UNMIXX: UNTANGLING HIGHLY CORRELATED SINGING VOICES MIXTURES

Jihoo Jung, Ji-Hoon Kim, Doyeop Kwak, Junwon Lee, Juhan Nam, Joon Son Chung

Korea Advanced Institute of Science and Technology, South Korea

ABSTRACT

We introduce UNMIXX, a novel framework for multiple singing voices separation (MSVS). While related to speech separation, MSVS faces unique challenges: data scarcity and the highly correlated nature of singing voices mixture. To address these issues, we propose UNMIXX with three key components: (1) musically informed mixing strategy to construct highly correlated, music-like mixtures, (2) cross-source attention that drives representations of two singers apart via reverse attention, and (3) magnitude penalty loss penalizing erroneously assigned interfering energy. UNMIXX not only addresses data scarcity by simulating realistic training data, but also excels at separating highly correlated mixtures through cross-source interactions at both the architectural and loss levels. Our extensive experiments demonstrate that UNMIXX greatly enhances performance, with SDRi gains exceeding 2.2 dB over prior work.

Index Terms— singing voices separation, reverse attention

1. INTRODUCTION

As vocal layering–a technique that stacks multiple vocal tracks–has become a standard practice in contemporary music production, real-world music predominantly features multiple vocal tracks rather than a single line. However, most works on singing voices, such as singing information retrieval [1] and singing voice synthesis [2], assume a single-vocal setting, limiting their applicability to real-world scenarios. Multiple Singing Voices Separation (MSVS) addresses this gap by disentangling individual vocal tracks from complex mixtures, thereby enabling existing methods on singing voices to extend to real-world multi-vocal music.

MSVS is similar to speech separation in that both aim to separate acoustic sources within the same modality-singing voices and speech, respectively. However, MSVS poses greater challenges for two main reasons. First, suitable training datasets are scarce. As shown in Table 1, unlike the vast speech separation datasets, MSVS datasets contain only about an hour of audio. Second, singing voices exhibit a highly correlated nature. They are often aligned in note onsets and offsets, share harmonic components, contain similar lyrics, and even include segments sung by the same singer.

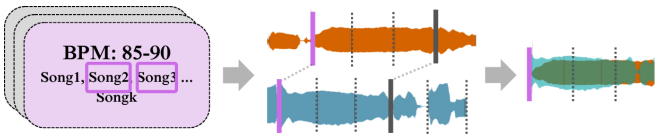
Table 1: Public datasets for speech separation and MSVS.

Category	Corpus/Dataset	Duration (hours)
Speech Separation	WSJ0-2mix [3]	43
	Libri2Mix [4]	292
MSVS (Choral Music)	jaCappella [5]	0.9
	ESMUC Choir [6]	0.5
MSVS (Pop Music)	MedleyVox [7]	1.0

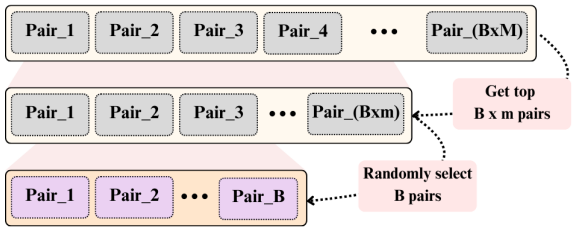
Prior studies on MSVS can fall into two categories. One targets choral music [8–10], separating soprano, alto, tenor, and bass from mixtures with four or more voices. The other focuses on pop music, typically with two singers, which is more practical and relevant to real-world use. However, to our knowledge, MedleyVox [7] is the only existing work addressing pop music separation. While it presents an evaluation dataset and baseline study, it still suffers from the two aforementioned challenges of MSVS. First, to compensate for scarce multi-singer training data, they primarily create synthetic mixtures by randomly mixing two monophonic vocals. This approach, however, struggles to capture the complex correlations of real multi-singer mixtures. Second, they largely rely on speech separation frameworks [11, 12], which are inadequate for disentangling highly correlated mixtures. As a result, remnants of one singer's voice often remain audible in the other's output, a phenomenon referred to as *interference*.

