ADVANCES IN CYBER-PHYSICAL SYSTEMS

Vol. 10, No. 2, 2025

EDGE-READY SPEECH SEPARATION WITH SUDO-TASNET

Andrii Tsemko[1,2], Ivan Karbovnyk[1]

[1]Ivan Franko National University of Lviv, 50, Drahomanov Str, Lviv, 79005, Ukraine,
[2]Infineon Technologies, 20, Luhanska Str, Lviv, 79000, Ukraine.
Authors' e-mails: *andrii.tsemko@lnu.edu.ua, ivan.karbovnyk@lnu.edu.ua*

https://doi.org/10.23939/acps2025.02.202

Submitted on 11.09.2025

***Abstract*: This article presents a hybrid speech separation model designed for efficient deployment on edge devices, focusing on optimizing both performance and computational resources. This study proposes a novel hybrid architecture that combines the strengths of Conv-TasNet and SuDoRM-RF models, leveraging their fully-convolutional structures to achieve efficient separation with minimal resource usage. The proposed model has obtained a separation performance of 10.59 db in SI-SDRi for clean Libri2Mix dataset for only 1.17 M parameters with only 0.92 GMACs/s.**

***Index terms*: Speech Separation, Audio Processing, Neural Networks**

I. INTRODUCTION

Today's state-of-the-art headphones offer real-time synchronous translation of the speaker and often deliver impressive results that help erase the language barrier. The speech enhancement component of this pipeline improves automatic speech recognition (ASR) performance, making translation more accurate; however, reliable operation also requires speech separation, which should primarily extract the main speaker and feed that signal to the ASR system. Speech separation must handle situations with simultaneous speech from your conversation partner and nearby bystanders, as well as other overlapping voices. A speech separation module is designed to separate multiple talkers into distinct channels, which can then be analyzed by ASR systems to select the correct speaker for downstream translation.

Recent investigations in neural networks for speech separation demonstrate a remarkable level of performance that would have been unimaginable a decade ago in single-channel experimental setups. Usually, speech separation models can be trained for both clean and noisy conditions. This is typically enabled by datasets such as LibriMix [1] and WSJ0-2mix [2], which contain two-speaker mixtures augmented with an additional noise source at SNRs ranging from -10 to 10 db. Even under such noisy conditions–closely aligned with the cocktail party problem–these models separate two speakers (often with different energy levels) while suppressing background noise, making them suitable not only for extending existing speech enhancement pipelines but also for replacing the enhancement stage when separation is required. More complex, real-world scenarios also include speech reverberation. In this setting, models are trained to perform dereverberation, denoising and separation by processing the mixture with a neural network and targeting fully clean speech. One state-of-the-art dataset for such training is the WHAMR! dataset [3].

While end-to-end speech separation systems already exist – designed to replace the entire speech enhancement pipeline by training n datasets such as WHAMR! – it is still recommended to maintain a pipeline-based design, as such systems generally demonstrate higher performance with ASR and make it possible to keep models relatively small and fast in terms of parameter count and MAC operations [4]. In our investigation, we focus on the pure speech separation task without speech enhancement.

Different attention-based model architectures have demonstrated breakthrough performance in challenging two-speaker conditions with noise. Such studies typically fall within the domain of speech enhancement, which is an important component of automated speech recognition (ASR) systems. Although state-of-the-art models achieve high levels of separation and denoising, they often rely on complex architectures. The main drawback of these models is their size and the number of MAC operations, which makes them too heavy to run on edge devices. For example, MossFormer2 [5] and SFSRNet [6] both deliver a similar level of separation of about 24 dB SI-SDR [7] on the WSJ0-2mix dataset, but require approximately 55 million parameters (≈220 MB). Other models, such as TF-GridNet [8] and SPMamba [9], use only 14 million and 6 million parameters, respectively, yet require 445 and 238 GMAC/s, making them not real-time friendly for edge devices. Model sizes and MAC/s demands in state-of-the-art systems are currently the main reason why speech separation models are not edge-friendly.

