Comparative Evaluation of Fine-Tuned Hybrid Transformer and Band-Split Recurrent Neural Networks for Music Source Separation

Ken Kalang Al Qalyubi*[1], Nur Ahmadi*†[2], Dessi Puji Lestari*†[3]
*School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung, 40132, Indonesia
†Center for Artificial Intelligence (U-CoE AI-VLB), Bandung Institute of Technology, Bandung, 40132, Indonesia
Email: [1]alqalyubiken@gmail.com, [2]nahmadi@itb.ac.id, [3]dessipuji@itb.ac.id

***Abstract*—In recent advancements within the field of music source separation (MSS), state-of-the-art models such as Hybrid Transformer Demucs (HT Demucs) and Band-Split Recurrent Neural Networks (BSRNN) have been at the forefront. Although the pre-trained HT Demucs model is capable of separating six sources—drums, bass, guitar, piano, vocals, and others—it underperforms on guitar, piano, and other sources compared to its performance on bass, drums, and vocals, as measured by the utterance-level Signal-to-Distortion Ratio (uSDR) metric. To date, there has been no evaluation of the BSRNN model's ability to separate these six sources. This paper seeks to address this gap by investigating and comparing the performance of the BSRNN and HT Demucs models for six-source separation. Using the MoisesDB dataset, both models were developed and fine-tuned for this task. Their performance was then evaluated to identify the superior model for six-source separation. Experimental results reveal that the fine-tuned HT Demucs model surpasses the BSRNN model, achieving average uSDR and cSDR scores of 6.26 dB and 5.88 dB, respectively, compared to 5.52 dB and 5.38 dB for the BSRNN model. Moreover, the fine-tuned HT Demucs model outperforms its pre-trained counterpart on piano and other sources by 1 dB and 0.3 dB, respectively.**

I. Introduction

The task of music source separation (MSS) has drawn more and more attention in the community due to its wide application in music field [1]. MSS is a crucial technology in music information retrieval, aimed at isolating one or more target music sources from their mixture. Target music sources typically refer to various musical instruments such as bass, drums, and vocals, while the mixture refers to the combination of these source signals [2]. One of the popular models in music source separation (MSS) is Spleeter which is designed for ease and speed in separation tasks for its users [3]. However, this model has relatively low scores compared to current state-of-the-art models. In recent research, Demucs achieved state-of-the-art status with its latest development, Hybrid Transformer Demucs architecture [4]. In their latest study, Sparse HT Demucs achieved an average Signal-to-Distortion Ratio (SDR) score of 9.20 dB. The most competitive baseline model is BSRNN, which achieved better SDR scores for the other and vocal sources [4]. In the comparison of scores for each sound source, Demucs has an advantage in separating drum and bass sources, while the vocal and other sources are still better separated by the BSRNN model, with SDR scores of 10.47 and 7.08 compared to Demucs scores of 9.47 and 6.41. Currently, MSS models are largely limited to applications involving only four sources: bass, drums, vocals, and other. Other sound sources are grouped or merged into the other category. The lack of publicly available datasets in this domain also contributes to the limitation of these sound sources.

One of the datasets commonly used to compare the performance of MSS models is MUSDB18. This dataset only contains data for bass, drums, vocals, and other sources. Although MUSDB18 [5], [6] has significantly contributed to advancements in this task, its source grouping is still too coarse for many real-world remix applications [7]. A recent study developed a dataset specifically to address the challenges of MSS tasks, called MoisesDB. This dataset facilitates the creation and evaluation of detailed source separation systems, surpassing the limitations of using only 4 sources (drums, bass, other, and vocals) due to a lack of data [8]. It consists of 240 music tracks from different artists and genres with a total duration of over 14 hours. The study tested the pre-trained HT Demucs model on six source classes (drums, bass, other, vocals, guitar, and piano). The results showed that HT Demucs achieved an average SDR score of 6.24 dB across all sources. Specifically, for each source, HT Demucs obtained SDR scores of 9.55 dB for vocals, 11.93 dB for bass, 11.02 dB for drums, 0.28 dB for other, 1.60 dB for piano, and 3.07 dB for guitar. The SDR scores for guitar, piano, and other are still relatively lower compared to sources like bass, drums, and vocals.