In this paper, we propose **UNMIXX**, a comprehensive framework which mitigates challenges in MSVS with three key components. First, we propose a musically informed mixing strategy that constructs mixtures by combining two songs with strong temporal and harmonic correlation. This produces highly correlated, music-like mixtures resembling real-world multi-singer tracks. Second, we propose cross-source attention, which forces the representations of two singers to diverge via reverse attention [13]. Third, we propose a magnitude penalty loss that explicitly penalizes spectrogram regions contaminated by interference. Both cross-source attention and magnitude penalty loss enforce mutual exclusivity between outputs through cross-source interactions at architectural and loss levels. This reduces interference and yields cleaner outputs, even in highly correlated mixtures. Experiments validate the effectiveness of UNMIXX, demonstrating consistent improvements on both duet and unison subsets of MedleyVox test set. Audio samples are available here[1].

[1]https://unmixx.github.io/



(a) Illustration of temporal alignment. Two songs are randomly sampled from a tempo group, each cropped to a fixed length starting from one of downbeat positions and then mixed. Dotted, solid, and purple lines denote beats, downbeats, and selected downbeats position, respectively.



(b) Illustration of Harmonic alignment. The first row shows $B \times M$ audio pairs sorted in descending order by harmonic overlap score. From the top $B \times m$ pairs, the final B pairs are randomly selected.

Fig. 1: Musically informed mixing process.

2. OVERALL ARCHITECTURE

Our architecture builds upon TIGER [14], a lightweight speech separation model, augmented with cross-source attention module in order to identify and suppress interference. The input mixture is first converted into a time-frequency representation using STFT. The frequency axis is then split into non-uniform sub-bands, with each sub-band projected into a fixed-dimensional space. It is then processed by two key modules–multi-scale selective attention and full-frequency-frame attention (F^3A)–applied first along the frequency dimension and then along the time dimension. In our method, F^3A incorporates both self-attention and cross-source attention. After repeating this interleaved process eight times, the full-band representation is restored to generate a mask for each singer. These masks are then applied to the input mixture to obtain separated waveforms via inverse STFT.

3. UNMIXX

3.1. Musically Informed Mixing (MIM)

We propose musically informed mixing (MIM) to address the scarcity of multi-singer training data. Similar to [15], rather than randomly mixing two monophonic vocals, MIM selects pairs of songs with strong temporal and harmonic correlation to generate highly correlated mixtures. Temporal alignment serves as a global data mining strategy to create mixtures with better synchronized note on/offsets and harmonic alignment as a local strategy to produce harmonically coherent mixtures. Figure 1 illustrates the pipeline.

Temporal Alignment. As part of global data mining, we enhance rhythmic consistency by selecting songs with similar tempi and synchronizing segments at downbeat positions. Specifically, before training, we extract beat and downbeat timings from all songs using a recent beat tracking model [16]. Each song's BPM is estimated from the median inter-beat interval, and songs with similar BPM are grouped together. During training, we perform dynamic mixing by randomly choosing a tempo group, sampling two songs from it, and cropping them to a fixed length. Each segment starts at one of downbeat positions rather than an arbitrary point, and the two segments are then mixed to form a training sample.

Harmonic Alignment. As part of local data mining, we enhance harmonic consistency by constructing each batch exclusively from audio pairs exhibiting strong harmonic correlation. We measure harmonic correlation using harmonic overlap score [10], which quantifies coinciding partials across the first 16 overtones of the two sources. During training, $B \times M$ candidate pairs are first sampled, where B denotes the batch size and M is a multiplicative factor. Harmonic overlap scores are computed for all pairs, sorted in descending order, and the top $B \times m$ candidates ($m < M$) are retained. From these, a final batch of B pairs is randomly sampled. Here, m is a hyperparameter that controls the degree of harmonic alignment, with smaller values enforcing stronger alignment. In our experiments, we set $M = 16$ and $m = 8$.

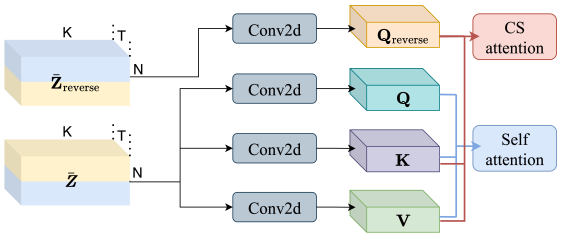


Fig. 2: Cross-source attention mechanism.

3.2. Cross-Source (CS) Attention

To promote divergence between the representations of two singers and mitigate interference, we propose cross-source (CS) attention. CS attention first divides the intermediate representation along the channel dimension into two halves, each corresponding to one singer. Then it suppresses regions exhibiting high similarity between the two representations leveraging reverse attention [13], a variant of cross-attention in which the logits are negated prior to softmax.

