Exploiting time-frequency patterns with LSTM-RNNs for low-bitrate audio restoration

Jun Deng[1] · Björn Schuller[1] · Florian Eyben[1] · Dagmar Schuller[1] · Zixing Zhang[1] · Holly Francois[2] · Eunmi Oh[3]

Abstract

Perceptual audio coding is heavily and successfully applied for audio compression. However, perceptual audio coders may inject audible coding artifacts when encoding audio at low bitrates. Low-bitrate audio restoration is a challenging problem, which tries to recover a high-quality audio sample close to the uncompressed original from a low-quality encoded version. In this paper, we propose a novel data-driven method for audio restoration, where temporal and spectral dynamics are explicitly captured by a deep time-frequency-LSTM recurrent neural networks. Leveraging the captured temporal and spectral information can facilitate the task of learning a nonlinear mapping from the magnitude spectrogram of low-quality audio to that of high-quality audio. The proposed method substantially attenuates audible artifacts caused by codecs and is conceptually straightforward. Extensive experiments were carried out and the experimental results show that for low-bitrate audio at 96 kbps (mono), 64 kbps (mono), and 96 kbps (stereo), the proposed method can efficiently generate improved-quality audio that is competitive or even superior in perceptual quality to the audio produced by other state-of-the-art deep neural network methods and the LAME-MP3 codec.

Jun Deng
jdeng@audeering.com

Björn Schuller
bs@audeering.com

Florian Eyben
fe@audeering.com

Dagmar Schuller
ds@audeering.com

Zixing Zhang
zzhang@audeering.com

Holly Francois
h.francois@samsung.com

Eunmi Oh
sait@samsung.com

[1] audEERING GmbH, Gilching, Germany

[2] Samsung Research UK, Staines, UK

[3] Samsung Research, Seoul, Republic of Korea

1 Introduction

Since MPEG-1 Layer-3 (MP3) was standardized in 1991, perceptual audio coding has quickly emerged as the most dramatic and important development in digital audio coding over the last decades, thanks in part to the proliferation and growing demands of digital music services, such as online music streaming services (i. e., Spotify, Pandora, and Apple Music), mobile devices (i. e., smart-phones and tablets), and online audio storage [31, 43]. Today perhaps all popular audio codecs, such as MP3, MPEG-2/4 advanced audio coding (AAC) and Dolby Adaptive Transform Coder 3 (AC-3), are rooted in perceptual audio coding. A powerful feature of perceptual audio coding is its capability to effectively reduce storage space or bandwidth required for transmission for audio data while retaining near-transparent CD-quality [5, 43, 56]. This is achieved by exploiting perceptual irrelevancies and audio data statistic redundancies [43].

However, perceptual audio coders may inject audible coding artifacts when encoding audio at low bitrates. This is commonly due to the fact that the amount of bits required to keep the amplitude of quantization noise below the masked threshold is insufficient, leading to audible noise, so-called spectral holes, or due to signal bandwidth reduction [5, 56]. Audio restoration and audio enhancement methods can be used to alleviate this problem and thus improve the perceptual quality. The most well-known audio restoration method is Spectral Band Replication (SBR) [11, 12] that is devised to improve the perceptual quality of highly compressed audio signals by restoring the high-frequency information which was lost [24, 61]. Such high-frequency spectrum extension technology has played a key role in improving coding efficiency while keeping the bitrate requirements low. SBR has advanced MPEG-4 AAC to form MPEG-4 High Efficiency Advanced Audio Coding (HE-AAC) [31, 60]. As a result, HE-AAC can obtain near-transparent CD-quality at a very low bitrate. The existing SBR approaches exploit the fact that there is harmonic redundancy in the frequency domain. That is, the higher frequencies can be replicated by the lower frequencies with proper guidance information provided.

Despite the widespread use and remarkable success of SBR methods in audio coding, SBR methods are faced with some unresolved issues, which likely degrade the quality of audio signals [31, 38, 39, 62]. It is observed that SBR may introduce audible artifacts like tonal spikes to the high frequency of the signals [31]. Another underlying issue is the mismatch between the harmonic structures caused by the process of the band replication to create the missing high-frequency content [61]. Moreover, the side information used for the high-frequency spectral content reconstruction needs to be transmitted, yielding the increase in the data storage space or transmission bandwidth.

Due to the rise of machine learning, there has been a new trend toward developing data-driven techniques for speech enhancement[34], active noise reduction [37], hydrological process modeling [1], noise exposure level prediction [2], as well as audio restoration [22, 28, 32, 47, 48, 61]. The idea is to enhance the quality of audio signals by learning to predict the missing values from a large amount of data. One apparent advantage over SBR is that it eliminates the need of any side information required for the process of the band replication. For example, one approach is to employ a Gaussian mixture model (GMM) to estimate the energy of high-frequency spectral envelop from wideband to super-wideband in the context of mobile wideband audio communication systems [32]. Similarly, GMM is also used to extend the bandwidth of telephone speech to the frequency range 0–300 Hz [47, 48]. In addition to the use of GMM, unsupervised learning *k*-means and supervised learning Support Vector Regression (SVR) are combined to build high-frequency envelop predictors for signals with similar characteristics [61]. More recently, a spectral mapping approach based on deep feed-forward networks [28] is proposed to directly estimate the missing high-frequency spectrum from narrow-band speech. Instead of spectral mapping, a novel deep residual convolutional network is present to enhance the quality by directly mapping audio inputs at low sampling rates into higher-quality signals with an increased resolution in the time domain [22]. Despite the aforementioned learning-based methods generally work well for speech signals, they may not be directly applicable to music [61]. This is simply because music has a more complicated excitation signal and spectral shape when compared to speech [22, 61].

In this paper, we propose a novel deep learning-based audio restoration method for music signals encoded at a low bitrate, which is called Time-Frequency Long Short-Term Memory Recurrent Neural Networks (TF-LSTM-RNNs). The proposed method fundamentally extends the basic autoencoder structure [7, 8, 10, 18], which directly learns a nonlinear spectral mapping from compressed audio spectrum to uncompressed audio spectrum, allowing to reconstruct high-quality audio from a low-quality version.

Our approach is motivated by the evident observation that in audio signals, a certain harmonic correlation or similarity of the spectrum can be found both in frequency and in time. Hence, for frequency content (i. e., spectral holes) removed by the lossy compression process, it is plausible to estimate the missing frequency content from the remaining (correlated) spectral information and adjacent frames. To leverage this observation, our proposed TF-LSTM-RNNs exploit harmonic correlations of time-frequency representations in both time and frequency directions, adaptively capturing the two-dimensional time-frequency input information to facilitate restoring the original audio from the compressed audio. The objective and subjective evaluations suggest that the proposed method can improve the subjective quality of music signals at low bitrates (e. g., 64 kbps) and achieve better performance than other representative deep neural network-based audio restoration methods.

Our contributions are summarized as follows:

1. Encouraged by the constant success of the data-driven methods for automatic speech recognition and speech enhancement, in this paper, we focus on making use of deep learning methods for music audio restoration at low bit rate. Accordingly, we systematically conduct investigations into a variant of state-of-the-art deep neural networks, such as VGG networks, U-CNN networks and WaveNets, for this audio restoration task.
2. In addition, we propose novel deep TF-LSTM-RNNs to remove the audible artifacts introduced by the lossy audio compression process in the hope of enhancing

the perceptual quality. To the best of our knowledge, this is the first work on deep learning, which has been successfully tested for music audio restoration.