Guitar and piano sources are derived from splitting the other stem where the previous stems group consists of drums, vocals, other, and bass. Although HT Demucs study shows that BSRNN model outperforms in the other class [4], there has been no research conducted on separating guitar and piano stems using this model. In this case, BSRNN might surpass HT Demucs model in separating guitar and piano sources. This highlights the need to investigate the BSRNN model in separating 6 stem sources. Additionally, it is necessary to rebuild the HT Demucs model for comparison to determine which model is the best for separating 6 stem sources.

II. Related Works

In this literature, we found several studies that have done experiments related to our study. We use these works as basis to select some of our experiments.

A. Hybrid Transformers for Music Source Separation

Hybrid Transformer Demucs (HT Demucs), which enhances music source separation by combining temporal and spectral processing achieved state-of-the-art results on MUSDB with extra training data, with 9.20 dB of SDR [4]. HT Demucs integrates Transformer layers into the existing Hybrid Demucs framework, utilizing self-attention within and cross-attention across temporal and spectral domains. This approach significantly improves the model's ability to handle long-range dependencies in music signals, resulting in superior performance compared to the original Hybrid Demucs, particularly when trained with additional data. The model achieves state-of-the-art results in the Music Demixing Challenge, demonstrating its effectiveness.

Furthermore, the paper discusses the advantages of using self-attention mechanisms to capture contextual information over long sequences, which is critical for accurate music source separation. By leveraging these techniques, HT Demucs achieves more precise separation of musical components, such as vocals, drums, bass, and other instruments. The research highlights the potential of Transformer-based models in advancing the field of music source separation.

B. Music Source Separation with Band-Split RNN (BSRNN)

BSRNN architecture significantly improves performance compared to state-of-the-art models in the Music Demixing Challenge 2021 [9]. This model divides the spectrogram of a music mixture into different frequency bands, which allows for more precise processing tailored to the characteristics of different instruments. The BSRNN leverages the distinct properties of musical signals by optimizing the separation process based on their frequency components.

In addition to the BSRNN architecture, the paper introduces a semi-supervised finetuning pipeline that utilizes unlabeled data to further enhance the model's performance. By combining labeled and unlabeled data, the model achieves superior results in music source separation. This innovative approach demonstrates significant advancements in the field, providing a more effective solution for isolating individual instruments from a music mixture, ultimately contributing to the improvement of music production and analysis tools.

C. MoisesDB: A Dataset for Source Separation Beyond 4-Stems

MoisesDB is a dataset specifically created to address the limitations of datasets in the field of Music Source Separation (MSS). This dataset will facilitate the creation and evaluation of source separation that goes beyond the use of only four sources (drums, bass, others, and vocals) due to data scarcity [8]. MoisesDB contains a total of 240 songs, each with separate sound sources. Another dataset that has been a standard in source separation is MUSDB18. Although MUSDB18 has significantly contributed to the progress in this task, its source grouping is too coarse for many real-world remix applications [7]. Additionally, MedleyDB dataset [10], [11] suffers from poor labeling [8]. These issues inspired the creation of MoisesDB, which is not limited to just four sound sources.

TABLE I
Dataset description

Item	Value
Number of data (songs)	88
Total duration (hours)	5
Average duration (minutes)	3

III. Methods

In this study, we aim to investigate the performance of BSRNN and HT Demucs models in separating six stem sources. We conduct several experiments to find the best model and configuration for each model.

A. Data

Data in MoisesDB will first be filtered to ensure that each song contains at least the following sources: guitar, piano, vocals, drums, bass, and one other sound source classified as other. Additionally, there will be a mixed data set, which is the combination of all sound sources. After filtering, the data for each sound source in each song will be checked to ensure they have the same duration. Any sound sources with differing durations will be trimmed to match the shortest sound source duration in the respective song. We then split the processed dataset accordingly with 70:20:10 proportions resulting in 62 train set, 18 test set, and 8 validation set. The dataset description is shown in Table I.

