IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 12, 2017, accepted October 5, 2017. Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.2766675

Experimental Study on Extreme Learning Machine Applications for Speech Enhancement

TASSADAQ HUSSAIN[1,2], SABATO MARCO SINISCALCHI[3,4], CHI-CHUN LEE[5], SYU-SIANG WANG[6], YU TSAO[6], (Member, IEEE), AND WEN-HUNG LIAO[2]

[1]Taiwan International Graduate Program, Social Network and Human Centered Computing Program, Institute of Information Science, Academia Sinica, Taipei 115, Taiwan
[2]Department of Computer Science, National Chengchi University, Taipei 11605, Taiwan
[3]Department of Computer Engineering, Kore University of Enna, 94100 Enna, Italy
[4]Department of Electrical Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
[5]Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan
[6]Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan

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

This work was supported by the National Science Council of Taiwan under Grant MOST104-2221-E-001-026-MY2.

ABSTRACT In wireless telephony and audio data mining applications, it is desirable that noise suppression can be made robust against changing noise conditions and operates in real time (or faster). The learning effectiveness and speed of artificial neural networks are therefore critical factors in applications for speech enhancement tasks. To address these issues, we present an extreme learning machine (ELM) framework, aimed at the effective and fast removal of background noise from a single-channel speech signal, based on a set of randomly chosen hidden units and analytically determined output weights. Because feature learning with shallow ELM may not be effective for natural signals, such as speech, even with a large number of hidden nodes, hierarchical ELM (H-ELM) architectures are deployed by leveraging sparse auto-encoders. In this manner, we not only keep all the advantages of deep models in approximating complicated functions and maintaining strong regression capabilities, but we also overcome the cumbersome and time-consuming features of both greedy layer-wise pre-training and back-propagation (BP)-based fine tuning schemes, which are typically adopted for training deep neural architectures. The proposed ELM framework was evaluated on the Aurora–4 speech database. The Aurora–4 task provides relatively limited training data, and test speech data corrupted with both additive noise and convolutive distortions for matched and mismatched channels and signal-to-noise ratio (SNR) conditions. In addition, the task includes a subset of testing data involving noise types and SNR levels that are not seen in the training data. The experimental results indicate that when the amount of training data is limited, both ELM- and H-ELM-based speech enhancement techniques consistently outperform the conventional BP-based shallow and deep learning algorithms, in terms of standardized objective evaluations, under various testing conditions.

INDEX TERMS Speech enhancement, artificial neural networks, extreme learning machine, hierarchical extreme learning machines.

I. INTRODUCTION

The goal of a speech enhancement algorithm is to ameliorate the intelligibility (the percentage of words correctly recognized by listeners and/or the quality and level of residual noise in that signal) of a corrupted signal in adverse conditions [1]. In the past several decades, the problem of speech enhancement has attracted considerable research interest [2], owing to the wide dissemination of voice-based solutions for real-world applications, such as automatic speech recognition [3]–[5], speaker recognition [6], [7], speech coding [8], hearing aids [9], [10], and cochlea implants [11], [12]. As new applications are deployed, the definition of speech enhancement has broadened to include not only the classical noise reduction problem, but also the signal separation and reverberation problems. In this work, we are concerned with the reduction of background (ambient) noise, which is generally broadband and non-stationary. In real-world applications, the level of background noise may significantly diminish the quality and intelligibility of a speech signal acquired by a microphone

to the point that it becomes useless for subsequent processing.

Several single-channel speech enhancement methods are available in the literature. However, the performance of speech enhancement in real acoustic environments is not always satisfactory, because improving intelligibility and quality concurrently is a challenging problem. A class of speech enhancement methods, termed spectral restoration, aims to design a filter or transformation that attenuates the noise components to generate clean speech. Notable techniques include the Wiener filter and its extensions [13]–[15], the minimum mean square error spectral estimator (MMSE) [16]–[18], the maximum a posteriori spectral amplitude estimator (MAPA) [19], [20], the maximum likelihood spectral amplitude estimator (MLSA) [21], [22], and generalized MAPA [23]. Another popular class of speech enhancement methods adopts speech models for speech enhancement. Notable examples include the harmonic model [24], the linear prediction (LP) model [25], [26], and the hidden Markov model (HMM) [27]. A common limitation of most of these conventional methods is that they rely on either the additive nature of the background noise, or the statistical properties of speech and noise signals. As a consequence, these methods fail to properly contrast the non-stationary noise of real-world scenarios in unexpected acoustic conditions.

Rather than assuming an explicit model, methods based on non-linear mapping have also been adopted to address noise reduction tasks. In such an approach, stereo training data is generally used to learn a non-linear mapping function between noisy and clean speech. In the non-linear mapping category, artificial neural networks (ANN) have been shown to be a viable solution to effectively address background noise issues [28], [29]. For example, in [30] a single-hidden-layer with 160 neurons was employed to estimate the instantaneous signal-to-noise ratio (SNR) level of amplitude modulation spectrogram (AMS), and then the noise was suppressed according to the estimated SNRs of different channels. Alternatively, in [31]–[33] shallow ANNs were used to determine a mapping between the noisy and clean speech signals. Unfortunately, a lack of depth hindered a comprehensive exploitation of the relationships between noisy and clean speeches. By leveraging a greedy layer-wise unsupervised learning algorithm [34], often referred to as pre-training [35], the training of deep neural networks (DNNs) can now be successfully designed, and the strong regression capabilities of deep models can be better explored. For example, deep/stacked denoising autoencoders were used to model the relationship between clean and noisy features in [36] and [37]. Deep recurrent neural networks and long-short term memory (LSTM) networks have also been adopted in feature enhancement [38], [39]. In [40], a deep belief network (DBN) with a restricted Boltzmann machine (RBM) was used to design a facial expression recognition (FER) system. Akhtar *et al.* [41] further exploited the performance of neural networks by generating a K-support norm-based noise model, to train neural networks for adversarial noise. Meanwhile, convolutional neural networks, which have a better capability of modeling local temporal-spectral structures of speech signals, have been adopted as a fundamental model for the speech enhancement task in [42], and a deeper structure of convolutional neural network (DCNN) was used for hand gesture recognition in [43]. A common issue with ANN-based speech enhancers is the degraded performance in the presence of unexpected noise. A simple, yet effective solution to this problem is to cover many different types of noise in the training set, as proposed in [44]. In addition to ANN, a generalized single hidden layer feedforward network (GSLFN) [45] has been proposed for regression problems in which the traditional single layer feedforward network (SLFN) is extended by exploiting the polynomial functions of inputs as output weights. In [46], the universal enhancing capabilities of deep models were more thoroughly investigated. In particular, the authors proposed a regression DNN-based speech enhancement framework via training a deep and wide neural network architecture using a large collection of heterogeneous training data with four noise types. Although DNNs can achieve outstanding noise reduction results, deep neural models have two notable limitations: (1) DNN considers a multilayer architecture as a whole that is initialized by a computation-heavy unsupervised initialization and fine-tuned by several passes of back-propagation (BP) based fine tuning in order to achieve reasonable learning capabilities - such a training scheme is cumbersome and time consuming; and (2) huge amounts of training data are needed to attain optimal performance [46], which may limit the deployment of DNN-based solutions in many real-world applications, especially when operated in wearable or mobile client sides.

In this work, we propose an alternative speech enhancement framework based on the unique and effective characteristics of the extreme learning machine (ELM) algorithm [47], namely extremely fast training, good generalization, and a universal approximation/classification capability. ELMs can play a key role in many machine learning applications, such as traffic sign recognition [48], gesture recognition [49], video tracking [49], object classification [50], data representation in big data [51], water distribution and wastewater collection [52], opal grading [53], nonlinear time-series modeling [54] and adaptive dynamic programming [55]. In [56], the authors have also demonstrated that ELMs are suitable for a wide range of feature mapping applications, rather than just the classical ones. Moreover, to take advantage of multi-layer models, we deploy a speech enhancement algorithm with hierarchical ELMs (H-ELMs). To the best of our knowledge, this is the first work to apply ELM and H-ELM to the speech enhancement task. To evaluate the noise reduction capability of ELM and H-ELM, we conducted a series of experiments on the standardized Aurora–4 noisy speech corpus [57]. Notably, the amount of training data in the Aurora–4 speech corpus is relatively limited in comparison to that used in [46]. Aurora–4 also provides a subset of the test data that allows

an assessment in mismatch (SNR and channel) conditions. The contributions of our results are as follows: (i) We have demonstrated that ELMs are indeed a viable solution for extracting clean speech features from the noisy counterpart, and ELM-based speech enhancement is effective even when testing data involving noisy type and SNR levels that are not seen in the training data, and; (ii) when the amount of training data is limited, the proposed ELM speech enhancement algorithm outperforms the algorithms based on more conventional BP-based neural networks under different testing conditions, in terms of the perceptual evaluation of speech quality (PESQ, a standardized speech quality evaluation metric), and segmental signal to noise ratio improvement (SSNRI, a standardized objective speech quality evaluation metric).