We extend the self-attention-based F^3A module by integrating CS attention. Let the input of the F^3A module be $\bar{\boldsymbol{Z}} \in \mathbb{R}^{N \times K \times T}$, where N, K, and T denote the channel dimension, the number of frequency sub-bands, and the number of frames, respectively. As illustrated in Figure 2, $\bar{\boldsymbol{Z}}$ is split along the channel dimension N, and the front and back halves are swapped to construct a reversed input $\bar{\boldsymbol{Z}}_{\text{reverse}} \in \mathbb{R}^{N \times K \times T}$. To compute reverse attention, a 1×1 convolution is applied to $\bar{\boldsymbol{Z}}_{\text{reverse}}$ to obtain $\boldsymbol{Q}_{\text{reverse}} \in \mathbb{R}^{(A \times E) \times K \times T}$, while three separate 1×1 convolutions are applied to $\bar{\boldsymbol{Z}}$ to generate $\boldsymbol{Q} \in \mathbb{R}^{(A \times E) \times K \times T}$,

$\boldsymbol{K} \in \mathbb{R}^{(A\times E)\times K\times T}$ and $\boldsymbol{V} \in \mathbb{R}^{(A\times N/A)\times K\times T}$, where A is the number of attention heads and E the embedding dimension per head. CS attention weights are computed as:

$$\boldsymbol{A}_{\text{cs}} = \text{Softmax}\left(-\frac{\boldsymbol{Q}_{\text{reverse}}\boldsymbol{K}^{\top}}{\sqrt{E\times T}}\right).$$

This formulation down-weights regions of high similarity between the two representations through the negative sign. At the same time, self-attention weights $\boldsymbol{A}_{\text{self}}$ are computed from $\boldsymbol{Q}$ and $\boldsymbol{K}$ without negation. The final output of the F^{3}A module is the average of self- and CS attention: $\boldsymbol{O} = \frac{1}{2}\left(\boldsymbol{A}_{\text{self}}\boldsymbol{V} + \boldsymbol{A}_{\text{cs}}\boldsymbol{V}\right)$. Here, self-attention preserves the internal consistency of each representation, while CS attention drives the two representations apart. With repeated application of F^{3}A, the representations of the two singers gradually learn to capture mutually exclusive information from the mixture, effectively suppressing interference.

3.3. Magnitude Penalty Loss

To further enhance separation quality, we propose a magnitude penalty loss ($\mathcal{L}_{\text{Penalty}}$) that suppresses interference in the predicted magnitude spectrogram. We identify interfering components by comparing each predicted spectrogram with the ground-truth spectrograms of the target and non-target sources. This cross-source constraint enables effective separation even for strongly entangled mixtures.

For each target source i, we first construct a binary interference mask I_i by locating time-frequency bins that satisfy two conditions: (1) the non-target source j's ground-truth magnitude spectrogram $M_{t,f}^{(j)}$ exhibits high energy ($> \tau_{\max}$), and (2) the target source i's ground-truth magnitude spectrogram $M_{t,f}^{(i)}$ exhibits low energy ($< \tau_{\min}$). Intuitively, I_i captures regions that are strongly present in the non-target source but absent in the target source, and thus should not appear in the estimated magnitude spectrogram $\hat{M}^{(i)}$. The magnitude penalty loss is then computed by multiplying this mask I_i with the estimated magnitude spectrogram $\hat{M}^{(i)}$ and normalizing by the number of interfering bins. In this way, magnitude penalty loss explicitly penalizes $\hat{M}^{(i)}$ by capturing undesired energy. This can be formulated as:

$$\mathcal{L}_{\text{Penalty}} = \sum_{i=1}^{2} \mathbb{E}_{\hat{M}^{(i)}}\left[\frac{\|\hat{M}^{(i)} \odot I_i\|_2^2}{\|I_i\|_1 + \epsilon}\right],$$

$$I_i(t,f) = \begin{cases} 1\,, & \text{if } M_{t,f}^{(j)} > \tau_{\max} \text{ and } M_{t,f}^{(i)} < \tau_{\min}, \\ 0\,, & \text{otherwise.} \end{cases}$$

We combine the proposed magnitude penalty loss with the conventional signal-to-noise ratio (SNR) loss ($\mathcal{L}_{\text{SNR}}$) and a magnitude loss ($\mathcal{L}_{\text{Mag}}$). Magnitude loss is defined as the L2 distance between the ground-truth and estimated magnitude spectrograms. The overall training objective is given by

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{SNR}} + \lambda_{\text{mag}} \cdot \mathcal{L}_{\text{Mag}} + \lambda_{\text{penalty}} \cdot \mathcal{L}_{\text{Penalty}},$$

where λ_{mag} and λ_{penalty} are non-negative weights balancing the magnitude and penalty losses. We set $\tau_{\max} = 1.0$, $\tau_{\min} = 0.5$, $\lambda_{\text{mag}} = 0.1$, and $\lambda_{\text{penalty}} = 0.02$, with the penalty loss applied after half of the training process.

