$F0$ Estimation and Voicing Detection With Cascade Architecture in Noisy Speech

Yixuan Zhang, Heming Wang, *Student Member, IEEE*, and DeLiang Wang, *Fellow, IEEE*

***Abstract*—As a fundamental problem in speech processing, pitch tracking has been studied for decades. While strong performance has been achieved on clean speech, pitch tracking in noisy speech is still challenging. Severe non-stationary noises not only corrupt the harmonic structure in voiced intervals but also make it difficult to determine the existence of voiced speech. Given the importance of voicing detection for pitch tracking, this study proposes a neural cascade architecture that jointly performs pitch estimation and voicing detection. The cascade architecture optimizes a speech enhancement module and a pitch tracking module, and is trained in a speaker-independent and noise-independent way. It is observed that incorporating the enhancement module improves both pitch estimation and voicing detection accuracy, especially in low signal-to-noise ratio (SNR) conditions. In addition, compared with frameworks that combine corresponding single-task models, the proposed multi-task framework achieves better performance and is more efficient. Experimental results show that the proposed method is robust to different noise conditions and substantially outperforms other competitive pitch tracking methods.**

***Index Terms*—Complex domain processing, densely-connected convolutional recurrent neural network, multi-task learning, neural cascade architecture, pitch tracking, voicing detection.**

I. Introduction

PITCH tracking, also known as fundamental frequency ($F0$) estimation, is a fundamental research problem in the domain of speech and music processing. Pitch tracking plays an important role in many applications including speech synthesis, speaker identification, and speech analysis. While high $F0$ estimation accuracy has been achieved for speech in clean conditions, pitch tracking in noisy conditions is challenging. In noisy conditions, the harmonic structure of speech signals is corrupted, making $F0$ estimation difficult. In addition, non-stationary noises complicate the task of classifying voiced and unvoiced intervals. Strictly speaking pitch and fundamental frequency are different concepts: pitch is a perceptual attribute while $F0$ is an acoustic property of a signal. The two terms are often used interchangeably in the literature, which is followed in this article for convenience.

Existing pitch tracking approaches can be classified into three categories based on input features. Time-domain methods extract $F0$ information by estimating the periodicity in a signal. Typical time-domain methods, such as YIN [1], RAPT [2], and PRAAT [3], are based on auto-correlation functions. Frequency domain methods such as SAFE [4] and PEFAC [5] determine $F0$ by analyzing harmonic structure in the frequency domain. PEFAC, for example, attenuates noises on a spectrogram by applying non-linear amplitude compression, and selects pitch candidates from harmonic peaks. Time-frequency methods such as Wu et al. [6] perform pitch tracking in both the time and frequency domain. The signal is often decomposed into sub-bands and temporal analysis is applied to each frequency band. For conventional methods, after determining pitch candidates, post-processing algorithms such as dynamic programming [5] and hidden Markov models [6], [7] are usually applied to produce the most probable pitch track from the pitch candidates.

Deep neural networks (DNNs) have been introduced to pitch tracking in recent years. DNN-based methods have achieved substantial improvements over traditional signal processing methods. These methods formulate pitch tracking as either a multi-class classification problem or a regression problem. In the first such study, Han and Wang [8] focused on pitch tracking in very noisy conditions, and proposed to estimate probabilistic outputs of pitch states with DNNs. Viterbi decoding was then used to produce the pitch contours based on the DNN outputs. The trained model shows robustness to different noise conditions and better results over conventional methods. Different from Han and Wang [8] whose model operates on spectral inputs, recent DNN methods such as CREPE [9], FCN [10], DeepF0 [11], and penn [12] perform pitch tracking on raw waveform. These methods use convolutional neural network (CNN) models and estimate the probabilistic outputs of pitch states. In the evaluation process, the final pitch estimates are obtained directly from the network outputs without post-processing. However, it is worth noting that studies such as [9], [10] include Viterbi decoding as an optional procedure in their posted code. In addition, these methods use synthesized data for training where ground-truth $F0$ is guaranteed. The use of synthesized speech for training resolves the issue of how to generate ground-truth pitch labels for speech signals. However, a common drawback of these methods is that they focus on pitch tracking in voiced frames and ignore voicing detection. For a pitch tracker, the ability to

Manuscript received 5 September 2022; revised 7 June 2023 and 16 August 2023; accepted 27 August 2023. Date of publication 13 September 2023; date of current version 20 October 2023. This work was supported in part by NIDCD under Grant R01DC012048 and in part by Ohio Supercomputer Center. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Juan Ignacio Godino-Llorente. *(Corresponding author: Yixuan Zhang.)*

Yixuan Zhang and Heming Wang are with the Department of Computer Science and Engineering, Ohio State University, Columbus, OH 43210 USA (e-mail: zhang.7388@osu.edu; wang.11401@osu.edu).

DeLiang Wang is with the Department of Computer Science and Engineering and the Center for Cognitive and Brain Sciences, Ohio State University, Columbus, OH 43210 USA (e-mail: dwang@cse.ohio-state.edu).

Digital Object Identifier 10.1109/TASLP.2023.3313427

classify voiced/unvoiced intervals is crucial, especially in noisy conditions. For human speech, around 25% of speech signals are unvoiced [13]. When producing a pitch track, unvoiced intervals must be classified as such. In noisy conditions, voicing detection becomes more difficult. Studies such as [14], [15] incorporate voicing detection as a separate task for pitch tracking. For example, in [15], a pitch tracker is integrated into a speech enhancement framework for hearing aids, which is trained to produce pitch and voiced probability estimates.

In a preliminary study [16], we proposed a densely-connected convolutional recurrent neural network (DC-CRN) model for pitch tracking in noisy speech, motivated by the DC-CRN model [17] for speech enhancement. The model takes complex-domain short-time Fourier transform (STFT) [18], [19] as input and produces probabilistic pitch state outputs. Furthermore, a neural cascade architecture [20] is employed to jointly optimize speech enhancement and pitch tracking DC-CRN model. The current study expands the preliminary study mainly in the following aspects. First, we consider voicing detection in pitch tracking, which was not considered in the preliminary study, and extend the previous architecture to perform multi-task learning for $F0$ estimation and voicing detection. Second, we analyze the differences between single-task learning and multi-task learning, and establish the superiority of the latter for $F0$ estimation as well as voicing detection. Third, we evaluate the influence of incorporating phase information in input features on the multi-task learning framework. Fourth, we evaluate the proposed models on both synthesized and recorded speech data. Experimental results show that the models trained on synthesized speech generalize well to real speech recordings and incorporating phase in input features improves model performance. Compared to frameworks that combine corresponding single-task models, the multi-task framework also saves computational costs. Our neural cascade model substantially outperforms the baseline methods on both $F0$ estimation and voicing detection.

The rest of this article is organized as follows. Related work is discussed in the next section. In Section III, we formulate the $F0$ estimation problem. In Section IV, we provide the details of model architecture and DNN configurations. Section V describes the experimental setup, and Section VI provides evaluation and comparison results. Section VII concludes this article.

II. Related Work

In recent years, multi-task learning has been used in pitch tracking and shown to have advantages over single-task models. In [21], the task of speaker separation is optimized simultaneously with multi-pitch tracking. The results show that the two tasks can benefit each other. In [22], the authors investigated different training architectures to jointly perform singing voice separation and pitch tracking, and showed that a stacked architecture which first performs vocal separation outperforms other joint training models. In [14], a multi-task DNN is developed for pitch tracking in noisy speech, where a convolutional encoder followed by two fully-connected layers is designed for joint voiced/unvoiced classification and pitch regression. Our preliminary study mentioned above also performs multi-task learning that jointly optimizes a speech enhancement module and a pitch tracking module. We adopted DC-CRN [17] as the speech enhancement module. The DC-CRN model has an encoder-decoder structure with skip connections from the encoder to the decoder. The encoder is composed of several convolutional DC blocks. The output from the convolutional encoder is first reshaped into a sequence of 1D features and then fed to a 2-layer BLSTM. The output of BLSTM is reshaped to 3-D representation again and fed to the decoder which has several deconvolutional DC blocks. Convolutional DC block-based skip pathways are used from the encoder to the decoder, similar to U-Net++ [23]. The skip pathways help to enrich the feature maps from the encoder.