The remainder of this paper is organized as follows. Section II introduces related work. Section III presents the ELM/H-ELM based speech enhancement algorithms. Section IV presents our experimental setup and results. The conclusions from this study are discussed in Section V.

II. RELATED WORK

In general, speech enhancement techniques can be categorized into two main groups, namely signal processing solutions and data-driven approaches. In the following sections, we discuss the underpinnings of both approaches by describing some prominent techniques in both groups. First, the speech enhancement problem will be introduced more formally through the spectral restoration method. Next, we will briefly discuss key data-driven methods.

A. CONVENTIONAL SPECTRAL RESTORATION METHODS

Speech enhancement algorithms involve a transformation of a noisy speech signal into the spectral domain to recover the desired clean signal. A noisy speech signal $y[n]$ is composed of a clean speech signal $x[n]$, and additive noise signal $v[n]$,

$$y[n] = x[n] + v[n], \tag{1}$$

where n is the time index. A noisy signal is converted into short time Fourier transform (STFT) domain to determine its frequency and phase components. In STFT, the speech signal is divided into short frames using a window function $w(n)$. The corresponding STFT speech signal can be expressed as

$$Y[m, l] = X[m, l] + V[m, l], \tag{2}$$

where $Y[m, l]$, $X[m, l]$, and $V[m, l]$ are the mth frequency bins of the noisy speech, clean speech, and noise spectra of the lth frame, respectively, corresponding to frequency ω_m, where $\omega_m = 2\pi m/M$, $m = 0, 1, \ldots, M-1$. The aim of noise reduction (NR) approaches is to restore $x[n]$ (or $X[m, l]$) from $y[n]$ (or $Y[m, l]$). For spectral restoration, a gain function $G[m, l]$ is estimated based on the computed a priori SNR statistic and *a posteriori* SNR statistic. The enhanced speech, $\widehat{X}[m, l]$, is obtained by filtering $Y[m, l]$ through $G[m, l]$. The phase of the noisy speech is copied and used to prepare the phase of the enhanced speech. An inverse STFT (ISTFT) is applied to convert $\widehat{X}[m, l]$, $m = 0, 1, \ldots, M-1$; $l = 1, 2, \ldots, L$ and the phase, to obtain the enhanced speech $\widehat{\boldsymbol{x}}$. Some of the notable techniques mentioned in the Section I, namely MMSE, MLSA, and MAPA, are based on this approach.

B. DATA DRIVEN METHODS

1) NONNEGATIVE MATRIX FACTORIZATION

In nonnegative matrix factorization (NMF) based speech enhancement, a speech data matrix $\boldsymbol{Y} \in \boldsymbol{R}^{M\times L}$ with M frequency bins and L speech frames is projected to a space that is a linear combination of a set of vectors, i.e., $\boldsymbol{Y} \approx \boldsymbol{WH}$, where $\boldsymbol{W} = [\boldsymbol{W}_X \boldsymbol{W}_V] \in \boldsymbol{R}^{M\times(p_x+p_v)}$ ($\boldsymbol{W}_X$ and $\boldsymbol{W}_V$ denote the basis matrices of speech and noise, respectively) and $\boldsymbol{H} = [\boldsymbol{H}_{\widehat{X}}^T \boldsymbol{H}_{\widehat{V}}^T]^T \in \boldsymbol{R}^{(p_x+p_v)\times L}$. Here, $p_x, p_y \leq min(M, L)$ are the numbers of encoding vectors for speech and noise ($\boldsymbol{H}_{\widehat{X}}$ and $\boldsymbol{H}_{\widehat{V}}$ denote the encoded coefficient matrices of speech and noise, respectively). NMF approximation is achieved by using two alternative minimizing criteria: (1) the least square criteria to minimize $\|\boldsymbol{V} - \boldsymbol{WH}\|^2$ w.r.t $\boldsymbol{W}$ and $\boldsymbol{H}$; and (2) the generalized Kullback-Leibler (KL) divergence to minimize $D(\boldsymbol{V}\|\boldsymbol{WH})$ [58], [59].

During the speech enhancement training stage, NMF is applied separately on clean and noisy data, in which magnitude spectrums of the speech ($|X[m, l]|$) and noise ($|V[m, l]|$) are computed. Subsequently, the Euclidean distance between the magnitude spectrum and the factored matrices is minimized by the following update rule [58]:

$$\begin{aligned} \boldsymbol{H} &\leftarrow \boldsymbol{H} \otimes \frac{\boldsymbol{W}^T\boldsymbol{Y}}{\boldsymbol{W}^T\boldsymbol{WH}} \\ \boldsymbol{W} &\leftarrow \boldsymbol{W} \otimes \frac{\boldsymbol{YH}^T}{\boldsymbol{WH}H^T} \end{aligned} \tag{3}$$

In the enhancement stage, a spectral gain is estimated and the enhanced speech is obtained as

$$\widehat{X}[m, l] = G[m, l]Y[m, l] \tag{4}$$

where the gain function $G[m, l]$ is formulated using a specific statistical model and optimality criterion.

2) DEEP DENOISING AUTOENCODER (DDAE)

Recently, deep denoising autoencoders (DDAEs) have demonstrated a tremendous performance in the field of speech enhancement. DDAE is trained as a noisy-clean pair to learn the statistical information between the clean and noisy speech signals [60]. The aim of DDAE is to transform the noisy speech signal to a clean speech by minimizing the reconstruction error between the predicted signal $\widehat{\boldsymbol{X}}$ and the reference clean signal $\boldsymbol{X}$, such that

$$\theta^* = \arg\min_{\theta}(E(\theta) + \rho C(\theta)) \tag{5}$$

with

$$E(\theta) = \|\phi(\boldsymbol{Y}) - \boldsymbol{X}\|_F^2 \tag{6}$$

where ρ is a constant that controls the tradeoff between the reconstruction accuracy and regularization term $C(\theta)$ [37],

$\phi(\boldsymbol{Y})$ denotes the transformation function of DDAE. During the training phase, a DDAE is trained in a greedy layer-wise manner and then used to estimate clean speech given noisy speech signals as

$$\begin{aligned} h_1(\boldsymbol{Y}[l]) &= \boldsymbol{\sigma}(\boldsymbol{W}_1\boldsymbol{Y}[l] + \boldsymbol{b}_1), \\ &\vdots \\ h_{D-1}(\boldsymbol{Y}[l]) &= \boldsymbol{\sigma}(\boldsymbol{W}_{D-1}h_{D-2}(\boldsymbol{Y}[l]) + \boldsymbol{b}_{D-1}), \\ \widehat{\boldsymbol{X}}[l] &= \boldsymbol{W}_D h_{D-1}(\boldsymbol{Y}[l]) + \boldsymbol{b}_D \end{aligned} \tag{7}$$

$\boldsymbol{Y}[l] = [log(|Y[1,l]|) \dots log(|Y[m,l]|) \dots log(|Y[M,l]|)]^T$ and $\widehat{\boldsymbol{X}}[l] = [log(|\widehat{X}[1,l]|) \dots log(|\widehat{X}[m,l]|) \dots log(|\widehat{X}[M,l]|)]^T$ are the lth logarithm amplitude vectors of the input noisy speech and estimated clean speech, respectively; $\{\boldsymbol{W}_1 \dots \boldsymbol{W}_D\}$ are the weight matrices, $\{\boldsymbol{b}_1 \dots \boldsymbol{b}_D\}$ are the corresponding bias vectors, and $\widehat{X}[m,l]$ is the logarithmic amplitude vector of the enhanced speech. Furthermore, $\boldsymbol{\sigma}$ is the vector-wise non-linear activation function. The relationship in Eq.(5) can be optimized by using any unconstrained optimization algorithm. In particular, the Hessian-free algorithm was adopted in [61] to compute this. During the enhancement phase, the ISTFT is applied to the magnitude spectrum together with the phase spectrum from the original signal to reconstruct the waveform [46], [60]. The difference between DDAE [60] and DNN [46] lies in the initialization, where DDAE formulates the noise reduction (NR) task as an encoding-decoding process, and DNN considers it as a regression task.

III. EXTREME LEARNING MACHINE IN A NUTSHELL

A. THE ELM MODEL

The extreme learning machine (ELM) was proposed by Huang *et al.* [47] for single layer feed-forward networks (SLFNs), to overcome issues of the BP algorithm. ELM provides an efficient and quick learning process, which does not require the massive fine-tuning of parameters [56].

