta_D)$ [48] and is determined through the validation set in the training process. In this study, to simplify and compare with other methods, we set ρ to 0.

TABLE 1. Summary of NN models used in this study. The W and S in feature type column represent waveform-based and spectral-based input, respectively. Conv is the abbreviation of convolution.

	Models used in Section IV				Relevant Works
	Ref.	Feature type	NN layer	Highlight	
FCN-based	[52]	W	Conv	The input length can be varied, and the local feature structures can be preserved.	[53], [57], [59]
DDAE-based	[48]	S	Dense	The model structure is simple.	[46], [47], [49]
LSTM-based	[51]	S	LSTM Dense	LSTM is known for capturing temporal features.	[50]
CRNN-based	[54]	S	Conv LSTM Dense	The CRNN-based model takes advantage of both the convolutional and LSTM layers.	[61]

Given noisy spectral features X, the DDAE estimates a clean speech signal by:

$$\begin{aligned} h_1(X) &= \sigma(W_1 X + b_1), \\ &\vdots \\ h_{L-1}(X) &= \sigma(W_{L-1} h_{L-2}(X) + b_{L-1}), \\ \widehat{S} &= W_L h_{L-1}(X) + b_L, \end{aligned} \tag{7}$$

where $W_1 \dots W_L$ and $b_1 \dots b_L$ are the weight matrices and bias vectors, respectively, and L is the number of layers. In addition, σ is a vector-wise non-linear activation function sigmoid.

3) LSTM-BASED SE MODEL

Because LSTM can capture the temporal relation of speech signals, it has proven to deliver promising results in SE [51]. The objective function of the LSTM-based SE model is close to that of the DDAE model, which is to find the best model parameters of LSTM θ_L that can minimize:

$$\mathcal{L}(\theta_L) = \|\widehat{S} - S\|^2, \tag{8}$$

In this study, we used the LSTM unit defined as follows:

$$\begin{aligned} i_n &= \sigma(W_i X_n + U_i h_{n-1} + b_i), \\ o_n &= \sigma(W_o X_n + U_o h_{n-1} + b_o), \\ f_n &= \sigma(W_f X_n + U_f h_{n-1} + b_f), \\ g_n &= \tanh(W_g X_n + U_g h_{n-1} + b_g), \\ c_n &= f_n \odot c_{n-1} + i_n \odot g_n, \\ h_n &= o_n \odot \tanh(c_n) \end{aligned} \tag{9}$$

where X_n, f_n, i_n, o_n, g_n, c_n, and h_n represent the input, forget gate, input gate, output gate, cell input activation, cell state, and hidden state vectors, respectively, and the subscript n indexes the frame step. In addition, W_q and b_q denote the weights and biases, respectively, where the subscript $_q$ can either be the input gate i, output gate o, forget gate f, or memory cell g, and $\odot$ represents element-wise multiplication.

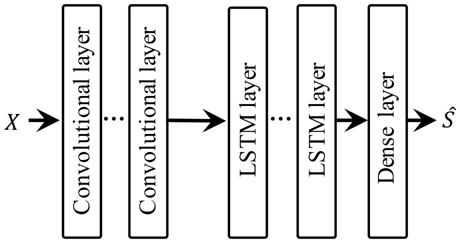


FIGURE 3. CRNN-based SE architecture.

4) CRNN-BASED SE MODEL

The CRNN in this work combines a CNN, LSTM, and Dense layer. Previous work indicated that CRNN could lead to better objective intelligibility and perceptual quality than an LSTM model with fewer trainable parameters [54]. The architecture of the CRNN-based SE model is shown in Fig. 3.

C. MODEL ADAPTATION

When operating SE in a real-world scenario, unknown noise types and new users are often encountered. Therefore, in many cases, the testing data may not be adequately covered by the trained SE model. Such training/testing mismatches in acoustic characteristics may considerably degrade SE performance. To effectively address this mismatch issue, an adaptation of an SE model is required. Thus far, various MA approaches have been proposed [69]–[73]. The main concept of MA is to adjust the parameters of a pretrained model (prepared using training data) based on a small set of testing data to reduce the influence of training/testing mismatches. Because the adapted SE models match the testing conditions, the SE performance can be improved.

III. THE PRESENTED CITISEN APPLICATION

CITISEN has three functions, including SE, MA, and BNC. For SE, CITISEN can enhance the quality and intelligibility of noise signals by reducing noise components from the speech signals. Then, for MA, CITISEN can further improve the results of SE by fine-tuning the SE model with uploaded data. Finally, for BNC, CITISEN can replace the original background noise with specified background noise. The functions of CITISEN are illustrated in Fig. 4.

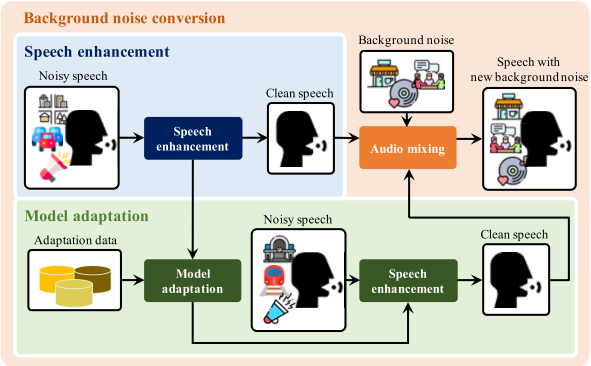


FIGURE 4. The SE, BNC, and MA functions in CITISEN.

A. SE FUNCTION

SE is a major function of CITISEN. As shown by the blue block in Fig. 4, given the noisy speech signal, the SE function removes background noises and generates the enhanced speech signal with improved quality and intelligibility. The SE models were trained in a cloud server, and the trained models were loaded into mobile devices. Because the model is trained and saved in a cloud server, mobile devices do not need to have a huge computational resource. When connected to the Internet, mobile devices automatically download updated SE models. A third-party module, called okhttp3, was used to save and manage the SE models. In addition, for SE, CITISEN has two recording modes: prerecording and instant recording. In the prerecording mode, CITISEN records the entire speech signal before processing, whereas in the instant recording mode, CITISEN records and processes the speech signal simultaneously. CITISEN is a standardized SE software with a user interface that can support pretrained SE models trained by various machine learning frameworks, including Keras, PyTorch, and TensorFlow. In addition, the SE models can have different architectures or input acoustic feature formats.

Fig. 5 (a) shows the implementation of the SE function in CITISEN, which contains four steps, including audio-recording, pre-processing, speech enhancing, and post-pressing. The details of SE function steps in CITISEN are described as follows.

1) AUDIO RECORDING

In this step, the application interface is implemented using the Java/Android application programming interface (API) AudioRecord. The AudioRecord saves an audio signal at a sampling rate of 16000 Hz in a single channel. In the instant recording mode, as AudioRecord processes and analyzes audio data in every 5120 bytes, which is equivalent to 320 sample points per second, the instant recording will approximately have a 20 ms delay. The configuration of the AudioRecord in CITISEN is presented in Table 2.

TABLE 2. AudioRecord configuration in CITISEN. (PCM: pulse-code modulation).

Parameter	Value
Sampling rate	16000 Hz
Audio channel	Mono
Audio format	PCM in 16 bits
Audio buffer size	5120 bytes

2) PRE-PROCESSING

In this step, CITISEN transfers the data format of the mobile input (byte) to the data format of the SE model input (float). For waveform-based SE models, such as FCN, the preprocessing step transfers the format of time-domain audio signals from bytes to float. For spectral-based SE models, such as DDAE, an additional STFT is required to transfer time-domain signals into frequency-domain signals. CITISEN performs STFT by calling Java/Android API DoubleFFT_1D in the JTransforms library. By calling this API, a one-dimensional time-domain signal is transferred into a complex matrix. The energy part of the complex matrix is presented as a spectrogram, which is used as the input for spectral-based SE models. The phase part of the complex matrix is reserved and used later to convert the enhanced spectrogram back to the time-domain audio signals.

3) SPEECH-ENHANCING

To operate the SE model on mobile devices, the pretrained SE model needs to be packaged into a.pb file. Then, CITISEN calls the Java API, which is built in TensorFlow: TensorFlow-InferenceInterface, and passes the assetManager (.pb file) and modelFilename (model Name) to the API. Finally, CITISEN loads the SE model and calculates the enhanced speech signal. This part requires the microprocessor of the mobile device to participate in the calculation, and thus different types of mobile phone models will have different time delays. Currently, we have implemented FCN-based and DDAE-based SE in CITISEN; however, the available SE models can be easily extended by uploading the SE models using the same method.

4) POST-PROCESSING

For spectral-based SE models, such as DDAE, the output of the SE model are reconstructed to a time-domain signal. The waveform reconstruction method in CITISEN is the iSTFT, which is implemented with the DoubleFFT_1D function. For waveform-based SE models, such as FCN, the output is already a time-domain signal and does not require additional conversions. Finally, the data type of the enhanced speech signals is converted to a playable form (from float to byte).

