A TIME-DOMAIN CONVOLUTIONAL RECURRENT NETWORK FOR PACKET LOSS CONCEALMENT

Ju Lin[1,*], *Yun Wang*[2], *Kaustubh Kalgaonkar*[2], *Gil Keren*[2], *Didi Zhang*[2], *Christian Fuegen*[2]

[1] Clemson University [2] Facebook AI

ABSTRACT

Packet loss may affect a wide range of applications that use voice over IP (VoIP), e.g. video conferencing. In this paper, we investigate a time-domain convolutional recurrent network (CRN) for online packet loss concealment. The CRN comprises a convolutional encoder-decoder structure and long short-term memory (LSTM) layers, which have been shown to be suitable for real-time speech enhancement applications. Moreover, we propose lookahead and masked training to further improve the performance of the CRN framework. Experimental results show that the proposed system outperforms a baseline system using only LSTM layers in terms of two objective metrics – perceptual evaluation of speech quality (PESQ) and short-term objective intelligibility (STOI); it also reduces the word error rate (WER) more than the baseline when used as a frontend for speech recognition. The advantage of the proposed system is also verified in a subjective evaluation by the mean opinion score (MOS).

Index Terms— Voice over IP, packet Loss concealment, neural networks, long short-term memory

1. INTRODUCTION

With the widespread usage of the Internet, voice over Internet Protocol (VoIP) has become increasingly popular. However, IP packets may be lost due to delay and jitter during audio data transmission, which degrades the speech quality [1]. To address the unreliable delivery of voice packets over the Internet, many packet loss concealment (PLC) methods have been developed and refined during the last several decades. Apart from trivial methods such as zero filling and repeating the segment before a lost packet, more intelligent PLC algorithms make use of the redundant information embedded in neighboring packets. Examples include interpolation-based methods [2] and model-based algorithms, e.g. hidden Markov models (HMM) [3] and linear predictive coding (LPC) [4].

Recently, deep learning techniques have been introduced to PLC, because they are capable of learning complex hierarchical functions. In [5], a deep neural network (DNN) was used as a nonlinear regression function for PLC, in which the model was trained using the log-power spectral features. A recurrent neural network (RNN) based speech signal predictor was proposed in [6], which directly operates on time-domain speech samples. More recent contributions propose to use generative RNNs that predict a frame using the preceding frames [7].

PLC algorithms may be classified into online and offline methods. In online methods, the system must make predictions for lost frames in real time, and generally can only use its left context. The works above [5, 6, 7] all fall into this category. Online PLC systems enjoy the advantage of a short latency, which is usually no more than a frame (typically $10 \sim 20$ ms). By contrast, offline methods process larger chunks of audio containing lost packets, and can make use of the context in both directions. These methods trade latency for better speech quality. In [8], a convolutional U-Net based approach was introduced that uses deep feature losses during the training stage to recover the missing signal. Several auto-encoder based approaches were investigated recently [9, 10]. In [11, 12, 13], generative adversarial network (GAN) based frameworks were proposed for PLC.

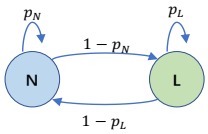
PLC algorithms may also be divided into time-domain and spectral-domain methods. The performance of spectral-domain approaches [14, 15] may be limited, because they need to recover the phase information in order to reconstruct the waveform. Time-domain based end to end methods [6, 7] overcome this limitation, and time-domain methods in the online setting are most suitable for real-time applications.

In this paper, we propose an online time-domain convolutional recurrent network (CRN) framework for PLC. The framework consists of a convolutional encoder-decoder architecture and multiple long short-term memory (LSTM) layers. CRNs [16] were originally proposed for speech enhancement operating on time-frequency (TF) representations; we adapt it to the time domain in this work. We also propose two techniques (lookahead and masked training) to further improve the performance. The lookahead technique allows the system to peek at one frame in the future; it earns a big improvement in the speech quality at the cost of a slightly increased latency. We note that lookahead for PLC was also proposed in [17]; our contribution here is showing that lookahead can be easily integrated into a deep learning framework. The masked training technique deals with the scenario of high packet loss rates. During inference, the context leading up to the frame to be predicted can contain many reconstructed frames instead of original frames, creating a mismatch between training and inference. To mitigate this mismatch, we replace original frames with reconstructed frames with a certain probability during training as well. The effectiveness of the proposed framework and techniques is verified with both objective and subjective evaluations.