3. The extensive experimental results demonstrate that the existing deep neural networks, such as WaveNets, which have been successfully applied for speech enhancement, have the limited capability of addressing the audio restoration task, especially for the low-bitrate codec music signal. However, our proposed TF-LSTM-RNNs method is capable of the challenging music audio restoration problem even when the bitrate of the codec music signal is below 64 kbps.

The remainder of this paper is organized as follows: Sect. 2 first discusses related work. We then describe the proposed deep TF-LSTM-RNN for low-bitrate audio restoration in Sect. 3. In Sect. 4, a number of deep neural network architectures for comparison, the music data sets, objective and subjective metrics are discussed. Finally, we draw a conclusion and point out directions for future work in Sect. 5.

2 Related work

2.1 Spectro-temporal Modeling using RNNs for audio processing

In the context of machine learning, there are a number of approaches available to extract patterns from a two-dimensional matrices such as spectro-temporal representations of audio. The typical approach is Convolutional Neural Networks (CNNs). Thanks to their capability of extracting representative features, CNNs have been widely and successfully applied in image classification [58]. Owing to these successes, CNNs have been increasingly employed for acoustic analysis as well, such as speech recognition [3], speaker analysis [36], music information retrieval [25], and denoising [44]. The most traditional way to implement CNNs is taking a sliding square window to segment a time-continuous spectrogram into sequential image series. Each segmented spectrogram is then considered as an independent image where a conventional image processing algorithm with CNNs could be used.

Most of these applications, however, lack a specific consideration with respect to audio. For image processing, translation invariance is one of the major concerns. To address this issue, local filter and pooling strategies are normally used with CNNs. In contrast, for a audio spectrogram, the translational invariance problem is largely reduced, since bins in the spectrograms have fixed positions associated with fixed functions/frequencies. Additionally, an audio spectrogram has intrinsic correlations which differ from, e.g., photographic images. Thus, the frequency information indeed has common patterns, which, however, have not yet been explored intensively in the past for CNNs.

To address this problem, Li et al. [26] firstly proposed to use Long Short-Term Memory (LSTM) RNNs to learn the frequency content information for improved speech recognition. The authors segmented the frequency bins at time step t into a series of subsets by applying an overlapped sliding window. The segmented frequency bins are then successively fed into the LSTM which is called F-LSTM, and all the outputs are concatenated as a new unified vector at time t. This unified vector is then fed into a cascaded LSTM which is called T-LSTM and operates as a traditional sequence processing LSTM. Therefore, the F-LSTM and T-LSTM recurrency and cell states model frequency context as well as the time context, respectively.

Rather than separately capturing the frequency and time context in a cascaded structure, several studies try to seek help from a more general RNN architecture specifically designed for multi-dimensional sequence processing, i. e., multi-dimensional RNNs (MDRNNs) [13]. The underlying idea of MDRNNs is to replace the single recurrent connection found in standard RNNs with as many recurrent connections as there are dimensions in the data [13] and have efficiently applied to, for example, image segmentation [57]. In speech processing, Li et al. [27] further modified the cascaded structure by concepts from MDRNNs, resulting in Time-Frequency domain LSTM, where a current output depends on the cell states of the previous time step, and the outputs from both previous time and frequency steps and the input of the current time and frequency step. In doing this, the Time-Frequency domain LSTM at each frequency step k and time step t has knowledge about the frequency information ranging from bin 0 to bin $k - 1$, and the time information ranging frame 0 from frame t, if it operates in a forward way. Due to its effectiveness, Time-Frequency domain LSTM was successfully applied to voice conversion [30] and pitch tracking [33].

Different from the aforementioned time-frequency networks, where the time-network and frequency-network are highly intervened when going through the spectrogram, our proposed audio enhancement network applies a separated scanning strategy. That is, the time-network and frequency-network are separately fed with time series data and frequency series data. This greatly facilitates the learning process.

Moreover, the proposed network structure is specifically inspired by bidirectional RNN (BRNN) [53]. Both structures include two sub-layers per layer. However, for BRNN the sub-layers, normally namely forward layer and backward layer, are used to scan the sequences in an opposite direction; for the proposed network the sub-layers, referred to as time layer and frequency layer, are designed to extract patterns from different domains, namely spectral and temporal domains. Hence, the sub-layers of the proposed networks can be further extended into bidirectional sub-layers. Therefore, different from BRNN which is inherently one dimensional, the proposed network is able to learn patterns from two-dimensional representations.

2.2 Audio restoration

Audio restoration has been often studied in the audio-processing field under the name *bandwidth extension* [11, 24, 60]. One commercial technique is spectral band replication [11], which has been integrated into HE-AAC [31, 60]. Besides, there is a variation of data-driven methods for bandwidth extension, including GMM [32, 47, 48], deep feed-word neural networks [28], *k*-means and SVR [61], and deep CNNs [22]. These bandwidth extension methods usually attempt to recover the high-frequency content (e. g., 4–8 kHz) from narrow-band audio (4 kHz in bandwidth) with or without the need of side information. Rather than only generating the high-frequency content, the proposed method in this paper aims to recover all the missing information in the frequency range from 0 up to 15 kHz, lost in the perceptual coding process. Moreover, our method is conceptually straightforward and is also the first work on deep learning, which has been successfully applied for music audio restoration.

3 Proposed method

3.1 Basic LSTM-RNNs

The LSTM-RNN model uses one or multiple LSTM blocks [19]. Every memory block consists of self-connected linear memory cells $\mathbf{c}$ and three multiplicative gate units: an input gate $\mathbf{i}$, a forget gate $\mathbf{f}$, and an output gate $\mathbf{o}$. Given an input $\mathbf{x}_t$ at the time step t, the activations of the input gate $\mathbf{i}_t$, the forget gate $\mathbf{f}_t$, the output gate $\mathbf{o}_t$, the candidate state value $\mathbf{g}_t$, the memory cell state $\mathbf{c}_t$ are separately computed by the following equations:

$$\mathbf{i}_t = \mathrm{sigm}(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{b}_i), \tag{1}$$

$$\mathbf{f}_t = \mathrm{sigm}(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{b}_f), \tag{2}$$

$$\mathbf{o}_t = \mathrm{sigm}(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{b}_o), \tag{3}$$

$$\mathbf{g}_t = \tanh(\mathbf{W}_{gx}\mathbf{x}_t + \mathbf{W}_{gh}\mathbf{h}_{t-1} + \mathbf{b}_g), \tag{4}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \tag{5}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{6}$$

where $\mathbf{W}$ is a weight matrix of the mutual connections; $\mathbf{h}_t$ represents the output of the hidden block; $\mathbf{b}$ indicates the block bias, $\odot$ indicates the element-wise multiplication operation.

The LSTM memory cell can store and access information over a long temporal range and thus efficiently avoid the vanishing gradient problem [19]. To further increase the LSTM-RNN's capability to access future context, LSTM-RNN can be extended to a bidirectional version [53]. That is, the network calculates its forward hidden layer activations $\mathbf{h}_t^f$ from the beginning to the end of a sequence, and its backward hidden layer activations $\mathbf{h}_t^b$ from the end to the beginning of a sequence, then updates the output layer by

$$\mathbf{y}_t = \mathbf{W}_{fy}\mathbf{h}_t^f + \mathbf{W}_{by}\mathbf{h}_t^b + \mathbf{b}_y, \tag{7}$$

where $\mathbf{W}_{fy}$, $\mathbf{W}_{by}$ stand for the forward and backward weight matrices, and $\mathbf{b}_y$ denotes the hidden bias vector. The forward and backward directed layers are connected to the same output layer, which therefore can access the whole context information.