Our investigation focuses on small or tiny speech separation models with approximately 1 million parameters and about 1 GMAC/s of compute, which should deliver sufficient performance to be useful and deployable on edge devices such as microcontrollers.

II. LITERATURE REVIEW AND PROBLEM STATEMENT

Speech separation, where several talkers speak simultaneously, is distinct from automatic speech recognition (ASR) systems, which usually rely on a speech enhancement pipeline aimed at improving intelligibility

by removing non-speech components from the audio signal. However, speech separation differs from conventional denoising tasks such as noise suppression, since it must separate multiple instances of the same speech class, even when talkers have highly similar acoustic characteristics. Although multi-channel (multi-microphone) setups are also common in ASR systems, our investigation focuses solely on the single-channel case. While multi-channel setups enable algorithmic approaches such as beamforming for speech separation, the single-microphone problem remained unsolved for a long time, with major progress achieved only through machine learning. Over the past decade, neural network architectures for speech separation have rapidly advanced, delivering high-quality results that were previously unimaginable.

However, most models that achieve high-quality separation and top performance require large memory capacity, substantial RAM, and a high number of operations, which makes them impractical for real-time systems on edge devices. Our investigation focuses on delivering an optimal model with appropriate performance while keeping resource usage and latency low, making it edge-friendly.

The first breakthrough model that demonstrated an interesting approach was Conv-TasNet [10], which has been widely used as a baseline for the speech separation task. This model popularized an encoder–decoder-based architecture for speech separation that uses Conv1D and TransposedConv1D layers for feature extraction, converting the time-domain signal into STFT-like feature matrices and then back to the time domain after applying the system's estimated masks. It uses a relatively small number of parameter, bout 5.1 million, with approximately 7.19 GMAC/s, achieving a separation performance of 15.30 dB SI-SDR. The model is fully convolutional and supports causal convolution layers, enabling online operation and making it feasible to run on microcontrollers with sufficient available memory.

Another fully convolutional model we investigate is the SuDoRM-RF [11], which proposes a more optimal architecture and even improves separation performance to 17 dB SI-SDR on WSJ0-2mix. This model uses the same encoder–decoder topology with U-Net-like separation blocks in the separator, instead of the TCN dilated-convolution blocks used by Conv-TasNet, making the model smaller at 2.7M parameters with only 3.85 GMAC/s. We prepare a comparative table for different separation models with their architectural specifications, parameter counts, and MAC operations.

We use information from the article Advances in Speech Separation [12], and extend their analysis with additional details about the number of operations per model, because we focus on relatively fast and small models that can be easily deployed on edge devices with limited resources and implemented in a real-time speech enhancement pipeline. SuDoRM-RF and Conv-TasNet are the only models presented in this article, because the latest models mostly focus on transformer-based and/or attention-based architectures.

For example, the SPMamba model demonstrates the highest performance of 22.50 dB SI-SDRi among models with fewer than 10M parameters and uses only 1 million parameters more than Conv-TasNet; however, it requires 238 GMACs, which is almost 40 times more than Conv-TasNet and 61 times more than SuDoRM-RF. This makes the SPMamba model impractical for edge devices and real-time systems.

As a recurrent-based model, we present DPRNN [13] with 2.9M parameters and 42.2 GMACs, achieving 18.30 dB SI-SDR. While this model is similar to SuDoRM-RF in terms of parameter count, it requires 21 times more operations that cannot be parallelized due to its recurrent layers. For an attention-based model, we chose TFPSNet [14], which is the same size as SuDoRM-RF and requires 29.6 GMAC/s, achieving 21.10 dB SI-SDRi.

The smallest state-of-the-art model, TIGER [15], has only 0.8 million parameters and a compute cost of 7.65 MACs/s. This model uses multi-scale selective attention layers and a recursive path that makes it possible to reuse the same parameters multiple times, thereby improving separation performance and achieving 18 dB SI-SDRi on the Libri2Mix dataset. Although this model is the smallest, it still requires almost twice the operations, and its recursive blocks complicate parallelization on the NPU cores of edge devices.