B. MA FUNCTION

The MA function of CITISEN aims to adapt the SE model to unknown noises or new speakers, or both. CITISEN provides three different MA modes: noise only (N), speaker only (S), and noise and speaker (N+S). Users can upload a short audio clip of the environment noise or their clean speech signals to the cloud server, and the parameters of the

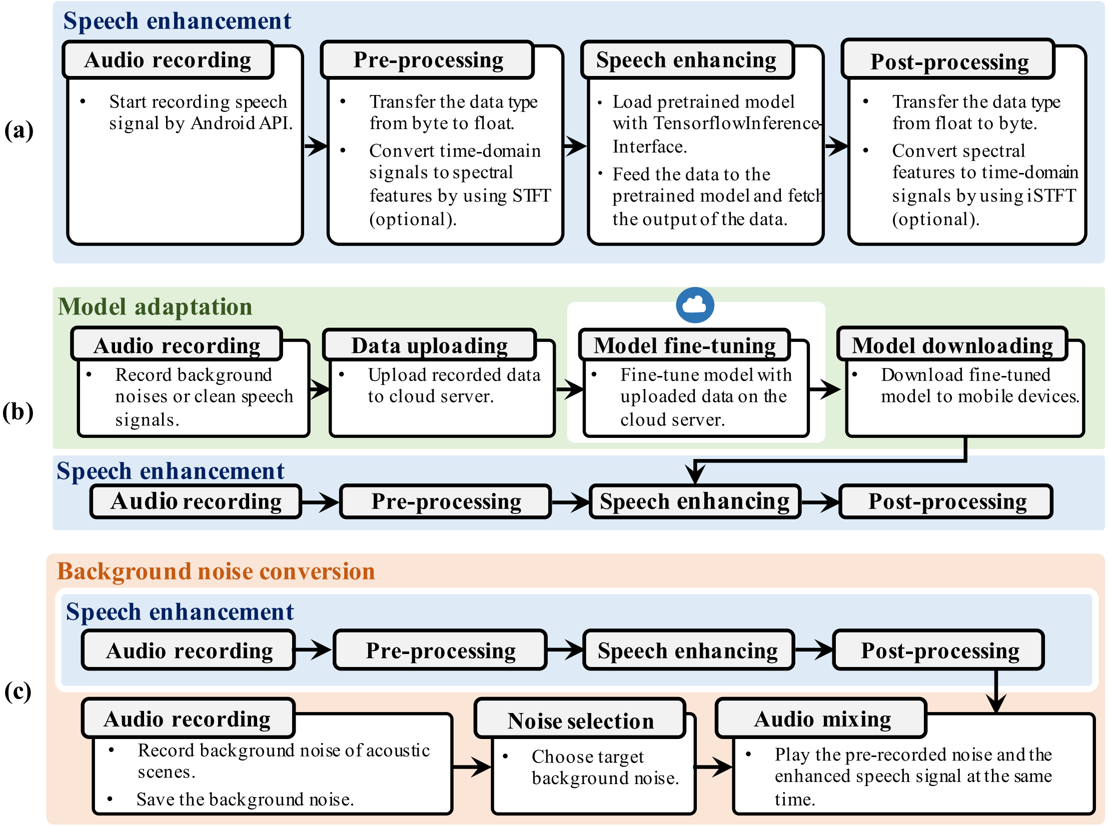

Speech enhancement
(a)
Audio recording
• Start recording speech signal by Android API.
Pre-processing
• Transfer the data type from byte to float.
• Convert time-domain signals to spectral features by using STFT (optional).
Speech enhancing
• Load pretrained model with TensorflowInference Interface.
• Feed the data to the pretrained model and fetch the output of the data.
Post-processing
• Transfer the data type from float to byte.
• Convert spectral features to time-domain signals by using iSTFT (optional).
Model adaptation
(b)
Audio recording
• Record background noises or clean speech signals.
Data uploading
• Upload recorded data to cloud server.
Model fine-tuning
• Fine-tune model with uploaded data on the cloud server.
Model downloading
• Download fine-tuned model to mobile devices.
Speech enhancement
Audio recording
Pre-processing
Speech enhancing
Post-processing
Background noise conversion
(c)
Speech enhancement
Audio recording
Pre-processing
Speech enhancing
Post-processing
Audio recording
• Record background noise of acoustic scenes.
• Save the background noise.
Noise selection
• Choose target background noise.
Audio mixing
• Play the pre-recorded noise and the enhanced speech signal at the same time.

FIGURE 5. Implementation details of three functions in CITISEN. Figure (a), (b), and (c) are SE, MA, BNC functions, respectively. For SE, CITISEN downloads pretrained SE models on the cloud server and then uses them to enhance prerecordings or instant recordings provided by users. For MA, a few audio files of unseen speakers or noise types are recorded and uploaded to the cloud server then used to adapt the pretrained SE model. For BNC, CITISEN removes the original background noise using an SE model, then mixes the processed speech signal with new background noise.

original SE model will be fine-tuned using the uploaded data. Users can then download and use the adapted SE models in CITISEN. Currently, we suggest that users record their referenced target speech signal in a noise-free environment. However, previous studies [74], [75] have shown that some level of noise contained in the referenced target can also lead to an effective reconstruction of the clean waveform in an SE system. The implementation of the MA function is shown in Fig. 5 (b).

C. BNC FUNCTION

BNC is a new topic in the field of speech processing. This idea is similar to the changing background of an image or video [76]. With the BNC, users can artificially convert the background noise of their speech signal to another specified noise. To use the BNC function, the noises of the target background must be recorded and stored first. Users can record background noises in different environments in real-world scenarios, such as car engine sounds and train stations. Then, users need to select the target background noise before running the BNC function. When running the BNC function, CITISEN removes the original background noise by using SE first and mixes the enhanced speech signal with new background noises by playing them simultaneously. In addition to SE steps, BNC has three additional steps: audio recording (of background noise), noise selection, and audio mixing. Fig. 5 (c) illustrates the implementation of the BNC function.

D. CITISEN USER INTERFACE AND USAGE

CITISEN has four pages: "speech enhancement," "background noise conversion," "uploading," and "recording," as shown in Fig. 6. The page name and navigation buttons are on each page's top left and bottom, respectively.

1) SPEECH ENHANCEMENT PAGE

Fig. 7 shows the "speech enhancement" page. On this page, the user needs to specify the gender by the "gender" button. Because males and females usually have different voice characteristics, knowing the users' gender can help to improve the performance of SE models. Then, by pressing the "model switch" button, the user can choose different SE models from an SE model list. Currently, CITISEN provides several default SE models trained using our own collected speech datasets. By pressing the "preview" button, users can hear their instant recordings without using SE. By pressing the "activate" button, the SE function will be activated, and users can hear their enhanced instant recordings.

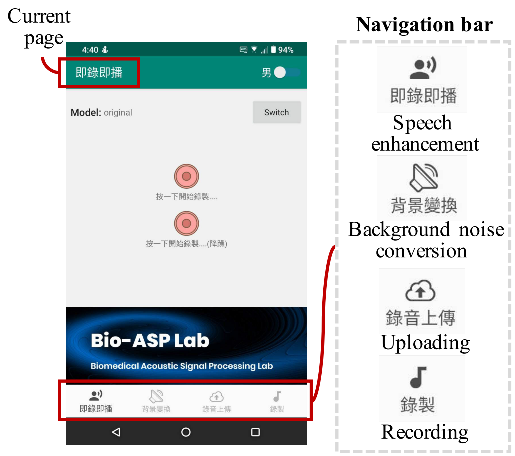


FIGURE 6. Four main pages in CITISEN.

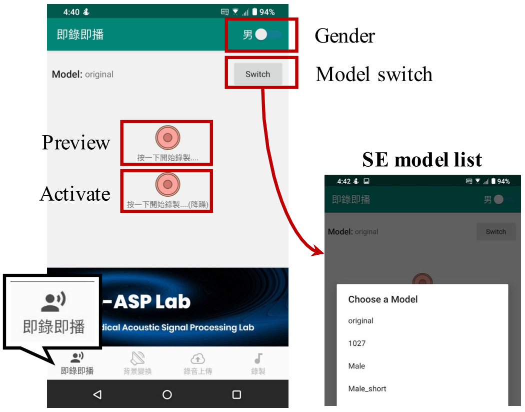


FIGURE 7. Speech enhancement page of CITISEN. The "gender" button on the upper-right corner is used to specify the user's gender. By pressing the "model switch" button, an SE model list will pop up, and users can change the SE model. After pressing the "preview" button, users will hear their original instant recordings, and after pressing the "activate" button, users will hear their enhanced instant recordings.

2) BACKGROUND NOISE CONVERSION PAGE

The "background noise conversion" page of CITISEN is shown in Fig. 8. On this page, CITISEN mixes the specified background noise with the enhanced speech signal to generate a new speech signal with the specified background noise. By pressing the "sound switch" button, users can choose the background noise they want to use on the pop-up background noise list. By pressing the "record noise" button, users can record and save a new background noise. In addition, by pressing the "activate" button, users will hear their enhanced instant recordings with the specified background noise. Moreover, the "background noise conversion" page has a volume bar, which allows users to adjust the volume of background noise and specify the SNR level of the converted speech signal accordingly.

3) UPLOADING PAGE

The "uploading" page is used for uploading the data for the MA function. As CITISEN provides both unknown noise adaptation and new speaker adaptation, there are two file upload buttons: "record speech" and "record noise." To start the recording, users can simply press one button. After finishing the recording by pressing the button again, CITISEN will pop up a submission window. Users can then name the audio file and upload the recorded audio to the server. After receiving the audio file, the server can adapt the SE model by fine-tuning the original SE model using the recorded audio data. The name of the audio file can also be used to call the adapted SE model, which is later sent from the server to the mobile device and appears on the SE model list on "speech enhancement" and "background noise conversion" pages. Accordingly, users can run the SE and BNC functions using the adapted SE model. The "uploading" page of CITISEN is shown in Fig. 9.

