INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

RNN-BLSTM Based Multi-Pitch Estimation

Jianshu Zhang[1], *Jian Tang*[1], *Li-Rong Dai*[1]

[1]National Engineering Laboratory for Speech and Language Information Processing
University of Science and Technology of China, Hefei, Anhui, P. R. China
xysszjs@mail.ustc.edu.cn, enjtang@mail.ustc.edu.cn, lrdai@ustc.edu.cn

Abstract

Multi-pitch estimation is critical in many applications, including computational auditory scene analysis (CASA), speech enhancement/separation and mixed speech analysis; however, despite much effort, it remains a challenging problem. This paper uses the PEFAC algorithm to extract features and proposes the use of recurrent neural networks with bidirectional Long Short-Term Memory (RNN-BLSTM) to model the two pitch contours of a mixture of two speech signals. Compared with feedforward deep neural networks (DNN), which are trained on static frame-level acoustic features, RNN-BLSTM is trained on sequential frame-level features and has more power to learn pitch contour temporal dynamics. The results of evaluations using a speech dataset containing mixtures of two speech signals demonstrate that RNN-BLSTM can substantially outperform DNN in multi-pitch estimation of mixed speech signals.

Index Terms: multi-pitch estimation, neural networks, RNN-BLSTM, PEFAC

1. Introduction

Pitch, or fundamental frequency, F_0, is an important characteristic of speech/music signals. The task of estimating the single F_0 of clean speech from a single speech signal has attracted a surprising amount of attention for decades [1], while estimating multi-pitch values from a mixture of two or more speech signals is still a particularly challenging task. However, work on the challenging task of multi-pitch estimation is now gaining momentum, fuelled by progress in signal processing techniques and new applications such as computational auditory scene analysis (CASA), speech enhancement/separation and mixed speech analysis.

The main idea of this work is to estimate multi-pitch in a one channel mixture of two speech signals, which can be extended to signals with more than two speech signals, assuming that the number of mixture sources is already known. A number of studies have addressed the problem. S. W. Lee [2] proposed a method to estimate multi-pitch by employing spectral harmonicity of speech for a model-based speech separation. Z. Jin and D. Wang [3] applied a channel selection method to extract periodicity features and calculate pitch scores which are fed into a hidden Markov model (HMM) to extract continuous pitch contours. Sha and Saul [4] modelled the instantaneous frequency spectrogram with nonnegative matrix factorization (NMF) and used the inferred weight coefficients to determine pitch candidates for one or more voices. M. G. Christensen [5] presented a method to estimate pitch by updating the signal statistics with an exponential forgetting factor and subsequent numerical optimization and then smoothing the estimates and separating the pitch into slowly and rapidly varying components with a Kalman filter. Recently Han and Wang [6] firstly quantized the plausible pitch frequency range into some bins, representing pitch states, then they have shown that two alternative neural networks (DNN and RNN) that model the pitch states given observations both produced accurate probabilistic outputs of pitch states. These frame-level pitch states are then connected into pitch contours by Viterbi decoding [6] [7]. Liu [8] then used DNN to develop speaker-dependent models for multi-pitch estimation. Moreover, in [9], an unsupervised method for obtaining multi-pitch tracks was proposed that modelled multi-pitch tracks by a type of Gaussian mixture model with time-varying means.

Acoustic harmonicity and pitch continuity are considered the main characteristics for use in pitch estimation. In particular, in comparison with single-pitch estimation, multi-pitch estimation has trouble maintaining pitch continuity. Motivated by the work of Han and Wang in [6], where DNN and simple RNN are used to model the posterior probability of pitch states for single-pitch estimation, in this work, to jointly model the acoustic harmonicity and pitch continuity, we propose RNN with bidirectional Long Short-Term Memory (RNN-BLSTM) to model the posterior probabilities of a pair of pitch states from frame-level observations of two speech signals. Compared with DNN (which relies on static frame-level acoustic features) and simple RNN (which has a vanishing gradient problem for long-context modelling), RNN-BLSTM [10] trains on sequential frame-level features and is capable of learning temporal dynamics. In addition, it has the power to address the vanishing gradient and exploding gradient problems [11]. Therefore, it is expected that RNN-BLSTM may generate more accurate pitch states probabilities than DNN [6] or simple RNN, especially for multi-pitch estimation task. Therefore, in contrast to the preceding methods, we employ the PEFAC algorithm [12] to extract the harmonic features for multi-pitch estimation. Further, our approach utilizes an advanced classifier, RNN-BLSTM, to generate accurate probabilistic outputs of pitch states and boost multi-pitch estimation performance, and the proposed models are speaker-independent, which avoids the need to acquire abundant data for the target speakers. After producing the pitch state probability for each frame, the pitch states are connected into pitch contours by Viterbi decoding.

