IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2021, accepted April 27, 2021, date of publication May 4, 2021, date of current version May 18, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3077281

DeepLPC-MHANet: Multi-Head Self-Attention for Augmented Kalman Filter-Based Speech Enhancement

SUJAN KUMAR ROY[1], **(Graduate Student Member, IEEE),**
AARON NICOLSON[2], **AND KULDIP K. PALIWAL**[1]
[1]Signal Processing Laboratory, Griffith University at Nathan, Brisbane, QLD 4111, Australia
[2]Australian e-Health Research Centre, Commonwealth Scientific and Industrial Research Organisation, Herston, QLD 4006, Australia
Corresponding author: Sujan Kumar Roy (sujankumar.roy@griffithuni.edu.au)

ABSTRACT Current augmented Kalman filter (AKF)-based speech enhancement algorithms utilise a temporal convolutional network (TCN) to estimate the clean speech and noise linear prediction coefficient (LPC). However, the multi-head attention network MHANet) has demonstrated the ability to more efficiently model the long-term dependencies of noisy speech than TCNs. Motivated by this, we investigate the MHANet for LPC estimation. We aim to produce clean speech and noise LPC parameters with the least bias to date. With this, we also aim to produce higher quality and more intelligible enhanced speech than any current KF or AKF-based SEA. To this end, we investigate MHANet within the DeepLPC framework. DeepLPC is a deep learning framework for jointly estimating the clean speech and noise LPC power spectra. DeepLPC is selected as it exhibits significantly less bias than other frameworks, by avoiding the use of whitening filters and post-processing. DeepLPC-MHANet is evaluated on the NOIZEUS corpus using subjective AB listening tests, as well as seven different objective measures (CSIG, CBAK, COVL, PESQ, STOI, SegSNR, and SI-SDR). DeepLPC-MHANet is compared to five existing deep learning-based methods. Compared to other deep learning approaches, DeepLPC-MHANet produced clean speech LPC estimates with the least amount of bias. DeepLPC-MHANet-AKF also produced higher objective scores than any of the competing methods (with an improvement of 0.17 for CSIG, 0.15 for CBAK, 0.19 for COVL, 0.24 for PESQ, 3.70% for STOI, 1.03 dB for SegSNR, and 1.04 dB for SI-SDR over the next best method). The enhanced speech produced by DeepLPC-MHANet-AKF was also the most preferred amongst ten listeners. By producing LPC estimates with the least amount of bias to date, DeepLPC-MHANet enables the AKF to produce enhanced speech at a higher quality and intelligibility than any previous KF or AKF-based method.

INDEX TERMS Speech enhancement, Kalman filter, augmented Kalman filter, LPC, temporal convolutional network, multi-head attention network.

I. INTRODUCTION

Speech corrupted by background noise (or noisy speech) can reduce the efficiency of communication between speaker and listener. A speech enhancement algorithm (SEA) can be used to suppress the embedded background noise and increase the quality and intelligibility of noisy speech [1]. SEAs are useful in many applications where noisy speech is undesirable and unavoidable. For example, speech communication systems, hearing aid devices, and speech recognition systems typically rely upon SEAs for robustness. Various SEAs, namely spectral subtraction (SS) [2]–[4], Wiener filter (WF) [5], minimum mean square error (MMSE) [6]–[8], Kalman filter (KF) [9], augmented KF (AKF) [10], computational auditory scene analysis (CASA) [11], and deep learning-based [12] have been introduced over the decades. This paper focuses on the AKF constructed from parameters estimated using deep learning.

The KF is an unbiased linear MMSE estimator, which was first introduced as a SEA by Paliwal and Basu [9]. In this

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

seminal work, each frame of the uncorrupted speech signal (i.e., clean speech) is represented by an auto-regressive (AR) process, whose parameters include the linear prediction coefficients (LPCs) and the prediction error variance. The LPC parameters as well as the additive noise variance are intrinsic to the KF recursive equations. For simplicity, the background noise was assumed to be stationary and white. Given a frame of noisy speech samples, the recursive equations of the KF estimate the clean speech samples. As demonstrated by Paliwal and Basu, it is difficult to accurately estimate the clean speech LPC parameters and additive noise variance in practice, with poor estimates resulting in enhanced speech with low quality and intelligibility.

In [10], Gibson *et al.* introduced the augmented KF (AKF) for speech enhancement in coloured noise conditions. For the AKF, both the clean speech and additive background noise are represented by AR processes. The clean speech and noise LPC parameters form an augmented matrix, which is used to construct the recursive equations of the AKF. In [10], the AKF processes the noisy speech iteratively (usually three to four iterations) to suppress the coloured background noise, yielding the enhanced speech. During this, the clean speech and noise LPC parameters for the current frame are estimated from the corresponding filtered speech frame of the previous iteration. Although this method demonstrated the ability to improve the signal-to-noise ratio (SNR) of noisy speech, the resultant enhanced speech suffered from *musical noise* and *speech distortion*. This is because the AKF is not robust to inaccurate LPC estimates [13], [14].

In [15], Roy *et al.* introduced a sub-band (SB) iterative KF (SBIT-KF) for SEA. With the assumption that the impact of noise in low-frequency SBs is negligible, SBIT-KF enhances only the high-frequency sub-bands (SBs) of the noisy speech using two KF iterations. However, low-frequency SBs can also be impacted by noise—typically when operating in real-life noise conditions. Moreover, the iterative processing employed by SBIT-KF produces *speech distortion* [10]. George *et al.* used a robustness metric to tune the AKF for coloured noise [13]. The authors demonstrated that inaccurate estimates of the clean speech and noise LPC parameters introduce bias in the AKF gain, leading to a degradation in speech enhancement performance. Typically, the adjusted AKF gain is under-estimated in speech regions, resulting in distorted speech.

In recent years, deep learning-based supervised methods have been used for speech enhancement. Many approaches utilise a time-frequency (T-F) representation derived from the unobserved clean speech and noise as the training target [11]. Inspired by the T-F masking of CASA [11], Wang and Wang proposed to use a deep neural network (DNN) to estimate the ideal binary mask (IBM) [16]. The estimated IBM can be used to estimate the T-F components of the clean speech. Later on, researchers found that the ideal ratio mask (IRM) produces higher objective quality scores than the IBM [17]. In [18], a post-processing method was employed after masking with the IBM, IRM, or ideal amplitude mask (IAM) [19], resulting in an improvement in objective quality and intelligibility. In [20], Williamson *et al.* introduced a complex ideal ratio mask (cIRM), which is capable to recover both the amplitude and the phase spectrum of the clean speech. In [21], Zheng *et al.* introduced a phase-ware SEA. Here, the phase information (converted to the instantaneous frequency deviation (IFD) is jointly used with the IAM to form the phase sensitive mask (PSM). The clean speech spectrum is then reconstructed using the estimated mask and the phase information (extracted from the IFD). Unlike masking-based methods, mapping-based methods utilise a deep neural network (DNN) to estimate the clean speech spectrum. In [12], Xu *et al.* employed a DNN to map the noisy speech log power spectra (LPS) to the clean speech LPS. In [22], Han *et al.* trained a DNN to learn a spectral mapping from the magnitude spectrum of noisy speech to that of clean speech.

Deep learning methods have also been proposed to improve the performance of statistical model-based SEAs, such as the MMSE short-time spectral amplitude (MMSE-STSA) estimator [6], MMSE log-spectral amplitude (MMSE-LSA) estimator [7], WF [1], and square-root WF (SRWF) [1]. Generally, the performance of these SEAs relies upon the accuracy of the *a priori* SNR estimate. In [23], Nicolson and Paliwal proposed Deep Xi—a deep learning framework to estimate the *a priori* SNR. In [24], Zhang *et al.* proposed the DeepMMSE framework for noise power spectral density (PSD) estimation. DeepMMSE uses the Deep Xi framework with a residual network (ResNet) temporal convolutional network (ResNet-TCN) to estimate parameters for the MMSE-based noise periodogram estimator. DeepMMSE was able to demonstrate better noise PSD tracking than other benchmark methods in various noise conditions.

In [25], an attention-based network was investigated for speech enhancement, namely the multi-head attention network (MHANet). This was motivated by the ability of multi-head attention to more efficiently model long-term dependencies than recurrent neural networks (RNNs) and TCNs [26]. The experimental results demonstrated that MHANet was able to attain significantly higher objective quality and intelligibility scores than a TCN and a long short-term memory (LSTM) network. This indicated that multi-head attention is more apt at modelling the long-term dependencies of the clean speech and background noise present in noisy speech than that of RNNs and TCNs.

Deep learning has also been employed for time-domain speech enhancement. In [27], Fu *et al.* proposed raw waveform-based speech enhancement using a fully convolutional neural network (RWF-FCNN). The FCNN maps noisy speech time-domain frames to clean speech time-domain frames. Different from noisy speech spectral mapping [22], RWF-FCNN maps each frame of the noisy speech waveform to the clean speech waveform. By estimating time-domain samples, RWF-FCNN also estimates the phase—spectral magnitude estimation methods [12]. In [28], the authors claimed that the discontinuities present at the boundaries of framed speech are detrimental to the enhancement process in [27]. Motivated by this, the authors proposed end-to-end utterance enhancement using an FCNN (EEUE-FCNN). In this SEA, an FCNN directly maps the noisy speech to the clean speech. It was shown that EEUE-FCNN [28] produces more intelligible enhanced speech than that of RWF-FCNN [27].

Deep learning has also been used for LPC estimation—a key stage for KF and AKF-based SEA [9], [10]. In [29], Pickersgill *et al.* proposed an LPC estimation method using a DNN. One drawback of this study is that results weren't given for lower SNR levels (below 10 dB). Moreover, only six noise recordings were used for training, reducing its generalisation capabilities for unseen noise conditions. For the KF SEA in [30], Yu *et al.* utilised a DNN to estimate LPC parameters from noisy speech frames. For training, only 10, 720 examples constructed from 670 speech recordings, four noise recordings, and four SNR levels were used. This limits the number of conditions observed by the DNN during training, thus reducing its generalisation capabilities to unseen conditions. Also, the additive noise variance is computed from the first noisy speech frame by assuming that there is no presence of speech. However, this does not account for conditions that have time-varying amplitudes. In [31], Yu *et al.* adopted a fully-connected feed-forward DNN (denoted as FNN) and an LSTM network to estimate the clean speech and noise LPCs, respectively, as well as multi-band spectral subtraction (MB-SS) post-processing [3] for coloured-noise AKF-based speech enhancement (FNN-CKFS, LSTM-CKFS). To estimate the prediction error variances for the AR processes of the AKF, the authors employed a maximum likelihood (ML) approach [32]. However, FNN-CKFS and LSTM-CKFS lack the ability to accurately estimate LPCs in various noise conditions—leading to the use of MB-SS for post-processing. This could be due to the small amount of training data used when fitting the FNN and LSTM networks.