3.2 Time-frequency-LSTM-RNNs for audio restoration

Time-Frequency-LSTM-RNNs (TF-LSTM-RNNs) are devised to perform recurrence in both time and frequency direction. Figure 1 illustrates an example of TF-LSTM-RNNs with two hidden RNN layers and one linear hidden layer. The two-dimensional time-frequency modeling is made possible by processing the time-frequency representations in both directions with two separate LSTM-RNN layers: Time-LSTM (T-LSTM) layer processes the data sequence along time for modeling temporal dynamics; the other one, which is called a Frequency-LSTM (F-LSTM) layer, processes the data sequence in frequency for capturing spectral dynamics. The output activations from both T-LSTM and F-LSTM layers are then fed to the same output layer, where they are merged. Moreover, outputs from T-LSTM are not fed to inputs of F-LSTM, and vice versa. It is to note, that without the recurrent connections in frequency, this architecture corresponds to a regular LSTM-RNN, which has been often used in various audio processing applications [9, 51]. If the recurrent connections in the time dimension are excluded, this leads to a regular LSTM-RNN in the frequency dimension. When both time and frequency dimensions are simultaneously taken into account in the same network, correlations in the temporal and spectral directions can directly and efficiently be used to minimize the objective function.

Based on the proposed network, we present a new data-driven method for audio restoration, where we learn a

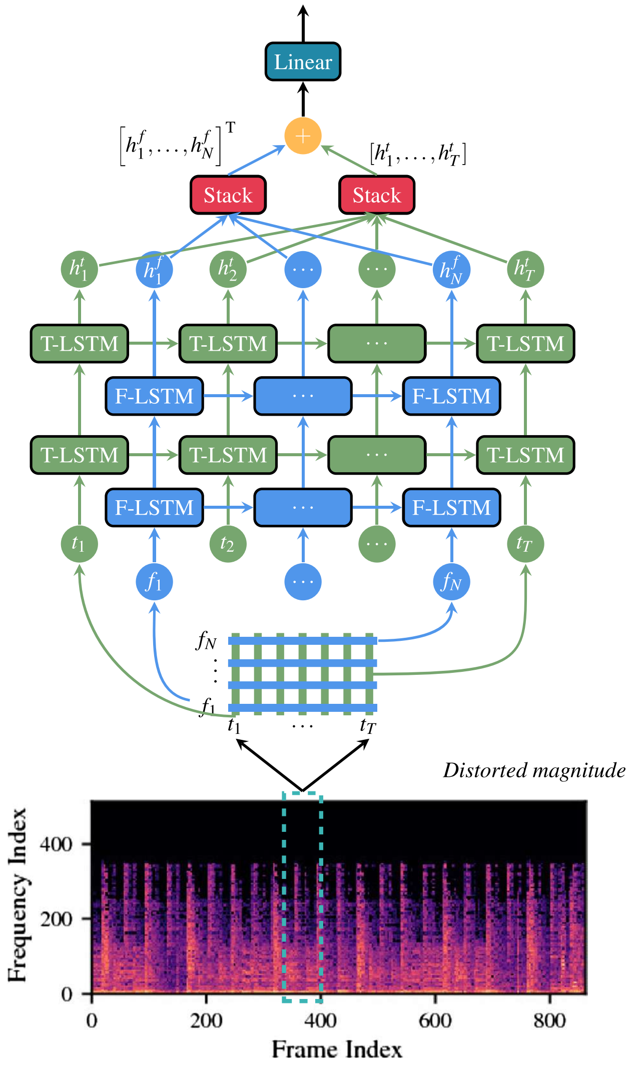


Fig. 1 A diagram of the unfolded time-frequency-LSTM (TF-LSTM) network proposed in this paper for audio restoration. It consists of two separate LSTM sub-networks: the Time LSTM (T-LSTM) that is responsible for the time direction and the Frequency-LSTM (F-LSTM) that is responsible for the frequency direction. Here, the T-LSTM is unfolded in time while the F-LSTM is unfolded in frequency. Inputs of F-LSTM do not come from outputs from T-LSTM, and vice versa

nonlinear mapping from low-quality magnitude to its original high-quality magnitude by exploiting temporal and spectral dynamics. The overall design of the proposed system using TF-LSTM is shown in Fig. 2, which consists of feature extraction, model learning, and audio waveform reconstruction.

The feature extraction module transforms a raw waveform signal into time-frequency representations by windowing with the Hann window (raised cosine) and performing STFT, yielding the magnitude of the Fourier coefficients and the phase information. In addition, we apply the power and log operations to the resulting magnitudes, ending up the log-spectral power magnitudes, which are then used as the input to TF-LSTM.

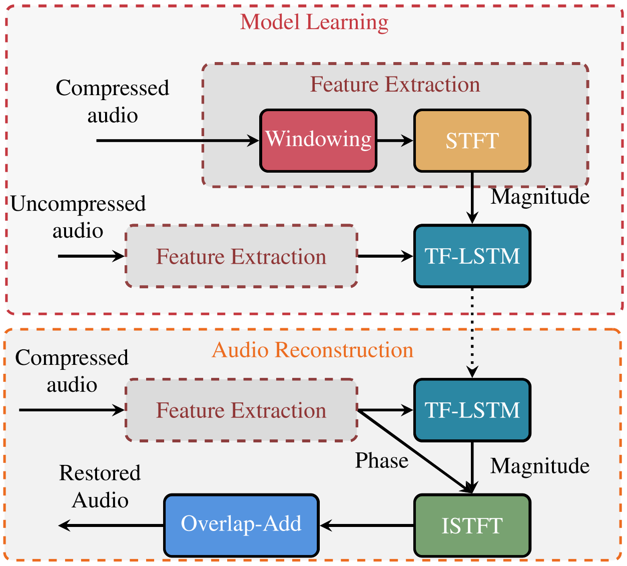


Fig. 2 The overall design of the proposed system using TF-LSTM for audio restoration

Next, given a pair of the log-spectral power magnitudes of a compressed music audio signal and the one of its corresponding uncompressed audio, the model training of TF-LSTM is to learn a mapping with the purpose of reconstructing the uncompressed magnitude from its compressed counterpart. Here, compressed audio is obtained from an MP3 codec.

Finally, given the magnitudes of a compressed music audio signal unseen in the training phase, the outputs of TF-LSTM are treated as an estimation of the restored magnitudes. Afterward, the inverse step of the log and power operations is performed on these restored magnitudes. The inverse Short-Time Fourier Transform (ISTFT) is computed from the combination of the restored magnitudes and the original phase information of the compressed audio. In addition, overlap-add with the same Hann window as applied during feature extraction is implemented to reconstruct the audio signal [16].