2. Review of RNN-BLSTM

A recurrent neural network (RNN) is a natural extension of the feedforward neural network (FNN). An FNN can map from input to output vectors only, whereas an RNN can in principle map from the entire history of previous inputs to each output. Because the RNN has hidden units with delayed connections to

http://dx.doi.org/10.21437/Interspeech.2016-117

each other, the output of class k at time t can be represented as:

$$y_k^t = \sum_{h=1}^{H} W_{hk} b_h^t \tag{1}$$

$$b_h^t = \sigma\left(\sum_{i=1}^{I} W_{ih} x_i^t + \sum_{h^*=1}^{H} W_{h^*h} b_{h^*}^{t-1}\right) \tag{2}$$

However, in practice, a standard architecture RNN is hard to train properly, and the accessible range of context is limited. The problem lies in the vanishing gradient and the exploding gradient as described in [11].

One effective method to address these problems is to use Long Short-Term Memory (LSTM) architecture [13], which uses memory blocks to control the flow of information. Each memory block contains one or more self-connected memory cells. An LSTM memory cell contains one self-connected cell and three controlling gates. The input and output gates manage information flow into and out of the memory cell. The multiplicative gates allow LSTM memory cells to store and access information over long periods of time. Meanwhile, to ensure that the original state of the subsequence is zero, a forget gate is added [14]. Furthermore, there are peephole weights connecting the gates to the cell, which are used to obtain more accurate Constant Error Carousel (CEC) information [10]. As illustrated in Figure 1, the equations of the LSTM memory blocks are as follows:

$$b_l^t = \sigma\left(\sum_{i=1}^{I} w_{il} x_i^t + \sum_{h=1}^{H} w_{hl} b_h^{t-1} + \sum_{c=1}^{C} w_{cl} s_c^{t-1}\right) \tag{3}$$

$$b_\phi^t = \sigma\left(\sum_{i=1}^{I} w_{i\phi} x_i^t + \sum_{h=1}^{H} w_{h\phi} b_h^{t-1} + \sum_{c=1}^{C} w_{c\phi} s_c^{t-1}\right) \tag{4}$$

$$a_c^t = \sum_{i=1}^{I} w_{ic} x_i^t + \sum_{h=1}^{H} w_{hc} b_h^{t-1} \tag{5}$$

$$s_c^t = b_\phi^t s_c^{t-1} + b_l^t \tanh\left(a_c^t\right) \tag{6}$$

$$b_w^t = \sigma\left(\sum_{i=1}^{I} w_{iw} x_i^t + \sum_{h=1}^{H} w_{hw} b_h^{t-1} + \sum_{c=1}^{C} w_{cw} s_c^t\right) \tag{7}$$

$$b_c^t = b_w^t \tanh\left(s_c^t\right) \tag{8}$$

where σ is the *sigmoid* function, and b_l^t, b_ϕ^t, b_w^t, a_c^t and s_c^t are respectively the input gate, forget gate, output gate, cell input activation, and cell state vectors, all of which are the same size as the hidden vector b_h^t.

Nevertheless, RNN with conventional LSTM is unidirectional and cannot model the future context. To address this issue, we investigate RNN with bidirectional LSTM [15] that does model the future context by processing the input vector sequence in both forward and backward directions. Generally, RNN-BLSTM is expected to achieve better performance for multi-pitch estimation than RNN with conventional LSTM or simple RNN.

3. Algorithm description

3.1. Feature extraction

The proposed multi-pitch estimation algorithms first extract spectral domain features from each frame and then employ RNN-BLSTM to compute the posterior probabilities of the multi-pitch states for each frequency bin. With probabilistic

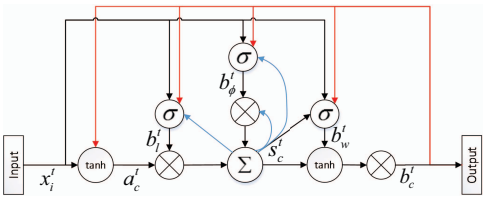

Figure 1: *The LSTM network architecture with one memory block.*

outputs in each time frame, we then use Viterbi decoding to connect the respective pitch states of both speech signals.