2. PACKET LOSS SIMULATOR

To create training and test data for PLC, it is essential to simulate realistic scenarios of packet loss. Algorithms for packet loss simulation include random sampling, two-state Markov chain models, and Gilbert-Elliot models [18]. We adopt a two-state Markov chain to model the packet loss behaviour. As shown in Fig. 1, the Markov chain consists of a "non-loss" state (N) and a "loss" state (L); transitions between the two states are dictated by two probabilities p_N and p_L. By configuring these two probabilities, we can generate speech signals with different expected packet loss rates, which can be calculated as $\frac{1-p_N}{2-p_N-p_L}$ (see Fig. 1). Note that the values in the table

*Work performed during internship at Facebook AI.



p_N	p_L	Expected loss%
0.9	0.1	10.0%
0.9	0.5	16.7%
0.5	0.1	35.7%
0.1	0.1	50.0%

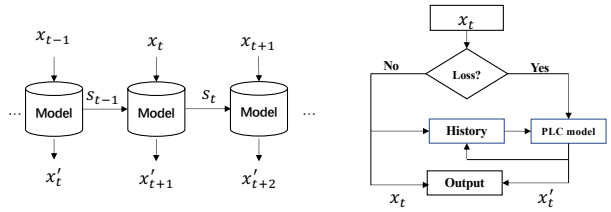
Fig. 1. Different configurations of the two-state Markov chain, and the resulting expected packed loss rates. N: non-loss state; L: loss state.

are *expected* packet loss rates; actual packet loss rates for individual sentences may vary.

3. MODELS

3.1. Generative Model for PLC

Training. Similar to [7], we regard PLC as a generative task with an auto-regressive training scheme. Given an audio signal $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ with T frames, we use the first $T-1$ frames ($\boldsymbol{X} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{T-1}$) as the input sequence to the model, and the last $T-1$ frames ($\boldsymbol{Y} = \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T$) as the target sequence. As shown in Fig. 2(a), at frame t, the model takes the hidden state of the previous frame $\boldsymbol{s}_{t-1}$ and the waveform of the current frame $\boldsymbol{x}_t$, makes a prediction $\boldsymbol{x}'_{t+1}$ for the waveform of the next frame, and generates a new hidden state $\boldsymbol{s}_t$. The model is trained to minimize the L_1 loss between the predicted sequence $\boldsymbol{X}' = \boldsymbol{x}'_2, \ldots, \boldsymbol{x}'_T$ and the target sequence $\boldsymbol{Y}$.



(a) Training (b) Inference

Fig. 2. Auto-regressive training flow and inference flow for packet loss concealment.

Inference. The flowchart of the inference stage is shown in Fig. 2(b). In our PLC system, whether each packet is lost or not is known as an input. If a frame $\boldsymbol{x}_t$ is not lost, we copy it directly to the output. Otherwise, we output the predicted signal $\boldsymbol{x}'_t$ at the previous frame. In either case, the outputted frame ($\boldsymbol{x}_t$ or $\boldsymbol{x}'_t$) is fed into the model to update the hidden state for future predictions.

3.2. Model Structures

Fig. 3 shows the structures of the models used in this study. We use an LSTM network without convolutional layers as a baseline system (Fig. 3(a)). The input waveform is fed directly into two LSTM layers, whose output is transformed with a fully connected layer with the tanh activation into the predicted waveform.

The first CRN architecture we explore employs convolutional encoder layers. As shown in Fig. 3(b), an input layer and a stack of convolutional blocks are inserted between the waveform input and the LSTM layers. The input layer uses filters of size 1 to increase the number of channels to 16; the convolutional blocks extract

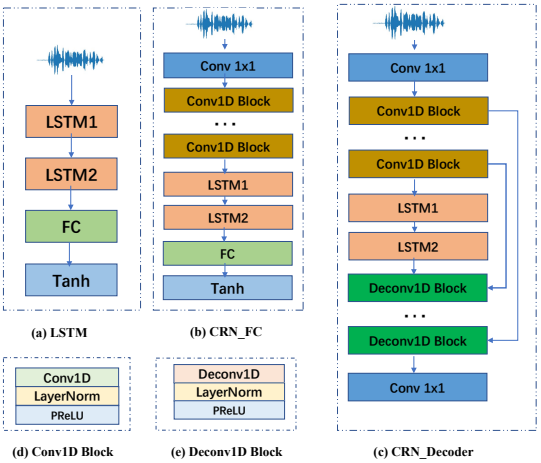


Fig. 3. The model architectures used for packet loss concealment.

high-level features from the input features. Each convolutional block consists of a 1-D convolutional layer, followed by layer normalization [19] and the PReLU activation [20]. We refer to this architecture as *CRN_FC*.