3.3 Loss function

Given a set of high-quality magnitudes $\{X_n\}$ and their corresponding low-quality magnitudes $\{Y_n\}$, the objective of the TF-LSTM-RNN method shown in Fig. 1 is to learn the nonlinear mapping function f from low-quality data $\{Y_n\}$ to high-quality data $\{X_n\}$. To this end, learning the nonlinear mapping function f is equivalent to the estimation of all the parameters θ of the TF-LSTM-RNN. In this work, this is achieved by minimizing the reconstruction error between the reconstructed magnitudes $f(Y;\theta)$ and the corresponding ground truth high-quality magnitudes X. That is, we use mean squared error (MSE) as the loss function:

$$J(\theta) = \frac{1}{N}\sum_{n=1}^{N} ||f(Y_n;\theta) - X_n||^2, \tag{8}$$

where N is the number of given training samples. The loss is minimized using stochastic gradient descent with the standard Back-Propagation Through Time (BPTT) algorithm [59].

4 Experiments

4.1 Comparison to state-of-the-art deep neural networks

We compare the following eight types of deep neural networks to evaluate our proposed TF-LSTM network in the context of the current state of the art. In the following, our proposed method is referred to as TF-LSTM. Note that for the fair comparison, we conducted generally hyper-parameters search for each type of deep neural networks, including the number of hidden units, the number of layers, the number of conventional maps. The learning rate and mini-batch have negligible effects of the performance due to the large training data. Tables 1 and 2 summarize the input audio representations and key hyper-parameters for all the deep neural networks used in the experiments.

Table 1 Audio representations were used for the different deep neural networks in the experiments

Net type	Audio representations
RNNs	
T-LSTM	Log-spectral magnitudes (0–15 kHz)
T-BLSTM	Log-spectral magnitudes (0–15 kHz)
F-LSTM	Log-spectral magnitudes (0–15 kHz)
TF-LSTM	Log-spectral magnitudes (0–15 kHz)
CNNs	
VGG-like CNN	Log-spectral magnitudes (0–15 kHz)
U-CNN	Waveform samples
WaveNet	Waveform samples
RNNs + CNNs	
LSTM + VGG-like CNN	Log-spectral magnitudes (0–15 kHz)
VGG-like CNN + LSTM	Log-spectral magnitudes (0–15 kHz)

4.1.1 Recurrent neural networks

Since TF-LSTM is derived from LSTM-RNNs, the comparison naturally starts with three representative LSTM-RNNs.

(1) *Time-LSTM* It uses a classic LSTM-RNN in place of the TF-LSTM-RNN model in Fig. 2, which only models temporal dynamics. This method is termed T-LSTM.

(2) *Time-BLSTM* Likewise, this comparison method uses BLSTM [14] to exploit temporal dynamics from the past and future context, termed T-BLSTM.

(3) *Frequency-LSTM* Frequency-LSTM learns only spectral dynamics via the recurrent connections, which is termed F-LSTM.

4.1.2 Convolutional neural networks

In addition to LSTM-RNNs, we evaluate three state-of-the-art CNN architectures for the first time for the task of compressed audio restoration. These CNNs have either produced state-of-the-art results on automatic speech recognition (ASR) or audio generation tasks [42, 52, 54] or audio super-resolution [22], which is related to compressed audio restoration but still quite different.

(4) *VGG-like CNN* The deep CNN used for comparison here is deeply rooted in the work of the VGG convolutional net, which was originally proposed for image classification in the ImageNet 2014 competition [55]. Recently, the VGG-inspired networks have been successfully adapted to ASR [52, 54], large-scale audio classification [17], speech anger detection [9]. The fundamental idea of the VGG net is to use small 3 × 3 convolutional kernels with Rectified Linear Unit (*ReLU*) [40] nonlinear functions without pooling between these layers. We apply this principle to constructing a VGG-inspired CNN network with 4 hidden convolutional layers and 1 fully connected layer for music restoration. Note that each convolution layer has 32 maps.

(5) *U-CNN* In [22], the authors trained a U-net like neural network introduced in [50] for enhancing the quality of audio signals such as speech or music by transforming inputs of a low sampling rate into signals with a higher sampling rate (up-sampling).

We strictly followed the work described in [22] to replicate the implementation of the U-net-like CNN for the music audio restoration. The replicated network takes 16 384 compressed samples to predict the corresponding 16 384 uncompressed samples. The network basically consists of a stack of down-sample blocks (*D block*) and up-sampling blocks (*U Block*). The down-sampling blocks perform a convolution, batch normalization [20], and apply a ReLU nonlinearity. We used a stride of two to reduce the dimensionality of the input. The up-sampling blocks apply a nearest neighbor up-sampling with a factor of 2 [41], a

Table 2 The hyper-parameters used for the experiments

Net type	# RNN layers	# Hid. units	# CNN maps (filters)	# CNN-layers
RNNs				
T-LSTM	2	256	–	–
T-BLSTM	2	256	–	–
F-LSTM	2	256	–	–
TF-LSTM	2	256	–	–
CNNs				
VGG-like CNN	–	–	32	4
U-CNN	–	–	32	6
WaveNet	–	–	32 and 256*	21
RNNs + CNNs				
LSTM + VGG-like CNN	2	256	32	3
VGG-like CNN + LSTM	2	256	32	3

*Two types of convolutional layers in the WaveNet have different numbers of convolutional maps. The number of residual convolutional maps are set to be 32 and the number of skip convolutional maps are set to be 256

convolution, batch normalization, and apply a ReLU nonlinearity. As shown in [22], skip connections from the down-sampling blocks to the up-sampling blocks are added. An additive residual connection between the input layer and the output layer is considered. Each convolution in down-sampling blocks uses a kernel size of 30 and the number of padding of 14, while each convolution in up-sampling blocks uses a kernel size of 31 and the number of padding of 15. Note that, in practice, we found that batch normalization does not help the net in terms of SNR, we normally disable it to save GPU memory. Similar to TF-LSTM, the MSE function is chosen as the objective of the network. In the following, this method is referred to as U-CNN.

(6) *WaveNet* The motivation for WaveNet is that WaveNet is powerful enough to generate realistic-sounding human-like speech and music audio [4, 42]. WaveNet's ability to generate raw waveforms suggests that it can model any kind of audio. In this work, we adopt WaveNet to the music restoration problem.

In [42], WaveNet uses a discrete softmax output to avoid making any assumption on the shape of the output's distribution. However, the preliminary experiments with discrete softmax outputs proved disadvantageous for music restoration. We found that 8-bit μ-law quantization used in WaveNet introduced noise to the enhanced music audio and disproportionately amplified the noise. For these reasons, we proposed to formulate the music restoration task as a real-valued regression problem by using the fundamental WaveNet architecture.

The fundamental WaveNet architecture consists of a stack of *dilated* causal convolutional layers. As suggested in [42], the dilation is doubled for every layer up to a limit and then repeated: in our implementation, the dilation levels are $1, 2, 4, \ldots, 512, 1, 2, 4, \ldots, 512$. In addition, we set up the number of skip maps to be 256 and the number residual maps to be 32.

It is worth noting that when compared to with our proposed TF-LSTM method, one major disadvantage of the WaveNet architecture is that the WaveNet is particularly hungry of computational resources for the inference phase. For example, in [4], the authors reported that the Floating Point Operations (FLOPs) of the WaveNet with 40 layers is approximately 55×10^9 for every second of the forward phase. In contrast, the FLOPs of the TF-LSTM with 2 layers that is used in our experiments (see Table 2) is 0.49×10^9 for every second of the forward phase.

4.1.3 Combinations of RNNs and CNNs

The remaining two deep networks used for comparison are the combinations of LSTM-RNNs and CNNS, which have attracted increasing attention in the audio processing community [9].