1) SHALLOW ELM

The input weights and biases of the hidden layer in SLFNs can be chosen randomly to learn N distinct observations [62]. Given N distinct observations $(\boldsymbol{y}_i, \boldsymbol{x}_i)$, where $\boldsymbol{y}_i = [y_{i1}, y_{i2} \dots y_{iJ}]^T \in \boldsymbol{R}^J$ and $\boldsymbol{x}_i = [x_{i1}, x_{i2} \dots x_{iI}]^T \in \boldsymbol{R}^I$, the outputs of the SLFNs can be modeled as

$$f(\boldsymbol{y}_i) = \sum_{q=1}^{Q} \boldsymbol{\beta}_q \sigma(\boldsymbol{w}_q \cdot \boldsymbol{y}_i + b_q) \tag{8}$$

where $\sigma(\cdot)$ is the activation function, $\boldsymbol{w}_q = [w_{1q}, w_{2q}, \dots, w_{Jq}]^T \in \boldsymbol{R}^J$ is the weight vector from the input nodes to the qth hidden node, b_q is the bias of the qth hidden node, $\boldsymbol{\beta}_q = [\beta_{q1}, \beta_{q2}, \dots, \beta_{qI}]^T \in \boldsymbol{R}^I$ is the weight vector from the qth hidden node to the output nodes, and Q is the number of hidden neurons. For the ith input vector, a standard SLFN aims to yield zero error, given as

$$\sum_{i=1}^{N} \|f(\boldsymbol{y}_i) - \boldsymbol{x}_i\| = 0 \tag{9}$$

The above relation can be shortened as

$$\boldsymbol{H}\boldsymbol{B} = \boldsymbol{X} \tag{10}$$

where

$$\boldsymbol{H} = \begin{bmatrix} \sigma(\boldsymbol{w}_1 \cdot \boldsymbol{y}_1 + b_1) & \cdots & \sigma(\boldsymbol{w}_Q \cdot \boldsymbol{y}_1 + b_Q) \\ \vdots & & \vdots \\ \sigma(\boldsymbol{w}_1 \cdot \boldsymbol{y}_N + b_1) & \cdots & \sigma(\boldsymbol{w}_Q \cdot \boldsymbol{y}_N + b_Q) \end{bmatrix}_{N \times Q},$$

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_Q^T \end{bmatrix}_{Q \times I}, \quad \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix}_{N \times I} \tag{10a}$$

The output weight matrix $\boldsymbol{B}$ is computed as

$$\boldsymbol{B} = \boldsymbol{H}^+\boldsymbol{X} \tag{11}$$

where $\boldsymbol{H}^+$ is the Moore-Penrose (MP) pseudoinverse of $\boldsymbol{H}$, which can be calculated using orthogonal projection methods such as $\boldsymbol{H}^+ = (\boldsymbol{H}^T\boldsymbol{H})^{-1}\boldsymbol{H}^T$, where $\boldsymbol{H}^T\boldsymbol{H}$ should be non-singular, or $\boldsymbol{H}^+ = \boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{H}^T)^{-1}$, where $\boldsymbol{H}\boldsymbol{H}^T$ should be non-singular.

In order to solve the linear inverse problem arising at the ELM output, in this study we adopted a fast-iterative shrinkage-threshold algorithm (FISTA) [63], which is an extension of the gradient algorithm, and offers better convergence properties for problems involving large amounts of data.

2) HIERARCHICAL ELM

Inspired by DNNs, where features are extracted using a multilayer framework with an unsupervised initialization, Tang *et al.* [49] extended ELM, and proposed H-ELM for multilayer perceptrons (MLPs). The overall structure of the H-ELM model is illustrated in Fig. 1. The H-ELM framework comprises two stages, i.e., unsupervised feature extraction and supervised feature regression. In unsupervised feature extraction, high level features are extracted using an ELM-based autoencoder by considering each layer as an autonomous layer. The input data is projected to ELM feature space for feature extraction, in order to make use of information from training data. The output of the unsupervised feature extraction stage can then be used as the input to the supervised ELM regression stage [49] for the final result, based on the learning from the two stages.

3) ELM AND H-ELM FOR SPEECH ENHANCEMENT

In this section, we describe the use of ELM and H-ELM for a regression model to perform speech enhancement. Fig. 2 illustrates the system architecture of the proposed ELM/H-ELM-based speech enhancement approach. The main concept is to use an ELM/H-ELM model to transform

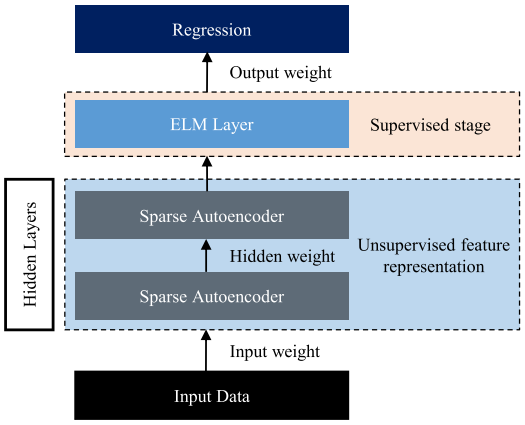

FIGURE 1. H-ELM architecture.

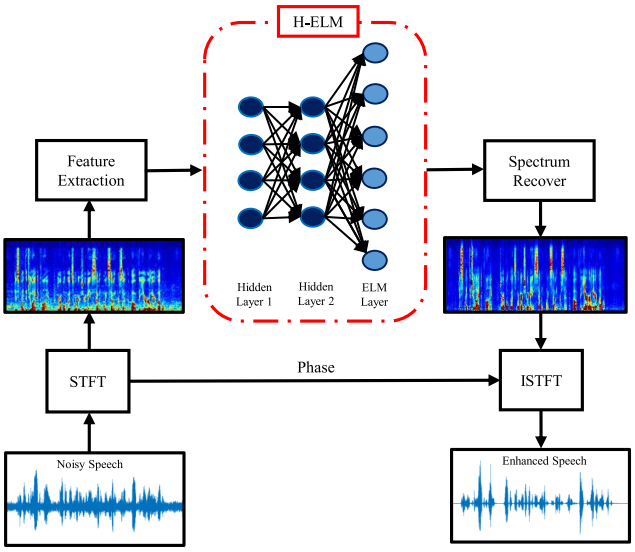

FIGURE 2. H-ELM-based speech enhancement architecture.

noisy speech to clean speech. The overall system includes offline and online stages.

During the offline stage, a set of noisy-clean speech pairs is prepared. The noisy and clean speech signals are first converted into the frequency domain using the STFT to determine the frequency and phase components of the signal. The logarithm power spectra (LPS) of the noisy and clean speech spectra are then placed at the input and output sides of the ELM model, respectively. More specifically, the goal of the ELM/H-ELM system is to reconstruct the clean speech signal from the noisy speech by minimizing the reconstruction error, such that

$$E = \|\boldsymbol{X} - \widehat{\boldsymbol{X}}\|_F^2 \tag{12}$$

where $\widehat{\boldsymbol{X}}$ is the estimated speech signal and $\boldsymbol{X}$ is the reference clean speech signal. According to ELM theory [56], any continuous target function can be approximated as $\sum_{l=1}^{N} \|f(\boldsymbol{Y}[l]) - \widehat{\boldsymbol{X}}[l]\| = 0$, where $\boldsymbol{Y}[l]$ and $\widehat{\boldsymbol{X}}[l]$ are the lth logarithm amplitude vectors of the input noisy speech and estimated clean speech described in Section II-B.2, respectively. The relationship in Eq.(8) can be written as

$$f(\boldsymbol{Y}[l]) = \sum_{q=1}^{Q} \boldsymbol{\beta}_q \sigma(\boldsymbol{w}_q \cdot \boldsymbol{Y}[l] + b_q) \tag{13}$$

where $\boldsymbol{w}_q$ is the weight vector, b_q is the bias and $\boldsymbol{\beta}_q$ is the output weight vector of the qth hidden node. The relation in Eq.(10) can be written compactly in matrix form as

$$\boldsymbol{H}\boldsymbol{B} = \widehat{\boldsymbol{X}} \tag{14}$$

where $\boldsymbol{H}$ is the hidden layer output, $\boldsymbol{B}$ is the output weight and $\widehat{\boldsymbol{X}}$ is the estimated speech signal, given as

$$\boldsymbol{H} = \begin{bmatrix} \sigma(\boldsymbol{w}_1 \cdot \boldsymbol{Y}[1]+b_1) & \cdots & \sigma(\boldsymbol{w}_Q \cdot \boldsymbol{Y}[1]+b_Q) \\ \vdots & & \vdots \\ \sigma(\boldsymbol{w}_1 \cdot \boldsymbol{Y}[N]+b_1) & \cdots & \sigma(\boldsymbol{w}_Q \cdot \boldsymbol{Y}[N]+b_Q) \end{bmatrix}_{N\times Q},$$

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_Q^T \end{bmatrix}_{Q\times M}, \quad \widehat{\boldsymbol{X}} = \begin{bmatrix} \widehat{\boldsymbol{X}}^T[1] \\ \vdots \\ \widehat{\boldsymbol{X}}^T[N] \end{bmatrix}_{N\times M} \tag{14a}$$

The corresponding output weight matrix for the estimated speech signal can be computed as

$$\widehat{\boldsymbol{B}} = \boldsymbol{H}^{+}\widehat{\boldsymbol{X}} \tag{15}$$

where $\boldsymbol{H}^{+}$ is the Moore-Penrose (MP) pseudoinverse of $\boldsymbol{H}$ and is described in Section III-A.1, $\widehat{\boldsymbol{B}}$ is the output weight matrix, and $\widehat{\boldsymbol{X}}$ is the estimated speech signal.

