# TUNET: A BLOCK-ONLINE BANDWIDTH EXTENSION MODEL BASED ON TRANSFORMERS AND SELF-SUPERVISED PRETRAINING

*Viet-Anh Nguyen[1], Anh H. T. Nguyen[1], and Andy W. H. Khong[2]*

[1]NextG, FPT Software, Vietnam
[2]Nanyang Technological University, Singapore
{anhnv79, anhnht3}@fsoft.com.vn, andykhong@ntu.edu.sg

## ABSTRACT

We introduce a block-online variant of the temporal feature-wise linear modulation (TFiLM) model to achieve bandwidth extension. The proposed architecture simplifies the UNet backbone of the TFiLM to reduce inference time and employs an efficient transformer at the bottleneck to alleviate performance degradation. We also utilize self-supervised pretraining and data augmentation to enhance the quality of bandwidth extended signals and reduce the sensitivity with respect to downsampling methods. Experiment results on the VCTK dataset show that the proposed method outperforms several recent baselines in both intrusive and non-intrusive metrics. Pretraining and filter augmentation also help stabilize and enhance the overall performance.

***Index Terms***— Bandwidth extension, transformer, self-supervised pretraining, speech enhancement

## 1. INTRODUCTION

Bandwidth extension (BWE), or audio super-resolution, enhances speech by generating a wideband (WB) signal from a narrowband (NB) signal. The NB signal is usually sampled below 8 kHz resulting in low auditory quality. Such sampling rate is widely used in G.711, G.729, and AMR audio codecs due to its efficient streaming. Including a BWE module at the receiver side will therefore improve audio fidelity.

Compared to conventional BWE approaches such as [1, 2, 3], recent end-to-end deep neural networks generate WB signals directly from NB signals without the need for feature engineering. For instance, inspired by the well-known UNet architecture [4] in image processing, AudioUNet [5] is a wave-to-wave BWE model that has outperformed traditional methods. In [6], the limitation of convolution on long-range dependency modeling in UNet is addressed by introducing the TFiLM layer that modulates blocks of convolution's feature maps with information learned by recurrent layers. Generative models such as the NU-Wave [7] neural vocoder relies on conditional diffusion models with modified noise level embedding and local conditioner. On the other hand, WSRGlow [8] models the distribution of the output conditioned on the input using normalizing flow.

While convolutional neural network architectures exhibit promising results for end-to-end BWE training, their effectiveness on long-range dependency modeling is still limited by receptive fields of convolution [9]. Stacking more convolution layers would help expand the receptive field at the expense of increased computation. In addition, training end-to-end BWE models requires high-rate target signals, making valuable low-rate data collected from telephony 8-kHz infrastructure unusable. It has also been observed that BWE models are susceptible to low-pass filtering [6, 10], generating severe distortion at the transition band of the anti-aliasing filter. This problem can be mitigated by data augmentation [10].

We propose a Transformer-aided UNet (TUNet)[1] by employing a low-complexity transformer encoder on the bottleneck of a lightweight UNet. Here, the Transformer assists such a small UNet with its captured global dependency while the UNet effectively downsamples waveform input with strided convolution to reduce computation that the Transformer must perform. In addition, inspired by masked language modeling in natural language processing [11], we propose *masked speech modeling* — a self-supervised representation learning scheme that reconstructs original signals from masked signals. The advantage of this pretraining is that it requires only low-rate data to make full use of telephony databases, allowing the model to learn the underlying statistics of the low-band speech and generalize to downstream tasks [12]. Finally, similar to [10], we make our model robust to downsampling methods by generating training data with different parameter sets of the Chebyshev Type I filter.

## 2. REVIEW OF TFILM-UNET AND PROPOSED TUNET ALGORITHM

### 2.1. TFiLM-UNet baseline

TFiLM-UNet is an offline UNet-based audio super-resolution model [6]. To assist convolution layers in capturing long-range information, Temporal Feature-Wise Linear Modulation (TFiLM) has been proposed. This layer acts as a normalization layer that combines maxpooling and long short-

---

[1]Source code and audio samples: https://github.com/NXTProduct/TUNet
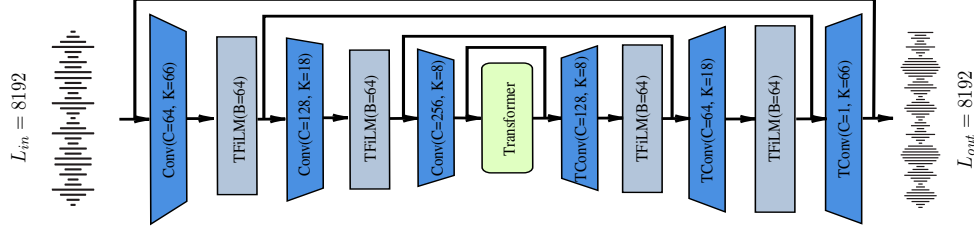
**Fig. 1**. TUNet architecture for speech enhancement. The encoder downsamples waveform input while the decoder does the reverse. A Transformer block is placed in the middle to model the attention of the bottleneck.

term memory (LSTM). While maxpooling reduces temporal dimension into $B$ blocks, LSTMs refine convolution's feature maps by captured long-range dependency.