III. Problem Formulation

$F0$ estimation and voicing detection are both important for pitch tracking. Following recent data-driven methods [8], [9], [10], we formulate $F0$ estimation as a multi-class classification problem. In this formulation, each class represents a pitch state with a unique $F0$. A DNN takes a noisy speech signal as input and outputs a vector in which each element represents the probability of $F0$ belonging to the corresponding pitch state. We adopt the $F0$ range described in FCN [10], which is from 30 Hz to 1000 Hz. This $F0$ range covers all possible $F0$ values in human voices, including scenarios such as soprano singing and vocal fry. Pitch states are selected from the target $F0$ range with 12.5-cent intervals. As a result, the target $F0$ vector $\boldsymbol{y}_p$ contains 486 elements $y_p^1, y_p^2, \ldots, y_p^{486}$, each of which corresponds to one pitch state. The value of the ith element y_p^i in the target vector $\boldsymbol{y}_p$ is calculated as,

$$y_p^i = \exp\left[-\frac{(p_i - p_{true})^2}{2 \cdot 25^2}\right], \tag{1}$$

where p_i and p_{true} are the $F0$ of the ith pitch state and $F0$ of the ground-truth pitch state in cents, respectively. Compared to a one-hot vector, the target vector is Gaussian-blurred with a standard deviation of 25 cents, mainly for reducing the penalty for near-correct estimates.

Having an estimated $F0$ vector $\hat{\boldsymbol{y}}_p$, the pitch estimate can be calculated as shown in (2) below. The index I corresponding to the maximum element in $\hat{\boldsymbol{y}}_p$ is first selected. With the index I determined, the pitch estimate is the weighted average of Ith pitch state and its neighboring pitch states $p_{I-4}, \ldots, p_{I+4}$. When the indices of some neighboring pitch states are out of range (i.e. < 1 or > 486), the pitch estimate is the weighted average of the pitch states whose indices are within the range. For the convenience of evaluation, the calculated $\hat{p}$ is converted to Hz,

$$I = \arg\max_i \ \hat{y}_p^i, \qquad \hat{p} = \frac{\sum_{i=I-4}^{I+4} \hat{y}_p^i p_i}{\sum_{i=I-4}^{I+4} \hat{y}_p^i}. \tag{2}$$

Natural speech contains both voiced and unvoiced intervals. A voiced interval often refers to an interval where the voice is periodic or quasi-periodic. An unvoiced interval, however, lacks harmonic structure and acoustically resembles noise [13]. In real speech recordings, silence and pure noise intervals also exist. In our study, we consider voicing detection as a binary classification problem. An audio frame is classified as a voiced frame only when it contains a periodic or quasi-periodic speech signal and its training target y_v is set to 1. The training target y_v is set to 0 for unvoiced speech, pure noise, and silence.

During inference, the estimated $\hat{y}_v$ is compared with the threshold of 0.5. If $\hat{y}_v$ is higher than 0.5, the frame is considered as a voiced frame; otherwise, the frame is viewed as a frame having no voiced speech. That is,

$$I_{\hat{y}_v} = \begin{cases} 1 & \text{if } \hat{y}_v > 0.5 \\ 0 & \text{if } \hat{y}_v \leq 0.5 \end{cases}, \tag{3}$$

where $I_{\hat{y}_v}$ is a binary number that indicates if the frame is a voiced frame.

The final pitch value $\hat{F}0$ is generated based on $\hat{p}$ and $I_{\hat{y}_v}$, given below

$$\hat{F}0 = \hat{p} \cdot I_{\hat{y}_v}. \tag{4}$$

In other words, the pitch values of all non-voiced frames are set to 0.

IV. Model Description

A. *Densely-Connected Convolutional Recurrent Network for Pitch Tracking*

In our previous study [16], we developed a densely-connected convolutional recurrent network (DC-CRN) for $F0$ estimation, but the previous model only considered $F0$ estimation in voiced frames. The present study takes voicing detection into account, and we expand the previous DC-CRN model to estimate both $\hat{y}_v$ and $\hat{\boldsymbol{y}}_p$. A multi-task learning objective is used to optimize voicing detection and $F0$ estimation simultaneously.

The proposed network architecture is shown in Fig. 1. The network is composed of 7 convolutional densely-connected (Conv-DC) blocks, a two-layer bidirectional long short-term memory (BLSTM) block, and two fully-connected layers with sigmoidal activation functions which produce $\hat{y}_v$ and $\hat{\boldsymbol{y}}_p$ respectively. The input to the network is a concatenation of the real and imaginary parts of the complex STFT of a noisy speech signal.

Fig. 2 shows the structure of a Conv-DC block in the DC-CRN network. The Conv-DC block consists of 4 composite layers followed by a gated convolutional layer. The input to each composite layer or gated convolutional layer is a concatenation of the outputs from all preceding layers. The dense connectivity enables a layer to reuse features computed in the preceding layers, which improves the information flow between layers. Each composite layer contains a 2-dimensional (2D) convolutional layer followed by batch normalization and the exponential linear unit (ELU). As shown in Fig. 2(b), the last layer is a gated convolutional layer that incorporates gated linear units developed in [24].

In order to reduce the number of trainable parameters and improve computational efficiency, a grouping strategy proposed by Gao et al. [25] is adopted. We observe that this method reduces computational complexity while not introducing much performance degradation. Fig. 3 shows an illustration of the grouping strategy for two-layer BLSTM. The group number is

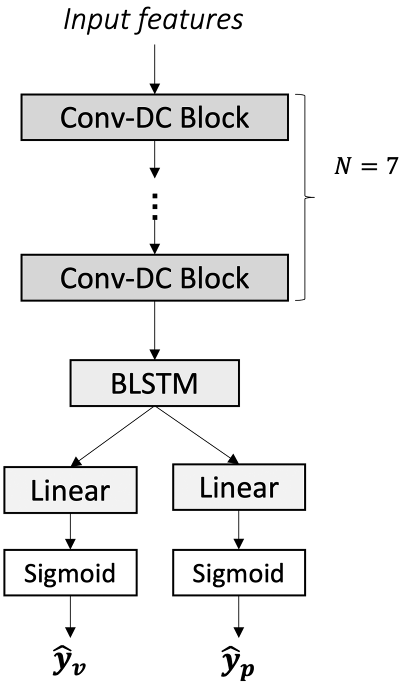
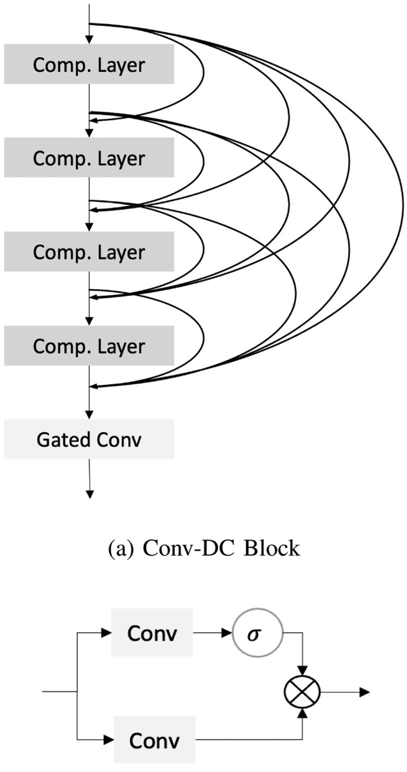


Fig. 1. DC-CRN architecture for pitch tracking. $\hat{\boldsymbol{y}}_p$ represents the output for $F0$ estimation and $\hat{y}_v$ the output for voicing detection. N denotes the number of Conv-DC blocks, and 'Linear' refers to fully-connected layer.