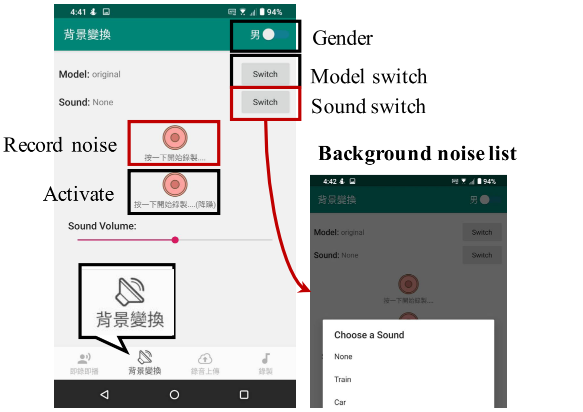


FIGURE 8. Background noise conversion page of CITISEN. By pressing the "sound switch" button, a background noise list will pop up. After pressing the "record noise" button, users can record and save a new noise signal. After pressing the "activate" button, users will hear the enhanced instant recording contaminated with the specified background noise. Note that the "gender" button and the "model switch" button have the same function as those on the "speech enhancement" page.

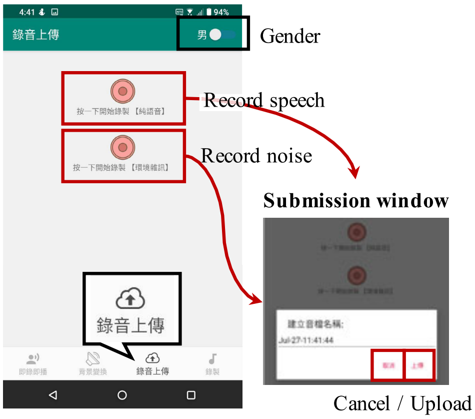


FIGURE 9. Uploading page of CITISEN. After recording a noise or speech signal, CITISEN asks the user to name and save the audio file and upload it to the cloud server.

4) RECORDING PAGE

The "recording" page supports prerecording and SE model evaluation. Specifically, on the "recording" page, users

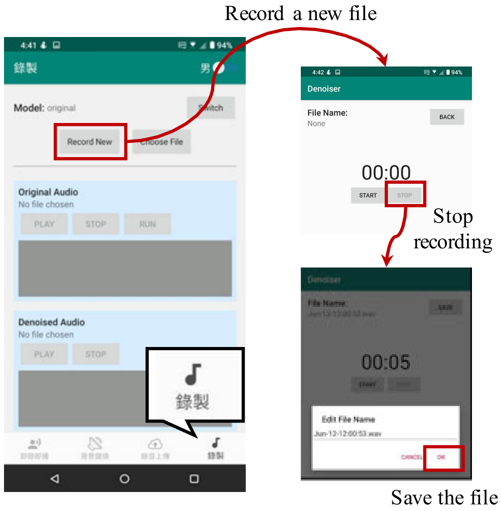


FIGURE 10. Recording page of CITISEN (Part I). A new audio file is recorded after pressing the "record new" button. The file can then be named and saved in a pop-up submission window.

can save, playback, and run SE on a saved speech signal. First, users can record new audio by pressing the "record new" button, and CITISEN will redirect to a processing page. After finishing the recording by pressing the "stop" button, users can name and save the record. The workflow is shown in Fig. 10. Then, users can choose an audio file, a model mode, and an SE model with the "choose file," "gender," and "model switch" buttons, respectively. Finally, by pressing the "run" button, an enhanced speech signal is generated. Because CITISEN demonstrates both the noisy and enhanced spectrograms, users can visually evaluate the SE results. In addition, users can aurally evaluate the results by pressing the "play" and "stop" buttons to listen to the original and the enhanced speech signals. An illustration showing more details about the "recording" page is shown in Fig. 11 and Fig. 12.

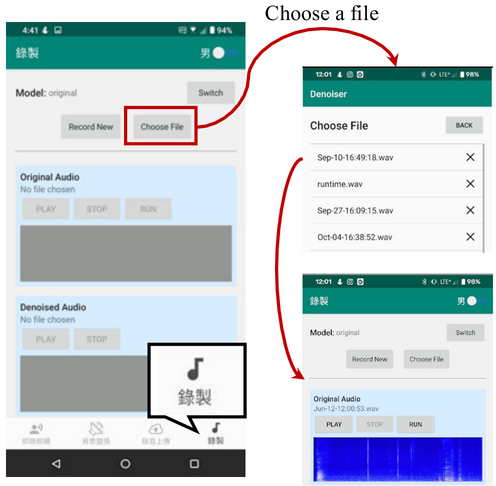


FIGURE 11. Recording page of CITISEN (Part II). By pressing the "choose file" button, users can choose an audio file on a pop-up window.

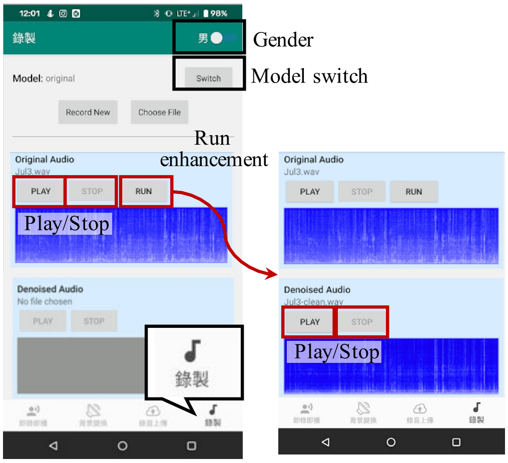


FIGURE 12. Recording page of CITISEN (Part III). Users can choose an SE model type and an SE model by using the "gender" and "model switch" buttons. In addition, users can evaluate the SE results visually and aurally.

IV. EXPERIMENTS

This section presents the setup, implementation details, and results of the experiments that tested the performance of the SE, MA, and BNC functions.

A. EXPERIMENTAL SETUP

In this study, TMHINT utterances [77] were used to prepare the training and testing sets, and the utterances were recorded at a 16 kHz sampling rate in a 16-bit format. Notably, the experiments are conducted offline on the cloud platform instead of the mobile platform for several reasons. First, the cloud platform provides a more stable communication and computation environment, which ensures the listening test can go smoothly. Second, because the performance of mobile phones varied too much, choosing one as the representative is hard. Moreover, mobile phones progress so fast that the current best mobile phone might be greatly outperformed by the new mobile phone next year. Finally, evaluating the results on the cloud platform provides the upper bound of these functions and makes the results comparable with other studies.

1) SE EXPERIMENTS

In the SE experiments, the training set was prepared using speech utterances from three males and three females. Each speaker read 200 TMHINT utterances in a quiet room, totaling 1200 clean utterances. Each utterance had approximately 3s and contained ten Chinese characters. Noisy utterances were generated by artificially contaminating these 1200 clean training utterances with five randomly sampled noise types from a 100-noise type dataset [78] at eight different SNR levels (±1dB, ±4dB, ±7dB, and ±10dB). Consequently, 48000 noisy-clean pair utterances were obtained. As for the testing set, we used the speech utterance from two other speakers (one male and one female, termed testing speaker in the following discussion), with 120 utterances for each speaker. We generated noisy utterances by artificially contaminating these 120 clean utterances with another set of five noise types (car, sea wave, take-off, train, and song) at two different SNR levels (0dB and 5dB). Notably, the speakers, speech content, and noise types differed between the training and testing sets. The performance of the SE was tested using both subjective listening tests and objective evaluations.

For the listening tests, we recruited 20 participants with a male-to-female ratio of 2 to 3. The group ages were between 20 and 38 years, with a mean age of 21.50 (standard deviation (SD) = 3.97). All participants were native Mandarin speakers with normal hearing to perceive the stimuli during the test. Each participant listened to 80 testing speech signals (40 for 0 dB and 40 for 5 dB) spoken by one male and one female testing speakers. These 80 speech signals had different contents with one of the five assigned background noises (car, sea wave, take-off, train, and song) under four conditions, including original noisy speech signals (without enhancement), enhanced by an MMSE-based SE method, enhanced by a DDAE-based SE model, and enhanced by an FCN-based SE model. These four conditions are denoted as noisy, MMSE, DDAE, and FCN, respectively, in the following discussion. Each participant tested 40 lower- and 40 higher-SNR speech signals. In addition, the subjects were instructed to repeat what they had heard verbally and were allowed to repeat the stimuli once. The character correct rate (CCR), which is calculated by dividing the number of correctly identified characters by the total number of characters, was used to evaluate the intelligibility of speech signals.

For the objective test, we evaluated the results of two more neural-network-based methods, including LSTM-based SE and CRNN-based SE. In the following discussion, the speech signals enhanced by these two methods are denoted as LSTM and CRNN, respectively. PESQ [67] and STOI [66] were used as objective evaluation metrics. PESQ was designed to evaluate the quality of the processed speech signal, and the score ranged from −0.5 to 4.5. A higher PESQ score indicates that an enhanced speech signal is closer to the clean speech signal. STOI was designed to compute speech intelligibility, and the scores ranged from 0 to 1. A higher STOI score indicates better speech intelligibility.

2) MA EXPERIMENTS

The performance of the MA function was evaluated under three modes: MA(N), MA(S), and MA(N+S). The training set of the MA experiments was prepared as follows: For MA(N), two new noises (machine beeping and air flowing) from a real hospital scenario were mixed with the same training clean utterances as the SE experiments to form the new noisy-clean speech signal pairs. For MA(S), we mixed 40 clean utterances of the testing speakers in the SE experiments (20 utterances for each speaker) with the same training noises as the SE experiments to form the new noisy-clean speech signal pairs. For MA(S+N), the testing speakers' clean utterances and new noise signals were mixed to form new noisy-clean speech signal pairs. In the SE experiments, the SNRs for performing noisy-training utterances were ±1 dB, ±4 dB, ±7 dB, and ±10 dB. These training data were then used to fine-tune the pretrained SE model in the SE experiments until the model converged. The testing set of MA experiments had the same testing clean utterances as the SE experiments mixed with machine beeping and air flowing noise at four different SNR levels (±2 dB, 0 dB, and 5 dB).