We aim to demonstrate the efficiency of fully convolutional models for speech separation and present a hybrid architecture that combines Conv-TasNet and SuDoRM-RF, which should deliver efficient separation with an optimal model size of slightly more than 1 million parameters. In our investigation, we focus exclusively on fully convolutional models because they require fewer operations and can be effectively accelerated by the NPU cores of edge devices.

III. SCOPE OF WORK AND OBJECTIVES

In this work we propose an efficient hybrid architecture neural network model for speech separation task focusing on efficient deployable for edge devices model. Investigated models are focused to be less then 1.5 million parameters and less than 1 GMACs/s.

The primary objective of this work is to design a hybrid speech separation model that achieves state-of0the-art separation performance with significantly reduced computational calculations compared to purely deep learning-based approaches.

IV. HYBRID MODEL CONV-TASNET AND SUDORM-RF

The core structure inherits the Conv-TasNet architecture, which comprises three processing stages: Encoding, Separation, and Decoding. This model is designed to process a single-channel, time-domain audio signal.

Table 1

Performance, parameters and MAC/s comparison of speech separation on the WSJ0-2Mix and Libri2Mix datasets

Model	Encoder/Decoder type	Separator type	Params (M)	GMAC/s	WSJ0-2Mix SI-SDRi (db)	Libri2Mix SI-SDRi(db)	Year
SuDoRM-RF	Convolution	CNN-based	2.7	3.85	17.0	13.5	2020
DPTNet	Convolution	Mixture-based	2.7	6.00	20.2	16.7	2022
Conv-TasNet	Convolution	CNN-based	5.1	7.19	15.3	12.2	2019
TIGER	STFT	Mixture-based	0.8	7.65	N\A	18.0	2025
S4M	Convolution	Mixture-based	3.6	8.00	20.5	16.9	2023
TDANet	Convolution	Mixture-based	2.3	9.19	18.6	17.4	2023
TFPSNet	STFT	Attention-based	2.7	29.60	21.1	19.7	2022
DPRNN	Convolution	RNN-based	2.9	42.20	18.8	14.1	2020
SepFormer	Convolution	Attention-based	26.0	59.50	22.3	16.5	2021
A-FRCNN	Convolution	Mixture-based	6.1	81.28	18.3	16.7	2021
TF-GridNet	STFT	Mixture-based	14.5	231.00	23.5	19.2	2023
SPMambda	STFT	Mixture-based	6.1	238.00	22.5	19.9	2024
MossFormer2	Convolution	Attention-based	55.7	331.44	24.1	21.7	2024
SFSRNet	Convolution	Attention-based	59.0	488.10	24.0	16.4	2022

The speech separation task, the input mixture is denoted x(t), which is the sum of speech sources $s_1(t)$, $s_N(t)$:

$$x\ t\ =\ \ \sum_{i=1}^{N} s_i\ t\ . \quad (1)$$

The model processes the mixture x(t) to estimate $\hat{s}_1(t)$, …, $\hat{s}_N(t)$. The objective is to minimize the discrepancy between these estimates and the ground-truth sources $s_1(t)$, … $s_N(t)$.

For a fully convolutional model, the key idea of the encoder–decoder is to transform the time-domain signal vector into an STFT-like feature matrix. Conv-TasNet, SuDoRM-RF, and our implementation all use a 1D convolution (Conv1D) for this operation. Analogous to the STFT, we use overlapping segments of length L with a stride of L/2. The resulting feature matrix is then processed with layer normalization (LN), as shown next:

$$x_{enc} = LN(Conv1D_{C,1,1}(x(t))). \quad (2)$$

The input vector is encoded as a matrix of shape (TF, ENC_N), where TF denotes the number of time frames and ENC_N denotes the number of Conv1D kernels. This matrix is analogous to a classical STFT representation, but unlike the traditional STFT—where frequencies are fixed on a uniform grid—the Conv1D layer employs trainable kernel weights. This approach has shown improvements in related signal-processing domains, such as BLE communication systems [16], where using different trainable kernels instead of merely tuning the granular frequency bins of the STFT can yield superior performance. In our implementation, however, we use a single Conv1D layer rather than separate real and imaginary components, which simplifies the encoding.