B. Band-Split Recurrent Neural Networks (BSRNN)

BSRNN architecture that we use for this experiment is the implementation found in this repository[1]. Therefore, the implementation script from that repository will be used and adjusted according to the goals of the research. The model will be built from scratch, without using pre-trained models. Six new models need to be constructed: vocal, drum, bass, guitar, piano, and other. For all experiments we train the model with an initial learning rate $1e^{-3}$ and a batch size of 2 [9]. We use the small model version because of the limitation of our computing resource, where we set the feature dimension N to be 64, hidden unit of BLSTM layers to be 2N = 128, the hidden size in the mask estimation MLP to be 4N = 256, and use 8 band and sequence modeling modules. BSRNN is a single-target model meaning that a model is only responsible for one stem [9]. Figure 1 depicts the inference process of the BSRNN model for separating different musical components from an input song. The input song is processed by separate

[1] https://github.com/magronp/bsrnn

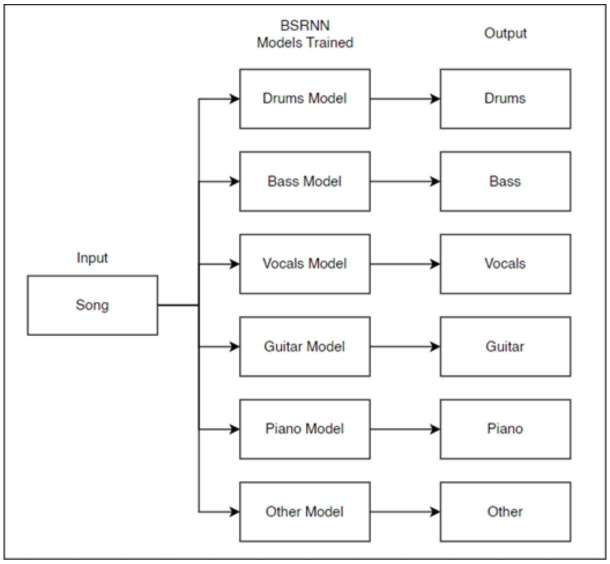

Fig. 1. BSRNN inference process for MSS.

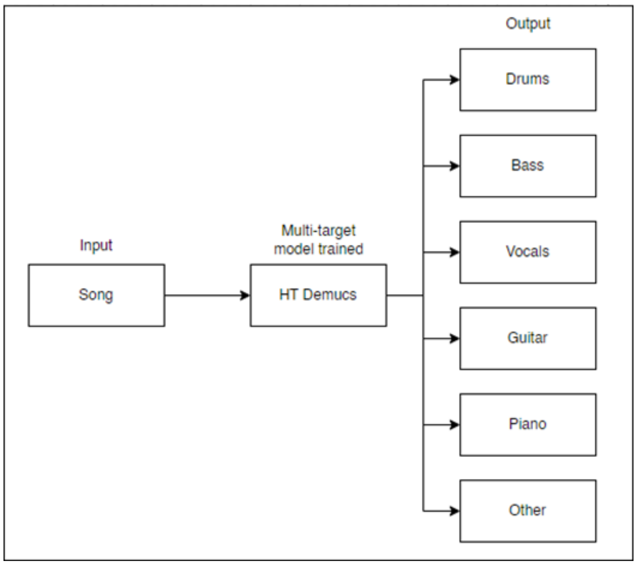

Fig. 2. HT Demucs multi-target inference process for MSS.

BSRNN models, each specifically trained to isolate one stem. Each model outputs the corresponding isolated track for its respective stem.

For building the new sound source models, specifically guitar and piano, a bandsplit configuration is required. Thus, the first step in developing the BSRNN model is to determine the bandsplit configuration that provides the best performance for separating guitar and piano sources as the rest of the stems already have their own configurations. Subsequently, models for each sound source will be constructed as part of the final training process. Each model for guitar and piano will be tested with three bandsplit configurations [9]. These configurations include vocals configuration (v7), bass configuration, and drums configuration. For speed, we train this experiment with a maximum epoch of 50 for speed.