(a) Conv-DC Block

(b) Gated Convolution

Fig. 2. Diagrams of (a) a DC-CRN block and (b) gated convolution. In (a), each composite (comp.) layer contains a convolutional layer followed by batch normalization and exponential linear unit (ELU) activation function. In (b), σ denotes a sigmoidal function and $\otimes$ represents element-wise multiplication.

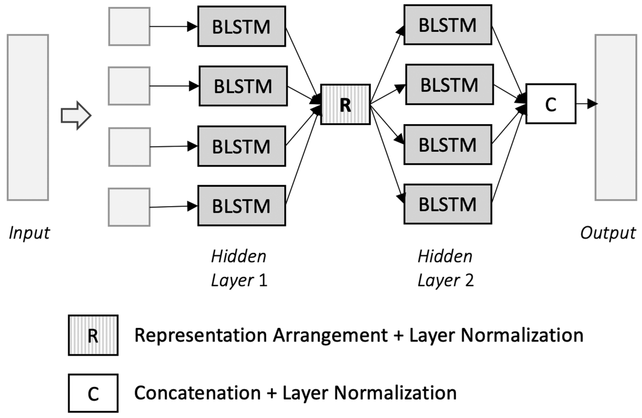

Input
BLSTM
BLSTM
BLSTM
BLSTM
R
BLSTM
BLSTM
BLSTM
BLSTM
C
Output
Hidden Layer 1
Hidden Layer 2
R Representation Arrangement + Layer Normalization
C Concatenation + Layer Normalization

Fig. 3. Grouping strategy for two-layer BLSTM. The group number is set to 4.

set to 4. The input features and hidden states of the first and second recurrent layers are first split into 4 disjoint groups. In the first recurrent layer, intra-group features are learned within each group. Then, the outputs from the first recurrent layer are rearranged and fed into the second recurrent layer to model inter-group dependency. Layer normalization is applied after each recurrent layer. The final output is a concatenation of the outputs from the second recurrent layer.

In order to learn the probabilistic outputs $\hat{y}_v$ and $\hat{\boldsymbol{y}}_p$, we train the DC-CRN network by minimizing the cross entropy loss $\mathcal{L}_v$ for voicing detection and $\mathcal{L}_p$ for $F0$ estimation. The loss functions are given as follows,

$$\mathcal{L}_v(y_v, \hat{y}_v) = -y_v \log \hat{y}_v - (1 - y_v) \log (1 - \hat{y}_v), \tag{5}$$

$$\mathcal{L}_p(\boldsymbol{y}_p, \hat{\boldsymbol{y}}_p) = \frac{1}{N} \sum_{i=1}^{N} [-y_p^i \log \hat{y}_p^i - (1 - y_p^i) \log (1 - \hat{y}_p^i)], \tag{6}$$

where N is the number of pitch estimates and equals 486 in our setting.

A multi-task learning objective is used to jointly optimize voicing detection and $F0$ estimation,

$$\mathcal{L}_{F0} = \mathcal{L}_v + \alpha \mathcal{L}_p, \tag{7}$$

where α is a coefficient for $\mathcal{L}_p$. The relative training efforts between the two tasks are controlled by tuning the coefficient.

B. Neural Cascade Architecture

Considering the rapid advances in speech enhancement in recent years, a natural question is: Can we use speech enhancement to help pitch tracking in noisy speech? It was observed in our preliminary study that estimating $F0$ from enhanced speech with a model trained on clean speech works reasonably well. However, the performance of such $F0$ estimation is limited because of the distortion introduced by a speech enhancement model. Inspired by a recent study [20], we develop a neural cascade architecture to incorporate speech enhancement into pitch tracking.

Fig. 4 shows the proposed cascade architecture. Our model is composed of a speech enhancement module and a pitch tracking module. For the speech enhancement module, we employ the DC-CRN model in [17]. For the pitch tracking module, the proposed DC-CRN model for pitch tracking is used. An input feature has three dimensions: frequency, time, and channel. To form the input feature, the real and imaginary parts $\mathbf{X}_r$, $\mathbf{X}_i$ of the complex STFT of the noisy speech signal are concatenated and viewed as two separate channels. The speech enhancement module estimates the real and imaginary parts $\hat{\mathbf{S}}_r$, $\hat{\mathbf{S}}_i$ of the complex STFT of clean speech. The input to the pitch tracking module is formed by concatenating $\hat{\mathbf{S}}_r$, $\hat{\mathbf{S}}_i$ with $\mathbf{X}_r$, $\mathbf{X}_i$ respectively. The pitch tracking module generates $\hat{y}_v$ and $\hat{\boldsymbol{y}}_p$, which will be utilized to calculate the final pitch track as described in Section III. Inspired by [21], the speech enhancement module and the pitch tracking module are jointly trained by minimizing $\mathcal{L}$, which is defined as,

$$\mathcal{L} = \beta \mathcal{L}_{enh} + \mathcal{L}_{F0}, \tag{8}$$

where $\mathcal{L}_{F0}$ is defined in (7), and β is a parameter to control the relative training effort between speech enhancement and pitch tracking.

The loss function for speech enhancement is defined as,

$$\begin{aligned}\mathcal{L}_{enh} = \frac{1}{TF} \sum_{t,f} [&|\hat{S}_r(t,f) - S_r(t,f)| \\ &+ |\hat{S}_i(t,f) - S_i(t,f)| \\ &+ ||\hat{S}(t,f)| - |S(t,f)||],\end{aligned} \tag{9}$$

where T denotes the number of time frames and F denotes the number of frequency bins, and the third term represents a magnitude-based loss. The network directly learns the real and imaginary parts of the complex STFT of the clean speech $\hat{\mathbf{S}}_r$, $\hat{\mathbf{S}}_i$ by minimizing $\mathcal{L}_{enh}$. We note that prior research [17], [26] suggests that integrating a magnitude term yields better results, which can be attributed to the greater significance of magnitude compared to phase.

C. Network Configurations

1) DC-CRN for Pitch Tracking: As portrayed in Fig. 1 and described in Section IV-A, the pitch tracking DC-CRN model has 7 convolutional densely-connected (DC) blocks followed by a two-layer BLSTM and two fully-connected layers with sigmoidal activation functions. In each convolutional densely-connected block, there are 4 composite layers and a gated convolutional layer. The convolutional layer in each composite layer uses a kernel size of 1×3 (time $\times$ frequency) and has 8 output channels. Zero padding of size 1 is applied along the frequency dimension to both sides. In the gated convolutional layer, the convolutions use a kernel size of 1×4, a stride of 2, and zero-padding of 1 for both sides along the frequency dimension. The convolutional DC blocks have 4, 8, 16, 32, 64, 128, and 256 output channels respectively. The two-layer BLSTM has 512 units in every direction.

2) DC-CRN for Speech Enhancement: To construct the speech enhancement module, we adopt the DC-CRN model

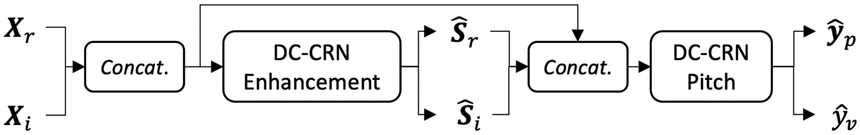


Fig. 4. Illustration of the neural cascade architecture. $\boldsymbol{X}_r$ and $\boldsymbol{X}_i$ denote real and imaginary parts of the complex STFT of an input mixture. The speech enhancement module takes $\boldsymbol{X}_r$ and $\boldsymbol{X}_i$ as input and estimates the complex STFT of the clean speech signal $\boldsymbol{S}$. The complex STFT of the enhanced signal and noisy input mixture are concatenated and fed into the pitch tracking module which jointly performs $F0$ estimation and voicing detection.