The main component of the model is the separation module. It is designed to estimate masks that are applied to the encoder output via element-wise multiplication. In our setup, this module estimates a fixed number of masks corresponding to two talkers. First, the encoder output is passed through a bottleneck layer implemented as a pointwise convolution (PWConv1D):

$$x_{bn} = PWConv1D(x_{enc}). \quad (3)$$

We retain the main design principle of the Separator by implementing it as a stack of sub-layers. Whereas Conv-TasNet and SuDoRM-RF employ architecture-specific nested blocks, we combine their respective modules—Conv-block from Conv-TasNet and UConv-block from SuDoRm-

RF—arranged alternately, starting with a Conv-block and then placing a UConv-block after it R times.

In Conv-TasNet, the Conv-block is formulated as a temporal convolutional network composed of stacked 1-D dilated convolutional blocks. This block comprises D stacked 1-D dilated convolutional layers, with the dilation steps defined according to a predetermined schedule:

$$dilation = 2^d. \quad (4)$$

where d is defined from 1 to D.

The overall Conv-block consists of pointwise (PWConv1D) and depthwise (DWConv1D) convolutions. This block processes the input sequence x_inp with a pointwise convolution followed by layer normalization and a LeakyReLU activation:

$$y = LeakyReLU(LN(PWConv1D(x\_inp))). \quad (5)$$

This output is then passed through a DWConv1D layer with dilation rate d, computed by Eq. (4) from the block index. The dilated convolution output is then passed through a LeakyReLU activation:

$$y = LeakyReLU(DWConv1D(y)). \quad (6)$$

Next, a bottleneck pointwise convolution followed by layer normalization is applied:

$$y = LN(PWConv1D(y)). \quad (7)$$

The block output is then computed as the sum of the bottleneck output y and a skip connection from the model input x:

$$conv\_block\_out = x + y. \quad (8)$$

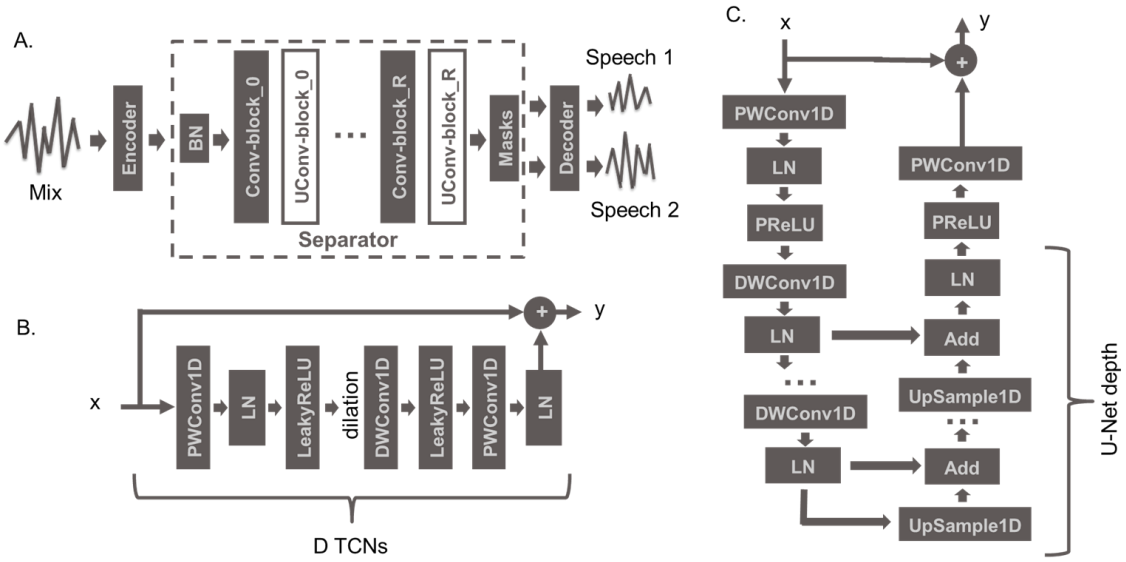

Fig. 1. A. SuDo-TasNet architecture. B. Conv-block architecture. C. UConv-block architecture.

