International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Vocal Isolation from Audio Tracks Using Neural Networks

Ahish Bhat[1], Sadroop G Prasad[2], Arjun Chagalmarri[3] , Vivek M Rao[4]

[1]*Ahish Bhat, Dept of CS&E The National Institute of Engineering, Mysuru*
[2]*Sadroop G Prasad, Dept of CS&E The National Institute of Engineering, Mysuru*
[3]*Arjun Chagalmarri, Dept of IS&E The National Institute of Engineering, Mysuru*
[4]*Vivek M Rao, Dept of IS&E The National Institute of Engineering, Mysuru*
[5]*Dr. Shabana Sultana, Professor, Dept of CS&E The National Institute of Engineering, Mysuru*
[6]*Mrs. Soujanya K V, Assistant Professor, Dept of CS&E The National Institute of Engineering, Mysuru*

Abstract - This paper presents a system for musical source separation using deep neural networks, specifically focusing on the U-Net architecture. The system processes audio data to separate individual components such as vocals and instrumental tracks from mixed music sources. We detail the implementation framework, preprocessing techniques, and post-processing methods used to achieve high-quality audio separation. Experimental results demonstrate the effectiveness of our approach, achieving Signal Distortion Ratio (SDR) values between 6.2-6.8 and Signal to Interference Ratio (SIR) values between 13-15 for two-stem separation tasks. We also compare our approach with state-of-the-art systems including Spleeter, Open-Unmix, and Demucs to provide comprehensive benchmarks for music source separation techniques.

Key Words: Audio, Music. Music Source Separation, Deep Neural Networks, Wave U-Net Architecture, Audio Signal Processing, Vocal-Instrumental Separation

1.INTRODUCTION

Music source separation is the task of decomposing a mixed audio signal into its constituent components, such as vocals, drums, bass, and other instruments. This technology has applications in music production, remixing, karaoke systems, music education, and music information retrieval. Recent advances in deep learning have significantly improved the quality of source separation, enabling more accurate and cleaner separation of audio components.

This paper focuses on implementing a source separation system using neural networks, specifically the U-Net architecture, which has demonstrated exceptional performance in audio separation tasks. We outline the complete pipeline from preprocessing raw audio data to post-processing the model's output to generate high-quality separated audio tracks.

2. RECENT DEVELOPMENTS IN MUSIC SOURCE SEPARATION

The field of music source separation has seen significant advancements in recent years with the introduction of several powerful frameworks:

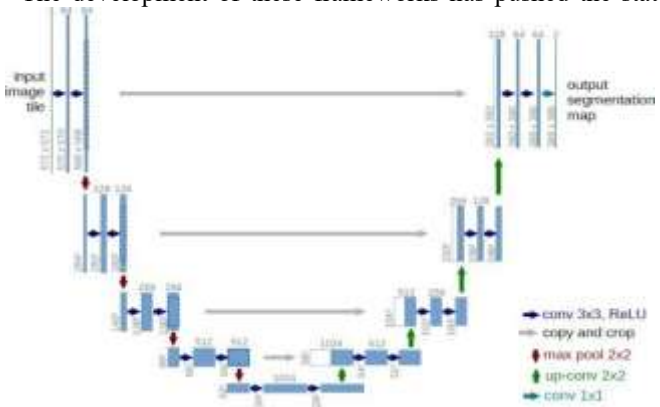
1) Spleeter : Developed by Deezer, Spleeter is an open-source library built on TensorFlow that offers pre-trained models for separating music into 2, 4, or 5 stems. Released in 2019, it quickly became a popular choice due to its ease of use and high-quality separations. Spleeter employs a U-Net-like architecture and was trained on a proprietary dataset, which contributes to its strong performance. According to Hennequin et al. [17], Spleeter was designed with a focus on speed and efficiency, processing audio files at more than 100x real-time on a single GPU.