Specifically, for MA(N), the training and testing speakers were independent, but the noises came from the same source. For MA(S), the training and testing speakers overlapped, but the training and testing noises were independent. In MA(S+N), the training speakers and testing speakers overlapped, and the noises came from the same source. Note that in every MA experiment, the contents of training speech signals and testing speech signals were different. In addition, the training and testing noises in MA(N) and MA(S+N) were from the same sources but recorded at different times.

3) BNC EXPERIMENTS

Based on our literature survey, there is no standard method for evaluating BNC. Because BNC aims to convert the original background noise into the target background noise, the accuracy (ACC), which is the number of correctly identified types of background noise divided by the total number of questions, was used to evaluate the BNC results. In addition, CCR was used to evaluate the maintenance of clarity and intelligibility of the converted speech signals. The ACC and CCR scores are estimated by both humans and machines. Specifically, we invited human listeners to conduct listening tests. We also trained an ASC model to analyze the ACC and used a pretrained automatic speech recognition (ASR) model to measure the CCR. For human evaluation, the CCR was the ratio of characters that a participant could correctly recognize. For machine evaluation, the CCR was calculated using the Levenshtein distance [79] between the predictions of a pretrained ASR system [80] and the ground truth. The details of the listening test and the ASC model are as follows.

a: LISTENING TEST

We asked the listeners to identify one out of five background noises (car, sea wave, take-off, train, and song) after listening to a converted speech signal. To avoid random guessing, listeners could choose "not clear" if they could not identify the background noise. During the test, participants were asked to repeat what they had heard, select the characters they had heard, and identify the background noise. Forty participants with a male-to-female ratio of 9 to 11 were recruited to participate in this set of listening tests. The group ages were between 14 and 43 years, with a mean age of 25.74 (SD = 8.68). All participants were native Mandarin speakers with normal hearing to perceive the stimuli during the test.

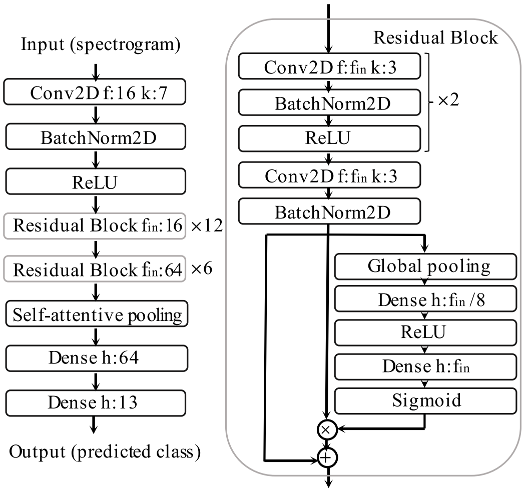


FIGURE 13. Structure of the ASC model. Conv2D represents the 2D convolution layer, where *f* is the number of filters, and *k* is the kernel size. The *h* in the dense layer denotes the size of each output sample.

The stimuli were Mandarin sentences spoken by one male and one female testing speaker. The testing speech signals were either processed using one of three SE methods (i.e., MMSE, DDAE, and FCN) or were not processed (i.e. the clean speech signals). Notably, the enhanced speech signals from SE experiments were used for BNC experiments, which means the original background noises were either car, sea wave, take-off, train, or song. The enhanced speech signals were then contaminated with the car, sea wave, take-off, train, or song noises, making 5 × 5 possible kinds of BNC conditions. To avoid the fatigue effect, we only tested the results of the 5 dB to 5 dB SNR condition. That is, the SNR of original and converted speech signals are both 5 dB. In total, each participant listened to 80 utterances.

b: ASC MODEL

We used the same dataset as the SE experiment to train the ASC model. Specifically, the training and testing utterances were the same as those in the SE experiments described in Section IV-A1. Thirteen noise types were used for the ASC model. Five of them were test noises used for the SE experiment, including car, sea wave, take-off, train, and song. The remaining eight noises were selected from the training noises of the SE experiment. Each noise segment was cut into two segments with a ratio of one to four. The shorter segment was used for testing, whereas the longer segment was used for training. The training and testing SNR levels were the same as the SE experiment.

Fig. 13 shows the details of the ASC model, which is based on [81]. The input of the model is the log1p spectrograms [82]. A Training epoch of 100, batch size of 128, optimizer Adam with a learning rate of 0.0001, and cross-entropy loss were used.

B. IMPLEMENTATION DETAILS OF SE MODELS

This section describes the structures and training details of the neural-network-based SE models. For spectral-based models, including DDAE, LSTM, and CRNN, the parameter settings of the STFT were as follows: the window length was 512, the hop length was 256, and the window type was the Hanning window. Then, the log1p spectrograms [82] were used as the input for the SE models. In inference, the noisy phase was reserved and combined with the enhanced spectral features to reconstruct the time-domain signals.

1) FCN

The FCN consisted of eight convolutional layers, where the filter number and kernel size of each of the first seven layers were 128 and 55, respectively. Batch normalization and the LeakyReLU were used to regularize the output of a hidden layer. The filter number and kernel size in the last layer were 1 and 55, respectively, with the hyperbolic tangent activation function applied to the FCN output. The number of training epochs was set to 60. In addition, batch size 1, optimizer Adam with a learning rate of 0.001, and mean square error (MSE) criteria were used.

2) DDAE

To incorporate contextual information, for each self-defined DDAE layer in this work, five adjacent frames of the input feature vector were concatenated to form the input of the next layer, whereas the output of each layer was a single frame. Also, the ReLU was used to regularize the output layer. The DDAE was composed of three DDAE layers with 257 output units in each layer, followed by a dense layer with single frames as the input and 825 output units, and another dense layer with 257 output units. Finally, the DDAE model had four more DDAE layers with 257 output units. The number of training epochs was 200. In addition, a batch size of 128, an Adam optimizer with a learning rate of 0.0001, and MSE criteria were used.

3) LSTM

The LSTM model used in this evaluation was constructed in the order of three stacked LSTMs and dense layers. Each LSTM layer contained 492 memory cells, and the size of the latest dense layer was 257. The number of training epochs was set to 20. The Adam optimizer with a learning rate of 0.001 and MSE criteria were used.

4) CRNN

The CRNN combines CNN and LSTM to enhance the input raw waveform. The CRNN comprised four convolutional blocks first, where each block was composed of three two-dimensional convolution layers. The ReLu activation function was applied to process the output of each layer. The kernel size for each convolutional layer was three, and the number of channel settings was arranged in the order of 16, 32, 64, and 64. In each block, the stride setting for the output convolutional layer along the speech feature dimension was three, and the setting for the remaining layers was one. Then, the convolutional block was followed by four LSTM layers with 384 memory cells and 257-dimensional dense layers

TABLE 3. FLOPs and number of model parameters for FCN, DDAE, LSTM, and CRNN models.

	FLOPs (M)	# of parameters (M)
FCN	10.8	5.4
DDAE	2.1	2.1
LSTM	5.5	5.5
CRNN	9.5	4.8

with the ReLu activation function. The input dimensions for the decoder were reshaped from the output of the encoder to 192 (3 × 64). In addition, the number of training epochs was 200, the batch size was 128, the optimizer was Adam with a learning rate of 0.0001, and MSE criteria were used.

C. EXPERIMENTAL RESULTS

In this section, we compare the complexity of the neural-network-based models and then perform a numerical analysis of the SE, MA, and BNC functions. Finally, we present the visualization results of processed speech signals.

1) COMPLEXITY ANALYSES

First, we evaluated the complexity of neural-network-based SE models in terms of floating-point operations (FLOPs[2]) and the number of model parameters. From the results in Table 3, we can observe that models with convolutional layers, such as the FCN and CRNN, require higher computational cost in terms of the FLOPs metric. The higher FLOPs imply that these models require more computational loading on hardware resources with similar parameter sizes.

Note that to avoid unstable communication and computation, we conducted experiments offline on a computer. However, we also tested whether the model with the highest FLOPs, the FCN model, could run on CITISEN. The results showed that the FCN model could successfully run on CITISEN.

2) SE EXPERIMENT

Table 4 presents the STOI and PESQ scores of noisy and enhanced speech signals processed using the MMSE, DDAE, FCN, LSTM, and CRNN models. From Table 4, all SE methods improved the PESQ scores, and except for MMSE and LSTM, other SE methods increased the performance of STOI. The increased PESQ along with the decreased STOI imply that some SE methods improve the quality, but the produced distortion might affect the intelligibility of a speech signal. The results also show that DDAE, CRNN, and FCN achieved higher scores than MMSE in terms of both STOI and PESQ, whereas FCN provided the highest PESQ and STOI scores among the evaluated methods. The results also demonstrate the effectiveness of using a deep-learning model for the SE task.

Table 5 presents the subjective listening test results for noisy and the three SE methods. From the table, it can be observed that MMSE yielded lower CCRs compared to noisy for both 0 dB and 5 dB SNRs, which is consistent with the findings of previous research and the STOI results reported in Table 1. That is, although some SE methods effectively remove background noise, speech intelligibility might be affected. In addition, the SE function is more helpful under low SNR situations, as noisy speech signals maintain high levels of intelligibility under high SNR situations. The one-way analysis of variance and Tukey post-hoc comparisons were applied to demonstrate the significance of improvements for analyzing the SNR-based CCR results of noisy, MMSE, FCN, and DDAE. The evaluations first revealed the significant difference across four SE systems, with $p < 0.001$ at 0 dB and 5 dB SNRs. The Tukey post-hoc tests further verified the significant differences for the following SE condition pairs at 0 dB: (FCN, DDAE), (DDAE, MMSE), (FCN, MMSE), and (noisy, MMSE), and at 5 dB: (MMSE, DDAE), (noisy, DDAE), (FCN, DDAE). Notably, the analysis on the scores of FCN and noisy indicated no significant difference, with $p > 0.05$ at both 0 dB and 5 dB SNRs. To achieve a significant difference from noisy speech signals to enhanced speech signals, a more advanced SE method performing under lower SNR conditions might be required.