In the online stage, the noisy speech signals are first converted into LPS and phase parts. The noisy LPS features are transformed to obtain the enhanced ones by following the steps in Eqs. (13) and (14) for the ELM/H-ELM models ($\boldsymbol{H}$ and $\widehat{\boldsymbol{B}}$) estimated in the offline stage. The phase of the noisy speech is used to prepare the phase of the enhanced speech. An ISTFT is applied to obtain the enhanced speech signals.

IV. EXPERIMENTS

In this section, we present our experimental setup and results.

A. EXPERIMENTAL SETUP

1) AURORA-4 DESCRIPTION

The Aurora–4 [57] dataset was used to evaluate the performance of the proposed ELM-based speech enhancement algorithm. The Aurora–4 dataset includes speech data recorded at two sampling rates, 8 kHz and 16 kHz. The 16 kHz speech data was used in this study. Aurora–4 contains two training sets: clean and multi-condition. Each set contains 7138 utterances, as shown in Table 1. In this study, we employed these two training sets to train the speech enhancement models (input data from the multi-condition training set, output data from the clean training set). The multi-condition training set was divided into two blocks, each consisting of 3569 utterances, where 893 were clean and the remaining 2676 were randomly contaminated

TABLE 1. Aurora–4 Training set description.

Training Set	Category	Description	
Training Set 1	Clean data	Clean Speech with Sennhesier microphone (3569 utterances)	
Training Set 2	Multi-condition data	Speech recorded with Sennhesier microphone (3569 utterances)	No noise (893 utterances)
			Speech contaminated with 6 different noises at at 10-20dB SNRs (2676 utterances)
		Speech recorded with 18 different microphones (3569 utterances)	No noise (893 utterances)
			Speech contaminated with 6 different noises at at 10-20dB SNRs (2676 utterances)

TABLE 2. Aurora–4 Test set description.

Test Set	Description	Category
Test Set 1	Clean speech with Sennhesier microphone	Set A
Test Sets 2-7	Noisy speech containing 6 noises at 5-15dB SNRs with Sennhesier microphone	Set B
Test Set 8	Clean speech using different microphones	Set C
Test Sets 9-14	Noisy speech containing 6 noises at 5-15dB SNRs with different microphone	Set D

with six different background noises at SNR levels varying from 10 to 20 dB. The first block of data was recorded using a Sennheiser microphone, and the second block was recorded using various microphones (so that the speech in the dataset contained interferences with two different channel conditions).

The testing set includes 4620 utterances, which were divided into 14 testing sets, each containing 330 utterances. The entire set was used to test the performance [57] under different noise and channel conditions. The testing data includes six different noises, namely babble, car, restaurant, street, airport, and train, with both matched and mismatched channel conditions. The testing dataset was further classified into four larger groups as shown in Table 2. Because Test Set 1 (Set A) contained clean speech only, the corresponding evaluation scores (PESQ, SSNRI, speech distortion index (SDI), and short-time objective intelligibility (STOI)) are not included for comparison in the following discussion. From Table 2, it can be noted that Set B covered speech with additive noise, Set C covered speech with convolutive noise, and Set D contained speech with both additive and convolutive noises. Test Sets C and D contained clean and noisy test utterances with mismatch channel conditions (channel distortions). By analyzing Tables 1 and 2, we can confirm that both sets (training and testing) are corrupted with the same noise types but with different SNR conditions. Thus, we can consider the task used in this study to be a training-testing mismatched task.

2) EVALUATION METRICS

Experiments were carried out in controlled conditions for an unbiased evaluation of the performances of different configurations. Four standardized objective metrics, PESQ, SSNRI, SDI, and STOI, were adopted for evaluation.

Among the four objective metrics, PESQ was used to evaluate the quality of processed speech [64], with a score ranging from −0.5 to 4.5. The higher the PESQ score, the closer the enhanced speech is to the original clean speech. SSNRI measures the difference in the segmental SNR between the processed speech and the noisy speech [65]. A higher SSNRI indicates a more significant SNR improvement. SDI corresponds to the ratio of the energies of the residual speech and clean speech signals. A low value of SDI indicates a smaller distortion between the enhanced and clean speech signals. STOI computes the speech intelligibility in human listening tests [66]. A higher STOI value indicates better speech intelligibility, and the score ranges from 0 to 1. In the following discussion, the scores across the testing sets (the clean test set, Test Set 1 in Table 1, was excluded) of the Aurora–4 task are reported.

The speech signal was processed using a moving window, with a size of 10 ms and a step of 5 ms. Then, the Mel-frequency power spectrum (MFP) feature was calculated for each speech frame. In this study, we used an 80-dimensional MFP feature.

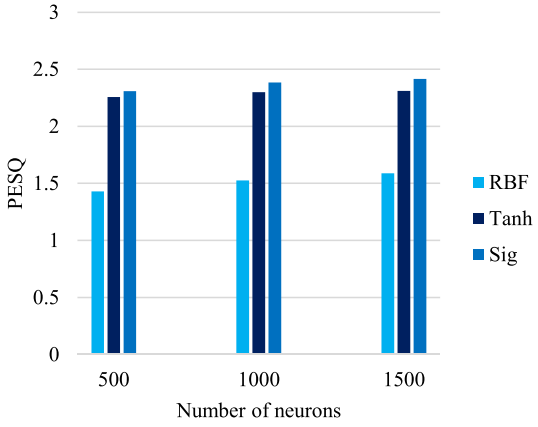

FIGURE 3. PESQ scores for ELM with different activation functions and numbers of hidden neurons.

B. EXPERIMENTAL RESULTS

1) ELM

In this section, the performance of ELM is investigated by varying the number of neurons (Q) in the hidden layer in Eq.(14) and the type of activation function. Fig. 3 shows the

TABLE 3. Single result abstracted from average objective evaluation scores of ELM [500] and H-ELM [200 200 500] configuration.

Test Set	ELM				H-ELM			
	PESQ	SDI	STOI	SSNRI	PESQ	SDI	STOI	SSNRI
Set B	2.4070	0.4240	0.8110	8.7280	2.5410	0.3820	0.8300	9.2360
Set C	2.6900	1.5690	0.7990	7.4570	2.8270	1.7220	0.8170	7.3940
Set D	2.2510	1.2700	0.7500	11.2810	2.3680	1.4600	0.7670	11.3750

PESQ scores for ELM using different activation functions, namely the sigmoid (Sig), hyperbolic tangent (Tanh), and radial basis function (RBF), with different numbers of neurons (Q = 500, 1000, and 1500). To assess which of these activation functions performs the best, we used the same set of training and test data. From Fig. 3, we note that the PESQ values for the above-mentioned activation functions monotonically increased with Q. These results demonstrate that the RBF, Tanh, and sigmoid functions all consistently returned performance improvements when the number of neurons was increased. Meanwhile, the sigmoid activation function achieves the best performance for different values of Q i.e., PESQ = 2.3570, 2.3847, and 2.4151, when compared with RBF (PESQ = 1.4293, 1.5261, and 1.5881) and Tanh (PESQ = 2.2563, 2.2998 and 2.3106). Thus, in the following experiments, the sigmoid function is used as the activation function for ELM.