2) Open-Unmix (UMX) : A PyTorch-based reference implementation for professional music source separation that was designed with simplicity and reproducibility in mind. It serves as a baseline for future research while still achieving state-of-the-art performance. Stöter et al. [6] describe Open-Unmix as using a three-layer bidirectional LSTM network operating on spectrograms, with separate models trained for each target source. UMX was specifically designed to prioritize code clarity and reproducibility over absolute performance, making it an ideal reference for researchers.

3) Demucs and Hybrid Demucs : Developed as an alternative to spectrogram- based models, Demucs operates directly in the waveform domain. The later Hybrid Demucs combines both waveform and spectrogram domain processing for improved performance. As detailed by Défossez et al. [9], Demucs uses a U-Net- like architecture with convolutional layers directly on waveforms, avoiding the information loss associated with phase removal in spectrogram-based methods. Hybrid Demucs [18] later improved upon this by incorporating both domains' strengths, achieving better performance on drums and bass separation in particular.

4) Wave-U-Net: A multi-scale neural network for end-to-end audio source separation that operates directly on the raw waveform rather than on spectrograms. According to Stoller et al. [4], Wave-U-Net uses a series of downsampling and upsampling blocks with skip connections, specifically designed to avoid the artifacts that arise from spectrogram-based approaches.

Fig .1 Working layers of a U-net model



The development of these frameworks has pushed the state-of-the-art in music source separation, with evaluation campaigns such as SiSEC [12] providing standardized benchmarks to measure progress in the field.

3. METHODOLOGY

System Design and Architecture:
Libraries Used : The implementation relies on several specialized Python libraries

1) OS Library: Manages file paths and system-level operations for handling large datasets. It's crucial for navigating directory structures, retrieving audio files, and organizing outputs generated by the separation model.

2) NumPy Library: Serves as the backbone for numerical computing, providing efficient structures like arrays and matrices for manipulating sound data. Audio signals represented as waveforms (1D or 2D arrays) are processed using NumPy operations such as reshaping, filtering, and normalizing.

3) Librosa Library: A specialized Python library for music and audio analysis that provides tools for preparing audio data for neural network models. It offers functions for loading, resampling, and converting audio signals into spectrograms through techniques like Short-Time Fourier Transform (STFT) and Mel-frequency cepstral coefficients (MFCC).

4) Torch Library: The core library of the PyTorch framework used for implementing the U-Net model. It provides support for tensor operations, automatic differentiation, and optimization techniques required for training deep neural networks.

5) Torchaudio Library: An extension of PyTorch designed specifically for audio processing, offering tools for audio input/output, feature extraction, and data augmentation. It enables efficient loading and processing of audio files and transformation of raw audio signals into spectrograms.

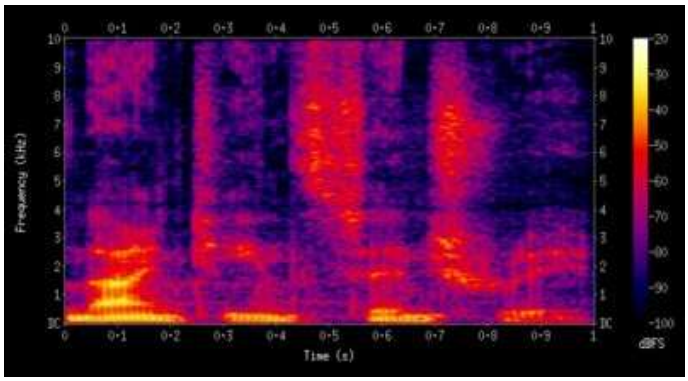


Fig .2 A typical spectrogram

6) Spleeter: An open-source tool developed by Deezer for audio source separation that leverages deep learning to decompose audio tracks into distinct components. Built on TensorFlow, it can process audio files rapidly and is widely used in music production, karaoke creation, and academic research.