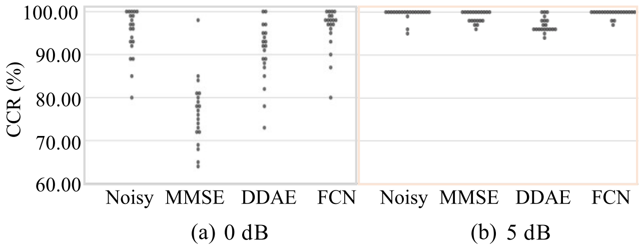


FIGURE 14. The subject-wise CCRs at (a) 0 dB and (b) 5 dB SNR conditions.

TABLE 4. Average STOI and PESQ scores for noisy and three SE methods over 0 and 5 dB SNR conditions. Noisy denotes the results of original noise without performing SE.

	STOI	PESQ
Noisy	0.6943	1.5188
MMSE	0.6497	1.6966
FCN	0.7666	2.2518
DDAE	0.7260	2.0366
LSTM	0.6610	1.6666
CRNN	0.7549	2.2144

TABLE 5. Average speech recognition results (CCRs) for noisy and three SE methods at 0 dB and 5 dB SNR conditions.

	0 dB	5 dB
Noisy	0.948	0.995
MMSE	0.764	0.989
DDAE	0.905	0.970
FCN	0.960	0.996

[2] https://github.com/Lyken17/pytorch-OpCounter

TABLE 6. Average STOI and PESQ scores for different SE models over −2, 0, 2, and 5 dB SNR conditions. Noisy denotes the results of original noise without performing SE, and baseline denotes the original FCN-based SE results.

	STOI	PESQ
Noisy	0.7392	1.7976
Baseline (w/o MA)	0.7858	2.3888
MA(N)	0.8256	2.6870
MA(S)	0.8090	2.4681
MA(N+S)	0.8317	2.6572

In addition to the averaged CCRs for all the participants, Fig. 14 (a) and (b) illustrate the subject-wise CCRs at 0 dB and 5 dB, respectively. Each gray circle in the figure represents the CCR score of an individual participant. According to both sub-figures, we can observe a larger CCR variance for MMSE and DDAE than that for FCN and noisy. The results imply the effectiveness of the FCN model in enhancing noisy speech signals with less ambiguous content than that of MMSE and DDAE.

3) MA EXPERIMENT

For the MA experiment, we fine-tuned the FCN model used in the SE experiment and used the original SE results from the FCN model as our baseline. From Table 6, it can be seen that SE yielded higher STOI and PESQ scores as compared to noisy, thereby confirming the results in that SE can improve speech quality and intelligibility over noisy speech signals, although the noise types are unknown and different from those used in the training set.

Next, compared with the baseline (original SE model without MA), all three MA modes achieved higher PESQ and STOI scores. More specifically, MA(N), MA(S), and MA(N+S) yielded noticeable relative improvements of 5.06%, 2.94%, and 5.84% in terms of STOI, and relative improvements of 12.48%, 3.32%, and 11.24%, in terms of PESQ, respectively, as compared to the baseline. Thus, the results obtained confirmed the effectiveness of the MA function and indicated that intelligibility and quality improvements could be attained by adapting the SE model based on both noise and speaker information. From the experimental results, we also observe that MA(N) achieved a higher PESQ than MA(S) and MA(S + N). One of the possible reasons for this is that the data for MA(N) was more than that for MA(S) and MA(S+N). Specifically, the number of fine-tuned speech signals was 2 × 1200 × 8 (new noises × clean training utterances of the original SE model × SNRs), 5 × 40 × 8 (training noises of original SE model × clean utterances from new speakers × SNRs,) and 2 × 40 × 8 (new noise × clean utterances from new speakers × SNRs) for MA(N), MA(S), and MA(S+N), respectively.

4) BNC EXPERIMENT

We present human and machine evaluations of the BNC function. Human evaluation was performed by conducting a listening test, whereas the machine evaluation was performed using an ASC model and a pretrained ASR system [80]. We evaluated the BNC using machines for three major reasons. First, recruiting humans to perform the tests is expensive and time-consuming, whereas using a machine to evaluate the performance is relatively inexpensive and efficient. Second, the ASC model has potential in several applications, such as monitoring systems, context-aware mobile devices, and audio search. Third, the machine can assist in human judgment. Therefore, the performance of the ASC model is also important for the BNC function. The details of the ASC model used in this study are described in Section IV-A3.b.

TABLE 7. CCR and ACC scores based on the BNS function in CITISEN.

	CCR	ACC
BNC(clean)	0.983	0.865
BNC(MMSE)	0.946	0.549
BNC(FCN)	0.968	0.814
BNC(DDAE)	0.949	0.804

a: RESULTS OF HUMAN EVALUATION

Based on the three SE methods, namely, MMSE, FCN, and DDAE, three sets of converted speech signals were obtained, denoted as BNC(MMSE), BNC(FCN), and BNC(DDAE), respectively. In addition, we included the results of BNC(clean), which is a set of speech signals converted from a silent background. Notably, BNC(clean) represents the upper bound of the BNC results because it was converted from a clean speech signal. From Table 7, we find that BNC(clean) has an ACC of 86.5%. This result suggests that participants sometimes could not correctly identify the type of background noise, although other types of noise did not contaminate the original speech signal. The 54.9% ACC of the BNC(MMSE) indicated that the enhanced speech signals of MMSE still contained high noise components that hindered the identification of the new background. However, BNC(FCN) and BNC(DDAE) achieved approximately 80% of ACC, suggesting that FCN and DDAE can produce enhanced speech signals with low residual noise components for the BNC function. Finally, the high CCR scores of the BNC(FCN) and BNC(DDAE) indicate the maintenance of clarity and intelligibility of the converted speech signals.

Fig. 15 shows the ACC of different types of BNC conditions. The results were the average scores of BNC(FCN) and BNC(DDAE). We excluded the results of BNC(MMSE) because the ACC of BNC(MMSE) was considerably lower than that of BNC(FCN) and BNC(DDAE), and MMSE performed worse than other SE methods (Table 4). From the column "sea" and "take-off" in Fig. 15, we observed that participants were less able to identify the background "sea" and "take-off." The "sea" and "take-off" backgrounds are less recognizable than the other noises because participants must hear a nearly complete wave or take-off sound to confirm it. Conversely, from the column "song," we know

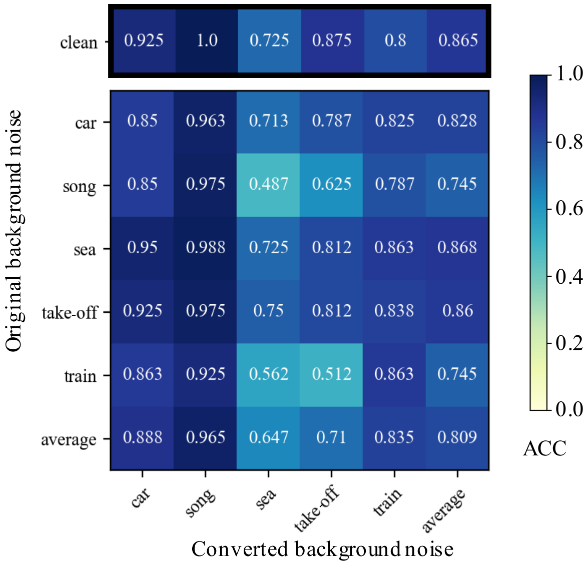


FIGURE 15. Listening test ACC of different kinds of BNC conditions. The results were the average scores of BNC(FCN) and BNC(DDAE).

that participants found it easier to identify the "song" background. This result might be because the "song" background contains music with a human voice, which is considerably different from other background noise. Evidently, the characteristics of the target background significantly affected the identification results. In addition, the original background noise affected the ACC because the noise type usually notably affects the SE performance.

b: RESULTS OF MACHINE EVALUATION

Because the BNC function focused on the background noise, the SNR level affected the performance. We make two assumptions about the effect of the SNR level of a speech signal on the performance of the BNC. The first assumption is that a higher original SNR will lead to a better ACC. That is, the target background noise is easier to identify if the converted speech signal is less affected by the original background noise. The second assumption is that a lower converted SNR will result in a better ACC. That is, the target background noise is easier to recognize if it is louder than the speech signals.

To test these two assumptions, we conducted four pairs of experiments that converted speech signals with the original SNR level a dB to speech signals with converted SNR level b dB, where $a \in \{0, 5\}$, and $b \in \{0, 5\}$. Fig. 16 shows the average ACC of the BNC(DDAE) and BNC(FCN). The figures in the same row and column represent the speech signals with the same original levels and converted SNR levels, respectively. That is, the influences of the original SNR level could be obtained by comparing the figures in different rows, whereas the effects of the converted SNR level could be determined by comparing the figures in different columns. In Fig. 16, we find that speech signals with an original SNR level of 5 dB (bottom row) outperform speech signals with an original SNR of 0 dB (top row), which confirms our first assumption that a speech signal with a higher original SNR has a better BNC result. Subsequently, speech signals with the converted SNR level of 0 dB (left column) performed better than speech signals with the converted SNR level of 5 dB (right column). This result verified our second assumption that a speech signal with a lower converted SNR yields a better BNC result.