Motivated by the performance improvement that Deep Xi offers to statistical model-based SEAs [23], the AKF in [14] employed the Deep Xi framework to estimate its parameters (named Deep Xi-AKF). Improving upon Deep Xi-AKF, the KF in [33] utilised the DeepMMSE framework [24] to estimate its parameters (named Deep Xi-KF, as DeepMMSE uses Deep Xi). This was motivated by DeepMMSE's ability to significantly reduce MMSE-based noise PSD estimation bias. Deep Xi-AKF and Deep Xi-KF also used significantly larger training sets than previous methods. For Deep Xi-AKF and Deep Xi-KF, the noise parameters are computed from the estimated noise PSD derived from Deep Xi and DeepMMSE, respectively. However, Deep Xi-AKF and Deep Xi-KF do not directly estimate the clean speech LPC parameters from the noisy speech. Rather, a whitening filter is constructed with its coefficients computed from the estimated noise PSD. The whitening filter is then applied to each noisy speech frame, yielding pre-whitened speech, from where the speech LPC parameters are computed. This leads to biased clean speech LPC estimates—thus impacting the quality and intelligibility of the enhanced speech produced by the AKF and KF.

Recently, a deep learning framework was proposed to estimate the clean speech and noise LPC power spectra (LPC-PS), called DeepLPC [34]. The clean speech and noise LPC-PS estimates are then used to the LPC estimates required to construct the AKF. As a result, DeepLPC produces clean speech LPCs with significantly less bias than the aforementioned methods. This leads to the production of the highest quality and most intelligible enhanced speech amongst current KF and AKF SEAs—outperforming Deep Xi-KF while using the same training set. However, a ResNet-TCN [24] was used to estimate the clean speech and noise LPC-PS (DeepLPC-ResNet-TCN). As mentioned previously, a ResNet-TCN is suboptimal for modeling the long-term dependencies of noisy speech.

Motivated by the shortcomings of previous deep learning-based KF and AKF SEAs (presented in Table 1), we propose DeepLPC-MHANet for AKF-based speech enhancement. DeepLPC-MHANet aims to produce clean speech and noise LPC parameters with the least bias to date.

TABLE 1. Summary of existing deep learning-based LPC estimation methods for the KF as well as AKF.

Methods	Summary	Limitations
DNN-LPC [29]	A traditional DNN [12] is used to estimate the clean speech LPC parameters.	Only six noise recordings were used for training the DNN, reducing its generalisation capabilities for unseen noise conditions.
DeepXi-AKF [14]	The AKF is constructed with the noise and clean speech LPC estimates derived from Deep Xi-ResNet-TCN and a whitening filter [13], respectively.	The whitening filter gives a biased estimate of the clean speech LPC parameters, which impacts the quality and intelligibility of enhanced speech.
DeepXi-KF [33]	The KF is constructed with the noise variance and speech LPC estimates derived from the DeepMMSE framework [24] and whitening filter [13], respectively.	As like [14], the biased clean speech LPC parameters derived from the whitening filter impact the quality and intelligibility of enhanced speech.
LSTM-CKFS [31]	The AKF is constructed with the speech and noise LPC parameters derived from an LSTM network and an ML-based approach [32].	LSTM network and an ML-based approaches in [31] exhibit a high amount of bias in the estimated speech and noise LPC parameters.
DeepLPC-ResNet-TCN [34]	The clean speech and noise LPC power spectra are jointly estimated using DeepLPC-ResNet-TCN—which is used to compute the clean speech and noise LPC parameters for the AKF	ResNet-TCN demonstrates deficiency when modeling the long-term dependencies of noisy speech—unlike the MHANet [25].

With this, we also aim to produce higher quality and more intelligible enhanced speech than any KF or AKF-based SEA. DeepLPC is selected as it avoids the issues associated with previous deep learning frameworks for KF and AKF SEAs, including the use of whitening filters, post-processing, and small training sets. MHANet is selected as it is better suited than TCNs for modelling the long-term dependencies of noisy speech. Together, DeepLPC and MHANet form an improved map from the noisy speech to the clean speech and noise LPC parameters.

The structure of this paper is as follows: background knowledge is presented in Section II, including the AKF for speech enhancement, an overview of DeepLPC framework, and MHANet. In Section III, we describe the proposed DeepLPC-MHANet. Following this, Section IV describes the experimental setup. The experimental results are then presented in Section V, along with a discussion. Finally, Section VI gives some concluding remarks.

II. BACKGROUND

A. AKF FOR SPEECH ENHANCEMENT

In this section, we overview the AKF for speech enhancement. First, we describe the signal model. The noisy speech $y(n)$, at discrete-time sample n, is given by:

$$y(n) = s(n) + v(n), \tag{1}$$

where $s(n)$ is the clean speech and $v(n)$ is assumed to be uncorrelated additive coloured noise. Next, a 32 ms rectangular window with 50% overlap is used to convert $y(n)$ into frames, denoted by $y(n, l)$:

$$y(n, l) = s(n, l) + v(n, l), \tag{2}$$

where $l \epsilon \{0, 1, \ldots, L-1\}$ is the frame index with L being the total number of frames, and $n \epsilon \{0, 1, \ldots, N-1\}$ where N is the total number of samples within each frame. For simplicity, the frame index is omitted from the following AKF recursive equations.

Each frame of the clean speech and noise signal in Equation (2) can be represented by p^{th} and q^{th} order AR models, as in [35, Chapter 8]:

$$s(n) = -\sum_{i=1}^{p} a_i s(n-i) + w(n), \tag{3}$$

$$v(n) = -\sum_{k=1}^{q} b_k v(n-k) + u(n), \tag{4}$$

where $\{a_i; i = 1, 2, \ldots, p\}$ and $\{b_k; k = 1, 2, \ldots, q\}$ are the LPCs, and $w(n)$ and $u(n)$ are Gaussian-distributed excitation noises with zero mean and variances σ_w^2 and σ_u^2, respectively.

Equations (2)-(4) form the augmented state-space model (ASSM) of the AKF [10], given by:

$$\boldsymbol{x}(n) = \boldsymbol{\Phi}\boldsymbol{x}(n-1) + \boldsymbol{r}\boldsymbol{z}(n), \tag{5}$$

$$y(n) = \boldsymbol{c}^\top \boldsymbol{x}(n). \tag{6}$$

In the above ASSM,

1) $\boldsymbol{x}(n) = [s(n) \; \ldots s(n-p+1)\, v(n) \; \ldots \; v(n-q+1)]^T$ is a $(p+q) \times 1$ state-vector,
2) $\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_s & 0 \\ 0 & \boldsymbol{\Phi}_v \end{bmatrix}$ is a $(p+q) \times (p+q)$ state-transition matrix with:

$$\boldsymbol{\Phi}_s = \begin{bmatrix} -a_1 & -a_2 & \ldots & -a_{p-1} & -a_p \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}, \tag{7}$$

$$\boldsymbol{\Phi}_v = \begin{bmatrix} -b_1 & -b_2 & \ldots & -b_{q-1} & -b_q \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}, \tag{8}$$

3) $\boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}_s & 0 \\ 0 & \boldsymbol{r}_v \end{bmatrix}$, where $\boldsymbol{r}_s = \begin{bmatrix} 1 \; 0 \ldots 0 \end{bmatrix}^\top$, $\boldsymbol{r}_v = \begin{bmatrix} 1 \; 0 \ldots 0 \end{bmatrix}^\top$,
4)
$$z(n) = \begin{bmatrix} w(n) \\ u(n) \end{bmatrix}, \tag{9}$$
5) $\boldsymbol{c}^\top = \begin{bmatrix} \boldsymbol{c}_s^\top \; \boldsymbol{c}_v^\top \end{bmatrix}$, where $\boldsymbol{c}_s = \begin{bmatrix} 1 \; 0 \ldots 0 \end{bmatrix}^\top$ and $\boldsymbol{c}_v = \begin{bmatrix} 1 \; 0 \ldots 0 \end{bmatrix}^\top$ are $p \times 1$ and $q \times 1$ vectors,
6) $y(n)$ is the noisy measurement at sample n.

For each frame, the AKF recursively computes an unbiased linear MMSE estimate $\hat{\boldsymbol{x}}(n|n)$ at sample n, given $y(n)$, by using the following equations [13]:

$$\hat{\boldsymbol{x}}(n|n-1) = \boldsymbol{\Phi}\hat{\boldsymbol{x}}(n-1|n-1), \tag{10}$$

$$\boldsymbol{\Psi}(n|n-1) = \boldsymbol{\Phi}\boldsymbol{\Psi}(n-1|n-1)\boldsymbol{\Phi}^\top + \boldsymbol{Q}_n \boldsymbol{r}\boldsymbol{r}^\top, \tag{11}$$

$$\boldsymbol{G}(n) = \boldsymbol{\Psi}(n|n-1)\boldsymbol{c}(\boldsymbol{c}^\top \boldsymbol{\Psi}(n|n-1)\boldsymbol{c})^{-1}, \tag{12}$$

$$\hat{\boldsymbol{x}}(n|n) = \hat{\boldsymbol{x}}(n|n-1) + \boldsymbol{G}(n)[y(n) - \boldsymbol{c}^\top \hat{\boldsymbol{x}}(n|n-1)], \tag{13}$$

$$\boldsymbol{\Psi}(n|n) = [\boldsymbol{I} - \boldsymbol{G}(n)\boldsymbol{c}^\top]\boldsymbol{\Psi}(n|n-1), \tag{14}$$

where $\boldsymbol{Q}_n = \begin{bmatrix} \sigma_w^2 & 0 \\ 0 & \sigma_u^2 \end{bmatrix}$ is the process noise covariance.

For a noisy speech frame, the error covariances $\big(\boldsymbol{\Psi}(n|n-1)$ and $\boldsymbol{\Psi}(n|n)$ corresponding to $\hat{\boldsymbol{x}}(n|n-1)$ and $\hat{\boldsymbol{x}}(n|n)$, respectively$\big)$ and the Kalman gain $\boldsymbol{G}(n)$ are continually updated on a samplewise basis, while $(\{a_i\}, \sigma_w^2)$ and $(\{b_k\}, \sigma_u^2)$ remain constant. At sample n, $\boldsymbol{g}^\top \hat{\boldsymbol{x}}(n|n)$ gives the output of the AKF, $\hat{s}(n|n)$, where $\boldsymbol{g} = \begin{bmatrix} 1 \; 0 \; 0 \ldots 0 \end{bmatrix}^\top$ is a $(p+q) \times 1$ column vector. As in [13], $\hat{s}(n|n)$ is given by:

$$\hat{s}(n|n) = [1 - G_0(n)]\hat{s}(n|n-1) + G_0(n)[y(n) - \hat{v}(n|n-1)], \tag{15}$$

where $G_0(n)$ is the 1^{st} component of $\boldsymbol{G}(n)$, given by [13]:

$$G_0(n) = \frac{\alpha^2(n) + \sigma_w^2}{\alpha^2(n) + \sigma_w^2 + \beta^2(n) + \sigma_u^2}, \tag{16}$$

where

$$\alpha^2(n) = \boldsymbol{c}_s^\top \boldsymbol{\Phi}_s \boldsymbol{\Psi}_s(n-1|n-1)\boldsymbol{\Phi}_s^\top \boldsymbol{c}_s, \tag{17}$$

and

$$\beta^2(n) = \boldsymbol{c}_v^\top \boldsymbol{\Phi}_v \boldsymbol{\Psi}_v(n-1|n-1)\boldsymbol{\Phi}_v^\top \boldsymbol{c}_v, \tag{18}$$

are the transmission of *a posteriori* error variances of the speech and noise augmented dynamic model from the previous sample $n-1$, respectively [13].