2) H-ELM VERSUS ELM

When compared with ELM, H-ELM leverages hierarchical training to generate a sparse representation of the input data. Then, the standard ELM, which is on the top of the hierarchical structure of H-ELM, performs the regression. To closely study the performances of ELM and H-ELM, Table 3 presents the PESQ, SDI, and SSNRI scores of ELM and H-ELM for each test set. The learning accuracy of ELM/H-ELM is dependent on a user specified regularization parameter which needs to be selected carefully during experiments. In our experiments, we tried different values of regularization parameter to determine its impact on the performance. Here, we only reported the best regularization parameter (=200) in our speech enhancement task. As displayed in Table 2, Set B, Set C, and Set D contained speech utterances with only additive noise, only convolutive noise, and with both additive and convolutive noises, respectively. From Table 3, it can be noticed that ELM yielded higher PESQ and STOI values and lower SDI and SSNRI scores for Set B than those for Set D, because Set D includes additional convolutive distortions with a mismatch channel. As Set C did not contain additive noise, the PESQ score of Set C is higher than those of Set B and Set D. In general, the same trend could be observed in the H-ELM results, while the overall performance of H-ELM is consistently better than that of ELM (higher PESQ, STOI, and SSNRI scores) across Sets B and D. However, H-ELM attained a lower SDI score for additive noises (Set B) and higher SDI score for Set C and Set D when compared to ELM, because of the channel mismatch, which increases the distortion index for convolutive noises in Set C and Set D.

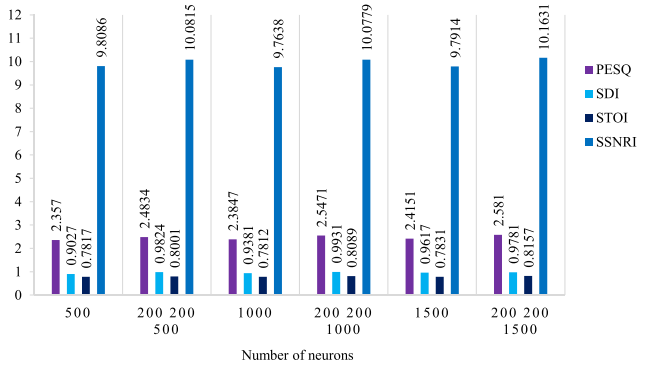

FIGURE 4. PESQ, SDI, STOI, and SSNRI average scores for ELM and H-ELM configurations.

Fig. 4 shows the average results for the 13 testing sets (Sets B, C, and D) across the four evaluation metrics, using different numbers of hidden neurons for the ELM ([500], [1000], [1500]) and H-ELM ([200 200 500], [200 200 1000], [200 200 1500]) configurations. For an impartial comparison with ELM, we used the same number of neurons in the regression stage (third layer) for H-ELM. Both ELM and H-ELM are tested against the same Aurora–4 testing dataset, using the sigmoid activation function. It can be seen that the H-ELM framework demonstrated significant improvements in terms of PESQ, STOI, and SSNRI, and maintained a stable performance against a higher number of neurons. However, for H-ELM the SDI score improved from 0.9824 to 0.9781 when the number of hidden neurons increased from [200 200 500] to [200 200 1500], whereas for ELM it jumped from 0.9027 to 0.9781 for an increase from [500] to [1500] hidden neurons. To determine the optimal size for the H-ELM hidden layers, we evaluated different configurations by changing the number of neurons in each layer. Experiments show that good results can be achieved by fixing the first two layers with the same number of hidden neurons and varying the number of hidden neurons in the third (ELM) layer. The configuration [200 200 X] was selected for H-ELM, where 'X' denotes the number of hidden neurons for the third layer, because compared to other configurations this achieved the best results with a low number of neurons during speech enhancement experiments.

It can be concluded by examining Table 3 and Fig. 4 that ELM provided less distortion for a low number of neurons,

but the distortion index deteriorated sharply as the number of neurons increased. However, the distortion index increased in H-ELM for a certain number of neurons, and then began to decrease.

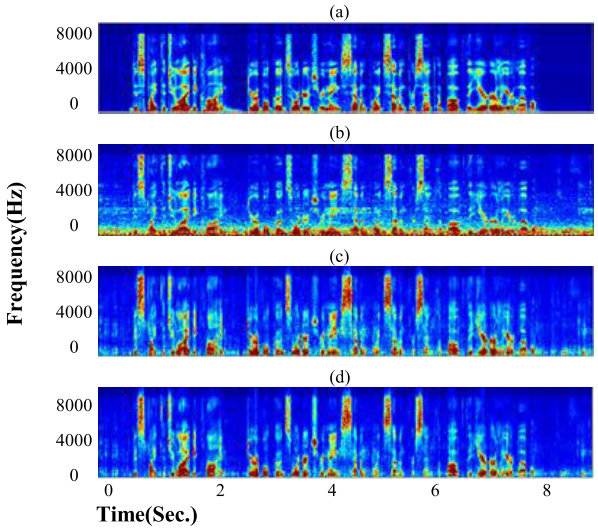

FIGURE 5. Spectrograms of an utterance (a) clean (PESQ = 4.6439), (b) noisy (PESQ = 2.2976), (c) ELM (PESQ = 2.3018), and (d) H-ELM (PESQ = 2.5489) contaminated with babble noise.

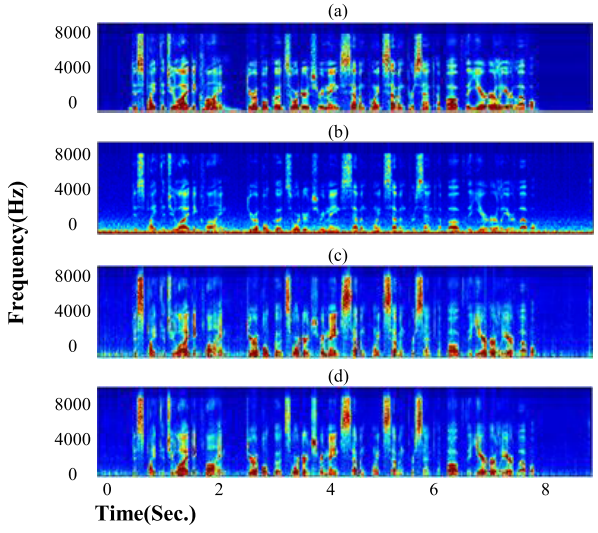

FIGURE 6. Spectrograms of an utterance (a) clean (PESQ = 4.6439), (b) noisy (PESQ = 2.4433), (c) ELM (PESQ = 2.5258), and (d) H-ELM (PESQ = 2.7345) contaminated with car noise.

3) SPECTROGRAM ANALYSIS

A spectrogram graphically represents the salient patterns of the speech signal and is used to analyze the signal over time at various frequencies. To visually compare the speech enhancement performances for both ELM and H-ELM, we plotted the spectrograms of the clean and noisy speech files for each enhanced speech signal. Fig. 5 and Fig. 6 present the spectrograms of the same utterance contaminated with two different noise types (babble and car, respectively) for Set D with mismatch channel conditions. Fig. 5(a) and (b) show the spectrograms of the clean and noisy speech signals, respectively, with babble noise. Fig. 5(c) and (d) present the enhanced speech signals for ELM and H-ELM using the [1500] and [200 200 1500] configurations, respectively. We can observe that both ELM and H-ELM successfully reduced the noise components, and H-ELM provides a better reconstructed speech signal than ELM. We also included the PESQ scores of the utterances in Fig. 5 and the scores show that H-ELM can more effectively improve speech quality than ELM. Moreover, Fig. 6(a) and (b) show the spectrogram plots for the corresponding clean and noisy speech, respectively, corrupted with car noise. Here, we note similar trends as that noted from Fig. 5. The H-ELM framework provides a higher PESQ (= 2.7345) than ELM (PESQ = 2.5258), where the noisy speech signal had (PESQ = 2.4433) and contained both additive and convolutive noises.