We employ the PEFAC [12] algorithm with minor modifications to extract the features of mixed speech signals. We first compute the log-frequency power spectrogram and then normalize it to the long-term speech spectrum to enhance robustness. A filter is then used to improve the harmonicity. For a periodic source containing the multi-pitches $f_0^{(1)}$ and $f_0^{(2)}$, the power spectral density in the log-frequency domain is given by

$$\begin{aligned} X(q) = &\sum_{k=1}^{K} b_k^{(1)} \delta(q - \log k - \log f_0^{(1)}) \\ &+ \sum_{k=1}^{K} b_k^{(2)} \delta(q - \log k - \log f_0^{(2)}) \end{aligned} \tag{9}$$

where $q = \log f$, $b_k^{(1)}$ and $b_k^{(2)}$ represent the power of the k-th harmonic voiced by two speakers, respectively, δ denotes the Dirac delta function and K is the number of harmonics. Note that, in the log-frequency domain, the spacing of the harmonics does not depend on the pitches ($f_0^{(1)}$, $f_0^{(2)}$). As a result, their energy can be summed by convolving $X(q)$ using a filter with broadened peaks that has an impulse response defined as follows:

$$h(q) = \begin{cases} \frac{1}{\gamma - \cos(2\pi e^q)} - \beta, & \log(0.5) < q < \log(K + 0.5) \\ 0 \quad , & otherwise \end{cases} \tag{10}$$

where β is chosen so that $\int h(q)dq = 0$, and γ controls the peak width. The convolution $\tilde{X}(q) = X(q) * h(-q)$ will result in two peaks at $q_0^{(1)} = \log f_0^{(1)}$, $q_0^{(2)} = \log f_0^{(2)}$ together with additional peaks that correspond to simple rational multiples and sub-multiples of $f_0^{(1)}$ or $f_0^{(2)}$. Ideally, the two pitches, $f_0^{(1)}$ and $f_0^{(2)}$ can be found by taking the two highest peaks in the output of the filter. However, in our work, we treat $\mathbf{x}_t = (\tilde{X}_t(q_1), \ldots, \tilde{X}_t(q_n))^T$ as the extracted feature and employ supervised learning to generate pitch probabilities, i.e., to learn the mapping from the features to the pitch frequencies. We expect supervised learning to yield better results.

3.2. RNN-BLSTM based multi-pitch estimation

We select RNN-BLSTM to address the multi-pitch estimation problem. The proposed RNN-BLSTM architecture is illustrated in Figure 2. As shown in Figure 2, the RNN-BLSTM model has two outputs for both male and female speaker groups in the current frame given the input features of mixed speech with an acoustic context. The goal of training this RNN-BLSTM is to generate the posterior probabilities that a pair of pitch states occur at frame m. To simplify the computation, we quantize the plausible pitch frequency range into M frequency bins, corresponding to M pitch states $s^1, \ldots, s^M$. We use 24 bins per

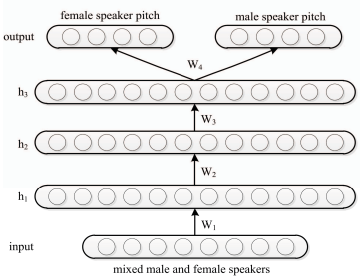

Figure 2: *Blockdiagram of our RNN-BLSTM architecture for multiple pitch estimation.*

octave in a logarithmic scale to quantize the plausible pitch frequency range (60 to 400Hz) into 67 bins, i.e., the quantized frequency of the m-th bin is $60 \times 2^{(m-1)/24} Hz$. In addition, we incorporate a non-pitch state s^0 that corresponds to an unvoiced speech or speech-free state. Therefore, there is a total of 68 states [16]. Because the RNN-BLSTM architecture in Figure 2 has two target speakers, the resulting output has 136 units, $[O_1^{(1)}, \ldots, O_{68}^{(1)}, O_1^{(2)}, \ldots, O_{68}^{(2)}]$.

To train the RNN-BLSTM classifier, each training sample is the feature vector $\mathbf{x}_t$ (DNNs need its neighbouring frames as well), and the corresponding target is a $2\,(M+1)$-dimensional vector of the two pitch states $[\mathbf{s}_t^{(1)}\ ,\ \mathbf{s}_t^{(2)}]$, whose element s_t^i is 1 when the groundtruth pitch falls into the corresponding frequency bin and 0 otherwise. The input for the RNN-BLSTM classifier is mixed speech signals of arbitrary speakers of different gender, while the outputs refer to the respective pitch state probabilities of the female and male speaker groups in a supervised manner. This architecture avoids the limitations of requiring abundant data from the target speakers to develop speaker-dependent models [8]. Moreover, our proposed RNN-BLSTM architecture improves the continuity of estimated pitch states along both time and frequency axes.