Equation (15) reveals that $G_0(n)$ has a significant impact on $\hat{s}(n|n)$. In practice, inaccurate estimates of $(\{a_i\}, \sigma_w^2)$ and $(\{b_k\}, \sigma_u^2)$ introduce bias into $G_0(n)$, which impacts $\hat{s}(n|n)$. In our previous work, we proposed the DeepLPC framework [34] to estimate $(\{a_i\}, \sigma_w^2)$ and $(\{b_k\}, \sigma_u^2)$ for the AKF, as described in following section.

B. DeepLPC FRAMEWORK

In this section, we review the DeepLPC framework [34]. DeepLPC was able to produce speech and noise LPC estimates with significantly less bias than previous methods by avoiding the use of a whitening filter, as used in earlier methods [14], [33]. This was accomplished by using deep learning to jointly estimate the clean speech and noise LPC-PS, denoted as $\hat{\boldsymbol{\lambda}}_s(l) = \{\hat{\lambda}_s(l,0), \hat{\lambda}_s(l,1), \ldots, \hat{\lambda}_s(l,M-1)\}$ and $\hat{\boldsymbol{\lambda}}_v(l) = \{\hat{\lambda}_v(l,0), \hat{\lambda}_v(l,1), \ldots, \hat{\lambda}_v(l,M-1)\}$, respectively, where M is the total number of discrete-frequency bins.

The DeepLPC framework is shown in Figure 1. DeepLPC is fed as input the single-sided noisy speech magnitude spectrum $|\boldsymbol{Y}(l)| = \{|Y(l,0)|, |Y(l,1)|, \ldots, |Y(l,M-1)|\}$. This is computed from the noisy speech in Equation 1 using the short-time Fourier transform (STFT):

$$Y(l,m) = S(l,m) + V(l,m), \tag{19}$$

where $Y(l,m)$, $S(l,m)$, and $V(l,m)$ denote the complex-valued STFT coefficients of the noisy speech, clean speech, and noise, respectively, for time-frame index l and discrete-frequency bin m. The Hamming window is used for analysis and synthesis.

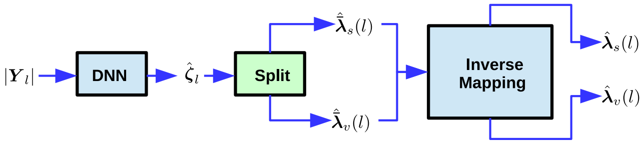
$|\boldsymbol{Y}_l|$
DNN
$\hat{\zeta}_l$
Split
$\hat{\bar{\boldsymbol{\lambda}}}_s(l)$
$\hat{\bar{\boldsymbol{\lambda}}}_v(l)$
Inverse Mapping
$\hat{\boldsymbol{\lambda}}_s(l)$
$\hat{\boldsymbol{\lambda}}_v(l)$

FIGURE 1. Block diagram of the DeepLPC framework.

DeepLPC then estimates the clean speech and noise LPC-PS in two stages. For the first stage, a DNN jointly estimates a mapped version of the speech and noise LPC-PS, $\boldsymbol{\zeta}_l = \{\bar{\boldsymbol{\lambda}}_s(l); \bar{\boldsymbol{\lambda}}_v(l)\}$, of size $M \times 2$, where $\{\cdot;\cdot\}$ denotes the concatenation operation, and $\bar{\boldsymbol{\lambda}}_s(l)$ and $\bar{\boldsymbol{\lambda}}_v(l)$ are computed from $\boldsymbol{\lambda}_s(l)$ and $\boldsymbol{\lambda}_v(l)$, respectively, by using a mapping function. As in [34], the cumulative distribution functions (CDF) of $\boldsymbol{\lambda}_s(l)$ and $\boldsymbol{\lambda}_v(l)$ are used as the mapping functions to compute $\bar{\boldsymbol{\lambda}}_v(l)$ and $\bar{\boldsymbol{\lambda}}_s(l)$, respectively. A description of how $\bar{\boldsymbol{\lambda}}_v(l)$ and $\bar{\boldsymbol{\lambda}}_s(l)$ are computed is provided in Appendix A. In [34], a ResNet-TCN was used to estimate $\hat{\boldsymbol{\zeta}}_l$. For the second stage, $\hat{\boldsymbol{\zeta}}_l$ is first split into the mapped clean speech and noise LPC-PS, $\hat{\bar{\lambda}}_s(l,m)$ and $\hat{\bar{\lambda}}_v(l,m)$, respectively. Next, the inverse mapping of $\hat{\bar{\lambda}}_s(l,m)$ and $\hat{\bar{\lambda}}_v(l,m)$ yields $\hat{\lambda}_s(l,m)$ and $\hat{\lambda}_v(l,m)$. The inverse mapping is described in Appendix B.

The |IDFT| of $\hat{\lambda}_s(l,m)$ and $\hat{\lambda}_v(l,m)$ yields an estimate of the autocorrelation matrices, $\widehat{R}_{ss}(\tau)$ and $\widehat{R}_{vv}(\tau)$, where τ is the autocorrelation lag. As in [34, eq. (26)-(27)], we construct Yule-Walker equations with the estimated $\widehat{R}_{ss}(\tau)$ and $\widehat{R}_{vv}(\tau)$. These are solved using the Levinson-Durbin recursion [35, Chapter 8], giving $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ $(p = 16)$ and $(\{\hat{b}_k\}, \hat{\sigma}_u^2)$ $(q = 16)$ for constructing the AKF.

C. MHANet

The MHANet proposed in [25] is overviewed from input to output in this section. For a detailed description of MHANet, we refer the reader to [25]. MHANet is shown in Figure 2 (left). The first layer is used to project the input to a size of d_{model}, and is realised as follows: $\max(0, \text{LN}(|\boldsymbol{Y}|\boldsymbol{W}^I + \boldsymbol{b}_s^I))$, where LN is frame-wise layer normalisation [36] $\big(\boldsymbol{W}^I \in \mathbb{R}^{M \times d_{model}}$ and $\boldsymbol{b}_s^I \in \mathbb{R}^{d_{model}}\big)$. Next, the positional encoding from [25] is added after the first layer, where the time-frame index indicates the position. The position encoding is learned using weight matrix W_p, with a maximum length of 2048 time-frames (i.e. $W_p \in \mathbb{R}^{2048 \times 256}$). This is followed B cascading blocks identical to those from the encoder of the Transformer [26], except that masked multi-head attention (MHA) is employed, to ensure causality.

Each block includes an MHA module, a two-layer feed-forward neural network (FNN) [12], residual connections [37], and frame-wise LN [36]. The MHA module of each block is shown in Figure 2 (middle). The MHA module takes three inputs belonging to a set of L queries $(\boldsymbol{Q}_s \in \mathbb{R}^{L \times d_{model}})$, keys $(\boldsymbol{K}_s \in \mathbb{R}^{L \times d_{model}})$, and values $(\boldsymbol{V}_s \in \mathbb{R}^{L \times d_{model}})$, where L is the number of frames, and d_{model} is the size of each query, key, and value. Each MHA module includes a total of H heads of *masked scaled dot-product attention*, where $h = \{1, 2, \cdots, H\}$ is the head index. For head h, $\boldsymbol{Q}_s$, $\boldsymbol{K}_s$, and $\boldsymbol{V}_s$ are linearly projected as: $\boldsymbol{\mathcal{Q}}_h = \boldsymbol{Q}_s\boldsymbol{W}_h^Q$, $\boldsymbol{\mathcal{K}}_h = \boldsymbol{K}_s\boldsymbol{W}_h^K$, and $\boldsymbol{\mathcal{V}}_h = \boldsymbol{V}_s\boldsymbol{W}_h^V$, where $\boldsymbol{W}_h^Q \in \mathbb{R}^{d_{model} \times d_k}$, $\boldsymbol{W}_h^K \in \mathbb{R}^{d_{model} \times d_k}$, and $\boldsymbol{W}_h^V \in \mathbb{R}^{d_{model} \times d_v}$ are learned weight matrices. The projected queries and keys are of size d_k, and the projected values are of size d_v, where $d_k = d_v = d_{model}/H$. Figure 2 (right) shows the masked scaled dot-product attention mechanism for head h, which takes as input $\boldsymbol{\mathcal{Q}}_h$, $\boldsymbol{\mathcal{K}}_h$, and $\boldsymbol{\mathcal{V}}_h$. Masked scaled dot-product attention is computed as:

$$\text{Attention}(\boldsymbol{\mathcal{Q}}_h, \boldsymbol{\mathcal{K}}_h, \boldsymbol{\mathcal{V}}_h) = \text{softmax}\Big(\boldsymbol{M}_s + \frac{\boldsymbol{\mathcal{Q}}_h\boldsymbol{\mathcal{K}}_h^\top}{\sqrt{d_k}}\Big)\boldsymbol{\mathcal{V}}_h. \tag{20}$$

The outputs from all of the heads are then concatenated and linearly projected using the learned weight matrix $W_h^O \in \mathbb{R}^{Hd_v \times d_{model}}$, forming the final output of the MHA module:

$$\text{MHA}(\boldsymbol{Q}_s, \boldsymbol{K}_s, \boldsymbol{V}_s) = \text{concat}(\boldsymbol{A}_1, \boldsymbol{A}_2, \cdots, \boldsymbol{A}_H)\boldsymbol{W}^O. \tag{21}$$

A residual connection is applied from the input to the output of the MHA module, which is followed by frame-wise LN.