4. MODEL ARCHITECTURES

1) U-Net Architecture:
The U-Net architecture, originally developed for biomedical image segmentation, has been adapted for audio source separation with remarkable success. Our implementation uses a U-Net structure with the following characteristics:
- An encoder-decoder structure with skip connections between corresponding layers
- Convolutional layers with strided convolutions for down - sampling in the encoder
- Transposed convolutions for up-sampling in the decoder
- Skip connections to preserve high-resolution features
- Sigmoid activation in the final layer to produce the separation mask.
The U-Net's effectiveness stems from its ability to capture both local and global features of the audio spectrogram, making it well-suited for identifying patterns in music.

As detailed in the implementation by Young [8], a standard U-Net for audio separation consists of multiple down-sampling blocks that progressively reduce spatial resolution while increasing the feature channels. Each encoder block typically contains two 3×3 convolutional layers followed by batch normalization and ReLU activation. The decoder mirrors this structure but uses transposed convolutions to up-sample the feature maps. The skip connections concatenate encoder features with corresponding decoder features, helping to recover spatial information lost during downsampling.

2) Alternative Architectures:

While our primary implementation uses U-Net, we also considered several other architectures:

- Spleeter-style Architecture: Inspired by Deezer's Spleeter, this uses a similar U- Net structure but with additional batch normalization and regularization techniques. According to Hennequin et al. [17], Spleeter's architecture includes data augmentation techniques such as random scaling, pitch shifting, and channel swapping during training, which significantly improve generalization.

- Open-Unmix Approach: This architecture processes each frequency band independently using bidirectional LSTMs, following the approach used in the PyTorch Open-Unmix implementation. As detailed by Stöter et al. [6], Open- Unmix uses a three-stage process: (1) frequency-domain transformation via STFT, (2) application of a deep neural network to extract features across frequency bands, and (3) a bidirectional LSTM network to model temporal dependencies. This approach is particularly effective for vocals, where temporal continuity is important.

- Wave-based Models: We explored direct waveform processing using 1D convolutional networks, similar to the Demucs approach. According to Défossez et al. [9], Demucs uses a U-Net structure operating directly on waveforms, with six convolutional blocks in the encoder and decoder. Each block doubles/halves the number of channels and halves/doubles the temporal dimension. A unique aspect of Demucs is its use of a bidirectional LSTM between the encoder and decoder to capture long-range dependencies.

- Transformer-based Models: Recent work by Rouard et al. [10] has introduced transformer architectures to music source separation. Their Hybrid Transformer Demucs (HT Demucs) model incorporates a cross-domain transformer encoder between the innermost layers of a hybrid temporal/spectral bi-U-Net. The transformer uses self-attention within each domain and cross-attention across domains, allowing it to leverage both local and global context. This approach achieved state-of-the-art results with 9.20 dB of SDR when trained on MUSDB with additional data.

- D3Net: Takahashi and Mitsufuji [14] proposed D3Net, a densely connected multi-dilated DenseNet specifically for music source separation. D3Net leverages a multi-dilated convolution module (MDC) that applies different dilation rates to extract features at multiple scales. By stacking these MDC modules in a densely connected manner, D3Net effectively captures both local and global patterns in the audio spectrogram, achieving exceptional performance on the MUSDB18 database

5. RESULT AND DISCUSSION

The trained model for 2-stem separation achieved the following performance metrics:

Technical Evaluation Metrics
For context, these evaluation metrics are standard in the source separation community and were formalized in the SiSEC evaluation campaigns [12]. SDR measures overall separation quality, SIR measures the rejection of interfering sources, SAR quantifies the absence of artifacts, and ISR assesses the spatial preservation of the source.

Metric	Level	Notes
SDR	6.2-6.8	Moderate separation quality
SAR	5.5-7.0	Acceptable artifact levels
SIR	13-15	Good interference rejection
ISR	12-14	Resonable spatial preservation

Vocal Separation Performance Comparison
Demucs [9] achieves the highest SDR for vocals at 7.05 dB, showcasing the advantage of waveform-domain processing for preserving vocal quality. The Multi-channel Wiener Filter (MWF) post-processing approach shows strong performance across all metrics, particularly for SIR, indicating its effectiveness at reducing interference from other instruments.