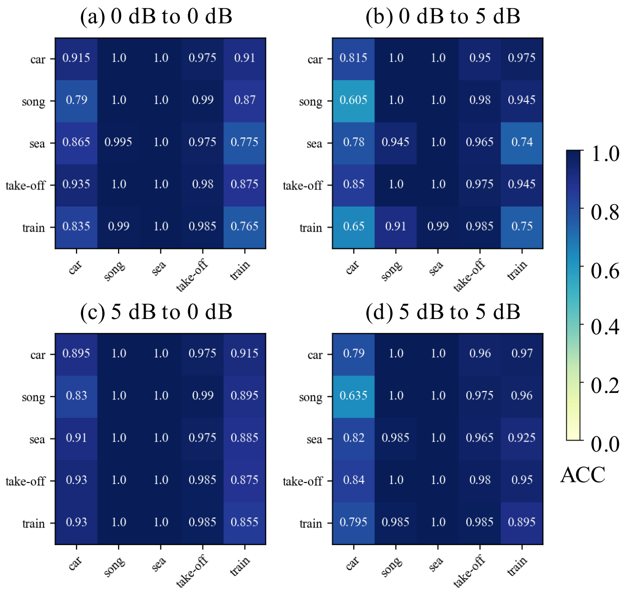


FIGURE 16. ACC at different BNC conditions. Comparing (c) and (d) (bottom row) with (a) and (b) (top row), the results showed that a speech signal with a higher original SNR had a better BNC result. In addition, comparing (a) and (c) (left column) with (b) and (d) (right column), the results indicated that a speech signal with a lower converted SNR had a better BNC result. The results were the average scores of BNC(FCN) and BNC(DDAE).

We evaluated the ACC of speech signals converted from a silent background (i.e., from a clean speech signal instead of an enhanced speech signal), which is the upper bound of BNC performance. Fig. 17 shows the results for different SNR levels. Unlike the results of the previous experiments, the converted SNR level did not affect the ACC of the BNC. None of the background noise conditions indicated that a lower converted SNR would lead to a better ACC. In addition, the average scores remained stable under different SNR levels. One possible reason is that, for an enhanced speech signal, a lower converted SNR can suppress the noise that was not removed by the SE models and make the target background easier to identify. Conversely, a lower converted SNR makes no difference for a clean speech signal because it does not contain other noise. Therefore, the target background is easy to recognize despite having a high converted SNR level. Notably, the ASC model achieves high ACC on the "sea" and "take-off" background, whereas the participants of the listening test have a lower identification rate for these two noises. The results suggest that the ASC model has potential

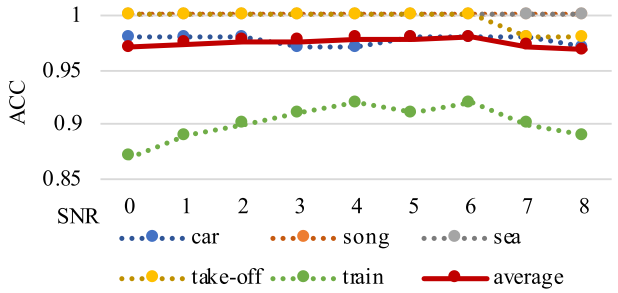


FIGURE 17. ACC vs. SNR of speech signals converted from a silent background. The ACC of "song" and "sea" overlapped in this figure. The results showed that SNR levels did not affect the ACC of BNC.

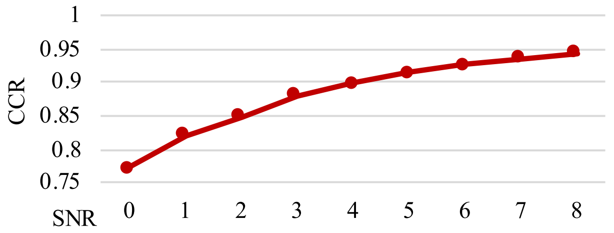


FIGURE 18. CCR vs. SNR of speech signals converted from a silent background. The results showed that the lower the SNR levels, the lower was the CCR.

TABLE 8. Summary of machine evaluations. The enhanced signal achieved more than 90% of ACC but had a great decline in CCR.

Source	Clean		Enhanced (5 dB)	
Converted SNR	5	0	5	0
ACC	0.978	0.970	0.937	0.953
CCR	0.914	0.771	0.756	0.605

to assist human listeners in recognizing noise that they cannot distinguish correctly. Finally, we present the CCR results using a pretrained ASR system [80]. As can be seen in Fig. 18, the SNR levels significantly affect the CCR. That is, the lower the SNR levels, the lower is the CCR.

Table 8 presents a summary of the machine evaluations. For the accuracy of BNC, enhanced speech signals performed worse than clean speech signals but still achieved more than 90% of accuracy. For the CCR, the performance decreased when using enhanced speech signals instead of clean speech signals. One possible reason is that the ASR system was not trained with the enhanced speech signals; therefore, the prediction of an enhanced speech signal is less accurate. In addition, the CCR is significantly affected by the language model of the pretrained ASR system. Specifically, despite having the same pronunciation, the ASR system might result in the wrong word, leading to a decline in CCR.

Subsequently, we used principal component analysis (PCA) [83] to visualize the embeddings of the ASC models in Fig. 19, where the clean and enhanced speech signals with background noise "n" are denoted as "c+n" and "en+n," respectively. We first found that different noise types were separated, indicating that the ASC model correctly recognized the background noise types. Then, we observed that the embeddings of clean speech signals with a new noise were close to those of enhanced speech signals with the same converted noise. Therefore, the BNC function can serve as a data augmentation method for the ASC model when clean speech signals are unavailable. Specifically, the BNC function can perform data augmentation by generating arbitrary numbers and SNR levels of training speech signals with specific background noise. In addition, the proposed BNC has the potential to open up new and interesting topics that have not yet received sufficient attention. For example, related studies include conversion of a new background noise naturally and the development of an ASC model that can distinguish between artificially converted and naturally recorded background noise.

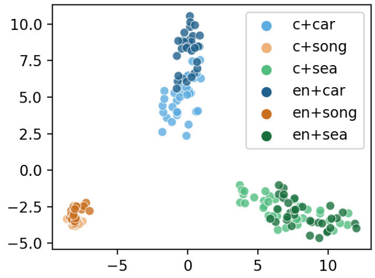
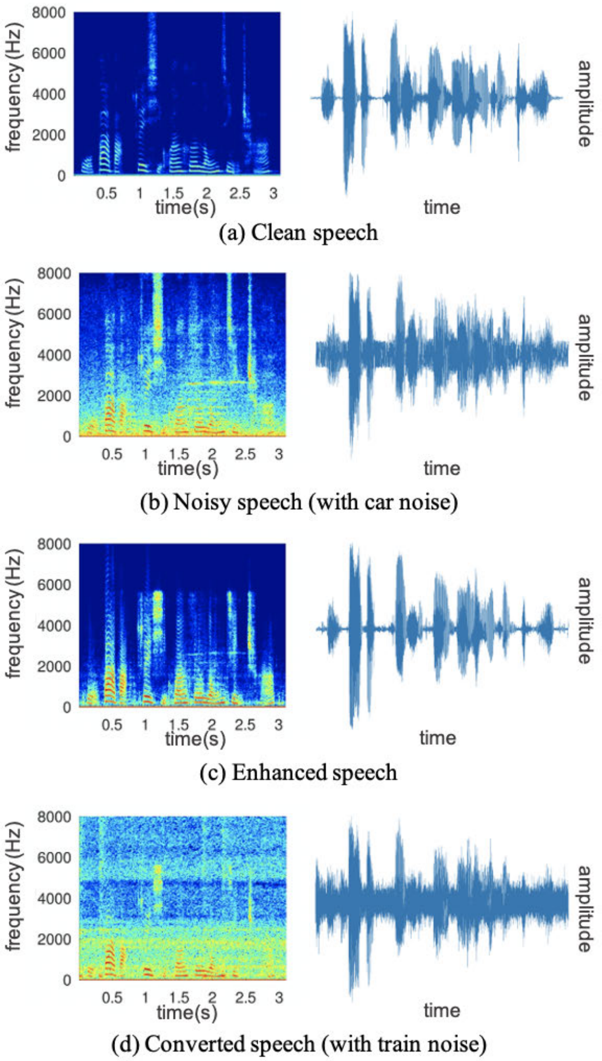


FIGURE 19. Visualization of the ASC embeddings. The original background noise of an enhanced speech signal was either "take-off" or "train." The embeddings of clean speech signals with new noise were close to those of enhanced speech signals with the same converted noise, which indicates that the BNC function might be used as a data augmentation method for the ASC model when a clean speech signal is unavailable.

5) VISUALIZATION RESULTS

Finally, we present the visualization results shown in Fig. 20. Figs. 20 (a), (b), (c), and (d) depict the spectrogram and waveform plots of the clean, noisy, enhanced, and BNC speech signals, respectively. For each sub-figure in Fig. 20, the left column depicts the spectrogram, and the right side depicts the associated waveform. Noisy speech signal (b) was produced by contaminating clean speech signal with car noise. Additionally, the BNC speech signal (d), which was produced by mixing the enhanced speech signal (c) with train noise, demonstrates the converted result from car noise to train noise.

The enhanced spectrogram shown in Fig. 20 (c) preserves several harmonic clean speech structures when compared with those presented in Figs. 20 (a). In addition, when comparing the waveforms in Figs. 20 (a), (b), and (c), the enhanced waveform presented in Fig. 20 (c) depicts considerably smaller noise components. Both observations demonstrate the effectiveness of SE in reducing noise from noisy input while providing detailed speech structures. The spectrogram shown in Fig. 20 (d) clearly illustrates different noise patterns in comparison with those presented in Fig. 20(b) and confirms the effectiveness of BNC.