(7) $LSTM + VGG-$likeCNN The architecture of the first combination uses multiple LSTM-RNNs on the input magnitude, then feeds the LSTM's output to the stacked VGG-like CNNs.

(8) $VGG - likeCNN + LSTM$ Likewise, the second combination is a convolutional feature extractor applied to the input magnitude, then multiple recurrent layers on top of the CNN's output.

4.2 Selected music data

For the experiments, we collected 1138 pop music tracks directly from music CDs. The CD-quality audio was then stored in stereo, 16-bit, 44.1 kHz sampling rate, PCM

WAV files. Further, for simplicity, we selected three different excerpts of 30 s from each track, resulting in an uncompressed audio dataset consisting of 3414 excerpts.

Next, the corresponding compressed audio data were created in the following way: first we encoded all excerpts with an MP3 codec; then we decoded all of the resulting compressed excerpts and saved the decompressed audio with 44.1 kHz sampling rate and 16-bit PCM WAV format. Both the MP3 encoding and MP3 decoding processes were done with LAME [46].

When the WAV files were converted into MP3 files, three low bitrates, mono, 96 kbps, mono, 64 kbps, and stereo, 96 kbps, which are commonly adopted for online music streaming services in our daily life, are considered. For clarity, here, we describe the process of the generation of mono MP3 audio at the three above bitrates:

1. MP3 (mono, 96 kbps): stereo audio is first downmixed to mono audio, then the mono audio is encoded at 96 kbps.
2. MP3 (mono, 64 kbps): the process is the same as the above one except the bitrate is set to 64 kbps.
3. MP3 (stereo, 96 kbps): stereo audio is first encoded at 96 kbps, then the only one channel is considered for restoration, in order to compare with the above two mono settings. Since the aforementioned deep learning-based audio restoration methods deal with the left and right channels independently, in the experiments, we simply selected the left channel for the music restoration process.

It is worth noting that the MP3 encoder for stereo 96 kbps MP3 data is essentially asked to compress the left and right channel information at 96 kbps, ending up the compressed left channel data (i. e., 48 kbps maximum) used for the experiments. In contrast, the encoder for mono 96 kbps MP3 data only requires to compress left channel data at 96 kbps. It turns out that the audio quality for stereo 96 kbps MP3 data is worsened than the one for mono 96 kbps MP3 data. The following experimental results in terms of SNR and LSD shown in Tables 3 and 5 confirms this statement.

For training and testing purpose, the entire data were randomly split into the training data (70%), the validation data (20%), and the test data (10%). All the hyper-parameters of deep neural networks were tuned based on the minimum loss value on the validation data.

4.3 Experimental setup

To perform STFT, we used a Hann window of 1 024 and a step size of 512. To find a trade-off between computational cost and efficiency for music audio processing, we chop a long music STFT sequence into a number of small subsequence, each which has only 100 frames. The feature extraction and music reconstruction are implemented by the librosa Python package [35].

Table 3 Objective results (in dB) for mono 96 kbps MP3 data. Higher SNR or LSD is better

Method	SNR	LSD
Baseline		
MP3 (mono 96 kbps)	22.07	11.61
RNNs		
T-LSTM	21.88	11.60
T-BLSTM	21.62	11.52
F-LSTM	23.46	12.25
CNNs		
VGG-like CNN	24.66	12.48
U-CNN [22]	18.70	11.63
WaveNet [42]	21.60	10.62
RNNs + CNNs		
LSTM + VGG-like CNN	20.85	11.10
VGG-like CNN + LSTM	19.99	11.21
Our proposed method		
TF-LSTM	23.68	12.39

For data pre-processing, we used statistics from the training data to perform mean subtraction and standard divide for the input data and the target data when the input audio representations are log-spectral power magnitudes. For the audio reconstruction, we first undo the data pre-processing effects for the reconstructed magnitudes and use such magnitudes and the phase information from the input (compressed) to run the ISTFT. For WaveNet and U-CNN, we do not perform any data pre-processing steps for the input and target.

We used the Adam optimiser [21] with the learning rate of $1e\ -3$ and a mini-batch size of 32 to update the parameters. We also reduced the learning rate by a factor of 2 when the loss on the validation set stops decreasing. All of the deep networks are implemented by the open-source PyTorch deep learning library [45].

4.4 Objective results

We apply two metrics to objectively measuring music audio quality, which have commonly used for assessing the quality of enhanced signals [6, 15, 23, 29]. First, signal-to-noise ratio (SNR) is defined as

$$\mathrm{SNR}(x, \hat{x}) = 10 \log_{10} \frac{||x||_2^2}{||x - \hat{x}||_2^2} \tag{9}$$

for a signal x and its approximation $\hat{x}$.

Next, log-spectral distance (LSD) [15] measures the reconstruction quality of individual frequency band, computed as follows:

$$\mathrm{LSD}(x,\hat{x}) = -10\log_{10}\left(\frac{1}{L}\sum_{l=1}^{L}\sqrt{\frac{1}{W}\sum_{f=1}^{W}\left(X(l,f)-\hat{X}(l,f)\right)^{2}}\right) \tag{10}$$

where X and $\hat{X}$ are the log-spectral power magnitudes of x and $\hat{x}$, respectively, L is the total number of frames, W is the total number of frequency bins. Note that, the higher SNR or LSD is, the better the reconstruction quality is.

Table 3 compares the performance of the proposed TF-LSTM approach with the other state-of-the-art approaches as outlined in Sect. 4.1 for the mono 96 kbps audio coding restoration. From the table, we can observe that the TF-LSTM can remarkably enhance the audio quality in a MP3 coding format by objective measures, leading to 23.68 dB SNR and 12.39 dB LSD. Moreover, the TF-LSTM performs better than all other approaches, except the VGG-like CNN (i. e., 24.66 dB SNR and 12.48 dB LSD).

More specifically, when evaluating the different kinds of RNNs, one can see that the F-LSTM outperforms the classic T-LSTM or T-BLSTM. This implicitly suggests that the distorted music because of the MP3 codec compression has a stronger frequency dynamics than the temporal dynamics, which is, however, seldom explored in previous works as discussed in Sect. 2. When integrating the T-LSTM and F-LSTM into a proposed TF-LSTM framework, the system further yields performance gain. This demonstrates that jointly exploring the temporal and frequency dynamics is of vital importance for the targeted task in this paper.

We further evaluate three types of CNN networks, i. e., VGG-like CNN, U-CNN, and WaveNet, for the enhancement systems. It can be seen that the VGG-like CNN can perform much better than the other two, and is also notably superior to the compressed MP3 coding format.

To take advantage of the CNNs for efficient feature extraction and the LSTM-RNNs for context-dependence learning, we incorporated the best CNN networks (VGG-like CNN) and the classic LSTM networks into a joint learning framework (i. e., LSTM + VGG-like CNN or VGG-like CNNS + LSTM). The achieved results are unfortunately not promising.

To access the robustness of TF-LSTM, we further carried out similar experiments in two more adverse scenarios by using the mono 64 kbps MP3 data or the stereo 96 kbps MP3 data. The results for both scenarios are shown in Tables 4 and 5, respectively. Similar finding is observed that the TF-LSTM approach can improve the quality of low-bitrate music, and is competitive to, and even superior to, the best recently reported approaches in terms of SNR and LSD.