Based on these configurations, the best-performing bandsplit configuration for the guitar and piano models will be determined. Once the optimal bandsplit configuration for guitar and piano as new sound sources is identified, the next step is to train all stems. These models include vocals, drums, bass, guitar, piano, and other. We also use the V7 configuration for the other stem, as previous research has done. For these experiments, we train the models with a maximum epoch of 100.

C. HT Demucs

HT Demucs model used is the official implementation from Meta AI, available on the GitHub repository[2] maintained by Facebook Research. Since Sparse HT Demucs is not publicly accessible, we have opted to use HT Demucs for this study. For all HT Demucs experiments, we train the models with a batch size of 2, optimized with Adam without weight decay. For HT Demucs we also use the small model where we set the channels to be 48 and the input/output dimension of the Transformer is 384. HT Demucs model can be train as a multi-target model or single-target model. Figure 2 shows how the inference process of the HT Demucs multi-target model for separating different musical components from an input song. The input song is processed by a single HT Demucs model trained for multi-target separation. The model outputs isolated tracks for each stem.

We first train HT Demucs as a multi-target model. We use learning rate of 3×10^{-4} as stated on [4]. We then tune the model to be single-target model. We follow the proposed procedure [4] where one copy of the multi-target model is fine-tuned on single target task with a learning rate of 10^{-4} and max epoch of 50. As a result, six separate models are developed, similar to the BSRNN approach. Subsequently, a comparison is made between the multi-target and single-target models.

D. Performance Metrics

We evaluate the performance of models with two following metrics:

1) uSDR: : uSDR corresponds to the modified utterance-level signal-to-distortion ratio metric, and used as the default evaluation metric in the Music Demixing (MDX) Challenge 2021 and MDX Challenge 2023. This was proposed in [12] and used in [9]. We report the mean across the SDR scores of all test songs.

2) cSDR: : cSDR corresponds to the chunk-level SDR calculated by the standard SDR metric in bss eval metrics [13] and served as the default evaluation metric in the Signal Separation Evaluation Campaign (SiSEC). It was also used on [9]. We use the official implementation which reports the median across the median SDR over all 1 second chunks in each song.

[2] https://github.com/facebookresearch/demucs

TABLE II
COMPARISON OF GUITAR AND PIANO MODEL USING BASS, VOCALS, AND DRUMS CONFIGURATIONS.

Instrument	Guitar		Piano	
	uSDR	cSDR	uSDR	cSDR
Bass	1.33	0.89	1.21	0.60
Vocals	1.83	1.38	1.51	0.78
Drums	**1.85**	**1.72**	**1.73**	**0.99**

TABLE III
COMPARISON OF BSRNN AND HT DEMUCS MODELS.

Instrument	Metric	Model		
		BSRNN	HT Demucs	Fine-tuned HT Demucs
Guitar	uSDR	2.39	2.46	**3.01**
	cSDR	2.38	2.16	**2.70**
Piano	uSDR	2.22	1.73	**2.75**
	cSDR	1.58	0.89	**2.03**
Vocal	uSDR	9.12	8.39	**9.28**
	cSDR	8.15	7.63	**8.72**
Bass	uSDR	9.45	10.37	**11.25**
	cSDR	10.39	10.87	**11.48**
Drum	uSDR	9.93	9.85	**10.41**
	cSDR	9.80	9.90	**10.12**
Other	uSDR	7.04×10^{-6}	0.69	**0.85**
	cSDR	1.34×10^{-7}	0.13	**0.26**
Overall	uSDR	5.52	5.58	**6.26**
	cSDR	5.38	5.26	**5.88**

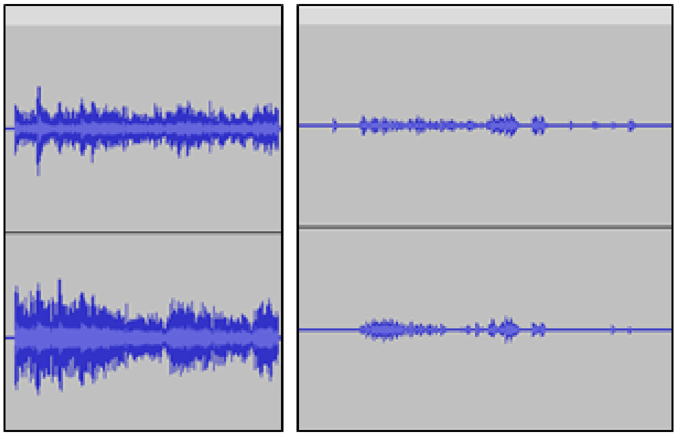

Fig. 3. Results of piano stem separation using the HT Demucs (left) and BSRNN (right).