(a) Clean speech

(b) Noisy speech (with car noise)

(c) Enhanced speech

(d) Converted speech (with train noise)

FIGURE 20. **CITISEN processed speech signals: (a) clean speech signal, (b) noisy speech signal (with car noise), (c) enhanced speech signal, and (d) converted speech signal (from car noise to train noise). For each sub-figure, the left and right columns show the spectrogram and waveform, respectively.**

V. CONCLUSION

In this study, we presented a speech signal processing mobile application called CITISEN. The contributions of CITISEN are as follows: (1) CITISEN was developed as a standardized SE tool with a user interface for performing SE on a prerecording or instant recording. In addition, experimental results confirmed the SE function of providing improved STOI and PESQ scores. (2) CITISEN has an MA function that allows users to adapt the SE models in terms of personalized testing conditions, and the MA function was proven to provide notable STOI and PESQ improvements as compared to the results without MA. (3) CITISEN provides a BNC function that converts the background noise of a speech signal into another noise. Notably, the BNC function is a novel concept for SE techniques and was implemented in mobile devices for the first time. The listening test results indicated that the BNC function could convert the background noise while maintaining the clarity and intelligibility of the converted speech signals. In addition, machine evaluation experiments showed that the ASC embeddings of clean speech signals with a new noise were close to those of enhanced speech signals with the same converted noise. Therefore, the BNC function can serve as a data augmentation method for the ASC model in the condition that clean speech signals are unavailable. (4) By simply replacing the settings with the associated model, CITISEN can run with other SE models that were not tested in this study. Therefore, CITISEN provides a suitable platform for evaluating deep-learning-based SE models and effectively reduces the development interval for converting deep-learning models to industrial applications.

REFERENCES

[1] A. Varga and H. J. M. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, no. 3, pp. 247–251, Jul. 1993.

[2] A. L. Giraud, S. Garnier, C. Micheyl, G. Lina, A. Chays, and S. Chéry-Croze, "Auditory efferents involved in speech-in-noise intelligibility," *NeuroReport*, vol. 8, no. 7, pp. 1779–1783, May 1997.

[3] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust.*, Oct. 2013, pp. 1–4.

[4] R. Beutelmann and T. Brand, "Prediction of speech intelligibility in spatial noise and reverberation for normal-hearing and hearing-impaired listeners," *J. Acoust. Soc. Amer.*, vol. 120, no. 1, pp. 331–342, Jul. 2006.

[5] H. J. M. Steeneken and T. Houtgast, "A physical method for measuring speech-transmission quality," *J. Acoust. Soc. Amer.*, vol. 67, no. 1, pp. 318–326, Jan. 1980.

[6] A. Sankar and C.-H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. Speech Audio Process.*, vol. 4, no. 3, pp. 190–202, May 1996.

[7] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2013.

[8] J. Benesty, S. Makino, and J. Chen, *Speech Enhancement*. Berlin, Germany: Springer, 2005.

[9] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust Automatic Speech Recognition: A Bridge to Practical Applications*. New York, NY, USA: Academic, 2015.

[10] A. El-Solh, A. Cuhadar, and R. A. Goubran, "Evaluation of speech enhancement techniques for speaker identification in noisy environments," in *Proc. ISMW*, Dec. 2007, pp. 235–239.

[11] Y.-H. Chin, J.-C. Wang, C.-L. Huang, K.-Y. Wang, and C.-H. Wu, "Speaker identification using discriminative features and sparse representation," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 8, pp. 1979–1987, Aug. 2017.

[12] J. Li, L. Yang, J. Zhang, Y. Yan, Y. Hu, M. Akagi, and P. C. Loizou, "Comparative intelligibility investigation of single-channel noise-reduction algorithms for Chinese, Japanese, and English," *J. Acoust. Soc. Amer.*, vol. 129, no. 5, pp. 3291–3301, May 2011.

[13] H. Levit, "Noise reduction in hearing aids: An overview," *J. Rehabil. Res. Develop.*, vol. 38, no. 1, pp. 111–121, 2001.

[14] T. Venema, *Compression for Clinicians*. Clifton Park, NY, USA: Thomson DelMar, 2006, ch. 7.

[15] Y. H. Lai, F. Chen, S.-S. Wang, X. Lu, Y. Tsao, and C.-H. Lee, "A deep denoising autoencoder approach to improving the intelligibility of vocoded speech in cochlear implant simulation," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 7, pp. 1568–1578, 2016.

[16] F. Chen, Y. Hu, and M. Yuan, "Evaluation of noise reduction methods for sentence recognition by Mandarin-speaking cochlear implant listeners," *Ear Hearing*, vol. 36, no. 1, pp. 61–71, 2015.

[17] P. Scalart and J. V. Filho, "Speech enhancement based on a priori signal to noise estimation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Conf.*, May 1996, pp. 629–632.

[18] E. Hänsler and G. Schmidt, *Topics in Acoustic Echo and Noise Control: Selected Methods for the Cancellation of Acoustical Echoes*. Berlin, Germany: Springer, 2006.

[19] J. Chen, J. Benesty, Y. A. Huang, and E. J. Diethorn, *Springer Handbook of Speech Processing*. Berlin, Germany: Springer, 2008, pp. 843–872.

[20] J. Li, S. Sakamoto, S. Hongo, M. Akagi, and Y. Suzuki, "Two-stage binaural speech enhancement with Wiener filter for high-quality speech communication," *Speech Commun.*, vol. 53, no. 5, pp. 677–689, May 2011.

[21] R. Frazier, S. Samsam, L. Braida, and A. Oppenheim, "Enhancement of speech by adaptive filtering," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 1976, pp. 251–253.

[22] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech Signal Process.*, vol. SSP-33, no. 2, pp. 443–445, Apr. 1985.

[23] R. McAulay and T. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. SSP-34, no. 4, pp. 744–754, Aug. 1986.

[24] T. F. Quatieri and R. J. McAulay, "Shape invariant time-scale and pitch modification of speech," *IEEE Trans. Signal Process.*, vol. 40, no. 3, pp. 497–510, Mar. 1992.

[25] U. Mittal and N. Phamdo, "Signal/noise KLT based approach for enhancing speech degraded by colored noise," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 2, pp. 159–167, Mar. 2000.

[26] Y. Ephraim, "Statistical-model-based speech enhancement systems," *Proc. IEEE*, vol. 80, no. 10, pp. 1526–1555, Oct. 1992.

[27] S. Suhadi, C. Last, and T. Fingscheidt, "A data-driven approach to a priori SNR estimation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 1, pp. 186–195, Jan. 2011.

[28] U. Kjems and J. Jensen, "Maximum likelihood based noise covariance matrix estimation for multi-microphone speech enhancement," in *Proc. EUSIPCO*, Aug. 2012, pp. 295–299.

[29] R. McAulay and M. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, no. 2, pp. 137–145, Apr. 1980.

[30] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, no. 4, pp. 561–580, Apr. 1975.

[31] B. Atal and M. Schroeder, "Predictive coding of speech signals and subjective error criteria," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 3, pp. 247–254, Jun. 1979.

[32] W. A. Jassim, R. Paramesran, and M. S. A. Zilany, "Enhancing noisy speech signals using orthogonal moments," *IET Signal Process.*, vol. 8, no. 8, pp. 891–905, 2014.

[33] T. Lotter and P. Vary, "Speech enhancement by MAP spectral amplitude estimation using a super-Gaussian speech model," *EURASIP J. Adv. Signal Process.*, vol. 2005, no. 7, pp. 1–17, Dec. 2005.

[34] X. Zou and X. Zhang, "Speech enhancement using an MMSE short time DCT coefficients estimator with supergaussian speech modeling," *J. Electron.*, vol. 24, no. 3, pp. 332–337, May 2007.

[35] B. M. Mahmmod, A. R. Ramli, T. Baker, F. Al-Obeidat, S. H. Abdulhussain, and W. A. Jassim, "Speech enhancement algorithm based on super-Gaussian modeling and orthogonal polynomials," *IEEE Access*, vol. 7, pp. 103485–103504, 2019.

[36] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Mag.*, vol. ASSP-3, no. 1, pp. 4–16, Jan. 1986.

[37] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. NIPS*, 2000, pp. 556–562.

[38] K. W. Wilson, B. Raj, P. Smaragdis, and A. Divakaran, "Speech denoising using nonnegative matrix factorization with priors," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2008, pp. 4029–4032.

[39] N. Mohammadiha, P. Smaragdis, and A. Leijon, "Supervised and unsupervised speech enhancement using nonnegative matrix factorization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 10, pp. 2140–2151, Oct. 2013.

[40] J.-C. Wang, Y.-S. Lee, C.-H. Lin, S.-F. Wang, C.-H. Shih, and C.-H. Wu, "Compressive sensing-based speech enhancement," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 2122–2131, Nov. 2016.

[41] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 1, pp. 1–37, 2011.

[42] S. Tamura, "An analysis of a noise reduction neural network," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, May 1989, pp. 2001–2004.

[43] F. Xie and D. Van Compernolle, "A family of MLP based nonlinear spectral estimators for noise reduction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 1994, p. 53.

[44] E. A. Wan and A. T. Nelson, "Networks for speech enhancement," *Handbook of Neural Networks for Speech Processing*, vol. 139. Boston, MA, USA: Artech House, 1999, p. 1.

[45] J. Tchorz and B. Kollmeier, "SNR estimation based on amplitude modulation analysis with applications to noise suppression," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 3, pp. 184–192, May 2003.

[46] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 1–38, 2010.

[47] Y. Wang and D. Wang, "Towards scaling up classification-based speech separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[48] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder," in *Proc. Interspeech*, Aug. 2013.

[49] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[50] M. Wollmer, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, "Feature enhancement by bidirectional LSTM networks for conversational speech recognition in highly non-stationary noise," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6822–6826.

[51] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. Schuller, "Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR," in *Proc. LVA/ICA*, 2015, pp. 91–99.

[52] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Dec. 2017, pp. 006–012.

[53] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 9, pp. 1570–1584, Sep. 2018.