More specifically, in the case of using mono 64 kbps MP3 data, TF-LSTM beats all other state-of-the-art approaches, and achieves the best results of 19.22 dB SNR and 9.78 dB LSD. In the case of using stereo 96 kbps MP3 data, TF-LSTM also yields competitive results (i. e., 18.19 dB SNR and 9.41 dB LSD) to the best ones achieved

Table 4 Objective results (in dB) for the mono 64 kbps MP3 data. Higher SNR or LSD is better

Method	SNR	LSD
Baseline		
MP3 (mono, 64 kbps)	18.43	9.39
RNNs		
T-LSTM	18.53	9.56
T-BLSTM	18.66	9.64
F-LSTM	18.51	9.60
CNNs		
VGG-like CNN	18.96	9.68
U-CNN [22]	16.50	9.24
WaveNet [42]	18.98	9.44
RNNs + CNNs		
LSTM + VGG-like CNN	17.35	9.20
VGG-like CNN + LSTM	18.44	9.64
Our proposed method		
TF-LSTM	19.22	9.78

Table 5 Objective Results (in dB) for the stereo 96 kbps MP3 data. Higher SNR or LSD is better

Method	SNR	LSD
Baseline		
MP3 (stereo, 96 kbps)	17.81	9.15
RNNs		
T-LSTM	17.63	9.21
T-BLSTM	18.16	9.40
F-LSTM	17.84	9.33
CNNs		
VGG-like CNN	17.42	9.13
U-CNN [22]	15.83	8.99
WaveNet [42]	18.33	9.35
RNNs + CNNs		
LSTM + VGG-like CNN	17.17	9.03
VGG-like CNN + LSTM	17.78	9.34
Our proposed method		
TF-LSTM	18.19	9.41

by WaveNet (i. e., 18.33 dB SNR and 9.35 dB LSD), whereas the later requires higher computational cost. This indicates that TF-LSTM is not only effective but also robust to capture the temporal and frequency dynamics for music restoration in different distortion scenarios.

4.5 Subjective results

The subjective listening tests were blind and crowdsourced. Fifty music clips not included in the training data were used for evaluation. In total, 10 subjects participated in the Mean Opinion Score (MOS) listening test based on the CrowdMOS toolkit and methodology given in [49]. Test stimuli were randomly chosen and presented for each subject. After listening to each stimulus, the subjects were asked to rate the quality of the stimulus in a five-point Likert scale score (1: bad, 2: poor, 3: fair, 4: good, 5: excellent).

To reduce the experimental workload, we particularly selected the best, rather than all, state-of-the-art approaches evaluated by objective metrics, and compared them with the proposed TF-LSTM in terms of MOS. Tables 6 and 7 present the MOS test results for the mono 96 kbps and stereo 96 kbps data. From both tables, it can be seen that all approaches achieve higher 5-scale MOSs than the mono 96 kbps and stereo 96 kbps MP3 data, suggesting that deep neural network-based methods for music audio restoration are feasible to improve the perceptual quality of music signals, distorted by MP3 encoder. Moreover, the TF-LSTM notably outperforms the WaveNet in terms of MOS when enhancing the stereo MP3 96 kbps data.

To intuitively demonstrate the performance of TF-LSTM, we illustrated the spectrogram of the uncompressed audio, the MP3 one (stereo, 96 kbps), the restored one by WaveNet, and the restored one by TF-LSTM in Fig. 3. First, it can be found that WaveNet can regenerate high-frequency information (15–22.05 kHz), but the spectral holes made by the MP3 encoding are still untouched and the generated frequency information is like random noise instead of harmonic patterns. In comparison, the spectral holes of TF-LSTM seem to be visibly “smeared”, indicating TF-LSTM indeed recovers the missing spectral information.

Table 6 Subjective 5-scale mean opinion scores (MOS) of music samples from 10 subjects for mono 96 kbps MP3 data

Method	MOS
Uncompressed (mono)	3.85 ± 0.89
MP3 (mono, 96 kbps)	3.74 ± 0.95
VGG-like CNN	3.83 ± 0.92
TF-LSTM	3.78 ± 0.93

Table 7 Subjective 5-scale mean opinion scores (MOS) of music samples from 10 subject for the stereo 96 kbps MP3 data

Method	MOS
Uncompressed (left channel)	3.87 ± 0.99
MP3 (stereo, 96 kbps)	3.69 ± 0.98
WaveNet [42]	3.76 ± 0.92
TF-LSTM	3.83 ± 0.89

Moreover, we calculated the average spectral distance between the uncompressed music audio, and the MP3 one (stereo, 96 kbps), the restored one by WaveNet, the restored one by TF-LSTM. Here, the absolute error between the average magnitude of x and that of $\hat{x}$ is computed as a measure of the average spectral distance. The average spectral distance is mathematically defined as follows:

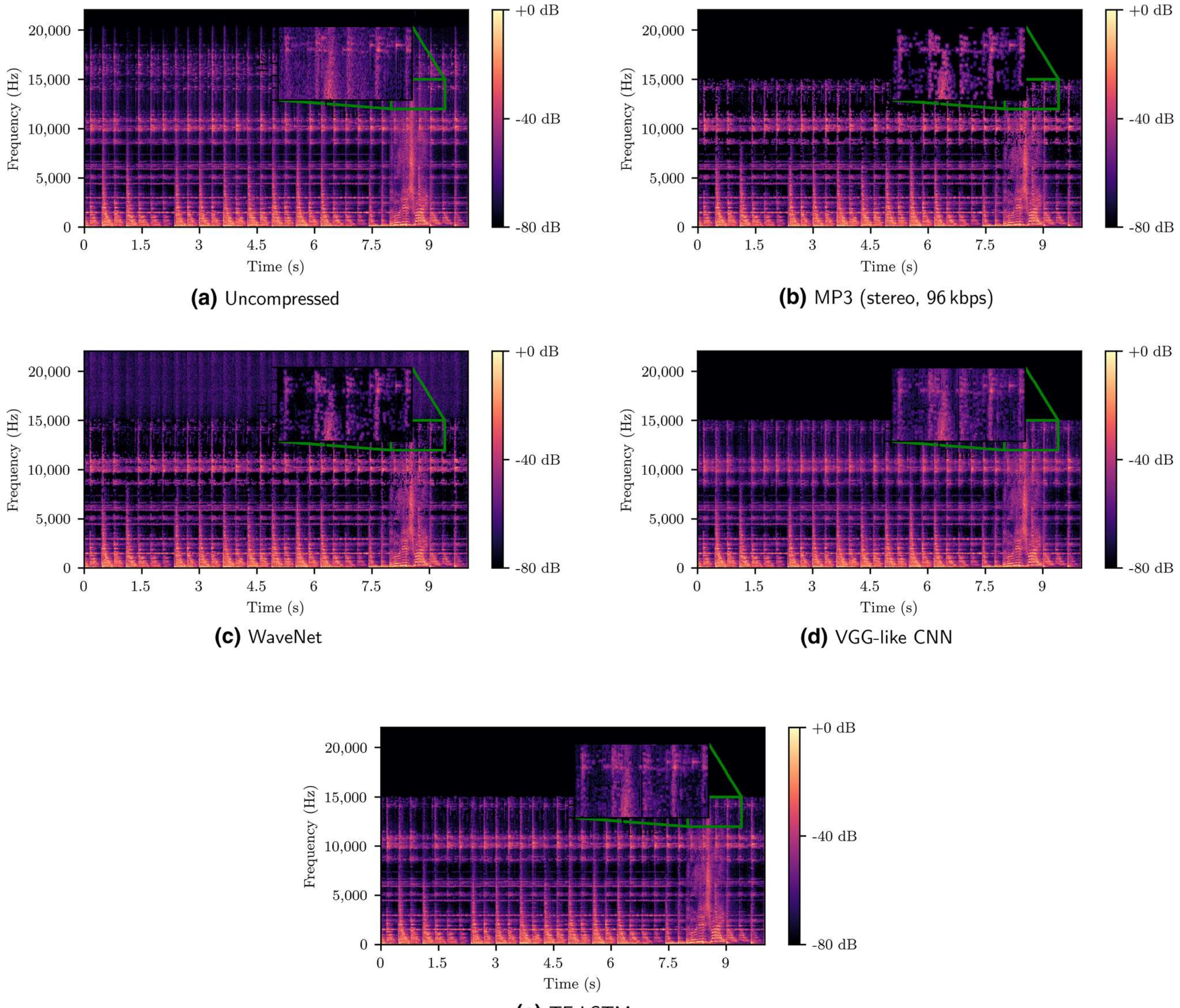
$$\mathrm{d}(x, \hat{x}; f) = \left| \frac{1}{L} \sum_{l=1}^{L} X(l,f) - \frac{1}{L} \sum_{l=1}^{L} \hat{X}(l,f) \right|. \tag{11}$$