We also investigate an encoder-decoder architecture for the CRN. As shown in Fig. 3(c), this architecture has the same input, encoder, and LSTM layers as CRN_FC, but contains additional decoder layers and an output layer. The decoder consists of deconvolutional blocks, and serves to convert the low resolution features to the target size. Each deconvolutional block consists of a 1-D transposed convolutional layer, followed by layer normalization [19] and the PReLU activation [20]. The CRN also includes skip connections from each encoder layer to its corresponding decoder layer, in order to avoid losing low-level details and to facilitate optimization. Finally, the output layer uses filters of size 1 to generate predicted frames in a single channel. We refer to this architecture as *CRN_Decoder*.

3.3. Lookahead

In our preliminary experiments, we found that the right end of lost frames were often not predicted ideally: the phase did not connect smoothly with the ensuing frame, and the amplitude was often attenuated. This indicates that some right context is essential for reconstructing lost frames. To avoid incurring a large latency, we allow the model to look ahead at one single future frame $\boldsymbol{x}_{t+2}$ when predicting $\boldsymbol{x}'_{t+1}$. This increases the latency by two frames. As shown in Fig. 4, the future frame $\boldsymbol{x}_{t+2}$ is concatenated with the current frame $\boldsymbol{x}_t$ as the input.

During inference, the future frame may not be always available because it may be a lost frame itself. In this case, we need to replace it with an all-zero frame. In order to avoid mismatch between training and inference, we replace the future frame by all zeros with a certain probability p during training as well. We choose this probability to be 40% in this work, because that is the highest packet loss rate we allow in our simulated data.

3.4. Masked Training

During inference, the left context leading up to the frame to be predicted may contain many predicted frames instead of original frames. This is especially pronounced when packet loss occurs in

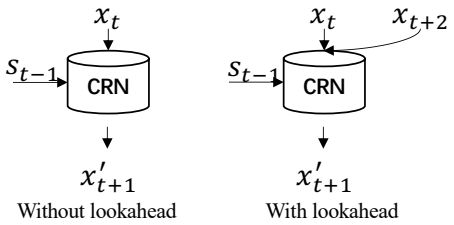


Fig. 4. The lookahead operation.

bursts: the current frame $\boldsymbol{x}_t$ may be a predicted frame itself. This again creates a mismatch between training and inference. To address this mismatch, we also replace original frames $\boldsymbol{x}_t$ with the predicted frame $\boldsymbol{x}'_t$ with a certain probability during training. We choose this probability empirically to be 30%.

4. EXPERIMENTS

4.1. Datasets

Packet Loss Concealment. The training data for PLC was taken from an in-house corpus, which is a de-identified dataset collected using mobile devices through crowd-sourcing from a data supplier for ASR. No personally identifiable information is contained in this dataset. The participants were instructed to pronounce utterances as they would talk to an in-home voice assistant on the topics of calling friends, setting timers, playing music, etc. We selected 500K and 1.5K sentences from the in-house data as training and test sets, respectively. We degraded the test set using our packet loss simulator, picking p_N and p_L randomly from Fig. 1, but only retained degraded sentences having an actual packet loss rate below 40%. We measured the objective metrics (PESQ and STOI) on this test set.

Automatic Speech Recognition. To verify the ASR performance of the proposed method, we conducted experiments on the LibriSpeech corpus [21]. LibriSpeech is an open-source corpus containing 960 hours of speech derived from audiobooks in the LibriVox project. We trained two ASR acoustic models: one using the original 960 hours of training data; the other one in a multi-style fashion using the original 960 hours of data plus 360 hours of distorted data generated with our packet loss simulator. We also selected 1,000 sentences from the "test-clean" set as test data. We processed them with the packet loss simulator in the same way as we processed the in-house data, and measured word error rate (WER) on this test set.

4.2. PLC Model Setup

All models in this work operate directly on raw audio sampled at 16 kHz. A 20 ms sliding window is used to extract frames of speech waveform of 320 samples each. When lookahead is applied, the current frame and the future frame are concatenated to form a 640-sample input to the model.