In the TFiLM-UNet model, the encoder contains four downsampling (D) blocks, each comprising a convolution layer, maxpooling layer, ReLU activation, and TFiLM layer, consecutively. In the decoder, upsampling (U) blocks follow sequential operations: convolution, dropout, ReLU, DimShuffle, and TFiLM, in which the DimShuffle layer doubles time dimension by manipulating the feature shape. Stacking and additive skip connections are applied between D/U blocks and input/output, respectively.

## 2.2. Lightweight UNet with Transformer

With reference to Fig. 1, our proposed model follows the same waveform-to-waveform UNet to that of TFiLM. As opposed to TFiLM-UNet, the proposed model is significantly smaller due to the use of fewer convolution filters and higher dimensional reduction rates. Precisely, the encoder consists of three strided 1D convolution layers, each having $C$ filters of kernel size $K$. Stride $S$ of all these layers is set at 4, resulting in the time dimension of the bottleneck being 64 times shorter than the length $L_{in}$ of the input. Consequently, the bottleneck features can be processed efficiently in the follow-up Performers [13] blocks. We employ Performers since its self-attention mechanism has linear time complexity compared to the quadratic complexity of the conventional attention [14]. On the decoder side, three transposed 1D convolution layers commensurating the downsampling rates of the encoder are used to generate output signals that have the length $L_{out} = L_{in}$. We use Tanh activation for the last transposed convolution and LeakyReLU [15] for the rest. TFiLM layers are applied after convolution layers except for the last encoder layer that is replaced by the Performer blocks. To smooth the loss landscape [16], skip connections that connect TFiLM encoders to the corresponding decoders are employed.

Compared to the TFiLM-UNet, our model has four key differences: i) Our encoder and decoder require one fewer layer and four times fewer filters than the baseline; ii) Each encoding layer reduces time dimension by four times instead of two to assist quick input compression; iii) We replace
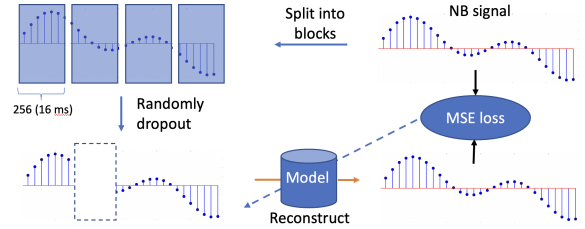


**Fig. 2**. Masked speech modeling pretraining pipeline.

downsampling and upsampling blocks in TFiLM with strided convolution and transposed convolution layers, respectively; and iv) compared to the stacking skip-connection in TFiLM, we employ additive skip connection which further reduces the number of parameters in the decoder. These modifications ensure that our model is significantly lighter than the baseline while preserving learning capability.

## 2.3. Masked speech modeling

We propose masked speech modeling (MSM) pretraining as illustrated in Fig. 2. Since audio signals possess fine granular characteristics, instead of masking the sequence at sample level, we mask 20% of 256-sample blocks to create the masked input. The model will optimize the mean squared error between the output and the masked input. Compared to the masked reconstruction pretraining in [17], both encoder and decoder are pretrained in our proposed approach.

## 2.4. Improving robustness to downsampling methods by augmentation

The performance of BWE models is highly sensitive to different anti-aliasing filters when downsampling methods in testing differ from training [5, 6, 10]. Similar to [10], to improve the robustness of our model, we generate the low-rate signals by downsampling the high-rate speech dataset with random anti-aliasing filters. More specifically, we adopt the Chebyshev Type I anti-aliasing filter and randomize its ripple and order parameters. This helps in creating variations in the transition band of the anti-aliasing filter.

## 2.5. Learning objectives

Since the mean squared error (MSE) loss may not guarantee the good perceptual quality [18], we combine MSE loss with multi-resolution short-time Fourier transform (STFT) loss [19] in the Mel scale. Given a reconstructed signal $\hat{y}$ and a target signal $y$, the training loss is given by

$$\ell(\hat{y}, y) = \ell_{\mathrm{MR}}(\hat{y}, y) + \alpha \, \mathrm{MSE}(\hat{y}, y), \tag{1}$$

where $\alpha$ denotes the weight of the MSE loss, and $\ell_{\mathrm{MR}}$ is the multi-resolution STFT loss (MR loss).