[54] K. Tan and D. Wang, "A convolutional recurrent neural network for real-time speech enhancement," in *Proc. INTERSPEECH*, Sep. 2018, pp. 3229–3233.

[55] J. Qi, H. Hu, Y. Wang, C.-H.-H. Yang, S. M. Siniscalchi, and C.-H. Lee, "Exploring deep hybrid Tensor-to-Vector network architectures for regression-based speech enhancement," in *Proc. INTERSPEECH*, Oct. 2020, pp. 76–80.

[56] Z.-X. Li, L.-R. Dai, Y. Song, and I. McLoughlin, "A conditional generative model for speech enhancement," *Circuits, Syst., Signal Process.*, vol. 37, no. 11, pp. 5005–5022, Nov. 2018.

[57] H. Phan, I. V. McLoughlin, L. Pham, O. Y. Chến, P. Koch, M. De Vos, and A. Mertins, "Improving GANs for speech enhancement," *IEEE Signal Process. Lett.*, vol. 27, pp. 1700–1704, 2020.

[58] S. Pascual, A. Bonafonte, and J. Serra, "SEGAN: Speech enhancement generative adversarial network," in *Proc. INTERSPEECH*, Aug. 2017, pp. 3642–3646.

[59] F. Yang, Z. Wang, J. Li, R. Xia, and Y. Yan, "Improving generative adversarial networks for speech enhancement through regularization of latent representations," *Speech Commun.*, vol. 118, pp. 1–9, Apr. 2020.

[60] C. Tang, C. Luo, Z. Zhao, W. Xie, and W. Zeng, "Joint time-frequency and time domain learning for speech enhancement," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 3816–3822.

[61] H. Li and J. Yamagishi, "Noise tokens: Learning neural noise templates for environment-aware speech enhancement," in *Proc. INTERSPEECH*, Oct. 2020, pp. 2452-2456.

[62] J. Kim, M. El-Khamy, and J. Lee, "T-GSA: Transformer with Gaussian-weighted self-attention for speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6649–6653.

[63] S. Pascual, M. Park, J. Serra, A. Bonafonte, and K.-H. Ahn, "Language and noise transfer in speech enhancement generative adversarial network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 5019–5023.

[64] S. Wang, W. Li, S. M. Siniscalchi, and C.-H. Lee, "A cross-task transfer learning approach to adapting deep speech enhancement models to unseen background noise using paired senone classifiers," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6219–6223.

[65] R. Liang, Z. Liang, J. Cheng, Y. Xie, and Q. Wang, "Transfer learning algorithm for enhancing the unlabeled speech," *IEEE Access*, vol. 8, pp. 13833–13844, 2020.

[66] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.

[67] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)—A new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, May 2001, pp. 749–752.

[68] Y. Tsao and Y.-H. Lai, "Generalized maximum a posteriori spectral amplitude estimation for speech enhancement," *Speech Commun.*, vol. 76, pp. 112–126, Feb. 2016.

[69] S. Chopra, S. Balakrishnan, and R. Gopalan, "Dlid: Deep learning for domain adaptation by interpolating between domains," in *Proc. ICML*, 2013, pp. 1–8.

[70] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. NIPS*, 2014, pp. 1–9.

[71] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, 2016.

[72] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. ICML*, vol. 70, 2017, pp. 1126–1135.

[73] C.-F. Liao, Y. Tsao, H.-Y. Lee, and H.-M. Wang, "Noise adaptive speech enhancement using domain adversarial training," in *Proc. INTERSPEECH*, Sep. 2019, pp. 3148–3152.

[74] R. E. Zezario, T. Hussain, X. Lu, H.-M. Wang, and Y. Tsao, "Self-supervised denoising autoencoder with linear regression decoder for speech enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 6669–6673.

[75] T. Fujimura, Y. Koizumi, K. Yatabe, and R. Miyazaki, "Noisy-target training: A training strategy for DNN-based speech enhancement without clean speech," in *Proc. 29th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2021, pp. 436–440.

[76] M. Seki, H. Fujiwara, and K. Sumi, "A robust background subtraction method for changing background," in *Proc. 5th IEEE Workshop Appl. Comput. Vis.*, Dec. 2000, pp. 207–213.

[77] M.-W. Huang, "Development of Taiwan Mandarin hearing in noise test," Dept. Speech Lang. Pathol. Audiol., Nat. Taipei Univ. Nursing Health Sci., Taipei, Taiwan, Tech. Rep., 2005.

[78] G. Hu and D. L. Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, pp. 2067–2079, 2010.

[79] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Sov. Phys.-Dokl.*, vol. 10, no. 8, pp. 707–710, 1966.

[80] A. Zhang. (2017). *Speech recognition (Version 3.8)*. Accessed: Nov. 1, 2021. [Online]. Available: https://github.com/Uberi/speech_recognition#readme.

[81] J. S. Chung, J. Huh, S. Mun, M. Lee, H.-S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Proc. INTERSPEECH*, Oct. 2020, pp. 2977–2981.

[82] S.-Y. Chuang, Y. Tsao, C.-C. Lo, and H.-M. Wang, "Lite audio-visual speech enhancement," in *Proc. INTERSPEECH*, Oct. 2020, pp. 1131–1135.

[83] G. H. Dunteman, *Principal Components Analysis*, no. 69. Newbury Park, CA, USA: Sage, 1989.

YU-WEN CHEN received the B.S. degree in electrical engineering from the National Cheng Kung University, Tainan, Taiwan, in 2017, and the M.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, in 2019. She is currently a Research Assistant at Academia Sinica, Taipei. Her research interests include speech processing, human–computer interaction, multimodal learning, and machine learning.

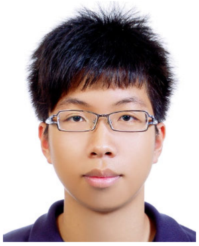
KUO-HSUAN HUNG received the B.S. degree from the National Chiao Tung University, in 2015, and the M.S. degree from the National Central University, Taiwan, in 2017. He is currently pursuing the Ph.D. degree with the Department of Biomedical Engineering, National Taiwan University. He is currently a Research Assistant with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. His research interests include bio-medical signal processing, noise reduction, speaker recognition, and deep learning.

YOU-JIN LI received the B.S. degree from the Department of Electronic Engineering, National Ilan University, Yilan, Taiwan, in 2014, and the M.S. degree from the Department of Electrical Engineering with Communications, National Ilan University, in 2016. He is currently pursuing the Ph.D. degree with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan. His research interests include signal processing, speech enhancement, beamforming, deep learning, and multi-channel compression.

ALEXANDER CHAO-FU KANG received the B.S. degree in electrical engineering from the University of California at Los Angeles, Los Angeles, CA, USA, in 2009, and the M.S. degree in telecommunication from the University of Maryland, College Park, MD, USA, in 2014. From 2014 to 2019, he was a Software Engineer in Silicon Valley, CA, USA, where he engaged in artificial intelligence related development. He is currently working as a Research Assistant at the Research Center for Information Technology Innovation, Academia Sinica, Taipei. His research interests include signal processing, speech enhancement, voice conversion, and deep learning.

YA-HSIN LAI received the Ph.D. degree in education from University of Bath, Bath, U.K., in 2020. From 2020 to 2021, she was a Postdoctoral Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan, where she engaged in a variety of research in automatic speech recognition, speech enhancement, and musical intervention for dyslexic children. She is currently an Assistant Professor at the Master Program of Youth and Child Welfare, Chinese Culture University, Taipei. Her research interests include psychometric instrument development and testing, parenting education, attachment relationships, and child and youth wellness.

KAI-CHUN LIU received the M.S. and Ph.D. degrees in biomedical engineering from the National Yang-Ming University, Taipei, Taiwan, in 2015 and 2019, respectively. He is currently a Postdoctoral Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei. His research interests include pervasive healthcare, wearable computing, machine learning, and bio-signal processing.



SZU-WEI FU received the Ph.D. degree from the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, in 2020. He is currently an Applied Scientist at Microsoft, Vancouver. His research interests include speech processing, speech enhancement, machine learning, and deep learning.

SYU-SIANG WANG received the B.S. degree from the Department of Electrical Engineering, National Changhua University of Education, Changhua, in 2008, the M.S. degree from the Department of Electrical Engineering, National Chi Nan University, in 2010, and the Ph.D. degree from the Graduate Institute of Communication Engineering, National Taiwan University, in 2018. He was a Research Assistant with Yu Tsao at the Research Center for Information Technology Innovation, Academia Sinica, where he was involved in robust speech feature extraction and speech enhancement. He is currently a Research Assistant with Yuan Ze University, Taiwan. His research interests include speech recognition, speech enhancement, audio coding, bio-signal processing, and deep neural networks.

YU TSAO (Senior Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1999 and 2001, respectively, and the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2008. From 2009 to 2011, he was a Researcher with the National Institute of Information and Communications Technology, Tokyo, Japan, where he engaged in research and product development in automatic speech recognition for multilingual speech-to-speech translation. He is currently a Research Fellow (Professor) and the Deputy Director with the Research Center for Information Technology Innovation, Academia Sinica, Taipei. He is also a Jointly Appointed Professor with the Department of Electrical Engineering, Chung Yuan Christian University, Taoyuan, Taiwan. His research interests include assistive oral communication technologies, audio coding, and bio-signal processing. He was a recipient of the Academia Sinica Career Development Award, in 2017, the National Innovation Awards from 2018 to 2021, the Future Tech Breakthrough Award 2019, and the Outstanding Elite Award, Chung Hwa Rotary Educational Foundation from 2019 to 2020. He is the corresponding author of a paper that receives the 2021 IEEE Signal Processing Society (SPS), Young Author, Best Paper Award. He is currently an Associate Editor for the IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, and LANGUAGE PROCESSING and IEEE SIGNAL PROCESSING LETTERS.