LSTM. The LSTM baseline model consists of two LSTM layers followed by a fully connected output layer. Each LSTM layer has 1024 memory cells. The output layer reduces the dimensionality from 1024 to 320, which is the length of an audio frame. To constrain the output values within [-1, 1], the tanh activation is applied.

CRN_FC. This CRN consists of 1 input layer and 7 Conv1d blocks. The number of output channels is [16, 16, 32, 64, 128, 128, 256, 256] for each layer, and their filter sizes are [1, 3, 3, 3, 3, 3, 3, 3]. The LSTM and output layers are identical to the LSTM baseline.

CRN_Decoder. The input, encoder and LSTM layers are identical to CRN_FC. The LSTM layers are followed by 6 Deconv1d blocks and one output layer. Their number of output channels is [256, 128, 128, 64, 32, 16, 1] respectively, and their filter sizes are [3, 3, 3, 3, 3, 4, 1].

All the models were trained using the Adam optimizer [22] with an initial learning rate of 0.0002, and a minibatch size of 160 sentences. All sentences were zero-padded to have the same length as the longest sentence within a minibatch.

4.3. ASR Setup

We adopt a chenone-based hybrid acoustic model [14]. It comprises six *latency-controlled bidirectional long short-term memory* (LC-BLSTM) layers, each having 1,000 memory cells for each direction. The model was trained on 80-dimensional log-Mel filterbank features extracted from 25 ms windows with a 10 ms frame shift. It was first trained with the cross-entropy loss for 25 epochs, then further trained with the LF-MMI [23] criterion for 8 epochs. An official unpruned 4-gram language model of Librispeech is used for decoding.

We trained two versions of acoustic models with different data. One was trained using all the 960 hours of LibriSpeech data, and we refer to it as *LF-MMI-Default*. The other one was trained in a multi-style fashion using the 960 hours of LibriSpeech data plus 360 hours of distorted data, and we refer to this model as *LF-MMI-MTR*.

4.4. Evaluation Metrics

Speech enhancement is commonly evaluated with the *perceptual evaluation of speech quality* (PESQ) score [24] and the *short-time objective intelligibility* (STOI) score [25]. Both scores can be automatically computed by comparing the enhanced speech signal with a clean reference signal, and act as objective proxies of the *speech quality* and *intelligibility* perceived by humans. The PESQ score ranges from -0.5 to 4.5, and the STOI score ranges from 0 to 1. We use the wide-band version of the PESQ score, because our signals are sampled at 16 kHz.

Enhanced signals generated by our systems may be consumed either by human listeners or by ASR systems. To evaluate the performance more directly, we conducted a subjective listening test and measured the speech quality using mean opinion scores (MOS). We also measured the intelligibility with word error rates (WER) from our ASR systems.

4.5. Results

Objective evaluation. Table 1 shows the wide-band PESQ score, STOI score, and WERs of the various systems. For PESQ and STOI, we report the overall score, and a breakdown by the packet loss rate.

From the overall PESQ and STOI scores, we can see that all the PLC models improve the speech quality and intelligibility of degraded speech. CRN-based methods outperform the LSTM baseline: CRN_FC slightly improves the PESQ score from 2.03 to 2.08, and the STOI score from 0.8788 to 0.8831. Furthermore, adding lookahead can improve the performance significantly for both CRN structures. For example, lookahead increases the PESQ and STOI scores of CRN_FC remarkably by 0.35% and 1.62% (absolute), respectively. This suggests that the additional right context helps to reconstruct the speech signal.

	#Params	WB-PESQ				STOI				WER(%)	
Packet loss rate	-	0-10%	10%-20%	20%-40%	Overall	0-10%	10%-20%	20%-40%	Overall	LF-MMI-Default	LF-MMI-MTR
Degraded	-	2.09	1.57	1.19	1.53	0.9233	0.8620	0.7252	0.8203	19.99	7.43
LSTM	14.24M	2.79	2.13	1.53	2.03	0.9526	0.9057	0.8134	0.8788	10.32	7.51
CRN_FC	17.30M	2.86	2.19	1.57	2.08	0.9544	0.9092	0.8199	0.8831	9.18	6.91
+look ahead	17.50M	**3.23**	**2.49**	1.93	**2.43**	0.9614	0.9182	0.8476	0.8993	8.02	6.29
+masked training	17.50M	3.16	2.41	**1.96**	2.41	0.9599	0.9124	0.8468	0.8992	8.13	6.36
CRN_Decoder	17.34M	2.79	2.15	1.54	2.05	0.9540	0.9012	0.8212	0.8840	8.49	6.54
+look ahead	17.93M	3.14	2.44	1.88	2.37	**0.9624**	**0.9220**	0.8535	0.9033	7.26	**6.19**
+masked training	17.93M	3.12	2.41	1.90	2.39	0.9618	0.9173	**0.8547**	**0.9044**	**7.13**	6.26

Table 1. The PESQ scores, STOI scores and WERs of the various networks. The best value in each column is boldfaced.