The whole Conv-block consists of D such sub-blocks implementing the operations defined by formulas (5)–(8). This Conv-block is designed as an optimized, fully convolutional separation module that leverages the TCN paradigm to replace recurrent layers, enabling the convolutional stack to capture both local feature relations and long-term dependencies, which enhances processing of time-sequenced data and supports efficient deployment on NPUs and microcontrollers. After the Conv-block, the sequence is processed by the UConv-block, which we define in the next section.

The UConv-block is a fully convolutional module patterned after a U-Net architecture. Its purpose is to capture information across multiple temporal resolutions via mirrored, structured sequences of downsampling and upsampling operations. At the beginning of the UConv-block, a pointwise 1D convolution followed by layer normalization acts as a bottleneck:

$$y\_0 = \mathrm{LN}(\mathrm{PWConv1D}(x)). \quad (9)$$

Next, a depthwise 1D convolution is applied, followed by layer normalization, and the output of each stage is stored to serve as skip connections in the subsequent upsampling path. The downsampling path comprises U successive blocks defined as:

$$y\_i = \mathrm{LN}(\mathrm{DWConv1D}(y\_i\text{-}1)), \quad (10)$$

where the i-th convolution operates on the output of the (i−1)-th stage, followed by layer normalization.

After the downsampling path, we use an UpSample1D operation that upsamples by a factor of 2 and adds the result to the corresponding i-th skip connection:

$$y\_i = y\_i + \mathrm{UpSample1D}(y\_i). \quad (11)$$

At the end, the model applies layer normalization with a shared-axis PReLU activation, followed by an output pointwise convolution, overall defined as:

$$y = \mathrm{PWConv1D}(\mathrm{PReLU}(\mathrm{LN}(y\_U))). \quad (12)$$

Similar to the Conv-block, the UConv-block employs a residual connection by adding the input skip-connected feature to the block output:

$$\mathrm{uconv\_block\_out} = x + y. \quad (13)$$

In our implementation, the model consists of R Conv-blocks and R UConv-blocks, arranged alternately as illustrated.

V. EXPERIMENTAL SETUP

A. Dataset

We focus on the pure speech separation task, excluding noise suppression, and train and evaluate the model on data of this type. For training, we implement a data generator that uses LibriSpeech [17] train-360 as a source of clean speech fragments and mixes them with uniformly random amplitude scaling, producing mixtures with signal-to-noise ratios (SNR) between -20 dB and 20 dB for 90% of the samples. For evaluation, we use the pre-generated Libri2Mix test subset. Both training and testing are conducted on audio with an 8 kHz sampling rate.

B. Experiment configurations

The networks are trained for 600 epochs on 4-second audio segments. Training starts with an initial learning rate of $1e^{-3}$, which is reduced by a factor of 1.1 every 40 epochs. The Adam optimizer [18] is used for training. The batch size is 16, and each epoch comprises 2,000 batches of generator-produced training pairs.

C. Training and evaluation

The training objective is to maximize the scale-invariant signal-to-distortion ratio (SI-SDR), which is commonly used as both the loss function and the evaluation metric for speech separation tasks. To address the permutation problem, utterance-level permutation-

invariant training (uPIT) is applied. The SI-SDR is calculated as:

$$\alpha = \frac{s^T s}{\|s\|^2}, \tag{14}$$

$$SI-SDR(s, \hat{s}) = 10\log_{10}\frac{\|\alpha s\|^2}{\|\alpha s - \hat{s}\|^2}, \tag{15}$$

where s is a target signal, and $\hat{s}$ is estimated speech.

$$L(S, \hat{S}) = -\max_{\pi \epsilon \Pi_N}\left(\frac{1}{N}\sum_{i=1}^{N} SI-SDR(s_i, \hat{s}_{\pi(i)})\right), \tag{16}$$

where S is the list of N target sources, $\hat{S}$ is the list of N estimated sources, and the max operation ensures the optimal assignment.