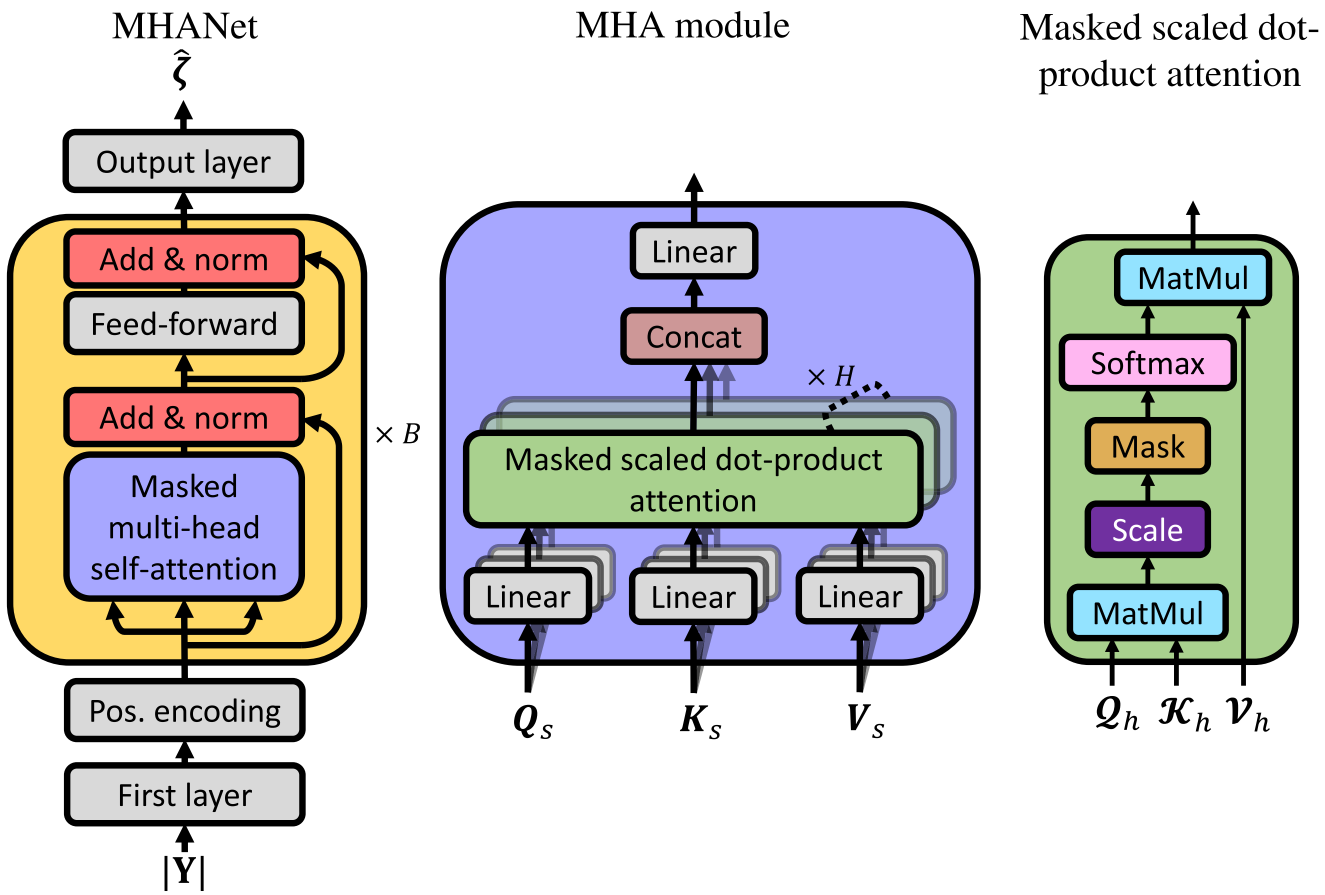

FIGURE 2. (left) DeepLPC-MHANet, (middle) multi-head attention (MHA) module, and (right) masked scaled dot-product attention.

The second half of the block includes a two-layer FNN:

$$\text{FNN}(\boldsymbol{Z}) = \max(0, \boldsymbol{Z}\boldsymbol{W}^1 + \boldsymbol{b}_s^1)\boldsymbol{W}^2 + \boldsymbol{b}_s^2, \tag{22}$$

where $\boldsymbol{Z} \in \mathbb{R}^{L\times d_{model}}$ is the input, $\boldsymbol{W}^1 \in \mathbb{R}^{d_{model}\times d_f}$, $\boldsymbol{b}_s^1 \in \mathbb{R}^{d_f}$, $\boldsymbol{W}^2 \in \mathbb{R}^{d_f\times d_{model}}$, and $\boldsymbol{b}_s^2 \in \mathbb{R}^{d_{model}}$. Hence, the inner layer has a size of d_f. A residual connection is applied from the input to the output of the FNN, which is followed by frame-wise LN. The last block is followed by the output layer, which is a sigmoidal feed-forward layer. For analyses regarding the behavior of the attention weights of the MHANet during speech enhancement, we refer the readers to [25, Figure 5].

III. PROPOSED DeepLPC-MHANet

Current deep learning-based AKF methods employ a TCN, for example, Deep Xi-KF, Deep Xi-AKF, and DeepLPC-ResNet-TCN-AKF. However, TCNs demonstrate deficiencies when modeling the long-term dependencies of noisy speech—unlike attention-based networks [25]. Hence, we investigate if an attention-based network can produce clean speech and noise LPC estimates with less bias and obtain higher quality and intelligibility scores than current deep learning-based KF and AKF SEAs. To this end, we compare the ResNet-TCN to the MHANet within the DeepLPC framework, as it has shown to outperform all other KF and AKF deep learning frameworks to date [34].

The MHANet was first investigated within the Deep Xi framework (Deep Xi-MHANet) [23], [25]. Specifically, Deep Xi-MHANet was used to estimate the *a priori* SNR from the noisy speech magnitude spectrum for statistical estimators, such as the MMSE-STSA estimator [6]. Our proposed method differs from Deep Xi-MHANet by jointly estimating the clean speech and noise LPC-PS instead of the *a priori* SNR. Simply, the differences between Deep Xi-MHANet and DeepLPC-MHANet is the training target. Deep Xi-MHANet estimates the *a priori* SNR for statistical estimators and DeepLPC-MHANet jointly estimates the clean speech and noise LPC-PS for the AKF. The novelty of our proposed method lies in the fact that it will be the first deep learning-based KF or AKF SEA to utilise an attention-based network.

The block diagram of the proposed SEA, DeepLPC-MHANet-AKF, is shown in Figure 3. It can be seen that DeepLPC-MHANet estimates $\boldsymbol{\zeta} = \{\bar{\boldsymbol{\lambda}}_s; \bar{\boldsymbol{\lambda}}_v\}$ from $|\boldsymbol{Y}|$. The hyperparameters for DeepLPC-MHANet are the same used in [25]: $B = 5$, $d_f = 1\,024$, $d_{model} = 256$, $H = 8$, and $\Gamma = 40\,000$. The training strategy as well as a complexity

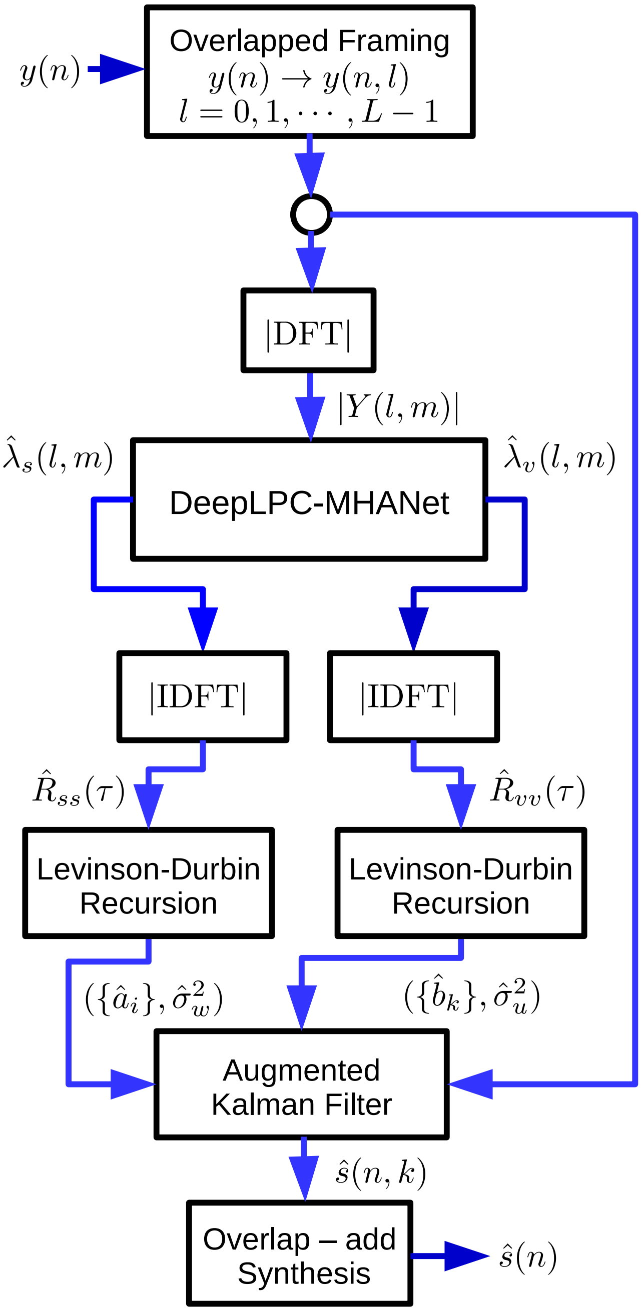

FIGURE 3. **Block diagram of the proposed SEA.**

and convergence analysis of MHANet are detailed in Sections IV-B and IV-C.

IV. SPEECH ENHANCEMENT EXPERIMENT

A. TRAINING & VALIDATION SET

The noisy speech for the training and validation sets are formed from clean speech and noise recordings. For the clean speech recordings, the *train-clean-100* set of the Librispeech corpus [38] (28 539), the CSTR VCTK corpus [39] (42 015), and the *si** and *sx** training sets of the TIMIT corpus [40] (3 696) were used, giving a total of 74 250 clean speech recordings. To form the validation set, 5% of the clean speech recordings (3 713) are randomly selected. Thus, 70 537 of the clean speech recordings are used for the training set. For the noise recordings, the QUT-NOISE dataset [41], the Nonspeech dataset [42], the Environmental Background Noise dataset [43], [44], the noise set from the MUSAN corpus [45], multiple FreeSound packs (https://freesound.org/),[1] and coloured noise recordings (with an value ranging from 2 to 2 in increments of 0.25) were used, giving a total of 16 243 noise recordings. For the validation set, 5% of the noise recordings (813) are randomly selected. The remaining 15 430 noise recordings are used for the training set. All the clean speech and noise recordings are single-channel with a sampling frequency of 16 kHz. To create the noisy speech for the validation set, each of the 3 713 clean speech recordings are corrupted by a random section of a randomly selected noise recording (from the set of 813 noise recordings) at a randomly selected SNR level (−10 to +20 dB, in 1 dB increments). The noisy speech for the training set was created using the method described in Section IV-B.

B. TRAINING STRATEGY

The following training strategy was employed to train DeepLPC-MHANet:

- Mean squared error is used as the loss function.
- The *Adam* optimiser [46] with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 10^{-9}$ is used for stochastic gradient descent optimisation, where the learning rate, α_r, is controlled over the course of training as in [26]:

$$\alpha_r = d_{model}^{-0.5} \cdot \min(\gamma^{-0.5}, \gamma \cdot \Gamma^{-1.5}), \tag{23}$$

  where γ is the training step and Γ is the number of warmup steps.
- Gradients are clipped between $[-1, 1]$.
- The number of training examples in an epoch is equal to the number of clean speech recordings used in the training set, i.e., 70 537.
- A mini-batch size of 8 training examples is used.
- The noisy speech signals are generated on the fly as follows: each clean speech recording is randomly selected and corrupted with a randomly selected noise recording at a randomly selected SNR level (-10 to +20 dB, in 1 dB increments).
- During training, we employ early stopping which monitors the validation loss with a patience of 30 epochs. Using this strategy, training was terminated at epoch 180 (where epoch 150 was used for testing).

C. COMPLEXITY AND CONVERGE ANALYSIS OF DeepLPC-ResNet-TCN

The complexity of a DNN usually depends on the number of training parameters, where MHANet has 4.27 million parameters and ResNet-TCN has 2.1 million parameters [34].

[1] Freesound packs that were used: 147, 199, 247, 379, 622, 643, 1 133, 1 563, 1 840, 2 432, 4 366, 4 439, 15 046, 15 598, 21 558.

However, the computational complexity of self-attention layers is less than convolutional layers [25, Table 1]. As a result, the amount of time taken to complete a single training epoch for the MHANet is 30 minutes, compared to 40 minutes for ResNet-TCN (using an NVIDIA GTX 1080 Ti GPU on the Deep Xi dataset) [25].