To learn the probabilistic outputs, we use cross-entropy as the objective function:

$$\mathcal{L} = -\alpha \sum_{m=0}^{M} y_m^{(1)} \ln f_m(\mathbf{x}) - \beta \sum_{m=0}^{M} y_m^{(2)} \ln f_m(\mathbf{x}) \qquad (11)$$

where α and β are ratio coefficient (usually $\alpha = \beta = 1$), $\mathbf{x}$ denotes the features extracted from mixed speech signals, $\mathbf{y} = (y_0^{(1)}, \ldots, y_M^{(1)}, y_0^{(2)}, \ldots, y_M^{(2)})^T$ is the desired output and $f_m(\cdot)$ is the actual output of the m-th neuron in the output layer. The activation function in the hidden layers is the *sigmoid* function and the output layer uses the *softmax* function for probabilistic outputs.

4. Experimental results and comparisons

4.1. Corpus and parameter selection

We evaluated the performance for the proposed approach using the SSC (Speech separation challenge) corpus [17]. This corpus consists of recordings of 500 sentences spoken by each of 34 speakers (18 males, 16 females). We chose ten male speakers (speaker No. 1, 2, 3, 5, 6, 8, 9, 10, 12, 13) and selected 2500 sentences from among these male speakers equally. Similarly, we picked 2500 sentences equally from among ten chosen female speakers (speaker No. 4, 7, 11, 15, 16, 18, 20, 21, 22, 23). These sentences, spoken by male and female speakers, were mixed one-to-one at signal-to-noise ratios (SNRs) (here, we consider female speech as noise) ranging from -2dB to 2dB in increments of 2dB. Altogether we prepared $2500 \times 3 = 7500$ sentences for the training set. We then chose five male and five female speakers (speaker No. 26, 27, 28, 30, 32; 25, 29, 31, 33, 34) and constructed 500 mixed sentences for the testing set at SNRs equal to 0dB. No utterances and speakers from the testing set existed in the training set. The groundtruth pitches were extracted from single-speaker utterances using STRAIGHT [18] before the signals were mixed.

Han [6] and Liu [8] have shown that DNN outperforms other traditional models both in single-pitch and multi-pitch estimation. Therefore, we compared our proposed methods with DNN based multi-pitch estimation. A short-time Fourier transform [19] was adopted to compute the discrete Fourier transform (DFT) for each overlapping windowed frame. We then extracted 192-dimensional PEFAC features per frame. For analysis purposes, to ensure that both DNN and RNN-BLSTM yield their best performances, we adjusted the parameters accordingly. In all experiments the DNN consisted of 960 input nodes (a stack of 5 neighbouring frames), 2 hidden layers, which used a *sigmoid* activation function with 512 nodes per layer and dual 68 output nodes (the probabilities of the pitch states of male and female targets), while the RNN-BLSTM consisted of 192 input nodes (no neighbouring frames), 2 hidden layers, which also used a *sigmoid* activation function but with 256 nodes per layer. The output layers are the same as those for the DNN.

We evaluated multi-pitch estimation results in terms of three measurements: precision rate (PR) [6], pitch decision error (PDE) [6] and estimation accuracy rate (EAR). The three measurements were all evaluated on single speakers. As an example of the precision rate for female speakers, PR(F), a single pitch estimate for female speakers is considered correct when the deviation of the estimated F_0 is within 5% of the female groundtruth F_0 (0 Hz denotes unvoiced frames), while the PDE denotes the percentage of frames that have been estimated but attributed to the wrong speaker. To calculate PDE(F) and PDE(M), we added another support measurement: the estimation accuracy rate (EAR). The EAR indicates the percentage of frames that have been estimated correctly–regardless of whether they have been misclassified to the wrong speaker. Consider that the target speaker is female, then

$$PR(F) = \frac{N_{f\to f}^{0.05}}{N} \qquad (12)$$

$$EAR(F) = \frac{N_{f\to f}^{0.05} + N_{f\to m}^{0.05}}{N} \qquad (13)$$

$$PDE(F) = EAR(F) - PR(F) \qquad (14)$$

Here, $N_{f\to f}^{0.05}$ denotes the number of frames in which the female speakers' estimated pitch frequency deviates less than 5% from the female groundtruth frequency, while $N_{f\to m}^{0.05}$ denotes the number of frames in which the male speakers' estimated pitch frequency deviates less than 5% from the female groundtruth frequency. N denotes the total number of frames in a sentence. PR(M), EAR(M), PDE(M) are calculated in a similar manner.