From the breakdown by packet loss rate, we see that masked training is only helpful for higher loss rates (20%-40%). This is in agreement with the motivation of reducing the training-inference mismatch when the packet loss rate is high. Comparing the CRN structures with and without a decoder, we can see that decoder layers improves intelligibility (STOI), but not speech quality (PESQ).

The observations above are corroborated by the ASR results. CRN_FC outperforms the LSTM baseline, and the decoder brings extra WER reduction thanks to the improved intelligibility. In the LF-MMI-Default setting, the lowest WER is achieved by the CRN_Decoder system equipped with both lookahead and masked training. LF-MMI-MTR gives us an even lower WER compared with LF-MMI-Default. This indicates that, even with packet loss concealment in place, multi-style training and data augmentation by simulating packet loss can still boost the ASR performance.

Subjective evaluation. We designed the MOS evaluation as follows to evaluate the quality of the enhanced speech. We selected a total of 20 sentences (15 from the in-house test set and 5 from the LibriSpeech test-clean set). For each sentence, we prepared five conditions: original non-degraded speech, degraded speech generated by the packet loss simulator, and enhanced speech produced by the LSTM baseline and the two CRN systems. We repeated each sentence 3 times, resulting in a total of $20 \times 5 \times 3 = 300$ test sentences. We recruited 15 coworkers as listeners, and assigned 20 sentences to each listener. The listeners were asked to rate each sentence on a scale of 1 to 5.

Fig. 5 shows the mean and the standard deviation of the MOS scores for each condition. We can clearly observe that CRN-based systems generate higher-quality speech than the LSTM baseline, but the difference between the two CRN structures is indiscernible. An ANOVA analysis confirms that all differences in Fig. 5 are significant, except that between the two CRN structures.

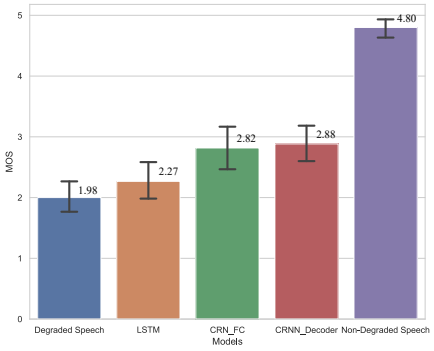


Fig. 5. MOS scores of the speech quality.

Visualization. Figure 6 shows an example of the waveforms of original speech, degraded speech, and the signal recovered by the various systems. In the degraded speech, two consecutive frames (total 40 ms) are dropped. The reconstruction results illustrate the advantage of lookahead. Without lookahead, the amplitude of the recovered speech tends to taper off toward the end of the dropped segment, but lookahead corrects this behavior thanks to the right context.

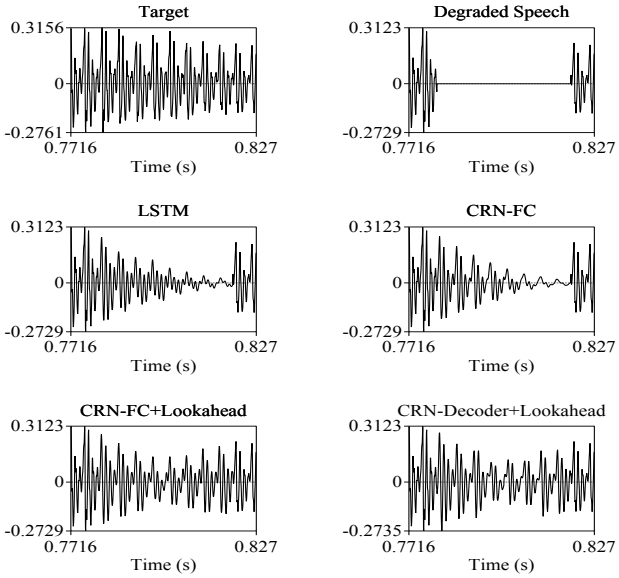


Fig. 6. Waveforms of degraded speech and enhanced speech by various approaches.