TABLE 4. Performance comparison of H-ELM frameworks using different window sizes.

ws	Framework	PESQ	SDI	STOI	SSNRI
1	[200 200 1500]	2.5810	**0.9780**	**0.8160**	10.1600
	[1000 1000 4000]	2.5669	1.1301	0.8105	10.0851
	[1000 1000 8000]	2.5938	1.1364	0.8116	10.0928
	[1000 1000 12000]	2.5979	1.1684	0.8124	10.0794
	[1000 1000 16000]	**2.6040**	1.1862	0.8126	**15.8031**
7	[200 200 1500]	2.6547	**1.0228**	0.8191	10.9900
	[1000 1000 4000]	2.7040	1.1405	0.8243	11.0340
	[1000 1000 8000]	2.7440	1.1499	0.8297	11.0527
	[1000 1000 12000]	2.7592	1.1450	0.8329	11.0753
	[1000 1000 16000]	**2.7698**	1.1576	**0.8345**	**15.7461**
11	[200 200 1500]	2.5880	**0.9930**	0.8130	11.1300
	[1000 1000 4000]	2.7060	1.1202	0.8250	11.2100
	[1000 1000 8000]	2.7310	1.1616	0.8292	11.2324
	[1000 1000 12000]	2.7585	1.1805	0.8320	11.2371
	[1000 1000 16000]	**2.7687**	1.1525	**0.8332**	**11.2656**

4) DEEPER AND WIDER H-ELM

In the previous sections, we have observed that H-ELM has superior capabilities as a regression model. Therefore, an H-ELM-based regression model is better suited for application to speech enhancement. To further scrutinize the H-ELM performance, we varied the size of the input speech vector by including more context at the input layer. In this manner, deeper H-ELM structures are introduced, and their performances measured. In particular, we considered the following four configurations: H-ELM1 with 6000 hidden neurons (hierarchical structure equal to [1000 1000 4000]), H-ELM2 with 10000 hidden neurons (hierarchical structure equal to [1000 1000 8000]), H-ELM3 with 14000 hidden neurons (hierarchical structure equal to [1000 1000 12000]), and H-ELM4 with 18000 hidden neurons (hierarchical structure equal to [1000 1000 16000]). Table 4 lists the resulting enhancements for the following five H-ELM configurations: H-ELM (hierarchical structure equal to [200 200 1500]),

H-ELM1, H-ELM2, H-ELM3, and H-ELM4, where the dimension of the input window size (i.e. *ws*) is changed from 80 to ($ws * 80$), in order to consider neighboring input speech vectors including left and right alongside the center speech vector. From Table 4, we can observe that H-ELM4 outperforms H-ELM, H-ELM1, H-ELM2, and H-ELM3 in terms of PESQ for a window size equal to 1 ($ws = 1$). However, H-ELM4 introduced more distortion (SDI = 1.1862) with less intelligibility (STOI = 0.8126) compared with the basic H-ELM configuration (H-ELM with configuration equal to [200 200 1500]) for a window size equal to 1. The table clearly illustrates a small improvement (overall improvement of 0.023) in the performance, with a PESQ increases from 2.5810 to 2.6040, for H-ELM configurations when the number of neurons increased from 1900 (H-ELM with configuration equal to [200 200 1500]) to 18000 (H-ELM4 with configuration equal to [1000 1000 16000]) in total, for a window size equal to 1. Moreover, the PESQ score for similar configurations escalated almost twofold, i.e. from 2.6547 to 2.7698 (overall improvement of 0.1151) when the *ws* increased from 1 to 7. Similarly, the performance further improved from 2.5880 to 2.7687 with an overall improvement of 0.1807, when the *ws* increased to 11. It is apparent from Table 4 that H-ELM demonstrated better speech enhancement capabilities when the size of the context window was increased. However, there was a sudden drop in the performances of the H-ELM frameworks when the input window size increased beyond 7, except for H-ELM1, where the PESQ enhances from 2.7040 to 2.7060. Although increasing the window size improved the intelligibility (STOI) and the SSNRI scores for the three H-ELM frameworks, it also introduced more distortion. The table tells us that the best results were achieved by H-ELM4 (configuration equal to [1000 1000 16000]) with an input window size of 7. It is worth mentioning that the deeper structures of H-ELM with a wider context window ($ws = 7$) proved to be more effective in terms of the speech quality (PESQ) and intelligibility (STOI) when comparing with an even larger context ($ws = 11$), which degraded the performance by considering irrelevant information.

5) H-ELM VERSUS DDAE

In this section, we compare H-ELM against a conventional deep denoising autoencoder (DDAE), where we have adopted a similar configuration to that reported in [60]. For deeper structures, the autoencoder is trained using clean and multi-condition data contaminated with six different background noises, as described in Section IV-A.1. We built four DDAE based speech enhancement systems, namely DDAE1, DDAE2, DDAE3 and DDAE4 with 3, 5, 7, and 9 nonlinear layers, respectively, each having 2048 hidden neurons. The deeper structures of DDAE were compared with our deeper H-ELM configurations. Namely, H-ELM1 was compared with DDAE1, that has a total of 6144 (= 2048*3) hidden neurons; H-ELM2 with 10000 hidden neurons was compared with DDAE2, which has 10240 (= 2048*5) hidden neurons; H-ELM3 with 14000 hidden neurons was compared with DDAE3, which has 14336 (= 2048*7); and H-ELM4 with 18000 hidden neurons was compared with DDAE4, which has 18432 (= 2048*9) hidden neurons. The learning rate during the training of the DDAE frameworks was set to 0.0002, with a batch size of 5000. The numbers of epochs for the four DDAE structures were set to 70. Table 5 lists the speech enhancement results for these deeper H-ELM and DDAE configurations with an input context window size equal to 7. This was selected because it gave the highest PESQ score (Section IV-B.4). By examining Table 5, we can confirm that H-ELM outperforms DDAE in terms of PESQ and SSNRI. However, H-ELM generated a higher distortion (SDI) with a low intelligibility (STOI) score compared with the DDAE frameworks. The table apparently demonstrates that the performance of H-ELM is consistent (increasing gradually) in terms of PESQ, STOI and SSNRI for higher number of neurons, while the DDAE performance showed inconsistency in terms of PESQ, SSNRI and SDI as more layers and neurons were introduced. That is, PESQ, SSNRI and SDI are degraded as the DDAE structure becomes larger. The table explicitly demonstrates the behavior of the DDAE structures by showing that adding more layers into the DDAE structures (DDAE3 and DDAE4) and injecting more neurons does not guarantee a good performance when the training data is limited, and a sufficient amount of data is necessary for DDAE structures to have a good generalization capability. On the other hand, the H-ELM structures proved to show a monotonically increasing performance for higher numbers of neurons.

TABLE 5. Objective evaluation scores of DDAE and H-ELM alongside traditional speech enhancement methods.

Method	PESQ	SDI	STOI	SSNRI
KLT	2.4907	1.2438	**0.8594**	9.5737
MMSE	2.5600	1.5212	0.8549	3.2246
RPCA	2.5615	1.8178	0.8426	1.6268
DDAE1	2.6767	1.0456	0.8293	10.6733
DDAE2	2.6783	1.0581	0.8330	10.6100
DDAE3	2.6731	**0.9686**	0.8385	10.5776
DDAE4	2.6664	1.0858	0.8401	10.3242
H-ELM1	2.7040	1.1405	0.8243	11.0340
H-ELM2	2.7440	1.1499	0.8297	11.0527
H-ELM3	2.7592	1.1450	0.8329	11.0753
H-ELM4	**2.7698**	1.1576	0.8345	**15.7461**

In addition, both learning algorithms are compared against three different classes of speech enhancement algorithms, i.e. a conventional spectral restoration approach in which we used an MMSE-based noise reduction technique [67], a subspace-based KLT [68] algorithm and noise reduction based on robust PCA (RPCA) [69], to verify the performances in speech enhancement tasks. It is evident that both learning algorithms have attained a significant improvement over the traditional methods, with improved PESQ, SDI, and

SSNRI scores. However, the intelligibility of the KLT is greater than for both of the above-mentioned learning algorithms. The results in Table 5 demonstrate that H-ELM with deeper configurations (H-ELM1, H-ELM2, H-ELM3, and H-ELM4) outscored the KLT, MMSE, RPCA, and DDAE methods with a reasonable margin. These results further confirm the advantages of H-ELM for achieving a satisfactory NR performance with relatively few training samples.

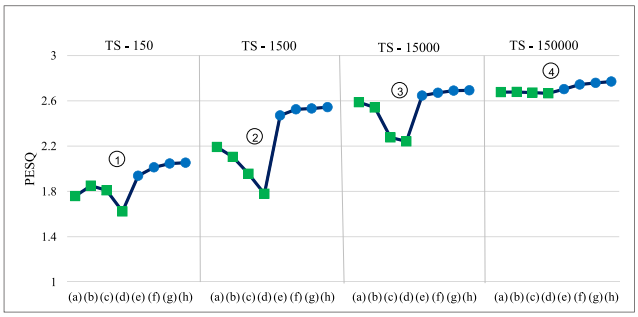

FIGURE 7. PESQ score for (a) DDAE1, (b) DDAE2, (c) DDAE3, (d) DDAE4, (e) H-ELM1, (f) H-ELM2, (g) H-ELM3 and (h) H-ELM4, using different amounts of training batch samples (TS).