4.2. Results and comparisons

Table 1 shows the results of multi-pitch estimation based on RNN-BLSTM applied to the testing set. The testing set includes a total of 5 female and 5 male speakers. Due to the diversity of body structure and pronunciation, there is a huge gap–upwards 40%–between the highest and lowest PR values.

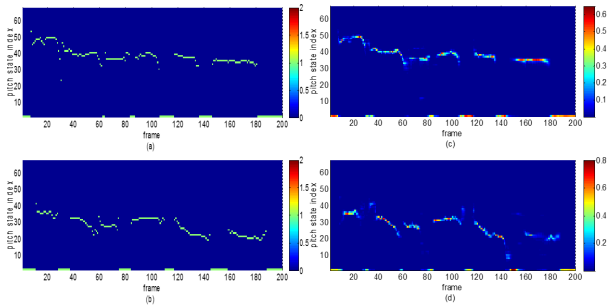

Figure 3: *(a) Groundtruth pitch states of a female speech signal. (b) Groundtruth pitch states of a male speech signal. (c) Probabilities of the female pitch states estimated by RNN-BLSTM. (d) Probabilities of the male pitch states estimated by RNN-BLSTM.*

However, the mean precision rates over 500 mixed input signals of female and male voices reach 81.51% and 83.42%, respectively, which is a satisfactory result. Figure 3 illustrates multi-pitch estimation results using RNN-BLSTM. The example is a mixture of utterances from the testing set whose PR(F) equals 83.89% and whose PR(M) equals 83.25%. Figure 3(a) and (b) show the groundtruth multi-pitch states extracted from the two single-speaker speeches using STRAIGHT [18]. In each frame, the probability of a pitch state is 1 if it corresponds to the female or male groundtruth pitch and 0 otherwise. Figure 3(c) and (d) show the probabilistic outputs of female and male utterances, respectively. Compared to Figure 3(a) and (b), the probabilities of the correct pitch states dominate at most time frames in both (c) and (d), demonstrating that the RNN-BLSTM successfully separated and predicted pitch states from mixed speech.

Table 1: *Precision rates (PR), in %, of multi-pitch estimation based RNN-BLSTM.*

observation	PR(F)	PR(M)
mean	81.51	83.42
best	94.78	97.54
worst	47.54	53.42

Table 2: *Comparison between RNN-BLSTM and DNN (EAF, PR, PDE, in %).*

model	gender	EAR	PR	PDE
DNN	female	78.18	68.10	10.09
DNN	male	86.45	73.55	12.90
RNN-BLSTM	female	87.32	81.51	5.81
RNN-BLSTM	male	91.26	83.42	7.85

The comparisons between methods based on DNN and RNN-BLSTM are shown in Table 2. RNN-BLSTM, which is capable of learning temporal dynamics, is expected to considerably outperform DNN in multi-pitch estimation, considering that pitch itself has a strong temporal continuity. First, estimation accuracy rate (EAR) represents the probability of identifying a single pitch candidate from a speech mixture, regardless of the unfortunate situations in which the female pitch candidates jump to a male pitch contour, or the male pitch candidates

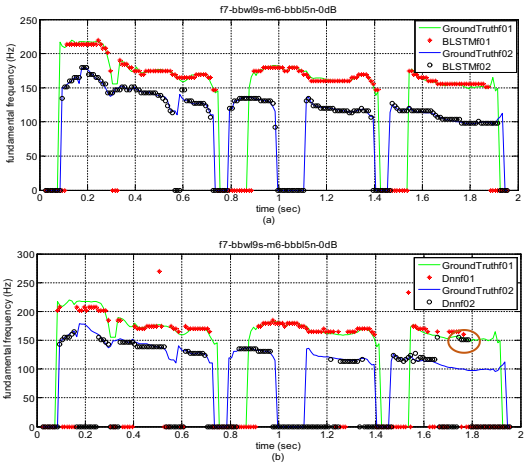

Figure 4: *multi-pitch estimation results of the f7-bbwl9s-m6-bbbl5n-0dB.wav file. (a) Probabilistic outputs from the RNN-BLSTM. (b) Probabilistic outputs from the DNN.*