For evaluation, we use the SI-SDRi metric, which measures the improvement in separation quality relative to the baseline—the original mixed signal (mix). The formula for SI-SDRi is:

$$\text{SI-SDRi}(s, \hat{s}, \text{mix}) = \text{SI-SDR}(s, \hat{s}) - \text{SI-SDR}(\text{mix}, \hat{s}). \tag{17}$$

D. Results and discussion

In Table 2, we demonstrate the separation performance of our proposed model versus Conv-TasNet and SuDoRM-RF models with the same number of parameters.

Table 2

Comparison of small models for Libri2Mix test

Models	SI-SDRi (db)	Params. (M)	GMACS/s
Conv-TasNet	9.52	1.20	0.96
SuDORM-RF	9.24	1.21	**0.90**
SuDo-TasNet (our)	**10.59**	**1.17**	0.92

The table additionally includes the size of the models in terms of parameters and computational requirements. All presented results are computed on the Libri2Mix test subset for the clean case (no noise). Our SuDo-TasNet model demonstrates improvement over 1 db for SI-SDRi compared to the Conv-TasNet model, while remaining a similar number of parameters and GMACS/s.

The SuDo-TasNet implementation is a fully convolutional model, which can be efficiently accelerated on lightweight NPUs such as the Ethos-U55 [19]. In contrast, the TIGER model relies on attention-based layers that are not widely supported by most NPUs, making our approach more suitable for deployment on edge devices.

VI. CONCLUSION

In this study, we introduced a hybrid speech separation network that combines Conv-TasNet and SuDoRM-RF layers. Our proposed model achieves an SI-SDRi of 10.59 db, providing an improvement of approximately 1 db compared to the original Conv-TasNet and SuDoRM-RF models, while maintaining the same number of trainable parameters and computational cycles.

Although the smallest state-of-the-art model, TIGER, uses only 0.8 M parameters, its computational cost is 8 times higher than that of our proposed model. It should also be noted that our experiments were conducted on 8 kHz audio samples, while TIGER operates on 16 kHz data. For 16 kHz, the SuDo-TasNet model can retain a similar number of parameters, but the required GMACS/s would approximately double, making our model about 4 times faster than TIGER.

VII. CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

VIII. DECLARATION ON GENERATIVE AI

During the preparation of this work, the author(s) used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] Cosentino, J., Pariente, M., Cornell, S., Deleforge, A., & Vincent, E. (2020). LibriMix: An open-source dataset for generalizable speech separation. *ArXiv preprint arXiv:2005.11262*. DOI: https://doi.org/10.48550/arXiv.2005.11262.

[2] Hershey, J. R., Chen, Z., Le Roux, J., & Watanabe, S. (2016). Deep clustering: Discriminative embeddings for segmentation and separation. *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2016*. DOI: https://doi.org/10.1109/ICASSP.2016.7471631.

[3] Maciejewski, M., Wichern, G., McQuinn, E., & Le Roux, J. (2020). WHAMR!: Noisy and reverberant single-channel speech separation. *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2020*. DOI: https://doi.org/10.48550/arXiv.1910.10279.

[4] Lichouri, M., Lounnas, K., Djeradi, R., & Djeradi, A. (2022). Performance of end-to-end vs pipeline spoken language understanding models on multilingual synthetic voice. In *Proc. 5th Int. Conf. Advanced Aspects of Software Engineering (ICAASE'22)* (Constantine, Algeria), Sep. 2022. DOI: https://doi.org/10.1109/ICAASE56196.2022.9931594.

[5] Zhao, S., Ma, Y., Ni, C., Zhang, C., Wang, H., Nguyen, T. H., Zhou, K., Yip, J., Ng, D., & Ma, B. (2024). MossFormer2: Combining transformer and RNN-free recurrent network for enhanced time-domain monaural speech separation. *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2024*. DOI: https://doi.org/10.48550/arXiv.2312.11825.

[6] Rixen, J., & Renz, M. (2022). SFSRNet: Super-resolution for single-channel audio source separation. *Proc. AAAI-22, 36th AAAI Conf. Artificial Intelligence*, 36(10), 11220–11228. DOI: https://doi.org/10.1609/aaai.v36i10.21372.