The results are plotted in Fig. 4. Among the frequency range [0, 15 kHz], the figure clearly shows that the TF-LSTM approach holds lower average spectral distance than MP3 and WaveNet, which consequently leads to a better quality performance in music restoration.

5 Conclusions and outlook

To enhance the quality of music audio based on spectral correlations, we have proposed a novel Time-Frequency-LSTM-RNN (TF-LSTM) architecture to advantageously exploit temporal and spectral dynamics. A TF-LSTM-RNN network is equipped with two separate LSTM-RNN layers so that it is capable of effectively modeling two-dimensional time-frequency information. Leveraging the power of TF-LSTM, our proposed system for audio restoration can learn a nonlinear mapping from the spectral magnitudes of low-quality (compressed) audio to those of high-quality (uncompressed) audio. Objective and subjective listening test results demonstrate that TF-LSTM restored audio samples outperform the MP3 audio and the audio signals enhanced by other state-of-the-art deep neural networks.

In the future, we plan to leverage psychoacoustic models so that deep nets can better focus on audible artifacts, e. g., by employing a perceptual loss function. Besides, the achieved performance improvement reported in this paper



(a) Uncompressed

(b) MP3 (stereo, 96 kbps)

(c) WaveNet

(d) VGG-like CNN

(e) TF-LSTM

Fig. 3 Illustration of spectral enhancement using the proposed TF-LSTM-RNNs

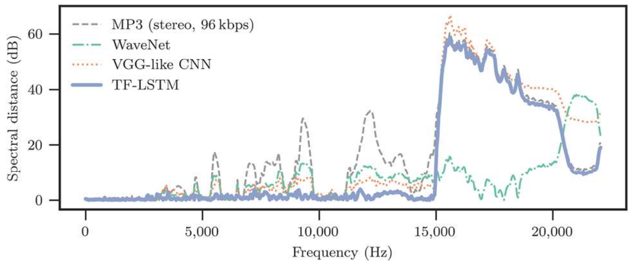


Fig. 4 Illustration of average spectral distance

was made by only recovering spectral magnitudes. Acoustic phase information is yet to be considered in future work. We believe that this will enable a new experience of compressed music enjoyment and perhaps allow for even better compression algorithms. Further, as the VGG-like CNN and CNN-based WaveNet establish the ability of addressing the music restoration task, we also plan to continue to investigate other CNN-based CNN architectures for this challenging problem.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Abbaszadeh P (2016) Improving hydrological process modeling using optimized threshold-based wavelet de-noising technique. Water Resour Manag 30(5):1701–1721
2. Aliabadi M, Golmohammadi R, Mansoorizadeh M, Khotanlou H, Hamadani AO (2013) An empirical technique for predicting noise exposure level in the typical embroidery workrooms using artificial neural networks. Appl Acoust 74(3):364–374
3. Amodei D, Ananthanarayanan S, Anubhai R, Bai J, Battenberg E, Case C, Casper J, Catanzaro B, Cheng Q, Chen G, et al (2016) Deep speech 2: end-to-end speech recognition in English and Mandarin. In: Proceedings of international conference on machine learning (ICML), New York City, NY, USA, pp 173–182
4. Arik SO, Chrzanowski M, Coates A, Diamos G, Gibiansky A, Kang Y, Li X, Miller J, Raiman J, Sengupta S, et al (2017) Deep voice: real-time neural text-to-speech. arXiv preprint arXiv:1702.07825
5. Brandenburg K (1999) MP3 and AAC explained. In: Audio engineering society conference: 17th international conference: high-quality audio coding. Audio Engineering Society
6. Cohen I, Gannot S (2008) Spectral enhancement methods. In: Springer handbook of speech processing. Springer, pp 873–902
7. Deng J, Zhang Z, Marchi E, Schuller B (2013) Sparse autoencoder-based feature transfer learning for speech emotion recognition. In: Proceedings of ACII, Geneva, Switzerland, pp 511–516
8. Deng J, Zhang Z, Eyben F, Schuller B (2014) Autoencoder-based unsupervised domain adaptation for speech emotion recognition. IEEE Signal Process Lett 21(9):1068–1072
9. Deng J, Eyben F, Schuller B, Burkhardt F (2017) Deep neural networks for anger detection from real life speech data. In: Proceedings of 2nd international workshop on automatic sentiment analysis in the wild (WASA 2017) held in conjunction with the 7th biannual conference on affective computing and intelligent interaction (ACII 2017), AAAC, IEEE, San Antonio, TX
10. Deng J, Xu X, Zhang Z, Frühholz S, Schuller B (2017) Universum autoencoder-based domain adaptation for speech emotion recognition. IEEE Signal Process Lett 24(4):500–504
11. Dietz M, Liljeryd L, Kjorling K, Kunz O (2002) Spectral band replication, a novel approach in audio coding. In: Audio engineering society convention 112
12. Disch S, Bäckström T (2017) Bandwidth extension. In: Speech coding: code- excited linear prediction. Springer, Cham, chap 11, pp 151–160
13. Graves A, Fernández S, Schmidhuber J (2007) Multi-dimensional recurrent neural networks. In: Proceedings of International conference on artificial neural networks (ICANN), Porto, Portugal, pp 549–558
14. Graves A et al (2012) Supervised sequence labelling with recurrent neural networks, vol 385. Springer, Berlin
15. Gray A, Markel J (1976) Distance measures for speech processing. IEEE Trans Acoust Speech Signal Process 24(5):380–391
16. Griffin D, Lim J (1984) Signal estimation from modified short-time fourier transform. IEEE Trans Acoust Speech Signal Process 32(2):236–243
17. Hershey S, Chaudhuri S, Ellis DP, Gemmeke JF, Jansen A, Moore RC, Plakal M, Platt D, Saurous RA, Seybold B, et al (2017) CNN architectures for large-scale audio classification. In: Proceedings Of ICASSP, New Orleans, USA, pp 131–135
18. Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507
19. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
20. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of ICML, Lille, France, pp 448–456
21. Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: Proceedings of ICLR, San Diego, USA
22. Kuleshov V, Enam SZ, Ermon S (2017) Audio super-resolution using neural nets. https://openreview.net/pdf?id=S1gNakBFx. Accessed 12 July 2017
23. Kuleshov V, Enam SZ, Ermon S (2017) Audio super resolution using neural networks. CoRR abs/1708.00853, arXiv:1708.00853
24. Larsen ER, Aarts RM (2004) Audio Bandwidth extension: application of psychoacoustics, signal processing and loudspeaker design. Wiley, Hoboken
25. Lee J, Park J, Kim KL, Nam J (2017) Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. arXiv preprint arXiv:1703.01789
26. Li J, Mohamed A, Zweig G, Gong Y (2015) LSTM time and frequency recurrence for automatic speech recognition. In: Proceedings of IEEE workshop on automatic speech recognition and understanding (ASRU), Scottsdale, AZ, USA, pp 187–191
27. Li J, Mohamed A, Zweig G, Gong Y (2016) Exploring multidimensional LSTMs for large vocabulary ASR. In: 2016 IEEE international conference on acoustics, speech and signal processing, ICASSP 2016, Shanghai, China, March 20–25, 2016, pp 4940–4944
28. Li K, Lee C (2015) A deep neural network approach to speech bandwidth expansion. In: 2015 IEEE International conference on acoustics, speech and signal processing, ICASSP 2015, South Brisbane, Queensland, Australia, pp 4395–4399
29. Li K, Huang Z, Xu Y, Lee C (2015) DNN-based speech bandwidth expansion and its application to adding high-frequency missing features for automatic speech recognition of narrowband speech. In: Proceedings of INTERSPEECH, pp 2578–2582
30. Li R, Wu Z, Ning Y, Sun L, Meng H, Cai L (2017) Spectro-temporal modelling with time-frequency LSTM and structured output layer for voice conversion. In: Proceedings of INTERSPEECH, Stockholm, Sweden, pp 3409–3413
31. Liu CM, Hsu HW, Lee WC (2008) Compression artifacts in perceptual audio coding. IEEE Trans Audio Speech Lang Process 16(4):681–695
32. Liu X, Bao C, Jia M, Sha Y (2010) A harmonic bandwidth extension based on gaussian mixture model. In: 2010 IEEE 10th international conference on signal processing (ICSP). IEEE, pp 474–477
33. Liu Y, Wang D (2017) Time and frequency domain long short-term memory for noise robust pitch tracking. In: Proceedings of IEEE international conference on acoustics, speech and signal processing (ICASSP), New Orleans, LA, USA, pp 5600–5604
34. Maas AL, Le QV, O'Neil TM, Vinyals O, Nguyen P, Ng AY (2012) Recurrent neural networks for noise reduction in robust ASR. In: Proceedings of INTERSPEECH
35. McFee B, Raffel C, Liang D, Ellis DP, McVicar M, Battenberg E, Nieto O (2015) librosa: audio and music signal analysis in python. In: Proceedings of the 14th python in science conference, pp 18–25
36. McLaren M, Lei Y, Scheffer N, Ferrer L (2014) Application of convolutional neural networks to speaker recognition in noisy conditions. In: Proceedings of INTERSPEECH, Singapore
37. Morzyński L, Makarewicz G (2003) Application of neural networks in active noise reduction systems. Int J Occup Saf Ergon 9(3):257–270
38. Nagel F, Disch S (2009) A harmonic bandwidth extension method for audio codecs. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, ICASSP 2009, 19–24 April 2009, Taipei, Taiwan, pp 145–148