Next, we analyse the convergence of the mean squared error between the predicted and true values for the training and validation sets of DeepLPC-MHANet on the Deep Xi dataset, as shown in Figure 4. It can be seen that the mean squared error reduces for the training set as well as the validation set after each epoch, until converging at around epoch 150. As the early stopping criterion with a patience of 30 is used, epoch 150 is chosen for testing.

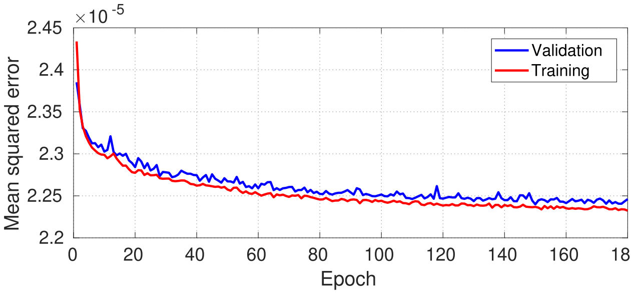

FIGURE 4. Mean squared error between the predicted and true values for the training and validation data sets of DeepLPC-MHANet.

D. TEST SET

For the objective experiments, 30 clean speech recordings belonging to six speakers (three male and three female) are taken from the NOIZEUS corpus [1, Chapter 12]. The noisy speech for the test set is formed by mixing the clean speech with real-world non-stationary (*voice babble, street*) and coloured (*factory* and *f16*) noise recordings selected from [43], [44] at multiple SNR levels varying from −5dB to +15 dB, in 5 dB increments. This provides 30 examples per condition with 20 total conditions. All clean speech and noise recordings in the test set are single channel with a sampling frequency of 16 kHz.

The NOIZEUS corpus was chosen to evaluate the proposed SEA as it has been used to evaluate many other deep learning-based KF and AKF SEAs [14], [33], [34]. An important attribute of the NOIZEUS corpus is that the clean speech recordings are phonetically balanced. This ensures that all phonemes are included in the evaluation. The NOIZEUS corpus and the selected noise recordings also guarantee an unbiased evaluation as they are different from those used in the training and validation sets.

E. SD LEVEL EVALUATION

The frame-wise spectral distortion (SD) (dB) [47] is used to evaluate the accuracy of the LPC estimates produced by DeepLPC-MHANet. Specifically, the estimated clean speech LPCs are evaluated. SD for l^{th} frame, D_l (in dB) is defined as the root-mean-square-difference between the LPC-PS estimate in dB, $\hat{\lambda}_s(l, m)_{[dB]}$, and the oracle case in dB, $\lambda_s(l, m)_{[dB]}$ as [47]:

$$D_l = \sqrt{\frac{1}{M}\sum_{m=0}^{M-1}\left[\lambda_s(l, m)_{[dB]} - \hat{\lambda}_s(l, m)_{[dB]}\right]^2}. \tag{24}$$

F. OBJECTIVE QUALITY AND INTELLIGIBILITY MEASURES

Objective measures are used to evaluate the quality and intelligibility of the enhanced speech with respect to the corresponding clean speech. The objective quality and intelligibility measures used in this paper are given in Table 2. We also analyse the enhanced speech spectrogram of the proposed SEA, to determine if it causes *speech distortion*, if any background noise is not suppressed (i.e. *residual background noise*), and if it introduces any *musical noise*.

TABLE 2. Objective measures, what each assesses, and the range of their scores. For each measure, higher is better.

Measure	Assesses	Range
CSIG [48]	Quality	$[1, 5]$
CBAK [48]	Quality	$[1, 5]$
COVL [48]	Quality	$[1, 5]$
PESQ [49]	Quality	$[-0.5, 4.5]$
STOI [50]	Intelligibility	$[0, 100]\%$
SI-SDR [51]	Quality	$[-\infty, \infty]$
SegSNR [52]	Quality	$[-\infty, \infty]$

G. SUBJECTIVE EVALUATION

The subjective evaluation was carried out through a series of blind AB listening tests [4, Section 3.3.4]. To perform the tests, we generated a set of stimuli by corrupting recordings *sp05* and *sp27* from the NOIZEUS corpus [1, Chapter 12]. The reference transcript for recording *sp05* is: "*Wipe the grease off his dirty face*", and is corrupted with *voice babble* at 5 dB. The reference transcript for recording *sp27* is: "*Bring your best compass to the third class*", and is corrupted with *factory* at 5 dB. Utterance *sp05* and *sp27* were uttered by a male and a female, respectively. In this test, the enhanced speech produced by seven SEAs, as well as the corresponding clean speech and noisy speech signals, were played as stimuli pairs to the listeners. Specifically, the test is performed on a total of 144 stimuli pairs (72 for each of recording) played in a random order to each listener, excluding the comparisons between the same method.

The listener gives the following ratings for each stimuli pair: perceptual preference for the first or second stimuli, or a third response indicating no preference. For pairwise scoring, a score of 100% is given to the preferred method, 0% to the other. A score of 50% is given to both methods when there is no preference. The participants were able to re-listen to the stimuli pair if required. Ten English speaking listeners participate in the blind AB listening tests.[2] The mean

[2] The AB listening tests were conducted on the approval of Griffith University Human Research Ethics: database protocol number 2018/671.

TABLE 3. Average SD (dB) level comparison for each of the LPC estimation methods. The boldface represent the lowest SD level. The used test set is described in Section IV-D.

		SNR level (dB)				
Noise	Methods	-5	0	5	10	15
Voice babble	Noisy	22.05	18.29	14.86	13.80	11.87
	DNN-LPC [29]	16.72	15.98	13.24	12.76	10.79
	LSTM-CKFS [31]	15.91	14.51	12.11	11.89	9.23
	Deep Xi-KF [33]	14.95	13.88	11.81	10.31	9.11
	DeepLPC-ResNet-TCN [34]	11.89	10.49	8.73	7.33	6.51
	Proposed	**10.84**	**8.49**	**6.71**	**5.60**	**4.89**
Street	Noisy	20.21	16.39	14.43	13.88	12.45
	DNN-LPC [29]	13.41	12.25	11.68	11.18	10.87
	LSTM-CKFS [31]	12.57	11.05	10.78	10.35	9.86
	Deep Xi-KF [33]	11.66	10.51	9.74	9.21	8.95
	DeepLPC-ResNet-TCN [34]	9.21	8.74	7.59	6.91	5.89
	Proposed	**7.84**	**6.32**	**4.88**	**4.56**	**4.49**
Factory	Noisy	29.46	25.21	21.16	18.36	16.83
	DNN-LPC [29]	18.74	17.15	16.47	15.79	14.67
	LSTM-CKFS [31]	16.39	15.91	14.61	13.60	13.12
	Deep Xi-KF [33]	15.10	14.98	13.87	12.72	12.33
	DeepLPC-ResNet-TCN [34]	12.29	10.89	9.48	8.21	7.89
	Proposed	**10.15**	**8.23**	**7.01**	**6.11**	**5.52**
F16	Noisy	28.81	24.56	20.54	17.78	15.32
	DNN-LPC [29]	18.93	17.78	16.55	15.23	13.22
	LSTM-CKFS [31]	16.78	15.36	14.65	13.13	12.78
	Deep Xi-KF [33]	14.21	13.01	12.59	11.96	10.81
	DeepLPC-ResNet-TCN [34]	12.13	10.46	9.49	8.63	7.83
	Proposed	**9.76**	**8.09**	**6.12**	**5.82**	**5.48**

subjective preference score (%) is used to compare the SEAs, which is the average of the preference scores given by the listeners.

H. SPECIFICATIONS OF THE COMPETITIVE SEAs

The performance of the proposed SEA is compared to the following SEAs (the following notation is used for convenience: (p, q) : is the order of $\{a_i\}$ and $\{b_k\}$, (σ_w^2, σ_u^2) are the prediction error variances of the speech and noise AR models, w_f is the analysis frame duration (ms), and s_f is the analysis frame shift (ms)).

1) **Noisy**: speech corrupted with additive noise.
2) **AKF-Oracle:** AKF, where $(\{a_i\}, \sigma_w^2)$ and $(\{b_k\}, \sigma_u^2)$ are computed from the clean speech and the noise signal, where $p = 16$, $q = 16$, $w_f = 32$ ms, $s_f = 16$ ms, and a rectangular window is used for framing.
3) **LSTM-CKFS** [31]: AKF constructed using $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ and $(\{\hat{b}_k\}, \hat{\sigma}_u^2)$ computed using an LSTM and ML-based approaches, where $p = 12$, $q = 12$, $w_f = 20$ ms, $s_f = 0$ ms, and a rectangular window is used for framing. LSTM-CKFS utilises multi-band SS post-processing [3].
4) **EEUE-FCNN** [28]: End-to-end utterance enhancement using a fully convolutional neural network.
5) **Deep Xi-KF** [33]: KF-based SEA, where $\hat{\sigma}_v^2$ is estimated using the DeepMMSE framework [24] and $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ are computed from pre-whitened speech corresponding to each noisy speech frame, where $p = 10$, $w_f = 32$ ms, $s_f = 16$ ms, and a rectangular window is used for framing. Specifically, the ResNet-TCN from [34] was used for DeepMMSE.
6) **Deep Xi-ResNet-TCN-MMSE-LSA**: The ResNet-TCN from [34] is used to form Deep Xi-ResNet-TCN [24]. Deep Xi-ResNet-TCN estimates the *a priori* SNR for the MMSE-LSA estimator [7], where $w_f = 32$ ms, $s_f = 16$ ms, and a square-root-Hann window is used for analysis and synthesis.
7) **DeepLPC-ResNet-TCN-AKF** [34]: AKF constructed with $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ and $(\{\hat{b}_k\}, \hat{\sigma}_u^2)$ derived from DeepLPC framework, where $p = 16$, $q = 16$, $w_f = 32$ ms, $s_f = 16$ ms, and a rectangular window is used for framing. The ResNet-TCN used for DeepLPC is described in [34].
8) **Proposed DeepLPC-MHANet-AKF:** Proposed SEA, where AKF is constructed from $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ and $(\{\hat{b}_k\}, \hat{\sigma}_u^2)$ computed using DeepLPC-MHANet, where $p = 16$, $q = 16$, $w_f = 32$ ms, $s_f = 16$ ms, and a rectangular window is used for framing.

V. RESULTS AND DISCUSSIONS

A. SD LEVEL COMPARISON

The average SD levels (found over all frames for each test condition) attained by the proposed method are given in Table 3. It can be seen that for both real-world non-stationary (*voice babble* and *street*) and coloured (*factory* and *f16*) noise conditions, the proposed method produced lower SD levels than DeepLPC-ResNet-TCN [34]. This demonstrates that an attention-based network is able to produce clean speech LPC estimates with less bias. This indicates that

TABLE 4. Mean objective scores on the NOIZEUS dataset in terms of CSIG, CBAK, COVL, PESQ, STOI, SegSNR, and SI-SDR. Apart from AKF-Oracle, the highest score amongst the methods for each measure is given in boldface.