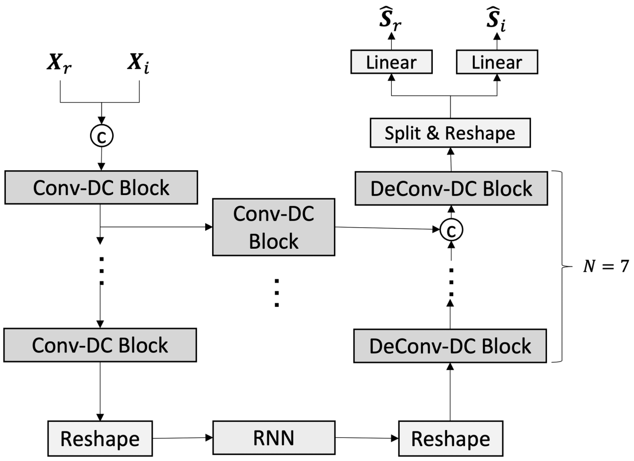


Fig. 5. DC-CRN architecture for speech enhancement. $\boldsymbol{X}_r$ and $\boldsymbol{X}_i$ denote real and imaginary parts of the complex STFT of an input mixture. $\hat{\mathbf{S}}_r, \hat{\mathbf{S}}_i$ denote the real and imaginary parts of estimated speech. © represents concatenation. N denotes the number of DeConv-DC blocks.

in [17] with some adjustments in the network configuration to fit input features. Fig. 5 shows a diagram of the network structure. The encoder and decoder is composed of seven convolutional and deconvolutional DC blocks respectively. The convolutional DC block in the speech enhancement module has the same configuration as the one in the pitch tracking module, depicted in Fig. 2. Compared to a convolutional DC block, a deconvolutional DC block's last layer is a gated deconvolutional layer where the convolutional layers in the DC block are replaced by the transposed convolutional layers. The convolutional or deconvolutional DC block shares the same hyperparameters as the convolutional DC block described in Section IV-C1 except that the output channels of the DC blocks in the encoder, decoder and pathways are 4, 8, 16, 32, 64, 128, 256 successively. In addition, the last layer in the convolutional DC blocks in the skip pathways has a stride size of 1 and a kernel size of 1×3. The two-layer BLSTM model has 512 units in each direction. The grouping strategy is applied to the BLSTM model, with the number of groups set to 4.

V. Experimental Setup

A. Data Generation With Speech Synthesis

Finding sufficient data with reliable ground-truth $F0$ labels for training is challenging. Some studies of pitch tracking in noisy [8] or multi-talker [27] scenarios use the estimated $F0$ contours obtained from applying conventional or pre-trained DNN pitch trackers to clean speech as ground-truth $F0$. Although a relatively accurate $F0$ track can be obtained, the estimation errors made by a pitch tracker cannot be neglected. A better approach uses speech datasets that provide laryngograph recordings. It is usually assumed that ground truth $F0$ can be obtained by applying a pitch estimator to laryngograph recordings. However, it has been observed that the pitch estimates from laryngograph data are not always reliable [10], [28]. In our experience, octave errors can occur. In addition, obtaining laryngograph data is laborious, which makes it difficult to collect large datasets for training.

To acquire a large number of signals with reliable ground-truth $F0$ annotations, recent studies [9], [10], [29] create datasets by synthesizing audio files based on given $F0$ annotations. This analysis-by-synthesis approach allows for complete control on ground-truth $F0$ since the ground-truth $F0$ track used for speech synthesis perfectly matches the given $F0$ annotations. For example, in [10], a vocoder PaN [30] is employed to generate synthesized speech with good quality, where a sequence of pulses is first generated based on the Liljencrants-Fant model of the glottal source [31] with the target $F0$ and then filtered by a vocal tract filter. The complete signal is formed with unvoiced components extracted from the original signal. In this study, we employ a high-quality speech synthesizer, WORLD [32], to create datasets of synthesized audio with given $F0$ labels. In the WORLD vocoder, a pitch track is first estimated by a pitch estimator such as DIO [33]. Then, the spectral envelope is estimated using CheapTrick [34] applied to the original waveform and the estimated $F0$ track. The excitation signal generated by PLATINUM [35] is used as an aperiodic parameter. Finally, the synthesizer takes the estimated pitch track, spectral envelope, aperiodic parameter, and the waveform as inputs and generates a synthesized audio signal. Note that we substitute the pitch tracker in WORLD with Torchcrepe [36], which is a Pytorch implementation of CREPE [9] with pre-trained models. It provides additional methods such as silence detection and removal of unreliable pitch estimates that we find to be helpful for detecting silence and unvoiced intervals more precisely.

B. Dataset Preparation

We first use the WORLD [32] vocoder to generate datasets of synthesized speech using recorded speech from LibriSpeech [37], more specifically the train-clean-360 subset. To build the training set, 4152 recordings from 921 speakers (439 male and 482 female) are selected for speech synthesis. The validation set is created with 1274 untrained recordings. All recordings longer than 6 s are chunked into 6-second segments. When we estimate the pitch track of a recording with Torchcrepe [36], the $F0$ estimates of the frames that are either silent or have a low estimation confidence score are set to 0. The synthesized speech is created with the WORLD synthesizer based on the original waveform and the estimated pitch track. In order to span the full range of target $F0$, each utterance is re-synthesized with pitch tracks that are an octave lower or an octave higher than the original estimated pitch track. Note that the re-synthesized signals that contain any pitch point out of the target $F0$ range are removed from the dataset and all synthesized audio files are downsampled from 16 kHz to 8 kHz. The noisy mixtures in the training set are generated by mixing the synthesized audio and random segments from 10,000 noises from a sound effect library with an SNR randomly chosen from $\{-5, -4, -3, -2, -1, 0\}$ dB. For the validation set, each synthesized audio file is mixed with a cafeteria noise from an Auditec CD at -5 dB SNR.

Since our models are trained on synthesized data, we generate two test sets for evaluation considering the possible bias caused by the synthesizer. One test set contains mixtures generated from synthesized audio. We pick 100 utterances from 10 speakers (5 males and 5 females) in LibriSpeech's clean-train-100 subset and generate synthesized audio with Torchcrepe and WORLD. The other test set is built on real recordings. We choose utterances from the FDA dataset [38] to build a test set of real recordings. Due to the existence of octave errors in the ground-truth $F0$ obtained from the laryngograph data of the FDA dataset, we adopt consensus ground-truth $F0$ from [28] for voiced frames, which derives ground-truth $F0$ from the consensus of state-of-the-art $F0$ estimation algorithms. It is observed that the provided consensus ground truth is broadly compatible with laryngograph-based ground truth and more representative in edge cases that might lead to octave errors such as obscured fundamentals. In our previous study [16], we focused on $F0$ estimation on voiced frames, and the voiced frames with confidence scores of consensus ground-truth $F0$ that are greater than 0.7 were considered for evaluation. In the present evaluation, since we perform voicing detection, we use ground-truth labels from laryngograph data to determine voiced/unvoiced intervals for the test set.

For both test sets, three noises are considered for creating mixtures: babble noise, factory noise from NOISEX92 [39] and cafeteria noise from an Auditec CD. These are all nonstationary and challenging noises for speech enhancement [20]. We consider four SNRs $\{-10, -5, 0, \text{and } 5\}$ dB for testing. For STFT computation, we use a Hamming window of 128 ms duration with a 10 ms frame shift and a 1024-point discrete Fourier transform.

C. Experimental Setup

During training, both the DC-CRN model and the cascade architecture are trained using the Adam optimizer [40] with a batch size of 4. The learning rate is initialized to 0.0005 and is halved if the validation loss does not decrease for 5 consecutive epochs. To avoid gradient explosion, we apply gradient clipping with a maximum value of 5. The maximum training epoch number is set to 80. All models converge within the designated number of training epochs. To find appropriate values of α and β, we explored three ways of tuning the hyperparameters: by fixing α and varying β, by fixing β and varying α, and by equally scaling both. Table I shows the corresponding results on 300 utterances randomly chosen from the validation set. From the results, we observe that $\alpha = 100$ and $\beta = 1$ yield the best result and they are also found to balance the losses in the late training stage. Thus these values are chosen.