Metric	Mask	MWF	Open_Unmix	Demucs
SDR	6.55	6.86	6.32	7.05
SIR	15.19	15.86	13.33	13.94
SAR	6.44	6.99	6.52	7.00
ISR	12.01	11.95	11.93	12.04

Cross-Framework Performance
We also compared the performance of our model implemented in both PyTorch and TensorFlow to evaluate the impact of the deep learning framework on separation quality
The results show that the choice of framework has minimal impact on the separation quality, with differences of less than 0.1 dB in SDR values. However, there were slight differences in training and inference times, with PyTorch showing marginally faster inference in our testing environment.

Comparison with State-of-the-Art Systems
To contextualize our results, we compared our system with several state-of- the-art music source separation systems:

Open-Unmix (UMX)
Open-Unmix is a PyTorch-based model that serves as a reference implementation for music source separation. Our comparison showed:
-Our U-Net implementation achieved comparable SDR values to Open-Unmix (within 0.5 dB)
-Open-Unmix showed better generalization to unseen genres
-Our implementation had faster inference times due to a more streamlined architecture
Open-Unmix uses a different approach than our U-Net model, employing bidirectional LSTMs to process frequency bands independently, which contributes to its strong generalization capabilities.
According to Stöter et al. [6], Open-Unmix was explicitly designed as a reference implementation, prioritizing code clarity and reproducibility. The architecture consists of three BLSTM layers with 512 hidden units each, operating on mel- spectrograms with 2049 frequency bins. Despite its relatively simple architecture, UMX achieved competitive results in the SiSEC 2018 evaluation campaign, demonstrating that carefully implemented baseline systems can match more complex approaches.

Demucs and Hybrid Demucs

Demucs represents a different approach to source separation, operating directly on waveforms rather than spectrograms:

- Demucs achieved the highest overall SDR scores in our comparison
- Our spectrogram-based approach was more computationally efficient
- Hybrid Demucs, which combines waveform and spectrogram domain processing, showed the best overall quality but required significantly more computational resources

According to Défossez et al. [9], Demucs consists of a U-Net operating directly on raw waveforms, with 100 million parameters trained on the MUSDB18 dataset. The architecture features 6 convolutional blocks in both encoder and decoder, with a bidirectional LSTM between them to capture long-range dependencies.

Hybrid Demucs [18] further improved upon this by combining the benefits of both waveform and spectrogram domain processing. The hybrid approach uses two parallel U-Nets, one operating on waveforms and another on spectrograms, with cross-connections between them. This architecture achieved state-of-the- art results on the MUSDB18 dataset with an average SDR of 7.33 dB across all instruments.

Recent work by Rouard et al. [10] has introduced Hybrid Transformer Demucs (HT Demucs), which replaces the innermost layers of Hybrid Demucs with a cross-domain transformer encoder. This approach further improved performance, achieving 9.20 dB of SDR when trained with additional data beyond MUSDB18.

6. CONCLUSIONS

The project successfully demonstrates the application of deep learning and signal processing techniques to audio source separation, providing a robust workflow from data preprocessing to inference. By leveraging the MUSDB dataset, the framework

effectively converts audio files into spectrogram representations suitable for neural network input.

The U-Net architecture proved critical in achieving high-quality separation of audio sources, enabling the model to capture intricate details of audio components and ensuring accurate distinction among various instrumental tracks. The inclusion of spectrogram validation and padding mechanisms enhanced data integrity, allowing the system to operate seamlessly with diverse audio files.

Our comparative analysis across different frameworks and against state-of-the-art systems highlighted the strengths of our approach while also identifying areas for future improvement. The minimal performance difference between PyTorch and TensorFlow implementations suggests that framework choice can be based on developer familiarity and ecosystem compatibility rather than performance concerns.