Methods	CSIG	CBAK	COVL	PESQ	STOI	SegSNR	SI-SDR
Noisy speech	2.41	2.27	2.12	1.64	67.87	0.89	6.39
LSTM-CKFS	2.63	2.55	2.42	1.99	77.58	6.54	11.15
EEUE-FCNN	2.76	2.66	2.56	2.05	79.45	6.93	11.59
Deep Xi-KF	3.11	2.83	2.72	2.16	81.89	7.14	12.15
Deep Xi-ResNet-TCN-MMSE-LSA	3.38	3.02	2.81	2.22	82.05	7.67	13.39
DeepLPC-ResNet-TCN-AKF	3.49	3.17	2.95	2.35	84.71	8.78	14.44
Proposed	**3.66**	**3.32**	**3.14**	**2.59**	**88.41**	**9.21**	**15.01**
AKF-Oracle	4.21	4.07	3.97	2.74	95.18	10.87	16.43

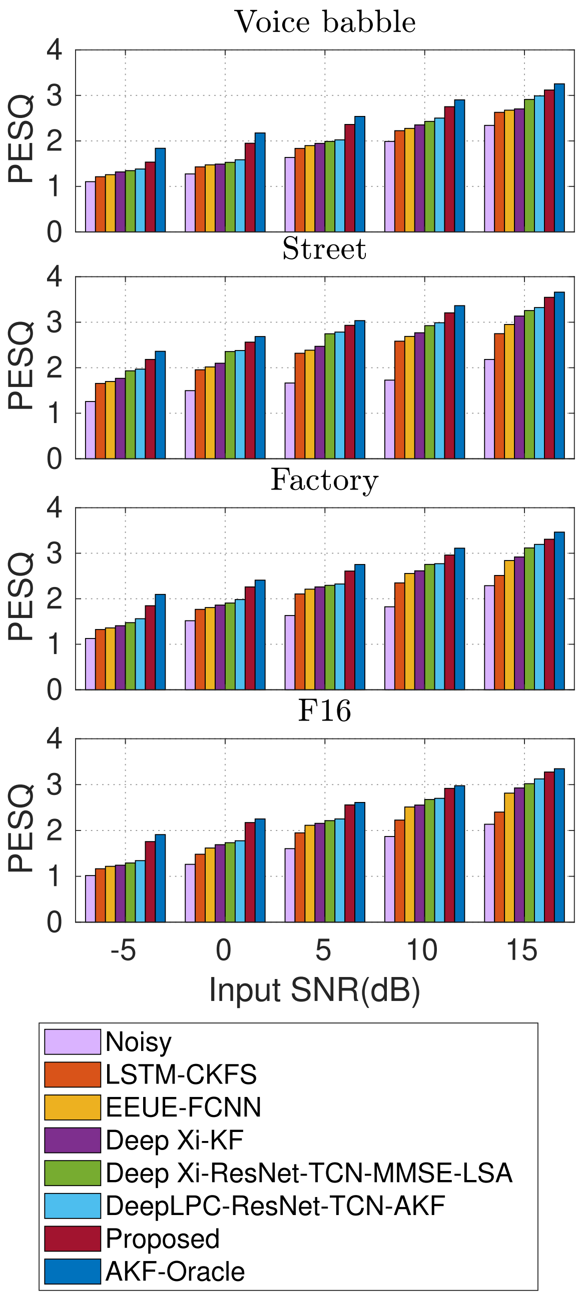

FIGURE 5. PESQ score for each SEA for each condition specified in Section IV-D.

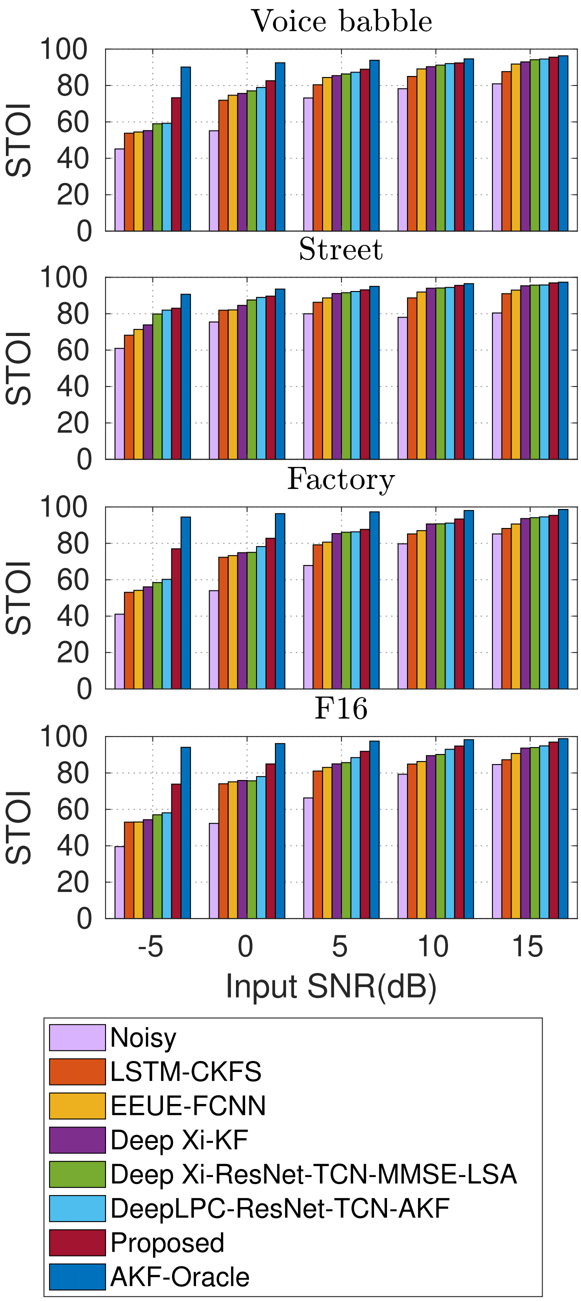

FIGURE 6. STOI score for each SEA for each condition specified in Section IV-D.

attention-based networks are more apt for clean speech LPC estimation than TCNs. This also indicates that the AKF constructed from the clean speech LPC estimates of the proposed method will produce enhanced speech at a higher quality and intelligibility than the competing methods.

B. OBJECTIVE EVALUATION

In this section, we evaluate the objective quality and intelligibility scores attained by the proposed method. The mean objective scores attained by each SEA on the NOIZEUS corpus are shown in Tables 4. It can be seen that AKF-Oracle produces the highest scores for all measures, which can be thought of as the upper boundary of performance. Noisy speech produced the lowest scores for all measures, indicating the lower boundary of performance. When comparing the proposed method to DeepLPC-ResNet-TCN-AKF, it can be seen it attains higher score for each objective measure. This demonstrates that the MHANet is better suited for the AKF than the ResNet-TCN. The proposed method also achieves higher objective scores than any of the competing methods, showing that it is currently the leading AKF in the literature.

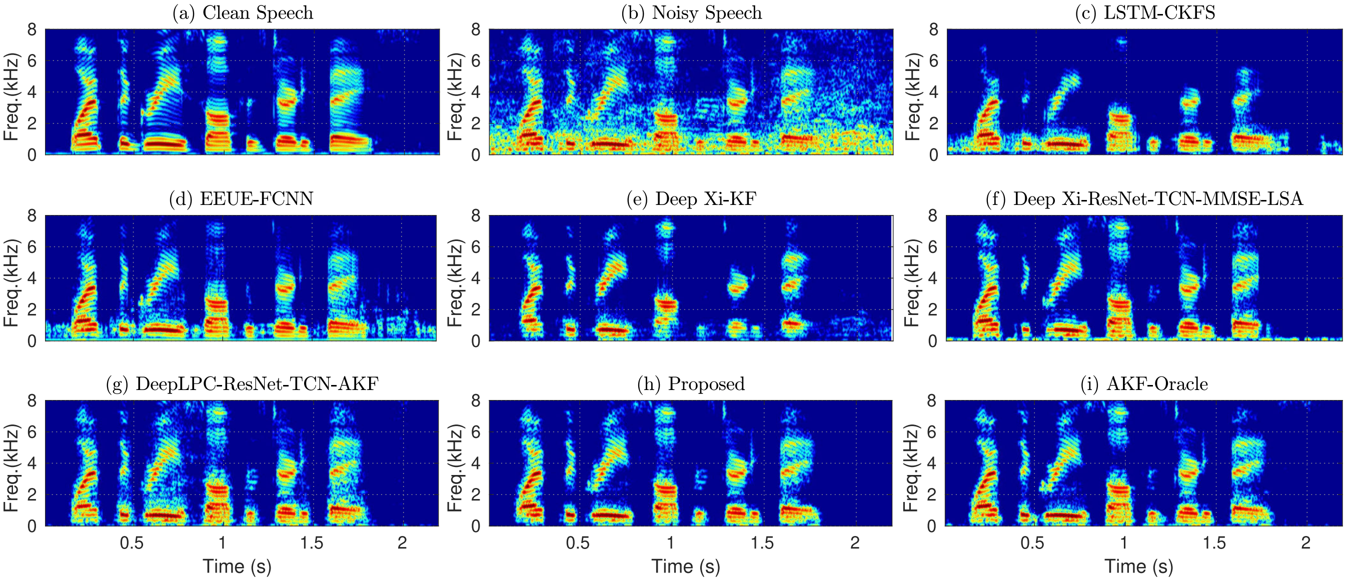

FIGURE 7. Spectrograms of: (a) clean speech (recording *sp05*), (b) noisy speech ((a) corrupted with 5 dB of *voice babble* noise), (c)-(i) enhanced speech produced by each SEA.

Figures 5 and 6 show the PESQ and STOI scores, respectively, of each SEA for multiple conditions. The proposed method produced higher PESQ and STOI scores than the competing SEAs for each condition. This demonstrates that the proposed method is able to produce higher objective quality and intelligibility scores than the competing methods—across multiple SNR levels and noise sources. The results also indicate that the performance of the MHANet is better able to generalise to different conditions than ResNet-TCN.

C. SPECTROGRAM ANALYSIS

In this section, we analyse the enhanced speech spectrograms produced by each SEA. Figure 7 (a) shows the spectrogram of the clean speech recording (male recording *sp05*). The clean speech is corrupted by *voice babble* noise at an SNR level of 5 dB to create the noisy speech shown in Figure 7 (b). This is a particularly tough condition for speech enhancement since the background noise exhibits characteristics similar to the speech produced by the target speaker.

The enhanced speech produced by LSTM-CKFS is shown in Figure 7 (c). It can be seen that LSTM-CKFS significantly reduced the amount of background noise in the noisy speech, although, it produced a significant amount of speech distortion. Figure 7 (d) shows the enhanced speech produced by EEUE-FCNN. This method produced less distorted speech than LSTM-CKFS (Figure 7 (c)), however, residual background noise remains. Less background noise is present in the enhanced speech produced by Deep Xi-KF (Figure 7 (e)) than the enhanced speech produced by EEUE-FCNN (Figure 7 (d)), however, the speech is more distorted. Deep Xi-ResNet-TCN-MMSE-LSA produced less distorted speech (Figure 7 (f)) than that of Deep Xi-KF (Figure 7 (e)). The enhanced speech produced by the DeepLPC-ResNet-TCN-AKF is shown in Figure 7 (g). It can be seen that the enhanced speech of DeepLPC-ResNet-TCN-AKF has less residual background noise and speech distortion than that of Deep Xi-ResNet-TCN-MMSE-LSA (Figure 7 (f)). The enhanced speech produced by the proposed method is shown in Figure 7 (h). It can be seen that there is less residual background noise in the enhanced speech than that of DeepLPC-ResNet-TCN-AKF (Figure 7 (g)). Finally, the enhanced speech produced by the AKF-Oracle method is shown in Figure 7 (i). The enhanced speech of AKF-Oracle is most similar to the clean speech in Figure 7 (a). This is due to AKF-Oracle using the clean speech and noise LPC parameters (which are unobserved in practice).