TABLE I
Effects of Hyperparameters of α and β in Terms of Raw Pitch Accuracy (RPA) and Voicing Decision Error (VDE) (See Section V-D for Definitions)

α	β	RPA (in %)	VDE (in %)
10	100	69.99	14.30
10	10	**72.49**	14.47
10	1	71.74	**14.01**
10	0.1	70.92	14.22
10	0.01	66.26	15.58
1000	1	68.29	17.39
100	1	**73.28**	**13.85**
10	1	71.74	14.01
1	1	66.37	14.66
0.1	1	57.97	14.14
100	10	**72.61**	15.24
10	1	71.74	**14.01**
1	0.1	60.28	14.18

D. Evaluation Metrics

We evaluate pitch tracking results in terms of two metrics: $F0$ estimation accuracy in voiced frames and the accuracy of frame voicing detection. Specifically, we use raw pitch accuracy (RPA) and voicing decision error (VDE) [41] for pitch tracking evaluations.

When calculating raw pitch accuracy, only the voiced frames are taken into account. In our case, an estimated $F0$ is only considered a correct estimate if the estimated $F0$ differs from ground-truth $F0$ by less than 50 cents.

$$RPA = \frac{N_{50}}{N_p}, \tag{10}$$

where N_p is the number of voiced frames, and N_{50} is the number of voiced frames whose estimated $F0$ is within 50 cents of the ground-truth $F0$.

Voicing decision error indicates the percentage of frames that are wrongly classified in terms of voicing.

$$VDE = \frac{N_{p\to n} + N_{n\to p}}{N}, \tag{11}$$

where N represents the total number of frames, $N_{p\to n}$ is the number of the voiced frames that are misclassified as non-voiced and $N_{n\to p}$ is the number of non-voiced frames that are misclassified as voiced.

TABLE II
Raw Pitch Accuracy (In %) on Synthesized Test Set

Method	Clean	Babble				Factory				Cafeteria			
		-10 dB	-5 dB	0 dB	5 dB	-10 dB	-5 dB	0 dB	5 dB	-10 dB	-5 dB	0 dB	5 dB
DC-CRN (magnitude)	92.27	39.79	65.14	79.80	86.65	57.39	77.64	85.62	89.11	36.04	60.54	78.03	84.95
DC-CRN (w/ non-pitch state)	95.14	39.03	62.75	79.13	87.40	51.25	72.91	84.59	90.30	29.00	54.88	75.53	84.16
DC-CRN	**97.86**	46.87	70.52	83.31	89.82	60.56	78.79	87.76	91.88	42.61	66.89	83.36	88.63
Single-task Combination	96.23	45.54	67.92	80.34	86.99	57.55	77.60	86.34	90.28	40.09	65.30	80.94	86.91
Cascade Architecture	96.78	**57.43**	**77.27**	**87.62**	**92.50**	**69.20**	**83.84**	**90.73**	**94.00**	**47.62**	**71.35**	**85.79**	**89.63**

TABLE III
Voicing Decision Error (In %) on Synthesized Test Set

Method	Clean	Babble				Factory				Cafeteria			
		-10 dB	-5 dB	0 dB	5 dB	-10 dB	-5 dB	0 dB	5 dB	-10 dB	-5 dB	0 dB	5 dB
DC-CRN (magnitude)	2.46	28.39	17.94	11.36	7.35	18.32	10.74	6.86	5.10	30.43	18.94	10.53	7.16
DC-CRN (w/ non-pitch state)	1.81	28.01	17.49	10.27	6.21	21.22	11.82	6.70	4.12	31.45	19.73	10.42	6.72
DC-CRN	**1.02**	26.01	15.59	9.15	5.19	16.64	9.29	5.36	3.45	25.07	15.11	7.53	**5.09**
Single-task Combination	1.29	28.30	18.98	11.47	6.84	18.97	10.82	6.78	4.50	26.02	16.02	8.70	6.07
Cascade Architecture	1.31	**22.48**	**12.76**	**7.40**	**4.57**	**13.86**	**7.94**	**4.81**	**3.23**	**23.78**	**13.70**	**7.21**	5.36

VI. Evaluation Results and Comparisons

We train the proposed models on synthesized data to have complete control of ground-truth $F0$ labels, and evaluation on both a synthesized test set and a test set with real recordings is conducted. In Section VI-A, we analyze the proposed models on the synthesized test set and explore phase information in pitch tracking. The differences between single-task learning and multi-task learning are also investigated. In Section VI-B, comparisons with baseline models and discussions are made on the test set of real recordings.

A. Model Analysis on Synthesized Test Set

Tables II and III show the evaluation results in terms of RPA and VDE respectively. With the increase of SNR, raw pitch accuracy gradually increases, and voicing decision errors gradually decrease. Next, we examine different input types on the multi-task DC-CRN model. We find that the model using complex STFT as input feature (DC-CRN) consistently outperforms the model trained with magnitude STFT input (DC-CRN magnitude) in terms of both RPA and VDE, especially in the more challenging babble and cafeteria noises and at lower SNRs. On average, for noisy speech, RPA is improved by 4.2% and VDE is decreased by 2.47% in absolute terms.

Our finding that better F0 estimation is obtained in the complex domain than in the magnitude domain joins a growing list of speech processing tasks with similar observations: speech enhancement [18], [42], speech dereverberation [43], speaker separation [44], and singing voice separation [45], [46]. It is well documented that pitch perception is closely related to the temporal fine structure of a signal, as opposed to the signal envelope [47]. Phase is a key characteristic of temporal fine structure, which may explain our observation that the real and imaginary spectrograms provide more discriminant features for pitch estimation than the magnitude spectrogram.

Even though multi-task learning is a simple and effective framework to train a pitch tracking model, it is uncertain if $F0$ estimation and voicing detection tasks can benefit each other and whether the multi-task framework is an optimal choice. To examine these questions, we train two DC-CRN single-task models, where one model is trained to perform $F0$ estimation only and the other is trained only for voicing detection. Two models are trained independently. We then combine the results from the two single-task models using (4) to generate pitch tracks. This approach is denoted as single-task combination. From the tables, we observe that the single-task combination model does not perform as well as the multi-task DC-CRN model. In addition, we find that the single-task DC-CRN model for voicing detection makes more false reject errors on voiced frames. For example, in the -10 dB cafeteria noise condition, with 1% VDE difference, false rejects in voiced intervals account for 14.13% out of 25.07% VDE errors for the multi-task model, and 18.5% out of 26.02% for the single-task model. Overall, the multi-task learning framework performs better and it also consumes less computation.

In addition, methods such as [8] incorporate a non-pitch state in the target vector and the probabilistic output is learned using a cross-entropy loss function. To examine this technique, we train a corresponding DC-CRN model under this setup (DC-CRN w/ non-pitch state in Tables II and III) and compare with multi-task learning. Post-processing is not considered in this comparison. From Tables II and III, one can observe that multi-task learning shows better results in terms of both RPA and VDE. We think that the multi-task framework has several advantages compared to the introduction of a non-pitch state. First, voicing detection is not influenced by wrong pitch estimation. Second, multi-task DC-CRN is trained with a binary cross entropy loss which enables using the Gaussian-blurred training target for pitch estimation and softens the penalty of near-correct predictions.

Evaluation results in Tables II and III also demonstrate that the proposed cascade architecture consistently outperforms the DC-CRN model in terms of both RPA and VDE in different noise conditions. Major improvements are observed in low SNR scenarios. For example, in the -10 dB case, the raw pitch accuracy is improved by 8.07% on average, and voicing decision error is reduced by 2.53% in absolute terms.