## 3. EXPERIMENTS

### 3.1. Setup

We focus on extending 4-kHz bandwidth (8 kHz sampling rate) to 8-kHz bandwidth (16 kHz sampling rate). Training data was segmented into smaller chunks with a window size of 8192 and 50% overlapping. We used the VCTK Corpus [20] for training and testing. This dataset includes 109 English speakers, in which recordings of the first 100 speakers were for training and the remaining for testing.

Besides VCTK, we further used the VIVOS dataset [21] to verify the effectiveness of our pretraining approach. This dataset consists of 15-hour speech recordings from 65 Vietnamese speakers, recorded in a quiet environment with high-quality microphones. We followed the dataset's default split: 46 speakers for training, 19 speakers for testing.

To evaluate the quality of the generated audio, we used four metrics: log-spectral distance (LSD), high-frequency log-spectral distance (LSD-HF), scale-invariant source-to-distortion ratio (SI-SDR) [22], and DNSMOS based on P.808 criterion [23]. LSD-HF computes LSD specifically on high-frequency bands, i.e., 4kHz - 8 kHz. As opposed to LSD, LSD-HF focuses only on the regeneration of the high-band spectrum and ignores artifacts or distortions in the low-band spectrum. A lower LSD/LSD-HF score implies a more similar spectral to the target, while a higher SI-SDR score indicates better performance. On the other hand, DNSMOS employs a deep learning model to predict the mean-opinion-score (MOS) of human raters. It has been shown to have excellent correlation to MOS [23]. A higher value of DNSMOS indicates better speech quality.

The $C$, $K$, $S$, and $B$ hyperparameters of our model are described in Fig. 1. The Performers[2] block has three hidden layers, two attention heads for each layer, and each head's dimension is 32; local window length is equivalent to bottleneck length divided by 8. Hyperparameters of MR loss such as resolutions were set with default values of the *auraloss*[3] v2.0.1 library. The MSE weight was set to $\alpha = 10000$. We trained our models for 150 epochs using the Adam optimizer, $3 \times 10^{-4}$ learning rate with 800 samples in each batch. For
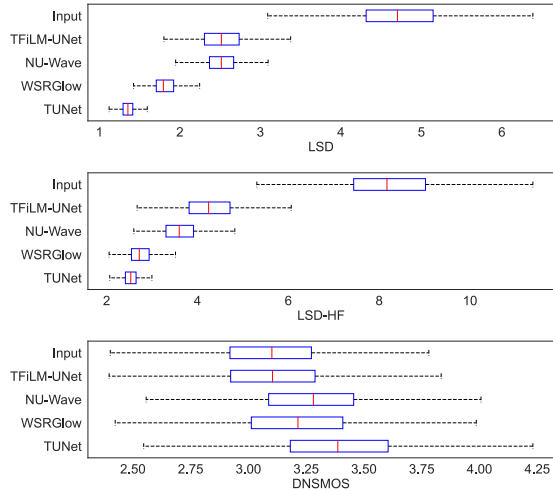
**Fig. 3**. Metric scores of the baselines and our model. Lower LSD/ LSD-HF is better, and higher DNSMOS is better.

**Table 1**. Model size and inference time on a single core CPU.

| System | #Params | Inference time (ms) |
|---|---|---|
| WSRGlow | 229M | 3146 |
| NU-Wave | 3M | 2431 (8 iters) |
| TFiLM-UNet | 68.2M | 1335 |
| TUNet | **2.9M** | **22.6** |

the baseline TFiLM-UNet model, while official implementation is available, we adopted an unofficial implementation[4] which reportedly produces slightly better results and much faster training.

### 3.2. Performance comparison with baselines

We compared our model's performance and inference speed with TFiLM-UNet and two recent generative models, NU-Wave [7] and WSRGlow [8]. The above baselines were trained on the VCTK dataset with low-rate data generated from 16-kHz data using only one 8th order Chebyshev Type I low-pass filter. In this experiment, MSM pretraining was excluded from our method.

Results in Fig. 3 show that our TUNet model achieved significantly higher performance than that of all the baselines. Compared to our TUNet, the WSRGlow model achieves tight LSD-HF scores but relatively worse in LSD, indicating that our model better preserves low frequencies. Despite the worst LSD score, the NU-Wave model achieves a considerable improvement in DNSMOS only after our proposed model.