[7] Le Roux, J., Wisdom, S., Erdoğan, H., & Hershey, J. R. (2018). SDR – half-baked or well done? *ArXiv preprint arXiv:1811.02508*. DOI: https://doi.org/10.48550/arXiv.1811.02508.

[8] Wang, Z.-Q., Cornell, S., Choi, S., Lee, Y., Kim, B.-Y., & Watanabe, S. (2023). TF-GridNet: Making time-frequency domain models great again for monaural speaker

separation. *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2023*. DOI: https://doi.org/10.48550/arXiv.2209.03952.

[9] Li, K., Chen, G., Yang, R., & Hu, X. (2024). SPMamba: State-space model is all you need in speech separation. *ArXiv preprint arXiv:2404.02063*. DOI: https://doi.org/10.48550/arXiv.2404.02063.

[10] Luo, Y., & Mesgarani, N. (2019). Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation. *IEEE/ACM Trans. Audio, Speech, Language Process.*, 27(8), 1256–1266. DOI: https://doi.org/10.1109/TASLP.2019.2915167.

[11] Tzinis, E., Wang, Z., & Smaragdis, P. (2020). Sudo rm -rf: Efficient networks for universal audio source separation. In *Proc. 2020 IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP)*. DOI: https://doi.org/10.1109/MLSP49062.2020.9231900.

[12] Li, K., Chen, G., Sang, W., Luo, Y., Chen, Z., Wang, S., He, S., Wang, Z.-Q., Li, A., Wu, Z., & Hu, X. (2025). Advances in speech separation: Techniques, challenges, and future trends. *ArXiv preprint arXiv:2508.10830*. DOI: https://doi.org/10.48550/arXiv.2508.10830.

[13] Luo, Y., Chen, Z., & Yoshioka, T. (2020). Dual-path RNN: Efficient long sequence modeling for time-domain single-channel speech separation. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2020*. DOI: https://doi.org/10.1109/ICASSP40776.2020.9054266.

[14] Yang, L., Liu, W., & Wang, W. (2022). TFPSNet: Time-frequency domain path scanning network for speech separation. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2022*. DOI: https://doi.org/10.1109/ICASSP43922.2022.9747554.

[15] Xu, M., Li, K., Chen, G., & Hu, X. (2025). TIGER: Time-frequency interleaved gain extraction and reconstruction for efficient speech separation. *Proc. International Conf. on Learning Representations (ICLR) 2025*. DOI: https://doi.org/10.48550/arXiv.2410.01469.

[16] Tsemko, A., Santra, A., Kapshii, O., & Pandey, A. (2024). Data-driven processing using parametric neural network for improved Bluetooth channel sounding distance estimation. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2025*, 1–5.

[17] Panayotov, V., Chen, G., Povey, D., & Khudanpur, S. (2015). LibriSpeech: An ASR corpus based on public domain audio books. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP) 2015*, 5206–5210. DOI: https://doi.org/10.1109/ICASSP.2015.7178964.

[18] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proc. 3rd Int. Conf. Learning Representations (ICLR) 2015*, San Diego, CA. DOI: https://doi.org/10.48550/arXiv.1412.6980.

[19] Arm Ethos-U55. [Electronic resource]. Arm Developer. Available: https://developer.arm.com/Processors/Ethos-U55

Andrii Tsemko was born in Melitopol, Ukraine, in 2000. Starting from 2023, he has been studying for a PhD in Computer Science at the Faculty of Electronics and Computer Technologies of Ivan Franko National University of Lviv. His research interests encompass machine learning, signal processing, embedded systems, and wireless communication technologies such as Wi-Fi and Bluetooth Low Energy (BLE).

Ivan Karbovnyk, PhD, Dr. Sci., born in Lviv, Ukraine, in 1978, is the Chair of Radiophysics and Computer Technologies Department at Ivan Franko National University of Lviv. With over 20 years of research experience, he specializes in automation, embedded systems, Internet of Things, computer modeling and electronics.