Fig. 6 illustrates pitch tracking results of an example utterance from the FDA test set mixed with babble noise at -10 dB SNR. Fig. 6(a),(b), and (c) show the spectrograms of noisy speech, clean speech, and the enhanced speech from the cascade model

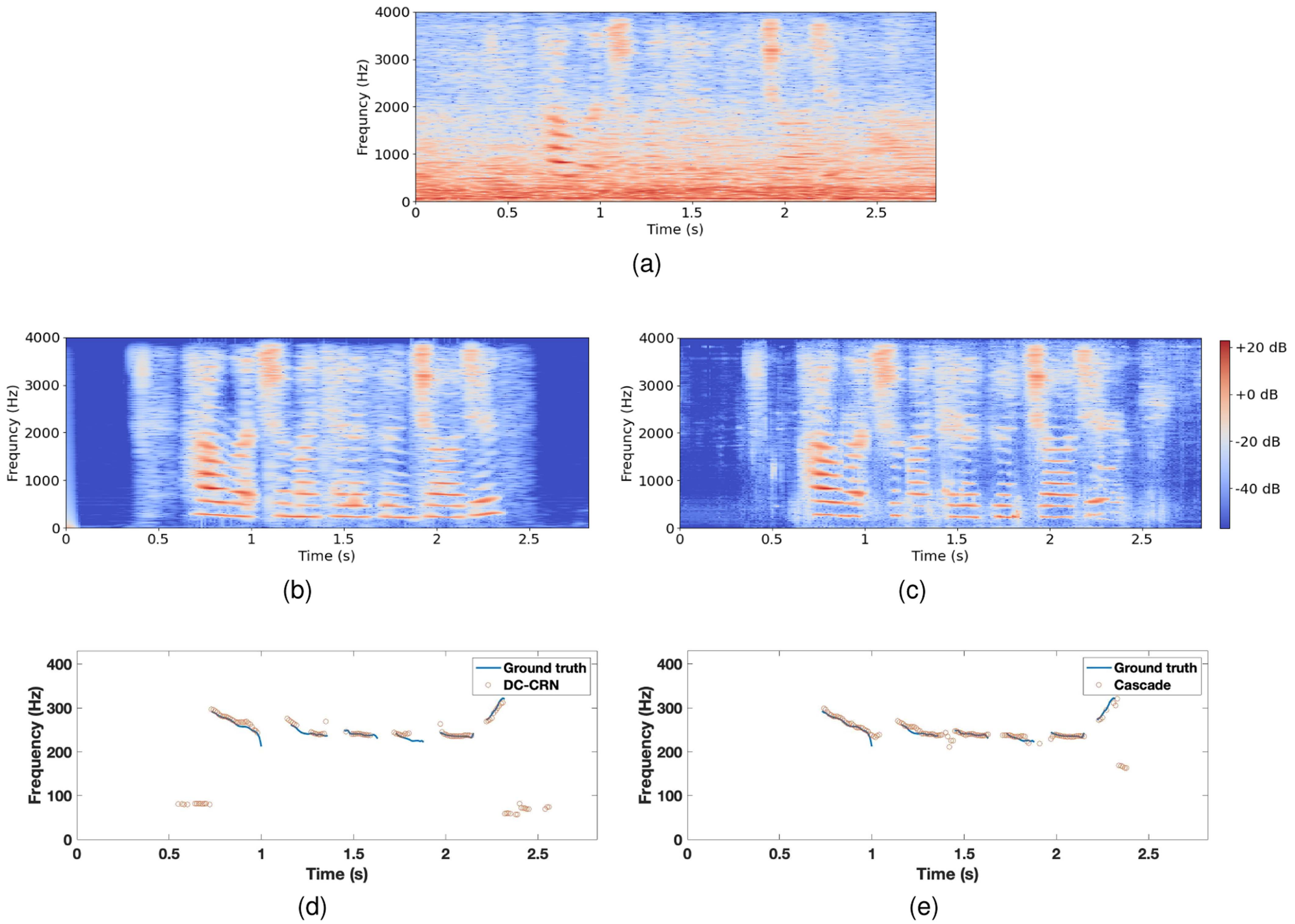


Fig. 6. Example of pitch tracking in noisy speech. The noisy speech is a female utterance from the FDA corpus mixed with babble noise at -10 dB SNR. (a) Spectrogram of noisy speech. (b) Spectrogram of clean speech. (c) Spectrogram of enhanced speech. (d) DC-CRN estimated pitch contours. The circles denote the estimated $F0$, and solid lines denote the ground-truth $F0$ contours. (e) Cascade architecture estimated pitch contours.

respectively. Fig. 6(d) and (e) show the estimated pitch contours from the DC-CRN model and the cascade architecture. We can see from Fig. 6(a) that, at -10 dB SNR, the harmonic pattern of clean speech is severely corrupted by the noise. By comparing Fig. 6(d) with (e) more robust voicing detection and accurate $F0$ estimation are observed with the cascade architecture.

B. Comparison With Baseline Methods on Real Recordings

We now compare several baseline methods and proposed methods on real recordings. Three competitive baseline methods are chosen for comparison and they are PEFAC [5], Han and Wang's RNN model [8], the FCN model [10] as discussed in Section I. PEFAC is a pitch tracking algorithm that identifies voiced frames in speech and estimates $F0$ in highly noisy conditions. In this method, voiced frames are identified using the likelihood ratio of two Gaussian mixture models trained on voiced and unvoiced frames respectively. In the RNN model [8], the network incorporates a no-pitch state in its output to identify unvoiced or speech-free frames. FCN (fully convolutional network) is an end-to-end model which takes a raw waveform as input and generates probabilistic outputs of pitch states. This model achieves state-of-the-art performance on clean speech. Specifically, we select FCN-929 for comparison as it is reported to have the highest pitch detection results on real recordings. For a fair comparison, we retrain the RNN model and FCN model on our training set. While the FCN model does not produce voicing decisions directly, one way to obtain voicing decisions is by thresholding the pitch class probability outputs. In our evaluation, we assess $F0$ estimation and voicing detection performance of FCN with this approach (noted as FCN with voicing detection (FCN w/ V.D.) in Fig. 7). Since FCN is not designed for voicing detection, for fair comparison, we also include an evaluation of $F0$ estimation in voiced frames.

We first compare the proposed models with pitch tracking methods that perform both $F0$ estimation and voicing detection. The models are evaluated in terms of raw pitch accuracy and voicing decision error. Fig. 7 shows the evaluation results on the FDA test sets with three noises and at 4 SNRs from -10 dB to 5 dB. It can be observed that both the DC-CRN method and cascade architecture outperform baseline methods substantially. The RNN model of Han and Wang achieves better RPA and VDE