4. EXPERIMENTAL SETUP

4.1. Datasets and Training Details

We train on around 400 hours of audio from 9 monophonic singing datasets [17–25]. For evaluation, we use the *unison* and *duet* subsets of MedleyVox evaluation dataset. The *unison* subset consists of mixtures of two voices with identical or octave-shifted melodies, same note on/offsets and lyrics, while the *duet* subset contains mixtures differing in melodies, note on/offsets, or lyrics. Both subsets contain mixtures of either two different singers or two parts sung by the same singer. Notably, 55% of duet mixtures and 97% of unison mixtures involve multiple parts sung by the same singer. All audio is resampled to 24 kHz. We compute STFTs with a 960-sample window, 240-sample hop, and 960-point FFT, and apply power-law compression to magnitude spectrograms to reduce dynamic range. Models are trained with a batch size of 8 using Adam optimizer at a 0.001 learning rate decayed by validation performance. Training runs for up to 500k steps, with early stopping after 60k steps of no improvement.

4.2. Evaluation Metrics

Following prior work [7], we use SDRi and SI-SDRi as evaluation metrics. However, since these metrics often underestimate quality in same-singer mixtures, we propose permutation-invariant segmental SNR (PSSNR) and its hybrid variant HSSNR as auxiliary metrics tailored for MSVS.

SDRi and SI-SDRi often yield misleadingly low scores for same-singer mixtures, as they heavily penalize singer assignment changes. Such cases arise when the model swaps singers (e.g., S_1 and S_2) across segments ($S_1S_2S_1$ and $S_2S_1S_2$) instead of producing consistent outputs ($S_1S_1S_1$ and $S_2S_2S_2$). While such penalties are appropriate in speech separation with distinct speakers, they are unnecessary in songs when a single singer performs multiple parts.

To address this, we propose PSSNR, which is identical to SSNR except that it recomputes the optimal permutation for each segment. PSSNR focuses solely on separation quality at segment level, ignoring assignment consistency across segments. To provide a unified score across different- and same-singer cases, we define HSSNR as the average of SSNR for different-singer cases and PSSNR for same-singer cases. In this way, HSSNR reflects necessary penalties from singer assignment changes while excluding redundant ones.

Table 2 presents the limitations of SDRi and SI-SDRi under singer assignment changes on the same-singer unison subset, while PSSNR mitigates them. We simulate assignment changes by randomly swapping segments between two

Table 2: Metric values on the unison subset in the same-singer case, obtained by swapping ground-truth signals at specified ratio.

Swap ratio(%)	SDRi (↑)	SI-SDRi (↑)	SSNR (↑)	PSSNR (↑)
10	7.36	6.95	31.04	34.74
20	3.74	2.95	26.86	34.74
30	1.02	-0.07	22.74	34.75
40	-0.52	-1.96	17.65	34.75
50	-1.04	-2.41	15.83	34.74

Table 3: Separation performance on duet and unison subsets of MedleyVox evaluation dataset. TIGER* denotes the TIGER trained on the same training data as used in the MedleyVox experiments.

Method	#params	Duet			Unison		
		SDRi	SI-SDRi	HSSNR	SDRi	SI-SDRi	HSSNR
MedleyVox	5M	15.10	14.20	13.33	4.90	4.40	7.65
TIGER*	947k	16.58	15.52	15.14	5.96	5.31	9.86
UNMIXX	951k	**17.52**	**16.47**	**15.96**	**7.16**	**6.58**	**10.50**

ground-truth signals at specified ratios. This induces assignment changes but is perceptually natural, as both parts are sung by the same singer and separation remains perfect. Nevertheless, SDRi and SI-SDRi drop sharply; beyond 30% swapping, SI-SDRi even becomes negative, implying that input mixture scores higher than the perfectly separated signals. SSNR shows a similar trend, whereas PSSNR remains stable. This shows that, unlike SDRi, SI-SDRi, and SSNR–which rely on a fixed global permutation–PSSNR removes spurious penalties and thus more reliable in same-singer mixtures.