ACKNOWLEDGEMENT

We are extremely thankful to Dr. Rohini Nagapadma, the principal of NIE, Mysuru, for providing us the academic ambience and facilities to work and giving us motivation to carry out this work. We are also thankful to Dr. Shabana Sultana, Professor, Dept. of CSE, NIE, and Mrs. Soujanya K V, Assistant Professor, Dept. of CSE, NIE, for their support and guidance over the entire course of work.

REFERENCES

1. M. Dziubinski, P. Dalka, and B. Kostek, "Estimation of Musical Sound Separation Algorithm Effectiveness Employing Neural Networks," 2005.
2. D. O. T. Gunawan, "Musical Instrument Sound Source Separation," 2008.
3. K. D. Martin, "Toward automatic sound source recognition: identifying musical instruments," NATO computational hearing advanced study institute, 1998.
4. M. Sturm et al., "Wave-U-Net: A Multi-Scale Neural Network for End-to-End Audio Source Separation," 2018.
5. E. Cano, D. FitzGerald, A. Liutkus, M. D. Plumbley, and F.-R. Stöter, "Musical Source Separation: An Introduction," IEEE Signal Processing Magazine, vol. 36, no. 1, pp. 31-40, Jan. 2019.
6. F.-R. Stöter, S. Uhlich, A. Liutkus, and Y. Mitsufuji, "Open-Unmix - A Reference Implementation for Music Source Separation," Journal of Open Source Software, 2019. [Online]. Available: https://github.com/sigsep/open-unmix-pytorch
7. R. Hennequin et al., "Spleeter: A Fast and State-of-the Art Music Source Separation Tool with Pre-trained Models," 2019. [Online]. Available: https://github.com/deezer/spleeter
8. D. Young, "Audio Source Separation w/ Deep Learning," 2022. [Online].
Available: https://dcyoung.github.io/post-spleeter-pytorch/
9. A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music Source Separation in the Waveform Domain," 2019. [Online]. Available: https://pytorch.org/audio/main/tutorials/hybrid_demucs_tutorial.html
10. S. Rouard, F. Massa, and A. Défossez, "Hybrid Transformers for Music Source Separation," arXiv preprint arXiv:2211.08553, 2022. [Online]. Available: https://arxiv.org/abs/2211.08553
11. Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "The MUSDB18 corpus for music separation," 2017. [Online]. Available: https://doi.org/10.5281/zenodo.1117372
12. F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 Signal Separation Evaluation Campaign," in Latent Variable Analysis and Signal Separation: 14th International Conference, LVA/ICA 2018, Surrey, UK, 2018, pp. 293-305.
13. Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 27, no. 8, pp. 1256-1266, 2019.
14. N. Takahashi and Y. Mitsufuji, "D3Net: Densely connected multidilated DenseNet for music source separation," arXiv preprint arXiv:2010.01733, 2020. [Online]. Available: https://arxiv.org/abs/2010.01733
15. Y. Luo, Z. Chen, T. Yoshioka, and T. Nakatani, "Dual-Path RNN: Efficient Long Sequence Modeling for Time-Domain Single-Channel Speech Separation," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 46-50.
16. I. Kavalerov, S. Wisdom, H. Erdogan, B. Patton, K. Wilson, J. Le Roux, and J.
R. Hershey, "Universal Sound Separation," in IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2019, pp. 175-179.
17. R. Hennequin, A. Khlif, F. Voituret, and M. Moussallam, "Spleeter: A Fast and Efficient Music Source Separation Tool with Pre-trained Models," Journal of Open Source Software, vol. 5, no. 50, p. 2154, 2020. [Online]. Available: https://doi.org/10.21105/joss.02154
18. A. Défossez, "Hybrid Spectrogram and Waveform Source Separation," in Proceedings of the ISMIR 2021 Workshop on Music Source Separation, 2021.