D. SUBJECTIVE EVALUATION

The mean subjective preference score (%) for each SEA is shown in Figures 8-9. The non-stationary (*voice babble*) noise experiment in Figure 8 reveals that the proposed method is widely preferred (72.23%) by the listeners to that of the competing methods, apart from the clean speech (100%) and the AKF-Oracle method (82.86%). DeepLPC-ResNet-TCN-AKF is found to be the most preferred method (68.43%) amongst the competing SEAs. Amongst the remaining SEAs, the listeners preferred the enhanced speech produced by Deep Xi-ResNet-TCN-MMSE-LSA (62.22%) the most, followed by Deep Xi-KF (53.71%), LSTM-CKFS (40%), and then EEUE-FCNN (38%). LSTM-CKFS was preferred by the listeners more than EEUE-FCNN, even though EEUE-FCNN attained higher objective scores. This may be due to the fact that LSTM-CKFS demonstrates superior noise suppression in regions of speech than EEUE-FCNN, as indicated in [13].

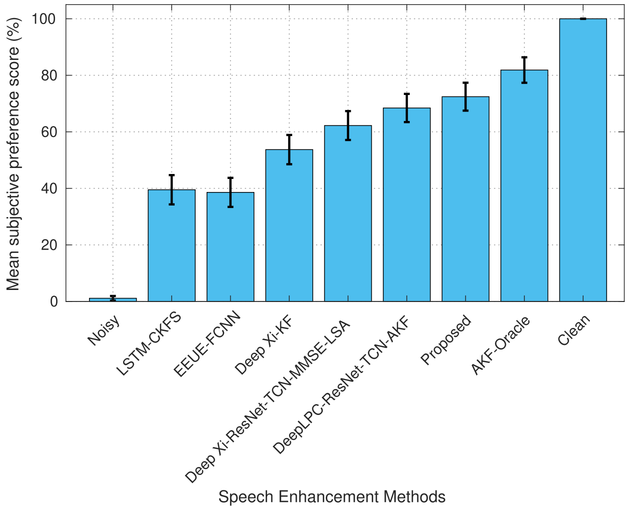

FIGURE 8. **The mean preference score (%) comparison between the proposed and benchmark SEAs for the recording sp05 corrupted with 5 dB non-stationary *voice babble* noise.**

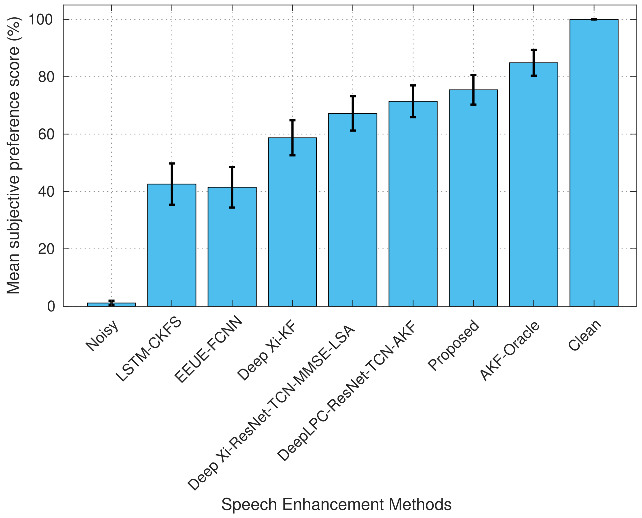

FIGURE 9. **The mean preference score (%) comparison between the proposed and benchmark SEAs for the recording sp27 corrupted with 5 dB coloured *factory* noise.**

For the coloured (*factory*) noise experiment (Figure 9), the listeners again preferred the proposed method (75%) over the competing SEAs, with only clean speech (100%) and AKF-Oracle (84.86%) being more preferred. As in the previous experiment, DeepLPC-ResNet-TCN-AKF was the most preferred amongst the competing methods (71.41%), with Deep Xi-ResNet-TCN-MMSE-LSA being the next most preferred (67.22%), followed by Deep Xi-KF (58.71%). As with the scores in Figure 8, the enhanced speech of LSTM-CKFS was preferred (42%) more than that of EEUE-FCNN (41%). In light of the blind AB listening tests, it is evident to say that the enhanced speech of the proposed method exhibits the best perceived quality amongst all tested methods for both male and female recordings corrupted by real-life non-stationary as well as coloured noises.

VI. CONCLUSION

In this study, we investigate if an attention-based network is more appropriate for AKF-based speech enhancement than a TCN. To this end, we replaced the ResNet-TCN used in the DeepLPC framework with the MHANet. Compared to DeepLPC-ResNet-TCN, the proposed method, DeepLPC-MHANet produces LPC estimates with less bias. Moreover, the AKF constructed with the clean speech and noise LPC parameters estimated from DeepLPC-MHANet is able to attain higher quality and intelligibility scores. We also compared the proposed method to all other deep learning-based KFs and AKFs, and DeepLPC-MHANet-AKF performed best.

The proposed method performs speech enhancement in the presence of additive background noise. However, in practice, speech can be corrupted by background noise and reverberation from surface reflections (or noisy-reverberant speech). Therefore, our future research direction is on Kalman filtering for speech enhancement in the presence of noisy-reverberant speech. Such a Kalman filter will be constructed from parameters estimated using the MHANet.

APPENDIX A
DeepLPC TRAINING TARGET

Here, we describe the training targets for DeepLPC [34]. The clean speech and noise LPC-PS, denoted as $\lambda_s(l, m)$ and $\lambda_v(l, m)$, respectively. During training, $\lambda_s(l, m)$ and $\lambda_v(l, m)$ are computed as in [35, Chapter 9]:

$$\lambda_s(l, m) = \frac{\sigma_w^2}{\left|1 + \sum_{i=1}^{p} a_i e^{-j2\pi im/M}\right|^2}, \tag{25}$$

$$\lambda_v(l, m) = \frac{\sigma_u^2}{\left|1 + \sum_{k=1}^{q} b_k e^{-j2\pi km/M}\right|^2}, \tag{26}$$

where $(\{a_i\}, \sigma_w^2)$ $(p = 16)$ and $(\{b_k\}, \sigma_u^2)$ $(q = 16)$ are computed from the clean speech, $s(n, l)$ and the noise signal, $v(n, l)$ using the autocorrelation method [35, Chapter 8], and $m\epsilon\{0, 1, \ldots, M - 1\}$ $(M = 257)$. As in [34], we used the speech and noise LPC order; $p = 16$ and $q = 16$, respectively.

Next, the dynamic range of $\lambda_s(l, m)$ and $\lambda_v(l, m)$ are compressed to the interval [0, 1] by using the cumulative distribution function (CDF) of $\lambda_s(l, m)_{[dB]}$ and $\lambda_v(l, m)_{[dB]}$, where $\lambda_s(l, m)_{[dB]} = 10\log_{10}(\lambda_s(l, m))$ and $\lambda_v(l, m)_{[dB]} = 10\log_{10}(\lambda_v(l, m))$ [34]. As shown in Figures 10 (a) and (c), $\lambda_s(l, 64)_{[dB]}$ and $\lambda_v(l, 64)_{[dB]}$ follow a Gaussian distribution. Hence, we assume that $\lambda_s(l, m)_{[dB]}$ and $\lambda_v(l, m)_{[dB]}$ are distributed normally with mean μ_s and μ_v, and variance σ_s^2 and σ_v^2, respectively $\left(\lambda_s(l, m)_{[dB]} \sim \mathcal{N}(\mu_s, \sigma_s^2)\right.$ and $\left.\lambda_v(l, m)_{[dB]} \sim \mathcal{N}(\mu_v, \sigma_v^2)\right)$.

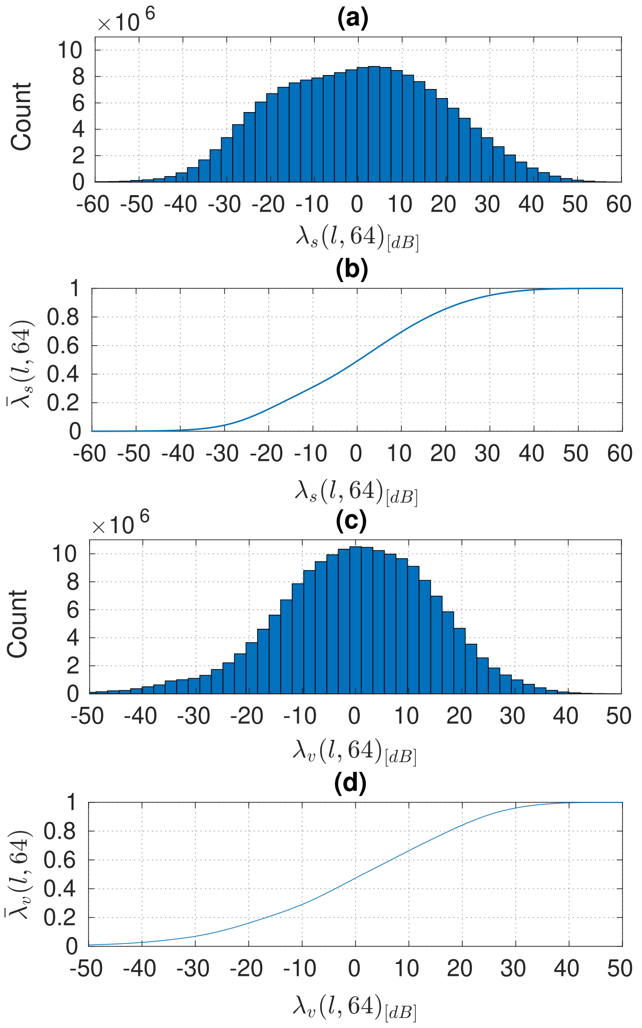

FIGURE 10. **The distribution of (a) $\lambda_s(l,64)_{[dB]}$ and (c) $\lambda_v(l,64)_{[dB]}$. The CDF of (b) $\lambda_s(l,64)_{[dB]}$ and (d) $\lambda_v(l,64)_{[dB]}$, where the sample mean and variance were found over the sample of the training set.**

The statistics of $\lambda_s(l,m)_{[dB]}$ and $\lambda_v(l,m)_{[dB]}$, i.e., (μ_s, σ_s^2) and (μ_v, σ_v^2) for each frequency bin, m were found over a sample of the training set. 2 500 randomly selected clean speech recordings were mixed with 2 500 randomly selected noise recordings from the training set (Section IV-A) with SNR levels ranging from -10 dB to $+20$ dB in 1 dB increments, giving 2 500 noisy speech signals. For each frequency bin, m, the sample mean and variances, (μ_s, σ_s^2) and (μ_v, σ_v^2) were computed from 2 500 concatenated clean speech recordings and scaled noise recordings, respectively. This sample was also used as the sample for Figure 10.