In terms of single-threaded inference time, we measured it on the AMD EPYC 7742 using ONNX inference engine. Our proposed model was significantly faster and more lightweight than the others. In Table 1, TUNet requires only 22.63 ms to

**Table 2**. Effectiveness of components on our model.

| Model | LSD | LSD-HF | SI-SDR |
|---|---|---|---|
| No Transformer | 1.45 | 2.64 | 21.61 |
| LSTMs bottleneck | 1.44 | 2.70 | 21.76 |
| No TFiLM | 1.44 | 2.69 | 21.89 |
| TUNet | **1.36** | **2.54** | **21.91** |

**Table 3**. BWE results on VCTK and VIVOS datasets when employing MSM pretraining.

| | Model | LSD | LSD-HF | LSD-LF | SI-SDR |
|---|---|---|---|---|---|
| VCTK | input | 4.75 | 8.27 | 1.23 | 20.32 |
| | w/o MSM | 1.36 | 2.54 | 0.18 | 21.69 |
| | MSM on VCTK | **1.28** | **2.45** | **0.11** | **22.08** |
| VIVOS | input | 5.59 | 9.79 | 1.39 | 21.75 |
| | w/o MSM | 1.36 | 2.49 | 0.23 | 25.08 |
| | MSM on VCTK | **1.29** | **2.42** | **0.16** | **26.15** |



**Fig. 4**. LSD scores of our models trained with a single and multiple anti-aliasing filter(s) on the VIVOS test set.

execute a single 512 ms audio frame while WSRGlow, NU-Wave (the default eight inference steps) and TFiLM-UNet took approximately 139, 107, and 59 times longer, respectively. Assuming each audio chunk being 87.5% overlapped, this amounts to 64 ms for a new block to arrive with a chunk size of 8192 and a sampling rate of 16 kHz. Since our inference time is shorter than 64 ms, this implies that the proposed method is more suited for semi-real-time applications compared to the baselines.

### 3.3. Ablation studies

To study the effects of its two main components, TFiLM layers and Performers blocks, we created three variations from TUNet: *'No Transformer'* — TUNet without Performers blocks on the bottleneck, *'LSTMs bottleneck'* — TUNet with the Transformer bottleneck replaced by a 3-layer, 256-unit (same as the Transformer) LSTM network, and *'No TFiLM'* — TUNet without TFiLM layers.
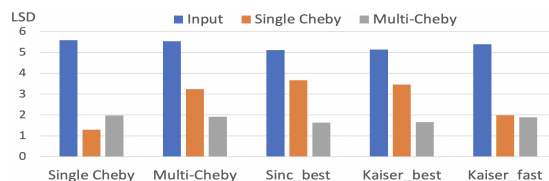
In Table 2, both Performer and TFiLM layers play significant roles in the proposed model since excluding these two components led to noticeably decreased scores on all metrics. The 'No Transformer' model, which excluded the Transformer from the bottleneck, performed worst in terms of LSD and SI-SDR, and the performance was only improved by a small margin even with LSTMs aided. The removal of TFiLM also led to a significant degradation but relatively less than the removal of the Transformer.

To determine the effectiveness of MSM pretraining, we pretrained TUNet on VCTK low-rate data with the pipeline described in Section 2.3. After obtaining a pretrained model, we subsequently trained it with the BWE task on the VCTK dataset. In this experiment, we used only one anti-aliasing filter in Section 3.2 to generate training data. To assess the generalization ability of MSM, we include an additional scenario where the pretraining dataset is VCTK, but the BWE training and test set are of a different language. We adopted one more metric — low-frequency log-spectral distance (LSD-LF) to measure the approximation error in the low band (0-4 kHz) caused by MSM pretraining.

Results in Table 3 show that models pretrained with MSM achieve significant improvements on spectral-based metrics while SI-SDR figures were modest. The scores indicate that the pretraining scheme not only enhanced high frequencies but also helped preserve low frequencies. Furthermore, the performance gain on the VIVOS was consistent with that of the VCTK. This implies that the BWE model adapted very well to the VIVOS dataset even though it was pretrained on a different language.

We next assessed sensitiveness to anti-aliasing filters of our models trained with and without filter augmentation. The first model, 'Single Cheby' is the best model obtained from the above experiments, which was trained with a single Chebyshev Type I anti-aliasing filter. The other 'Multi-Cheby' was trained with a set of random filters as described in Section 2.4. Both models employed the same MSM pretraining above. The BWE dataset used for this experiment was the VIVOS dataset. The test set was downsampled using all resampling methods available in the *resampy*[5] library. However, due to space constraints, we will only report the results on test sets generated by single/multiple Chebyshev filters (same as training of 'Single Cheby' and 'Multi-Cheby', respectively), Kaiser ('best' and 'fast' variations) filters, and the sinc downsampling.