5. EXPERIMENTAL RESULTS

5.1. Quality Comparison

Table 3 shows the performance of UNMIXX alongside two baselines: MedleyVox [7] and TIGER [14] trained on the same datasets as MedleyVox–i.e., our training dataset plus the speech dataset [4]. Across all metrics, UNMIXX consistently outperforms the baselines. Against MedleyVox, UNMIXX delivers substantial SDRi gains of +2.42 dB (duet) and +2.26 dB (unison). Compared to TIGER, it also achieves marked gains of +0.94 dB (duet) and +1.20 dB (unison), with only a marginal parameter increase. Moreover, UNMIXX shows clear improvements in HSSNR, confirming that such substantial SDRi and SI-SDRi gains result from genuine separation quality rather than favorable singer assignments.

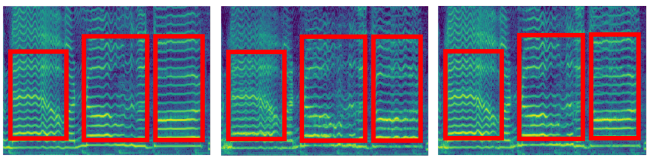
5.2. Ablation Studies

We verify the effectiveness of each component of UNMIXX through ablation studies, as reported in Table 4. Note that each component is individually added to the TIGER baseline.

The second block evaluates MIM. Excluding speech mixture from training data improves separation quality in both subsets. Building on this, temporal alignment and weak harmonic alignment (m=12) further enhance performance. Stronger harmonic alignment (smaller m) boosts unison but degrades duet performance. This is because excessive alignment produces overly correlated samples and reduces training data diversity, which can harm duet performance as they are inherently less correlated than unison.

Table 4: Ablation studies on the duet and unison subsets of MedleyVox evaluation dataset. Each proposed component was added to the baseline individually. Underline highlights the best score within each block. **Bold** marks the overall best result across the entire table.

	Method	Duet			Unison		
		SDRi	SI-SDRi	HSSNR	SDRi	SI-SDRi	HSSNR
(1)	TIGER*	16.58	15.52	15.14	5.96	5.31	9.86
(2)	- Speech dataset	16.57	15.46	15.71	6.54	5.90	9.89
	+ MIM (m=12)	17.11	16.05	15.43	7.03	6.43	10.06
	+ MIM (m=8)	16.79	15.75	15.83	**7.31**	**6.68**	**10.72**
	+ MIM (m=4)	16.09	14.96	14.58	7.12	6.48	9.50
(3)	+ CS attention	**18.01**	**17.00**	**16.02**	6.17	5.54	10.06
(4)	+ Mag loss	16.66	15.60	15.63	6.26	5.71	9.29
	+ Mag, Penalty loss	16.68	15.61	15.50	6.44	5.83	9.89



(a) Ground truth (b) +Mag loss (c) +Mag, Penalty loss

Fig. 3: Comparison of spectrograms using different objectives.

The third block evaluates CS attention, which yields consistent gains on both subsets, with particularly large improvements on duet subset. The fourth block examines magnitude-based objectives. Incorporating magnitude loss improves performance, and adding the magnitude penalty loss provides further gains–modest in duet but more pronounced in unison. This is because the duet subset is relatively easy to separate, leaving fewer regions for the penalty to act on, whereas the more challenging unison subset benefits more from such suppression. HSSNR shows a notable 0.51 dB increase in unison, highlighting the effectiveness of the magnitude penalty loss in eliminating residual interference and enhancing separation fidelity. We also visualize the output spectrograms to further demonstrate the effectiveness of the magnitude penalty loss. As shown in Fig. 3, adding the magnitude penalty loss yields clean spectrograms close to the ground truth, whereas the magnitude loss alone results in noisy spectrograms.

6. CONCLUSION

We propose UNMIXX, a comprehensive MSVS framework for separating individual vocal tracks from complex mixtures. We address the inherent challenges of MSVS with three key components–musically informed mixing, cross-source attention, and magnitude penalty loss–achieving notable gains over prior work. We verify the effectiveness of each component through extensive ablation studies with newly proposed evaluation metrics–HSSNR–tailored for MSVS.

7. ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) through the Korea Government (MSIT) under Grant RS-2023-00222383.

8. REFERENCES

[1] Ruiqi Li, Yu Zhang, Yongqi Wang, Zhiqing Hong, Rongjie Huang, and Zhou Zhao, "Robust singing voice transcription serves synthesis," in *Proc. of ACL*, 2024.

[2] Yu Zhang, Ziyue Jiang, Ruiqi Li, Changhao Pan, Jinzheng He, Rongjie Huang, Chuxin Wang, and Zhou Zhao, "TCSinger: Zero-shot singing voice synthesis with style transfer and multi-level style control," 2024.

[3] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016.

[4] Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent, "Librimix: An open-source dataset for generalizable speech separation," *arXiv*, 2020.

[5] Tomohiko Nakamura, Shinnosuke Takamichi, Naoko Tanji, Satoru Fukayama, and Hiroshi Saruwatari, "jacappella corpus: A japanese a cappella vocal ensemble corpus," in *Proc. ICASSP*, 2023.

[6] Helena Cuesta, "Data-driven pitch content description of choral singing recordings," 2022.

[7] Chang-Bin Jeon, Hyeongi Moon, Keunwoo Choi, Ben Sangbae Chon, and Kyogu Lee, "Medleyvox: An evaluation dataset for multiple singing voices separation," in *Proc. ICASSP*, 2023.

[8] Darius Petermann, Pritish Chandna, Helena Cuesta, Jordi Bonada, and Emilia Gómez, "Deep learning based source separation applied to choir ensembles," in *Proc. of ISMIR*, 2020.

[9] Matan Gover and Philippe Depalle, "Score-informed source separation of choral music," in *Proc. of ISMIR*, 2020.

[10] Saurjya Sarkar, Emmanouil Benetos, and Mark Sandler, "Vocal harmony separation using time-domain neural networks," in *Proc. Interspeech*, 2021.

[11] Yi Luo and Nima Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[12] Joel Rixen and Matthias Renz, "Sfsrnet: Super-resolution for single-channel audio source separation," in *Proc. AAAI*, 2022.

[13] Ruijie Tao, Xinyuan Qian, Yidi Jiang, Junjie Li, Jiadong Wang, and Haizhou Li, "Audio-visual target speaker extraction with reverse selective auditory attention," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 33, pp. 797–811, 2025.

[14] Mohan Xu, Kai Li, Guo Chen, and Xiaolin Hu, "Tiger: Time-frequency interleaved gain extraction and reconstruction for efficient speech separation," in *Proc. ICLR*, 2025.

[15] Yigitcan Özer and Meinard Müller, "Source separation of piano concertos using musically motivated augmentation techniques," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 32, pp. 1214–1225, 2024.

[16] Francesco Foscarin, Jan Schlüter, and Gerhard Widmer, "Beat this! accurate beat tracking without dbn postprocessing," in *Proc. of ISMIR*, 2024.

[17] Soonbeom Choi, Wonil Kim, Saebyul Park, Sangeon Yong, and Juhan Nam, "Children's song dataset for singing voice research," in *Proc. of ISMIR*, 2020.

[18] Zhiyan Duan, Haotian Fang, Bo Li, Khe Chai Sim, and Ye Wang, "The NUS sung and spoken lyrics corpus: A quantitative comparison of singing and speech," in *Proc. of APSIPA*, 2013.

[19] Julia Wilkins, Prem Seetharaman, Alison Wahl, and Bryan Pardo, "Vocalset: A singing voice dataset.," in *Proc. of ISMIR*, 2018.

[20] Shinnosuke Takamichi, Shota Ikawa, Naoko Tanji, and Hiroshi Saruwatari, "Jsut-song," 2022.

[21] Hiroki Tamaru, Shinnosuke Takamichi, Naoko Tanji, and Hiroshi Saruwatari, "Jvs-music: Japanese multispeaker singing-voice corpus," *arXiv*, 2020.

[22] Kilian Schulze-Forster, Clement SJ Doire, Gaël Richard, and Roland Badeau, "Phoneme level lyrics alignment and text-informed singing voice separation," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 29, pp. 2382–2395, 2021.

[23] AIHub, "Multi-timbre guide vocal dataset," aihub.or.kr, 2022.

[24] AIHub, "Multi-singer vocal dataset," aihub.or.kr, 2022.

[25] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner, "Musdb18-hq-an uncompressed version of musdb18," 2019.