5. CONCLUSION

In this paper, we have proposed to use CRN-based architectures for online packet loss concealment in the time domain. Compared with a baseline system using only LSTM layers, a CRN system with encoder layers achieves better performance in both objective and subjective evaluations. Adding decoder layers to the CRN can further improve the intelligibility of the reconstructed speech. By providing some right context for prediction, lookahead offers a significant improvement in all evaluation metrics. When the packet loss rate is high, masked training can reduce the mismatch between training and inference. Finally, we have found that multi-style training and data augmentation are still helpful for speech recognition even with packet loss concealment in place.

6. REFERENCES

[1] Akira Takahashi, Hideaki Yoshino, and Nobuhiko Kitawaki, "Perceptual qos assessment technologies for voip," *IEEE Communications Magazine*, vol. 42, no. 7, pp. 28–34, 2004.

[2] Colin Perkins, Orion Hodson, and Vicky Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE network*, vol. 12, no. 5, pp. 40–48, 1998.

[3] Bengt J Borgström, Per H Borgström, and Abeer Alwan, "Efficient hmm-based estimation of missing features, with applications to packet loss concealment," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[4] Björn W Schuller, *Intelligent audio analysis*, Springer, 2013.

[5] Bong-Ki Lee and Joon-Hyuk Chang, "Packet loss concealment based on deep neural networks for digital speech transmission," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 2, pp. 378–387, 2015.

[6] Reza Lotfidereshgi and Philippe Gournay, "Speech prediction using an adaptive recurrent neural network with application to packet loss concealment," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5394–5398.

[7] Mostafa M Mohamed and Björn W Schuller, "Concealnet: An end-to-end neural network for packet loss concealment in deep speech emotion recognition," *arXiv preprint arXiv:2005.07777*, 2020.

[8] Mikolaj Kegler, Pierre Beckmann, and Milos Cernak, "Deep speech inpainting of time-frequency masks," *arXiv preprint arXiv:1910.09058*, 2019.

[9] Andrés Marafioti, Nathanaël Perraudin, Nicki Holighaus, and Piotr Majdak, "A context encoder for audio inpainting," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2362–2372, 2019.

[10] Ya-Liang Chang, Kuan-Ying Lee, Po-Yu Wu, Hung-yi Lee, and Winston Hsu, "Deep long audio inpainting," *arXiv preprint arXiv:1911.06476*, 2019.

[11] Yupeng Shi, Nengheng Zheng, Yuyong Kang, and Weicong Rong, "Speech loss compensation by generative adversarial networks," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 347–351.

[12] Hang Zhou, Ziwei Liu, Xudong Xu, Ping Luo, and Xiaogang Wang, "Vision-infused deep audio inpainting," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 283–292.

[13] Andrés Marafioti, Piotr Majdak, Nicki Holighaus, and Nathanaël Perraudin, "Gacela–a generative adversarial context encoder for long audio inpainting," *arXiv preprint arXiv:2005.05032*, 2020.

[14] Duc Le, Xiaohui Zhang, Weiyi Zheng, Christian Fügen, Geoffrey Zweig, and Michael L Seltzer, "From senones to chenones: Tied context-dependent graphemes for hybrid speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 457–464.

[15] Seyed Kamran Pedram, Saeed Vaseghi, and Bahareh Langari, "Audio packet loss concealment using spectral motion," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 6707–6710.

[16] Ke Tan and DeLiang Wang, "A convolutional recurrent neural network for real-time speech enhancement.," in *Interspeech*, 2018, pp. 3229–3233.

[17] Juin-Hwey Chen, "Packet loss concealment based on extrapolation of speech waveform," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 4129–4132.

[18] Carlos Alexandre Gouvea Da Silva and Carlos Marcelo Pedroso, "Mac-layer packet loss models for wi-fi networks: A survey," *IEEE Access*, vol. 7, pp. 180512–180531, 2019.

[19] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[21] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[22] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi.," in *Interspeech*, 2016, pp. 2751–2755.

[24] ITU-T Recommendation, "Perceptual evaluation of speech quality (pesq): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," *Rec. ITU-T P. 862*, 2001.

[25] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.