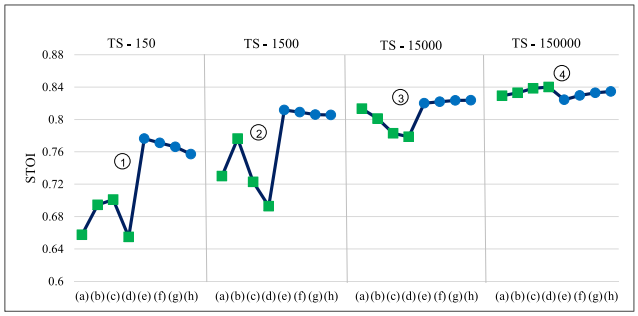

FIGURE 8. STOI score for (a) DDAE1, (b) DDAE2, (c) DDAE3, (d) DDAE4, (e) H-ELM1, (f) H-ELM2, (g) H-ELM3 and (h) H-ELM4, using different amounts of training batch samples (TS).

6) H-ELM SENSITIVITY TOWARDS THE TRAINING DATA

To analyze the sensitivity of the two learning algorithms (DDAE and H-ELM), we progressively decreased the sizes of the training batch samples (TS) in steps of 10%. Initially, we used 150000 MFP spectral patches of the training samples, which were reduced in 10% decrements to finally reach 150 MFP patches. The number of epochs was also reduced as the sizes of the training data were decreased. Initially, we used 70 epochs to train 150000 MFP DDAE frameworks, and which we then reduced the number of epochs to 40 epochs as the size of the training data was curtailed by 10% (i.e., 15000 MFP). We further reduced the epochs to 30 when the size of the training data was decreased to 1500 MFP and 150 MFP patches, respectively. The purpose of such an investigation is to evaluate the stability of each algorithm against the size of the training data. Fig. 7 and Fig. 8 present compact synopses of the two learning algorithms by means of PESQ and STOI, respectively, for $ws = 7$. Overall, there is a drop in the performance for both of the learning algorithms. However, the H-ELM frameworks provided a considerably substantial performance, even when the training samples reduced to 150 MFP patches in the end. On close examination, the graph in Fig. 7 shows an improvement in the performances of the DDAE frameworks when the size of the training samples (TS) was increased by 10%, from TS-150 to TS-1500. The PESQ score improved from 1.7588 to 2.1920 (from level ① to ②) for DDAE1, from 1.8507 to 2.1053 for DDAE2, from 1.8096 to 1.956 for DDAE3 and from 1.6227 to 1.7798 for DDAE4 when the TS is increased from 150 MFP patches to 1500 MFP patches. The same trend can be observed when the size of the training samples is increased from 15000 MFP patches to 150000 MFP patches (level ③ and level ④) for the DDAE frameworks. However, it can also be noted that the performances of DDAE3 and DDAE4 dropped rapidly as soon as the training data was reduced by 10% (from TS-150000 to TS-15000), i.e., from PESQ =2.6731 to 2.2777 for DDAE3 and from PESQ = 2.6664 to 2.2413 for DDAE4, which acutely describes the sensitiveness of deeper DDAE frameworks toward the training data. The performances for DDAE3 and DDAE4 degraded severely as the size of the training data was reduced by 20% (from TS-150000 to TS-1500).

On the other hand, H-ELM proved to be highly resilient against the reduction in the size of the training samples. The PESQ score for the H-ELM1 configuration escalated from 1.9377 to 2.4706 when the size of the training samples was only increased by just 10% (150 MFP to 1500 MFP patches), as shown in Fig. 7. Similarly, the PESQ score further improved from 2.6469 (MFP patches = 150000) to 2.7040 for the next increment in the size of the training samples (level ③ to level ④). Furthermore, the PESQ score for H-ELM2 increased from 2.0122 to 2.7440 when the size of the training samples was increased from 150 to 150000 MFP patches. The deeper structures of H-ELM (H-ELM3 and H-ELM4) provided a steady performance in terms of PESQ compared with the DDAE frameworks when the size of the training data was reduced. We also measured the effect of the reduction of training samples on the speech intelligibility, which measures the comprehensibility of the speech signal for the given conditions. Fig. 8 shows the intelligibility (STOI) of the test speech signals for each of the two learning algorithms with the limited training samples. The STOI score for the DDAE frameworks became very poor when the patches were reduced to 150 MFP. In contrast, H-ELM again proved to be very stable, even for a training sample size reduced to 150 MFP patches. The STOI for DDAE1 dropped from 0.8293 to 0.6575 (from level ④ to level ①), for DDAE2 the value decreased from 0.8330 to 0.6943, for DDAE3 it dropped from 0.8385 to 0.7009, and for DDAE4 it dropped from 0.8401 to 0.6547. However, for H-ELM the decrease was not so drastic. For H-ELM1, it declined from 0.8243 to 0.7764, for H-ELM2 it

declined from 0.8297 to 0.7710, for H-ELM3 it declined from 0.8329 to 0.7662 and for H-ELM4 it declined from 0.8345 to 0.7572.

Although both learning algorithms somehow maintained quality and intelligibility for the reduced training samples, DDAE, for which PESQ and STOI decreased most significantly compared with the H-ELM frameworks, revealed the sensitiveness of DDAE frameworks to the amount of the training samples.

V. CONCLUSION

The present study has introduced novel ELM/H-ELM-based speech enhancement methods, because we believe that the extreme learning machine framework offers a universal approximation capability through comparative measures. We carried out several experiments to investigate the optimal network structure. In addition, we used a hierarchical framework to ameliorate the ability of ELM by replacing the single layer with a multilayer model, whereby the distortion levels were appropriately controlled to provide a better generalization performance, alongside a desirable speech quality and speech intelligibility. To further verify the consistency, we compared the performance of H-ELM with DDAE and the traditional speech enhancement algorithms. Furthermore, acoustic context information was considered to analyze the performance against well-known learning algorithms. It turns out that H-ELM can yield comparable or even better results in terms of PESQ, STOI, SDI, and SSNRI compared with DDAE-based methods when the amount of training data is limited. To conclude, H-ELM is confirmed to be effective for speech enhancement tasks, especially when using limited training data.

Multi-task learning and transfer learning approaches have conventionally been adopted recently to improve the performances of deep learning models. Moving forward, we will adopt these two approaches in a new research study, to investigate the compatibility of H-ELM and achieve further improvements in the performance. Moreover, we intend to propose noise- and SNR-aware-based training criteria to effectively enhance the capabilities. Again, this could be another worthwhile future direction of research.

REFERENCES

[1] J. Benesty, S. Makino, and J. Chen, *Speech Enhancement*. New York, NY, USA: Springer, 2005.

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

[3] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust Automatic Speech Recognition: A Bridge to Practical Applications*. San Francisco, CA, USA: Academic, 2015.

[4] B. Li, Y. Tsao, and K. C. Sim, "An investigation of spectral restoration algorithms for deep neural networks based noise robust speech recognition," in *Proc. INTERSPEECH*, 2013, pp. 3002–3006.

[5] A. El-Solh, A. Cuhadar, and R. Goubran, "Evaluation of speech enhancement techniques for speaker identification in noisy environments," in *Proc. 9th IEEE Int. Symp. Multimedia Workshops (ISMW)*, 2013, pp. 235–239.

[6] J. Li *et al.*, "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, 2011.

[7] F. Yan, A. Men, B. Yang, and Z. Jiang, "An improved ranking-based feature enhancement approach for robust speaker recognition," *IEEE Access*, vol. 4, pp. 5258–5267, 2016.

[8] 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, 2011.

[9] T. Venema, *Compression for Clinicians*. Delmar, NY, USA: Delmar Pub., 2006.

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

[11] 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, Jul. 2017.

[12] 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.

[13] 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. (ICASSP)*, vol. 2. May 1996, pp. 629–632.

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

[15] J. Chen, J. Benesty, Y. Huang, and E. J. Diethorn, "Fundamentals of noise reduction," in *Springer Handbook of Speech Processing*. Berlin, Germany: Springer, 2008, pp. 843–872.

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

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

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

[19] 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.

[20] T. Lotter and P. Vary, "Speech enhancement by map spectral amplitude estimation using a super-Gaussian speech model," *EURASIP J. Appl. Signal Process.*, vol. 7, pp. 1110–1126, Jan. 2005.

[21] U. Kjems and J. Jensen, "Maximum likelihood based noise covariance matrix estimation for multi-microphone speech enhancement," in *Proc. 20th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2012, pp. 295–299.

[22] 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.

[23] Y.-C. Su, Y. Tsao, J.-E. Wu, and F.-R. Jean, "Speech enhancement using generalized maximum *a posteriori* spectral amplitude estimator," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 7467–7471.

[24] 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)*, vol. 1. Mar. 2013, pp. 251–253.

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

[26] 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.

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

[28] C.-T. Lin, "Single-channel speech enhancement in variable noise-level environment," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 33, no. 1, pp. 137–143, Jan. 2003.