39. Nagel F, Disch S, Wilde S (2010) A continuous modulated single sideband bandwidth extension. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, ICASSP 2010, Dallas, Texas, USA, pp 357–360
40. Nair V, Hinton GE (2010) Rectified linear units improve restricted Boltzmann machines. In: Proceedings of ICML, Haifa, Israel, pp 807–814
41. Odena A, Dumoulin V, Olah C (2016) Deconvolution and checkerboard artifacts. http://distill.pub/2016/deconv-checkerboard/. Accessed 12 July 2017
42. Oord A, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K (2016) Wavenet: a generative model for raw audio. arXiv preprint arXiv:1609.03499
43. Painter T, Spanias A (2000) Perceptual coding of digital audio. Proc IEEE 88(4):451–515
44. Park SR, Lee J (2016) A fully convolutional neural network for speech enhancement. arXiv preprint arXiv:1609.07132
45. Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A (2017) Automatic differentiation in pytorch. In: Proceedings of NIPS workshop
46. Project TL (2017) Lame. http://lame.sf.net. lAME 64 bits version 3.99.5
47. Pulakka H, Remes U, Palomäki KJ, Kurimo M, Alku P (2011) Speech bandwidth extension using gaussian mixture model-based estimation of the highband MEL spectrum. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, ICASSP 2011, May 22–27, 2011, Prague Congress Center, Prague, Czech Republic, pp 5100–5103
48. Pulakka H, Remes U, Yrttiaho S, Palomäki KJ, Kurimo M, Alku P (2012) Bandwidth extension of telephone speech to low frequencies using sinusoidal synthesis and a Gaussian mixture model. IEEE Trans Audio Speech Lang Process 20(8):2219–2231
49. Ribeiro FP, Florêncio DAF, Zhang C, Seltzer ML (2011) CROWDMOS: an approach for crowdsourcing mean opinion score studies. In: Proceedings of the IEEE international conference on acoustics, speech, and signal processing, ICASSP 2011, Prague, Czech Republic, pp 2416–2419
50. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241
51. Sak H, Senior AW, Beaufays F (2014) Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: INTERSPEECH 2014, 15th annual conference of the international speech communication association, Singapore, pp 338–342
52. Saon G, Sercu T, Rennie SJ, Kuo HJ (2016) The IBM 2016 english conversational telephone speech recognition system. In: Proceedings of INTERSPEECH, CA, USA, pp 7–11
53. Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE Trans Signal Process 45(11):2673–2681
54. Sercu T, Puhrsch C, Kingsbury B, LeCun Y (2016) Very deep multilingual convolutional neural networks for LVCSR. In: Proceedings of ICASSP, Shanghai, China, pp 4955–4959
55. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556
56. Spanias A, Painter T, Atti V (2006) Audio signal processing and coding. Wiley, Hoboken
57. Stollenga MF, Byeon W, Liwicki M, Schmidhuber J (2015) Parallel multi-dimensional LSTM, with application to fast biomedical volumetric image segmentation. In: Proceedings of Advances in neural information processing systems (NIPS), Montreal, Quebec, Canada, pp 2998–3006
58. Sze V, Chen YH, Yang TJ, Emer J (2017) Efficient processing of deep neural networks: a tutorial and survey. arXiv preprint arXiv:1703.09039
59. Werbos PJ (1990) Backpropagation through time: what it does and how to do it. Proc IEEE 78(10):1550–1560
60. Wolters M, Kjorling K, Homm D, Purnhagen H (2003) A closer look into mpeg-4 high efficiency AAC. In: Audio engineering society convention 115
61. Wu C, Vinton M (2017) Blind bandwidth extension using k-means and support vector regression. In: 2017 IEEE international conference on acoustics, speech and signal processing, ICASSP 2017, New Orleans, LA, USA, March 5–9, 2017, pp 721–725
62. Zernicki T, Domanski M (2008) Improved coding of tonal components in MPEG-4 AAC with SBR. In: 2008 16th European signal processing conference, EUSIPCO 2008, August 25–29, 2008, Lausanne, Switzerland, pp 1–5