IV. RESULTS AND DISCUSSION

A. Band-Split RNN (BSRNN)

We find that both guitar and piano achieve their optimal performance when utilizing the drums bandwidth configuration as can be seen in Table II. This phenomenon is attributed to the non-constant frequency ranges inherent in both instruments. Unlike vocals or bass, which typically maintain more constant frequency ranges, the guitar and piano produce a wide array of tones and harmonics. Moreover, drums configuration has more subband generated with its configuration compared to vocals and bass. This is why piano and guitar might be better when using drums configuration.

The lowest scoring stem in both uSDR and cSDR is the other stem. Most likely this is because BSRNN is unable to capture or separate inconsistent sources. The other stem is indeed a mix of rarely appearing sources such as flutes, tambourines, and even sound effects. Additionally, these sounds are often relatively non-dominant compared to main sounds like drums and vocals.

B. HT Demucs

Empirically, we find that there is not much difference between training for 200 epochs compared to 100 epochs, indicating that further training is unnecessary. We achieve an overall score of 5.58 dB in uSDR and 5.26 dB in cSDR when the model is trained for 200 epochs, with only a 0.02 dB difference compared to training for 100 epochs.

We then fine-tune the 200 epochs model on a single-target task. We find that following this pipeline shows a significant improvement by differences of 0.3 dB – 1 dB. We compare this model with the open-source 6 stems HT Demucs and found that the model trained for piano stem achieves a score of 2.75 dB on uSDR and 2.03 dB on cSDR, compared to the pre-trained model, which only achieves 1.76 dB uSDR and 0.72 dB on cSDR. Additionally, this improvement is also observed on other stem. However, the trained HT Demucs model has not yet surpassed the average score of the pre-trained model, with an average uSDR score of 6.48 dB compared to 6.26 dB for the trained model, and a cSDR score of 6.03 dB compared to 5.88 dB for the trained model.

C. Comparison of BSRNN and HT Demucs

We find that HT Demucs single-target (fine-tuned) model has successfully surpassed the average score of the HT Demucs multi-target version model and even the BSRNN model as can be seen in Table III. In both uSDR and cSDR metrics, HT Demucs single-target model demonstrates that it produces relatively clean signals and is consistently clean both in cSDR and uSDR when compared to other models.

We carried out further analysis to see more clearly the quality of the signals produced by BSRNN and HT Demucs. The results on Figure 3 and Figure 4 show that piano stem produced by HT Demucs has more complete signal compared to BSRNN. We also found that both the left-channel and right-channel signals demonstrate that the HT Demucs produces a balanced signal in the guitar source. However, when the bass source is played, the sound produced by HT Demucs still contains noise from the guitar stem. On the other hand, the output from the BSRNN model is cleaner of noise, although the sound is not balanced. This also applies to the vocal stem, where noise from the drums is still present in the output of the HT Demucs. For other stem, no sound is produced by BSRNN, whereas HT Demucs model still produces audible sound for the other stem.

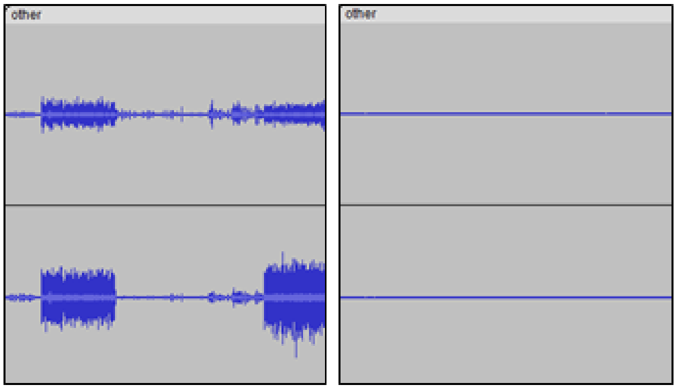

Fig. 4. Results of other stem separation using the HT Demucs (left) and BSRNN (right).