[29] C. F. Stallmann and A. P. Engelbrecht, "Gramophone noise detection and reconstruction using time delay artificial neural networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 6, pp. 893–905, Jun. 2017.

[30] 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.

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

[32] 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)*, vol. 2. Apr. 1994, p. II-53.
[33] 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.
[34] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[35] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, Feb. 2010.
[36] B. Xia and C. Bao, "Speech enhancement with weighted denoising auto-encoder," in *Proc. INTERSPEECH*, 2013, pp. 3444–3448.
[37] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, no. 12, pp. 3371–3408, Dec. 2010.
[38] A. L. Maas, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. 13th Annu. Conf. Int. Speech Commun. Assoc.*, 2012, pp. 22–25.
[39] M. Wöllmer, 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. (ICASSP)*, May 2013, pp. 6822–6826.
[40] M. Z. Uddin, M. M. Hassan, A. Almogren, A. Alamri, M. Alrubaian, and G. Fortino, "Facial expression recognition utilizing local direction-based robust features and deep belief network," *IEEE Access*, vol. 5, pp. 4525–4536, 2017.
[41] S. W. Akhtar *et al.*, "Improving the robustness of neural networks using k-support norm based adversarial training," *IEEE Access*, vol. 4, pp. 9501–9511, 2016.
[42] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai. (2017). "Raw waveform-based speech enhancement by fully convolutional networks." [Online]. Available: https://arxiv.org/abs/1703.02205
[43] Y. Kim and B. Toomajian, "Hand gesture recognition using micro-doppler signatures with convolutional neural network," *IEEE Access*, vol. 4, pp. 7125–7130, 2016.
[44] 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.
[45] N. Wang, M. J. Er, and M. Han, "Generalized single-hidden layer feedforward networks for regression problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1161–1176, Jun. 2015.
[46] 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.
[47] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, nos. 1–3, pp. 489–501, 2006.
[48] Z. Huang, Y. Yu, J. Gu, and H. Liu, "An efficient method for traffic sign recognition based on extreme learning machine," *IEEE Trans. Cybern.*, vol. 47, no. 4, pp. 920–933, Apr. 2017.
[49] J. Tang, C. Deng, and G.-B. Huang, "Extreme learning machine for multilayer perceptron," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 4, pp. 809–821, Apr. 2016.
[50] F. Sun, C. Liu, W. Huang, and J. Zhang, "Object classification and grasp planning using visual and tactile sensing," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 7, pp. 969–979, Jul. 2016.
[51] L. L. C. Kasun, H. Zhou, G.-B. Huang, and C. M. Vong, "Representational learning with ELMs for big data," *IEEE Intell. Syst.*, vol. 28, no. 6, pp. 342013–342031, Dec. 2013.
[52] W. Zhao, T. H. Beach, and Y. Rezgui, "Optimization of potable water distribution and wastewater collection networks: A systematic review and future research directions," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 659–681, May 2016.
[53] D. Wang, L. Bischof, R. Lagerstrom, V. Hilsenstein, A. Hornabrook, and G. Hornabrook, "Automated opal grading by imaging and statistical learning," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 2, pp. 185–201, Feb. 2016.
[54] N. Wang, M. J. Er, and M. Han, "Parsimonious extreme learning machine using recursive orthogonal least squares," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 10, pp. 1828–1841, Oct. 2014.
[55] D. Liu, Q. Wei, and P. Yan, "Generalized policy iteration adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 12, pp. 1577–1591, Dec. 2015.
[56] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 513–529, Apr. 2012.
[57] N. Parihar, J. Picone, D. Pearce, and H.-G. Hirsch, "Performance analysis of the Aurora large vocabulary baseline system," in *Proc. 12th Eur. Signal Process. Conf.*, 2004, pp. 553–556.
[58] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 556–562.
[59] L. Finesso and P. Spreij, "Nonnegative matrix factorization and i-divergence alternating minimization," *Linear Algebra Appl.*, vol. 416, nos. 2–3, pp. 270–287, 2006.
[60] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder.," in *Proc. Interspeech*, 2013, pp. 436–440.
[61] J. Martens, "Deep learning via hessian-free optimization," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 735–742.
[62] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
[63] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
[64] *Perceptual Evaluation of Speech Quality (PESQ): An Objective Method for End-to-End speech Quality Assessment of Narrow-Band Telephone Networks and Speech Codecs*, document Rec. ITU-T P. 862, 2001.
[65] S. Quackenbush, T. Barnwell, and M. Clements, *Objective Measures of Speech Quality*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.
[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] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, no. 6, pp. 1109–1121, Dec. 1984.
[68] Y. Hu and P. C. Loizou, "A generalized subspace approach for enhancing speech corrupted by colored noise," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 4, pp. 334–341, Jul. 2003.
[69] C. Sun, Q. Zhang, J. Wang, and J. Xie, "Noise reduction based on robust principal component analysis," *J. Comput. Inf. Syst.*, vol. 10, no. 10, pp. 4403–4410, 2014.

TASSADAQ HUSSAIN received the B.S. degree in computer engineering from the COMSATS Institute of Information Technology, Islamabad, Pakistan, in 2006, and the M.S. degree in electrical engineering from the Blekinge Institute of Technology, Sweden, in 2009. He is currently pursuing the Ph.D. degree with the Taiwan International Graduate Program-Social Network and Human Centered Computing, Institute of Information Science, Academia Sinica, Taiwan, and the Department of Computer Science, National Chengchi University, Taipei, Taiwan. His research interests cover signal processing, speech and speaker recognition, and deep learning.

SABATO MARCO SINISCALCHI received the Laurea and Ph.D. degrees in computer engineering from the University of Palermo, Palermo, Italy, in 2001 and 2006, respectively. In 2001, he was with STMicroelectronics, where he designed optimization algorithms for processing digital image sequences on very long instruction word architectures. In 2002, he was an Adjunct Professor with the University of Palermo and taught several undergraduate courses for computer and telecommunication engineering. In 2006, he was a Post-Doctoral Fellow with the Center for Signal and Image Processing, Georgia Institute of Technology, Atlanta, GA, USA, under the guidance of Prof. C.-H. Lee. From 2007 to 2009, he was with the Norwegian University of Science and Technology, Trondheim, Norway, as a Research Scientist with the Department of Electronics and Telecommunications under the guidance of Prof. T. Svendsen. From 2010 to 2015, he was an Assistant Professor with the University of Enna Kore, Enna, Italy. He is currently an Associate Professor with the University of Enna Kore and affiliated with the Georgia Institute of Technology. His main research interests include speech processing, in particular automatic speech and speaker recognition, and language identification. He is currently an Associate Editor of the IEEE/ACM Transactions on Audio, Speech, and Language Processing.

CHI-CHUN LEE (M'13) received the B.S. degree (*magna cum laude*) and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, CA, USA, in 2007 and 2012, respectively. He was a Data Scientist with the Idea Laboratory, ID Analytics, Inc., in 2013. He is currently an Assistant Professor with the Electrical Engineering Department, National Tsing Hua University, Taiwan. His research interests are in interdisciplinary human-centered behavioral signal processing, emphasizing the development of computational frameworks in recognizing and quantifying human behavioral attributes, and interpersonal interaction dynamics using machine learning and signal processing techniques. He received the USC Annenberg Fellowship from 2007 to 2009. He had led a team to participate and win the Emotion Challenge Classifier Sub-Challenge in Interspeech 2009. He has co-authored and received the Best Paper Award in Interspeech 2010. He is a member of ISCA, Tau Beta Pi, Phi Kappa Phi, and Eta Kappa Nu. He has been a Reviewer for multiple internationally-renowned journals and technical conferences.

SYU-SIANG WANG received the B.S. degree in electrical engineering from the National Changhua University of Education, Changhua, Taiwan, in 2008, and the M.S. degree in electrical engineering from National Chi Nan University, Nantou, Taiwan, in 2010. He is currently pursuing the Ph.D. degree with the Graduate Institute of Communication Engineering, National Taiwan University. He is currently a Research Assistant with the Research Center for Information Technology Innovation, Academia Sinica, Taiwan. His research interests include signal processing, speech recognition, and deep learning.

YU TSAO (M'09) 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, Japan, where he was engaged in research and product development in automatic speech recognition for multilingual speech-to-speech translation. He is currently an Associate Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. His research interests include speech and speaker recognition, acoustic and language modeling, audio-coding, and bio-signal processing. He received the Academia Sinica Career Development Award in 2017.

WEN-HUNG LIAO received the M.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA, in 1991 and 1996, respectively. He joined National Chengchi University, Taiwan, in 2000, where he is currently an Associate Professor and the Chairperson with the Computer Science Department. His research interests include computer vision, pattern recognition, human computer interaction, and multimedia signal processing.