The CDF of $\lambda_s(l,64)_{[dB]}$ over the sample is shown in Figure 10 (b), and is used to compress the dynamic range of $\lambda_s(l,64)_{[dB]}$. Similarly, the CDF of $\lambda_v(l,64)_{[dB]}$ over the sample is shown in Figure 10 (d), and is used to compress the dynamic range of $\lambda_v(l,64)_{[dB]}$. The CDFs of $\lambda_s(l,m)_{[dB]}$ and $\lambda_v(l,m)_{[dB]}$ are defined as [34]:

$$\bar{\lambda}_s(l,m) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{\lambda_s(l,m)_{[dB]} - \mu_s}{\sigma_s\sqrt{2}}\right)\right], \tag{27}$$

$$\bar{\lambda}_v(l,m) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{\lambda_v(l,m)_{[dB]} - \mu_v}{\sigma_v\sqrt{2}}\right)\right]. \tag{28}$$

APPENDIX B
DeepLPC INFERENCE

During inference, $\hat{\zeta}_l$ is first split into $\hat{\bar{\lambda}}_s(l,m)$ and $\hat{\bar{\lambda}}_v(l,m)$. The LPC-PS of the clean speech and the noise signal are then computed from $\hat{\bar{\lambda}}_s(l,m)$ and $\hat{\bar{\lambda}}_v(l,m)$ as:

$$\hat{\lambda}_s(l,m) = 10^{((\sigma_s\sqrt{2}\operatorname{erf}^{-1}(2\hat{\bar{\lambda}}_s(l,m)-1)+\mu_s)/10)}, \tag{29}$$

$$\hat{\lambda}_v(l,m) = 10^{((\sigma_v\sqrt{2}\operatorname{erf}^{-1}(2\hat{\bar{\lambda}}_v(l,m)-1)+\mu_v)/10)}. \tag{30}$$

Next, $(\{\hat{a}_i\}, \hat{\sigma}_w^2)$ and $(\{\hat{b}_k\}, \hat{\sigma}_u^2)$ are computed from $\hat{\lambda}_s(l,m)$ and $\hat{\lambda}_v(l,m)$, as described in [34].

REFERENCES

[1] P. C. Loizou, *Speech Enhancement: Theory Practics*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2013.

[2] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 2, pp. 113–120, Apr. 1979.

[3] S. Kamath and P. Loizou, "A multi-band spectral subtraction method for enhancing speech corrupted by colored noise," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2002, pp. 4160–4164.

[4] K. Paliwal, K. Wójcicki, and B. Schwerin, "Single-channel speech enhancement using spectral subtraction in the short-time modulation domain," *Speech Commun.*, vol. 52, no. 5, pp. 450–475, May 2010.

[5] 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. Proc.*, May 1996, pp. 629–632.

[6] 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.

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

[8] 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.

[9] K. Paliwal and A. Basu, "A speech enhancement method based on Kalman filtering," in *Proc. ICASSP 87. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1987, pp. 177–180.

[10] J. D. Gibson, B. Koo, and S. D. Gray, "Filtering of colored noise for speech enhancement and coding," *IEEE Trans. Signal Process.*, vol. 39, no. 8, pp. 1732–1742, Aug. 1991.

[11] G. J. Brown and D. Wang, *Separation of Speech by Computational Auditory Scene Analysis*. Berlin, Germany: Springer, 2005.

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

[13] A. E. W. George, S. So, R. Ghosh, and K. K. Paliwal, "Robustness metric-based tuning of the augmented Kalman filter for the enhancement of speech corrupted with coloured noise," *Speech Commun.*, vol. 105, pp. 62–76, Dec. 2018.

[14] S. K. Roy, A. Nicolson, and K. K. Paliwal, "Deep learning with augmented Kalman filter for single-channel speech enhancement," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Oct. 2020, pp. 1–5.

[15] S. K. Roy, W.-P. Zhu, and B. Champagne, "Single channel speech enhancement using subband iterative Kalman filter," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 762–765.

[16] 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.

[17] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[18] N. Saleem, M. I. Khattak, M. Al-Hasan, and A. B. Qazi, "On learning spectral masking for single channel speech enhancement using feedforward and recurrent neural networks," *IEEE Access*, vol. 8, pp. 160581–160595, 2020.

[19] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 708–712.

[20] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 3, pp. 483–492, Mar. 2016.

[21] N. Zheng and X.-L. Zhang, "Phase-aware speech enhancement based on deep neural networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 27, no. 1, pp. 63–76, Jan. 2019.

[22] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, and T. Zhang, "Learning spectral mapping for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 6, pp. 982–992, Jun. 2015.

[23] A. Nicolson and K. K. Paliwal, "Deep learning for minimum mean-square error approaches to speech enhancement," *Speech Commun.*, vol. 111, pp. 44–55, Aug. 2019.

[24] Q. Zhang, A. Nicolson, M. Wang, K. K. Paliwal, and C. Wang, "DeepMMSE: A deep learning approach to MMSE-based noise power spectral density estimation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1404–1415, 2020.

[25] A. Nicolson and K. K. Paliwal, "Masked multi-head self-attention for causal speech enhancement," *Speech Commun.*, vol. 125, pp. 80–96, Dec. 2020.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. U. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 5998–6008.

[27] 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.

[28] 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.

[29] C. Pickersgill, S. So, and B. Schwerin, "Investigation of DNN prediction of power spectral envelopes for speech coding & ASR," *Proc. 17th Speech Sci. Technol. Conf.*, Sydney, NSW, Australia, Dec. 2018, pp. 181–184.

[30] H. Yu, Z. Ouyang, W.-P. Zhu, B. Champagne, and Y. Ji, "A deep neural network based Kalman filter for time domain speech enhancement," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2019, pp. 1–5.

[31] H. Yu, W.-P. Zhu, and B. Champagne, "Speech enhancement using a DNN-augmented colored-noise Kalman filter," *Speech Commun.*, vol. 125, pp. 142–151, Dec. 2020.

[32] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, "Codebook driven short-term predictor parameter estimation for speech enhancement," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no. 1, pp. 163–176, Jan. 2006.

[33] S. K. Roy, A. Nicolson, and K. K. Paliwal, "A deep learning-based Kalman filter for speech enhancement," in *Proc. Interspeech*, Oct. 2020, pp. 2692–2696.

[34] S. K. Roy, A. Nicolson, and K. K. Paliwal, "DeepLPC: A deep learning approach to augmented Kalman filter-based single-channel speech enhancement," *IEEE Access*, early access, Apr. 23, 2021, doi: 10.1109/ACCESS.2021.3075209.

[35] S. V. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction*. Hoboken, NJ, USA: Wiley, 2006.

[36] L. J. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *CoRR*, vol. abs/1607.06450, pp. 1–4, Oct. 2016.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[38] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 5206–5210.

[39] C. Veaux, J. Yamagishi, and K. MacDonald, "CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit," in *Proc. Univ. Edinburgh. The Centre Speech Technol. Res. (CSTR)*, 2017.

[40] J. S. Garofolo, L. F. Lamel, W. M. Fisher, and J. G. D. S. Fiscus and Pallett, "DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 4930, Feb. 1993.

[41] D. B. Dean, S. Sridharan, R. J. Vogt, and M. W. Mason, "The QUT-NOISE-TIMIT corpus for the evaluation of voice activity detection algorithms," in *Proc. Interspeech*, 2010, pp. 3110–3113.

[42] G. Hu, "100 nonspeech environmental sounds," Dept. Comput. Sci. Eng., The Ohio State Univ., Columbus, OH, USA, Tech. Rep., 2004.

[43] F. Saki, A. Sehgal, I. Panahi, and N. Kehtarnavaz, "Smartphone-based real-time classification of noise signals using subband features and random forest classifier," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 2204–2208.

[44] F. Saki and N. Kehtarnavaz, "Automatic switching between noise classification and speech enhancement for hearing aid devices," in *Proc. 38th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2016, pp. 736–739.

[45] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," *CoRR*, vol. abs/1510.08484, pp. 1–8, Oct. 2015.

[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Tech. Rep., 2014.

[47] A. Gray and J. Markel, "Distance measures for speech processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, no. 5, pp. 380–391, Oct. 1976.

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

[49] 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 Process.*, May 2001, pp. 749–752.

[50] 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.

[51] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR–half-baked or well done?" in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 626–630.

[52] P. Mermelstein, "Evaluation of a segmental SNR measure as an indicator of the quality of ADPCM coded speech," *J. Acoust. Soc. Amer.*, vol. 66, no. 6, pp. 1664–1667, Dec. 1979.

SUJAN KUMAR ROY (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science and engineering from the University of Rajshahi, Bangladesh, in 2008 and 2010, respectively, and the Master of Applied Science (M.A.Sc.) degree in electrical and computer engineering from Concordia University, Canada, in May 2016. He is currently pursuing the Ph.D. degree with the School of Engineering, Griffith University, Brisbane, Australia. His research interests include speech processing, machine learning, and data science.

AARON NICOLSON was born in Brisbane, Australia, in 1994. He received the B.Eng. (Hons.) and Ph.D. degrees from Griffith University, Brisbane, Australia, in 2016 and 2020, respectively. He is currently a Postdoctoral Research Fellow with the Australian eHealth Research Centre, CSIRO. His research interests include speech, natural language, image, and multimodal processing using deep learning.

KULDIP K. PALIWAL was born in Aligarh, India, in 1952. He received the B.S. degree from Agra University, Agra, India, in 1969, and the M.S. degree from Aligarh Muslim University, Aligarh, in 1971, and the Ph.D. degree from the University of Bombay, Bombay, India, in 1978. He has worked with a number of organizations, including the Tata Institute of Fundamental Research, Bombay, India; the Norwegian Institute of Technology, Trondheim, Norway; the University of Keele, U.K.; AT&T Bell Laboratories, Murray Hill, NJ, USA; AT&T Shannon Laboratories, Florham Park, NJ, USA; and Advanced Telecommunication Research Laboratories, Kyoto, Japan. Since July 1993, he has been a Professor with the School of Microelectronic Engineering, Griffith University, Brisbane, Australia. He has also been carrying out research in the area of speech processing, since 1972. He has published more than 300 articles in these research areas. He has co-edited two books *Speech Coding and Synthesis* (Elsevier) and *Speech and Speaker Recognition: Advanced Topics* (Kluwer). His current research interests include speech recognition, speech coding, speaker recognition, speech enhancement, face recognition, image coding, pattern recognition, and artificial neural networks. He has served as a Founding Member for the IEEE Signal Processing Society's Neural Networks Technical Committee, from 1991 to 1995, and the Speech Processing Technical Committee, from 1999 to 2003. He is currently a Fellow of the Acoustical Society of India. He received the IEEE Signal Processing Society's Best (Senior) Paper Award for his paper on LPC quantization, in 1995. He was the General Co-Chair of the Tenth IEEE Workshop on Neural Networks for Signal Processing (NNSP2000). He was an Associate Editor of the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, from 1994 to 1997 and from 2003 to 2004. From 1997 to 2000, he also served as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS. From 2005 to 2011, he served as the Editor-in-Chief for the *Speech Communication* Journal (Elsevier). He is on the Editorial Board of the *IEEE Signal Processing Magazine*.