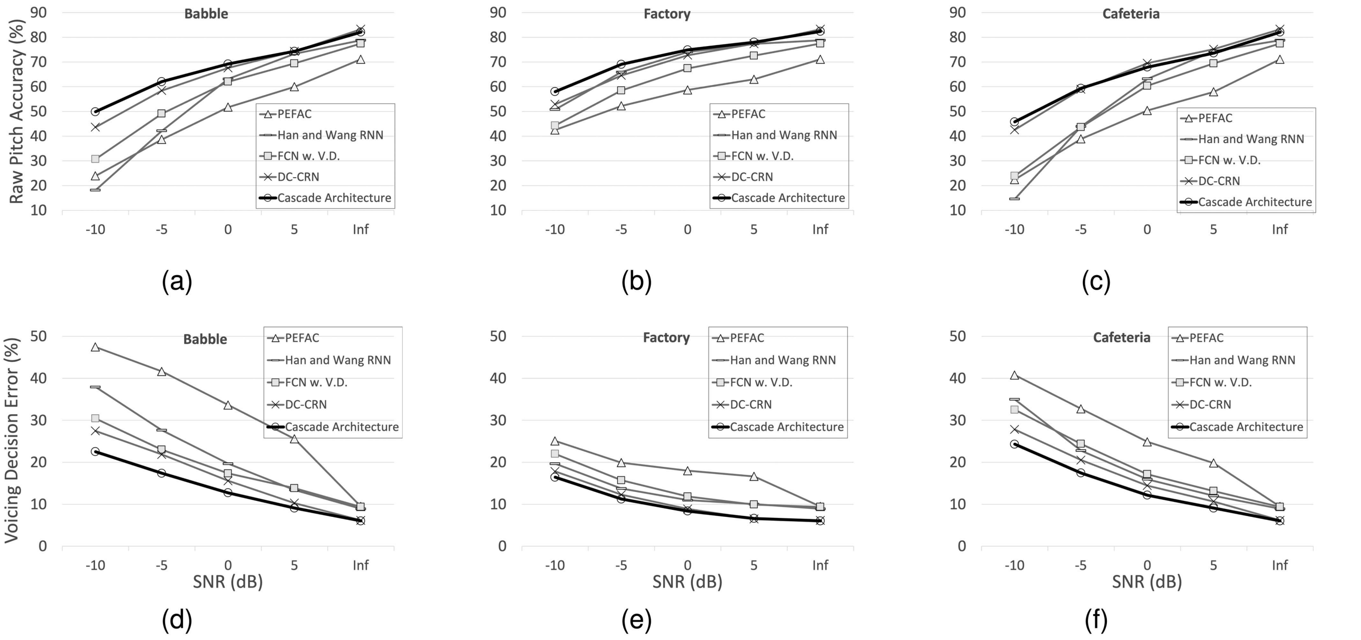


Fig. 7. Raw pitch accuracy and voicing decision error of $F0$ estimation on the FDA test set for three noises: (a) & (d) for babble noise, (b) & (e) for factory noise, and (c) & (f) for cafeteria noise.

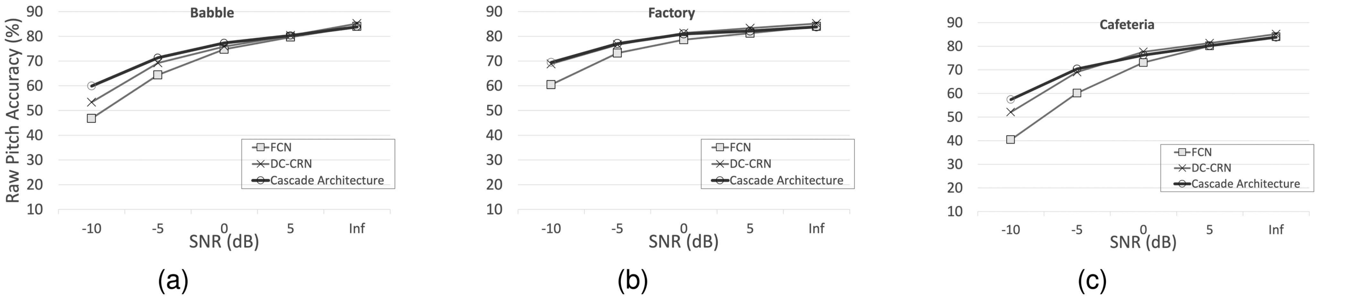


Fig. 8. Raw pitch accuracy of $F0$ estimation without the influence of voicing decision errors on the FDA test set for three noises: (a) babble noise, (b) factory noise, and (c) cafeteria noise.

than PEFAC in most cases and performs well in high SNR conditions. For FCN w/ V.D., we explore different thresholds ranging from 0 to 1, with increment of 0.1 and found 0.5 to be the best threshold. It can be seen that FCN w/ V.D. produces reasonable voicing detection results even though it is not trained for making voicing decisions and outperforms PEFAC in terms of both RPA and VDE. Compared with the DC-CRN model, the proposed cascade architecture has better performance, especially in low SNR scenarios. For example, in the -10 dB scenario, RPA is increased by 6.24%, 5.1%, and 3.21%, and VDE is decreased by 4.95%, 1.37% and 3.5%, for babble, factory, and cafeteria noises respectively. With the help of the speech enhancement module, consistent improvements of VDE are observed in the cascade architecture at all SNRs. In this comparison, RPA and VDE are calculated based on estimated pitch tracks. Note that, since $F0$ estimates incorporate voicing decisions, RPA results do not reflect the accuracy of $F0$ estimation entirely; for some voiced frames, the estimated $F0$ is set to 0 because of voicing decision error.

Recent DNN methods such as CREPE [9] and FCN [10] focus on $F0$ estimation in voiced frames, without detecting voiced/unvoiced intervals of target speech. To be consistent with this focus, when comparing with FCN, we calculate RPA according to $\hat{p}$ from $F0$ estimation in ground-truth voiced frames. Thus the calculated RPA represents the accuracy of $F0$ estimation without the influence of voicing decision error. Fig. 8 presents the RPA comparison results. We can see by comparing Figs. 8 to 7 that removing the influence from voicing detection increases the RPA scores by more than 10% on average for the proposed methods. We observe from Fig. 8 that FCN has strong performance in less noisy conditions but the $F0$ estimation results of the proposed methods are more robust in more noisy conditions. Note that in this study, we utilize ground-truth labels derived from laryngograph data to determine voiced/unvoiced intervals for the test set. This leads to inclusion of frames with lower confidence levels,and also lower RPA compared to [16]. In addition, the proposed DC-CRN model has 4.1 million trainable parameters, which are much fewer than FCN with 12.3 million parameters. In terms of the inference speed of FCN, DC-CRN, and the neural cascade architecture, we document the mean computation time over 500 runs on a 1-second audio on a CPU and a GPU. Specifically, we employ an Intel Xeon Gold 5115 2.4 GHz 10-core CPU, and a NVIDIA GeForce RTX 2080 Ti GPU. We observe that on the GPU, FCN exhibits the fastest speed, requiring only 4.9 ms. In contrast, DC-CRN and the neural cascade architecture take 17.8 ms and 18.3 ms, respectively. On the CPU, DC-CRN has the fastest speed at 101.9 ms, whereas the neural cascade architecture takes 109.1 ms, and FCN is the slowest at 170.7 ms. In the -10 dB scenario, the RPA of the DC-CRN model is 8.88% higher than the RPA of the FCN model on average. Again, the cascade architecture further outperforms DC-CRN in low SNR scenarios.

VII. Conclusion

In this study, we have proposed a neural cascade model that jointly performs $F0$ estimation and voicing detection. The cascade model takes complex STFT as input and optimizes a speech enhancement module and a pitch tracking module jointly. We have observed that introducing the speech enhancement module improves both $F0$ estimation and voicing detection accuracy, especially in very noisy conditions. Furthermore, we show that proposed multi-task learning is superior to a framework that combines the corresponding single-task models, and systematic comparisons show that the proposed method substantially outperforms other baseline methods. The pitch tracking models trained on synthesized speech show strong performance on real recordings. We thus believe that the proposed cascade architecture for multi-task training represents a significant advance in F0 estimation in noisy speech. For future work, we plan to explore pitch tracking in multi-talker and noisy speech mixtures, as well as the potential contribution of pitch estimation to speech enhancement.

Acknowledgment

The authors would like to thank Dr. Luc Ardaillon for providing details of the FCN method and insightful suggestions.

References

[1] A. De Cheveigné and H. Kawahara, “YIN, a fundamental frequency estimator for speech and music,” *J. Acoust. Soc. Amer.*, vol. 111, pp. 1917–1930, 2002.

[2] D. Talkin and W. B. Kleijn, “A robust algorithm for pitch tracking (RAPT),” *Speech Coding Synth.*, vol. 495, 1995, Art. no. 518.

[3] P. Boersma et al., “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound,” in *Proc. Conf. Inst. Phonetic Sci.*, 1993, vol. 17, pp. 97–110.

[4] W. Chu and A. Alwan, “SAFE: A statistical approach to F0 estimation under clean and noisy conditions,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 3, pp. 933–944, Mar. 2012.

[5] S. Gonzalez and M. Brookes, “PEFAC – A pitch estimation algorithm robust to high levels of noise,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 2, pp. 518–530, Feb. 2014.