As shown in Fig. 4, the 'Single Cheby' model achieved the best score when evaluated with the same filter. Although this model performed well on several downsampling methods such as 'kaiser_fast', its performance significantly degraded on test sets processed by the other downsampling methods such as the sinc algorithm. On the other hand, the 'Multi-Cheby' showed a stable performance across all the methods.

## 4. CONCLUSIONS

We have proposed a Transformer-aided UNet for bandwidth extension. Despite remarkable performance scores, our model remains lightweight and achieves fast processing. By leveraging only narrowband audio data for pretraining, we have achieved an overall improvement in performance. With multiple anti-aliasing filters applied, the model achieves robustness to different low-pass filters, an essential characteristic for real-world applications.

---

[5]https://github.com/bmcfee/resampy

# 5. REFERENCES

[1] Y. Qian and P. Kabal, "Wideband speech recovery from narrowband speech using classified codebook mapping," in *Proc. Australian Int. Conf. Speech Sci., Technol. (Melbourne)*, 2002.

[2] A. H. Nour-Eldin and P. Kabal, "Mel-frequency cepstral coefficient-based bandwidth extension of narrowband speech," in *Proc. Interspeech*, 2008.

[3] P. Jax and P. Vary, "On artificial bandwidth extension of telephone speech," *Signal Processing*, vol. 83, 2003.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput. Assist. Interv.*, 2015.

[5] V. Kuleshov, S. Z. Enam, and S. Ermon, "Audio super resolution using neural networks," in *Int. Conf. Learn. Representations, Workshop Track*, 2017.

[6] S. Birnbaum, V. Kuleshov, S. Z. Enam, P. W. Koh, and S. Ermon, "Temporal FiLM: Capturing Long-Range Sequence Dependencies with Feature-Wise Modulations," in *Proc. Neural Inf. Process. Syst.*, 2019.

[7] J. Lee and S. Han, "NU-Wave: A Diffusion Probabilistic Model for Neural Audio Upsampling," in *Proc. Interspeech*, 2021.

[8] K. Zhang, Y. Ren, C. Xu, and Z. Zhao, "WSRGlow: A Glow-based waveform generative model for audio super-resolution," in *Proc. Interspeech*, 2021.

[9] D. Linsley, J. Kim, V. Veerabadran, C. Windolf, and T. Serre, "Learning long-range spatial dependencies with horizontal gated recurrent units," in *Proc. Neural Inf. Process. Syst.*, 2018, p. 152–164.

[10] S. Sulun and M. E. P. Davies, "On filter generalization for music bandwidth extension using deep neural networks," *IEEE J. of Sel. Topics in Signal Process.*, vol. 15, no. 1, Jan 2021.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics*, 2019.

[12] A. Baevski, M. Auli, and A. rahman Mohamed, "Effectiveness of self-supervised pre-training for speech recognition," *ArXiv*, vol. abs/1911.03912, 2019.

[13] K. M. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Q. Davis, A. Mohiuddin, L. Kaiser, D. B. Belanger, L. J. Colwell, and A. Weller, "Rethinking attention with Performers," in *Proc. Int. Conf. Learn. Representations*, 2021.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Neural Inf. Process. Syst.*, 2017.

[15] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learn. for Audio, Speech and Lang. Process.*, 2013.

[16] L. Wang, B. Shen, N. Zhao, and Z. Zhang, "Is the skip connection provable to reform the neural network loss landscape?," in *Proc. Int. Joint Conf. Artif. Intell.*, 2020.

[17] W. Wang, Q. Tang, and K. Livescu, "Unsupervised pre-training of bidirectional speech encoders via masked reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020.

[18] J. Martín-Doñas, A. Gomez, J. Gonzalez Lopez, and A. Peinado, "A deep learning loss function based on the perceptual evaluation of the speech quality," *IEEE Signal Process. Lett.*, vol. PP, pp. 1–1, 09 2018.

[19] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel Wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020.

[20] J. Yamagishi, C. Veaux, and K. MacDonald, "CSTR VCTK Corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92)," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2019.

[21] H. T. Luong and H. Q. Vu, "A non-expert Kaldi recipe for Vietnamese speech recognition system," in *WLSI/OIAF4HLT@COLING*, 2016.

[22] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "SDR – Half-baked or well done?," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019.

[23] C. K. Reddy, V. Gopal, and R. Cutler, "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," *arXiv e-prints*, pp. arXiv–2010, 2020.