jump to a female pitch contour. As shown in Table 2, the results of EAR(F) and EAR(M) demonstrate that RNN-BLSTM is better at estimating pitch candidates than DNN. Second, both the PDE(F) and PDE(M) of RNN-BLSTM are smaller than those of DNN. This result highlights the temporal modelling capacity of RNN-BLSTM from another point of view, because pitch decision error (PDE) denotes the percentage of frames whose pitch candidate has been estimated successfully but attributes it to the wrong speaker. This is a typical case where temporal context is lacking. As a result, the precision rates (PR) of RNN-BLSTM are much higher than DNN for both female and male utterances. Figure 4 shows the comparison between RNN-BLSTM and DNN on the same test speech mixture. Observing the DNN result at approximately 1.78 s into this speech signal, the multiple black units denote the pitch candidates that should be estimated as female pitch states unfortunately jump to the male pitch contour, leading to a decrease in PR(F). Such results suggest that the reason RNN-BLSTM yields better probabilistic outputs than DNN is because it is better able to capture the temporal context; therefore, its outputs are smoother than those of DNN.

5. Conclusions

In this work, we use RNN-BLSTM to generate the posterior probabilities of pitch states for multi-pitch estimation. In comparison with the DNN based method, RNN-BLSTM takes advantage of the temporal continuity of pitch. The experimental results showed that RNN-BLSTM can not only outperform DNN when estimating pitch candidates but also effectively prevent pitch candidates from being misclassified to the wrong speaker.

6. Acknowledgements

We acknowledge the support of the following organizations or programs for research fundingNational Nature Science Foundation of China (Grant No. 61273264), Science and Technology Department of Anhui Province (Grant No. 15CZZ02007), Chinese Academy of Sciences (Grant No. XDB02070006), National Key Technology Support Program (2014BAK15B05)

7. References

[1] W. Hess, *Pitch Determination of Speech Signals*. Berlin: Springer-Verlag, 1983.

[2] S. W. Lee, F. K. Soong, P. C. Ching, and T. Lee, "Pitch tracking for model-based speech separation," in *International Symposium on Chinese Spoken Language Processing*, 2008, pp. 145–148.

[3] Z. Jin and D. Wang, "Hmm-based multipitch tracking for noisy and reverberant speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1091–1102, 2011.

[4] F. Sha and L. K. Saul, "Real-time pitch determination of one or more voices by nonnegative matrix factorization," in *Advances in Neural Information Processing Systems 17*, 2005, pp. 1233–1240.

[5] M. G. Christensen, "A method for low-delay pitch tracking and smoothing," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2012, pp. 345–348.

[6] K. Han and D. Wang, "Neural networks for supervised pitch tracking in noise," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2014, pp. 1488–1492.

[7] G. D. Forney, "The viterbi algorithm," *Proceedings of The IEEE*, vol. 61, no. 3, pp. 268–278, 1973.

[8] Y. Liu and D. Wang, "Speaker-dependent multipitch tracking using deep neural networks," in *INTERSPEECH 2015*, 2015, pp. 3279–3283.

[9] A. M. N, P. K. Ghosh, and K. Rajgopal, "Multi-pitch tracking using gaussian mixture model with time varying parameters and grating compression transform," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2014, pp. 1473–1477.

[10] A. Graves, *Supervised sequence labelling with recurrent neural networks*. Springer, 2012.

[11] Y. B. P. Simard and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[12] S. Gonzalez and M. Brookes, "Pefac-a pitch estimation algorithm robust to high levels of noise," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 22, no. 2, pp. 518–530, 2014.

[13] J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Computation*, vol. 12, pp. 2451–2471, 2000.

[15] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5, pp. 602–610, 2005.

[16] B. S. Lee and D. P. W. Ellis, "Noise robust pitch tracking by subband autocorrelation classification," in *INTERSPEECH 2012*, 2012.

[17] M. Cooke and T. W. Lee, "Speech separation challenge," in *http://staffwww.dcs.shef.ac.uk/people/M.Cooke/SpeechSeparationChallenge.htm*, 2006.

[18] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructing speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based f0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3, pp. 187–207, 1999.

[19] L. Durak and O. Arkan, "Short-time fourier transform: Two fundamental properties and an optimal implementation," *IEEE Transactions on Signal Processing*, vol. 51, no. 5, pp. 1231–1242, 2003.