V. Conclusion

In this paper, we investigate the performance of the Band-Split RNN (BSRNN) and Hybrid Transformer Demucs (HT Demucs) models in separating six stem sources. We find that BSRNN model configuration for guitar and piano that produced the best performance was when using the drum configuration. This is because the drum bandsplit configuration is more dispersed and results in more subbands compared to the vocal and bass configurations. It achieved an average uSDR score of 5.526 and a cSDR score of 5.387 across all sources. However, BSRNN model is not yet optimal for capturing sources that do not reside at constant frequencies, such as "other" as evidenced by evaluation scores that are close to zero for this category.

For HT Demucs, The single-target model achieved the best performance with a uSDR score of 6.264 and a cSDR score of 5.888. These results were achieved by training the multi-target model to focus on one specific audio source. The trained single-target HT Demucs model surpassed the performance of the pre-trained HT Demucs model on piano and other audio sources. For the piano source, the trained HT Demucs model outperformed with a difference of 1 dB in uSDR and 1.3 dB in cSDR metrics. For the "other" source, the trained model excelled with a difference of 0.3 dB in uSDR and 0.2 dB in cSDR.

Overall, the HT Demucs model outperformed BSRNN model based on the uSDR and cSDR evaluation metrics, with average scores of 6.26 dB and 5.88 dB, respectively. The analysis also showed that HT Demucs model excelled in producing balanced audio signals in both the left and right channels and was superior in signal completeness. On the other hand, the BSRNN model excelled in producing clearer audio with less noise, although the signals generated were less balanced and, in some cases, incomplete.

References

[1] O. Gillet and G. Richard, "Extraction and remixing of drum tracks from polyphonic music signals," in *2005 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2005, pp. 315–318.

[2] J. Qian, X. Liu, Y. Yu, and W. Li, "Stripe-transformer: deep stripe feature learning for music source separation," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2023, no. 1, p. 2, 2023.

[3] 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.

[4] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation," in *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[5] Z. Rafii, A. Liutkus, and F.-R. Stöter, "The musdb18 corpus for music separation," Zenodo: https: //doi.org/10.5281/zenodo.1117372, 2018.

[6] ——, "Musdb18-hq - an uncompressed version of musdb18," Zenodo: https://doi.org/10.5281/zenodo.3338373, 2019.

[7] E. Manilow, G. Wichern, and J. Le Roux, "Hierarchical musical instrument separation," in *2020 International Society for Music Information Retrieval Conference (ISMIR)*, 2020, pp. 376–383.

[8] I. G. Pereira, F. Araujo, F. Korzeniowski, and R. Vogl, "Moisesdb: A dataset for source separation beyond 4 stems," in *2023 International Society for Music Information Retrieval Conference (ISMIR)*, 2023, pp. 619–626.

[9] Y. Luo and J. Yu, "Music source separation with band-split rnn," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1893–1901, 2023.

[10] R. M. Bittner, J. Wilkins, H. Yip, and J. P. Bello, "Medleydb 2.0: New data and a system for sustainable data collection," *ISMIR Late Breaking and Demo Papers*, vol. 36, 2016.

[11] R. M. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. P. Bello, "Medleydb: A multitrack dataset for annotation-intensive mir research." in *2014 International Society for Music Information Retrieval Conference (ISMIR)*, vol. 14, 2014, pp. 155–160.

[12] Y. Mitsufuji, G. Fabbro, S. Uhlich, F.-R. Stöter, A. Défossez, M. Kim, W. Choi, C.-Y. Yu, and K.-W. Cheuk, "Music demixing challenge 2021," *Frontiers in Signal Processing*, vol. 1, p. 808395, 2022.

[13] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.