[6] M. Wu, D. L. Wang, and G. J. Brown, “A multipitch tracking algorithm for noisy speech,” *IEEE Speech Audio Process.*, vol. 11, no. 3, pp. 229–241, May 2003.

[7] Z. Jin and D. L. Wang, “HMM-based multipitch tracking for noisy and reverberant speech,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 5, pp. 1091–1102, Jul. 2011.

[8] K. Han and D. L. Wang, “Neural network based pitch tracking in very noisy speech,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 2158–2168, Dec. 2014.

[9] J. W. Kim, J. Salamon, P. Li, and J. P. Bello, “CREPE: A convolutional representation for pitch estimation,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 161–165.

[10] L. Ardaillon and A. Roebel, “Fully-convolutional network for pitch estimation of speech signals,” in *Proc. Interspeech*, 2019, pp. 2005–2009.

[11] S. Singh, R. Wang, and Y. Qiu, “DeepF0: End-to-end fundamental frequency estimation for music and speech signals,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 61–65.

[12] M. Morrison, C. Hsieh, N. Pruyne, and B. Pardo, “Cross-domain neural pitch and periodicity estimation,” 2023, *arXiv:2301.12258*.

[13] G. Hu and D. L. Wang, “Segregation of unvoiced speech from nonspeech interference,” *J. Acoust. Soc. Amer.*, vol. 124, pp. 1306–1319, 2008.

[14] D. N. Tran, U. Batricevic, and K. Koishida, “Robust pitch regression with voiced/unvoiced classification in nonstationary noise environments,” in *Proc. Interspeech*, 2020, pp. 175–179.

[15] H. Schröter, T. Rosenkranz, A. N. Escalante-B, and A. Maier, “LACOPE: Latency-constrained pitch estimation for speech enhancement,” in *Proc. Interspeech*, 2021, pp. 656–660.

[16] Y. Zhang, H. Wang, and D. L. Wang, “Densely-connected convolutional recurrent network for fundamental frequency estimation in noisy speech,” in *Proc. Interspeech*, 2022, pp. 401–405.

[17] K. Tan, X. Zhang, and D. L. Wang, “Deep learning based real-time speech enhancement for dual-microphone mobile phones,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 1853–1863, 2021.

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

[19] K. Tan and D. L. Wang, “Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 380–390, 2019.

[20] H. Wang and D. L. Wang, “Neural cascade architecture with triple-domain loss for speech enhancement,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 734–743, 2022.

[21] X. Li, R. Liu, T. Song, X. Wu, and J. Chen, “Single-channel speech separation integrating pitch information based on a multi task learning framework,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 7279–7283. .

[22] A. Jansson, R. M. Bittner, S. Ewert, and T. Weyde, “Joint singing voice separation and F0 estimation with deep U-net architectures,” in *Proc. 27th Eur. Signal Process. Conf.*, 2019, pp. 1–5.

[23] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, “UNet++: A nested U-net architecture for medical image segmentation,” in *Proc. Int. Workshop Deep Learn. Med. Image Anal.*, 2018, pp. 3–11.

[24] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, “Language modeling with gated convolutional networks,” in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 933–941.

[25] F. Gao, L. Wu, L. Zhao, T. Qin, X. Cheng, and T.-Y. Liu, “Efficient sequence learning with group recurrent networks,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2018, pp. 799–808.

[26] Z.-Q. Wang, G. Wichern, and J. Le Roux, “On the compensation between magnitude and phase in speech separation,” *IEEE Signal Process. Lett.*, vol. 28, pp. 2018–2022, 2021.

[27] Y. Liu and D. Wang, “Permutation invariant training for speaker-independent multi-pitch tracking,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 5594–5598.

[28] B. Bechtold, “Pitch of voiced speech in the short-time Fourier transform: Algorithms, ground truths, and evaluation methods,” Ph.D. dissertation, Dept. Med. Phys. Acoust., Carl von Ossietzky Universität Oldenburg, Oldenburg, Germany, 2021.

[29] J. Salamon, R. M. Bittner, J. Bonada, J. J. Bosch, E. Gómez Gutiérrez, and J. P. Bello, “An analysis/synthesis framework for automatic F0 annotation of multitrack datasets,” in *Proc. Int. Soc. Music Inf. Retrieval Conf.*, 2017.

[30] L. Ardaillon, “Synthesis and expressive transformation of singing voice,” Ph.D. dissertation, School Comput. Sci., Telecommun. Electron., UPMC-Paris 6 Sorbonne Universites, Paris, France, 2017.

[31] G. Fant et al., "A four-parameter model of glottal flow," *STL-QPSR*, vol. 26, pp. 1–13, 1985.

[32] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. Syst.*, vol. 99, pp. 1877–1884, 2016.

[33] M. Morise, H. Kawahara, and H. Katayose, "Fast and reliable F0 estimation method based on the period extraction of vocal fold vibration of singing voice and speech," in *Proc. 35th Int. Conf. Audio Eng. Soc.*, 2009, pp. 71–81.

[34] M. Morise, "CheapTrick, A spectral envelope estimator for high-quality speech synthesis," *Speech Commun.*, vol. 67, pp. 1–7, 2015.

[35] M. Morise, "Platinum: A method to extract excitation signals for voice synthesis system," *Acoust. Sci. Technol.*, vol. 33, pp. 123–125, 2012.

[36] M. Morrison, "Torchcrepe," 2020, Accessed: Aug. 2022. [Online]. Available: https://github.com/maxrmorrison/torchcrepe

[37] 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.*, 2015, pp. 5206–5210.

[38] P. C. Bagshaw, S. M. Hiller, and M. A. Jack, "Enhanced pitch tracking and the processing of F0 contours for computer aided intonation teaching," in *Proc. Eurospeech*, 1993, pp. 1003–1006.

[39] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, pp. 247–251, 1993.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

[41] T. Nakatani, S. Amano, T. Irino, K. Ishizuka, and T. Kondo, "A method for fundamental frequency estimation and voicing decision: Application to infant utterances recorded in real acoustical environments," *Speech Commun.*, vol. 50, pp. 203–214, 2008.

[42] Y. Hu et al., "DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement," in *Proc. Interspeech*, 2020, pp. 2472–2476.

[43] J. Zhang, M. D. Plumbley, and W. Wang, "Weighted magnitude-phase loss for speech dereverberation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 5794–5798.

[44] Y. Liu and D. L. Wang, "Divide and conquer: A deep CASA approach to talker-independent monaural speaker separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 2092–2102, Dec. 2019.

[45] Y. Zhang, Y. Liu, and D. L. Wang, "Complex ratio masking for singing voice separation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 41–45.

[46] W. Choi, M. Kim, J. Chung, D. Lee, and S. Jung, "Investigating U-Nets with various intermediate blocks for spectrogram-based singing voice separation," in *Proc. 21st Int. Soc. Music Inf. Retrieval Conf.*, 2020.

[47] B. C. Moore, "The role of temporal fine structure processing in pitch perception, masking, and speech perception for normal-hearing and hearing-impaired people," *J. Assoc. Res. Otolaryngol.*, vol. 9, pp. 399–406, 2008.

Yixuan Zhang received the bachelor's degree in electrical engineering from Southeast University, Nanjing, China, in 2016, and the M.S. degree in music technology from Carnegie Mellon University, Pittsburgh, PA, USA, in 2018. She is currently working toward the Ph. D. degree with Ohio State University, Columbus, OH, USA. Her research interests include audio source separation, pitch tracking, and acoustic echo cancellation.

Heming Wang (Student Member, IEEE) received the bachelor's degree in physics and the M.S. degree in applied mathematics from University of Waterloo, Waterloo, ON, Canada, in 2016 and 2018, respectively. He is currently working toward the Ph. D. degree with the Ohio State University, Columbus, OH, USA. His research interests include speech enhancement, speech super-resolution, and deep learning.

DeLiang Wang, biography not available at the time of publication.