MEL-ROFORMER FOR VOCAL SEPARATION AND VOCAL MELODY TRANSCRIPTION

Ju-Chiang Wang   Wei-Tsung Lu   Jitong Chen
ByteDance, San Jose, CA, USA
ju-chiang.wang@bytedance.com

ABSTRACT

Developing a versatile deep neural network to model music audio is crucial in MIR. This task is challenging due to the intricate spectral variations inherent in music signals, which convey melody, harmonics, and timbres of diverse instruments. In this paper, we introduce Mel-RoFormer, a spectrogram-based model featuring two key designs: a novel Mel-band Projection module at the front-end to enhance the model's capability to capture informative features across multiple frequency bands, and interleaved RoPE Transformers to explicitly model the frequency and time dimensions as two separate sequences. We apply Mel-RoFormer to tackle two essential MIR tasks: vocal separation and vocal melody transcription, aimed at isolating singing voices from audio mixtures and transcribing their lead melodies, respectively. Despite their shared focus on singing signals, these tasks possess distinct optimization objectives. Instead of training a unified model, we adopt a two-step approach. Initially, we train a vocal separation model, which subsequently serves as a foundation model for fine-tuning for vocal melody transcription. Through extensive experiments conducted on benchmark datasets, we showcase that our models achieve state-of-the-art performance in both vocal separation and melody transcription tasks, underscoring the efficacy and versatility of Mel-RoFormer in modeling complex music audio signals.

1. INTRODUCTION

Modeling musical audio signals with deep neural networks (DNNs) for MIR tasks has emerged as a vibrant and promising area of research [1–5]. Most of such DNN models are built upon the spectrogram, a fundamental frequency-time representation of audio signals. Traditional approaches typically treat the spectrogram as a sequence of spectra over time, with the frequency axis representing the feature dimension. However, recent advancements have encompassed explicit modeling of the frequency dimension as a sequence in their architecture designs [5–9], recognizing its rich semantic information in music audio signals, including melody, harmonics, and instrument timbres. These architectures have showcased state-of-the-art performance in various MIR tasks such as vocal melody extraction [5], section segmentation [10], instrument transcription [8], and music source separation [11], leveraging the model's ability to discern spectral patterns effectively.

The Transformer architecture [12] has demonstrated remarkable efficacy not only in Natural Language Processing but also in various MIR tasks, where it excels at modeling sequences to predict high-level musical semantics such as tags, beats, chords, sections, and notes [5, 8–10, 13–15]. However, its potential to accurately predict low-level audio signals remained uncertain. Lu et al. proposed a novel architecture, called BS-RoFormer [11], to tackle the task of music source separation (MSS), which aims to separate audio recordings into musically distinct sources such as vocals, bass, and drums [16, 17]. Inspired by the Band-Split RNN (BSRNN) model [7], BS-RoFormer adopts the interleaved sequence modeling, treating time and frequency dimensions as two separate sequences. Notably, it replaces Recurrent Neural Networks (RNNs) with Transformer encoders, demonstrating exceptional performance. This was evident in its first-place ranking and substantial margin of performance gain over the runner-up in the Music Separation track of the Sound Demixing Challenge 2023 (SDX'23) [18].

Another key attribute contributing to the success of BS-RoFormer is the band-split module at the front-end. Traditional Transformer-based models typically rely on a Convolutional Neural Network (CNN) front-end to extract features from the spectrogram for the succeeding Transformer blocks (e.g., [5, 13, 19]). However, CNNs are not inherently designed to model two spectral events that are far apart in frequency, which could limit the model to characterize detailed spectral patterns. In contrast, the band-split module divides the frequency dimension into a number of subbands and employs multi-layer perceptrons (MLPs) to directly project the raw subband spectrograms into a sequence of band-wise features for the succeeding Transformer to model it along the frequency axis. From another perspective, the band-split mechanism can be seen as a set of learnable band-pass filters, underscoring the importance of designing an effective band-division scheme.

In this paper, we introduce *Mel-RoFormer*, which is a successor of BS-RoFormer with an enhanced band-division scheme that leverages the Mel-scale [20] to improve the model's generalization ability. The Mel-scale is

Attribution: J.-C. Wang, W.-T. Lu, and J. Chen, "Mel-RoFormer for Vocal Separation and Vocal Melody Transcription", in *Proc. of the 25th Int. Society for Music Information Retrieval Conf.,* San Francisco, United States, 2024.

engineered to mimic the non-linear perception of sound by the human ear, exhibiting higher discrimination at lower frequencies and reduced discrimination at higher frequencies. It has a long history as a reference for designing acoustic features [21] such as MFCC and mel-spectrogram in audio signal processing. We develop the Mel-band mapping based on the Mel-scale, resulting in a band-division that generates overlapping subbands in frequency. In contrast, the band-split scheme in BS-RoFormer is defined empirically and produces non-overlapping subbands.

Mel-RoFormer is applied to address two fundamental MIR tasks: vocal separation and vocal melody transcription, which involve isolating singing voices from audio mixtures and transcribing their lead melodies, respectively. Mel-RoFormer demonstrates superior performance compared to BS-RoFormer and other MSS models in experiments. For vocal melody transcription, we propose a two-step approach instead of training a unified model. We first pretrain a vocal separation model and then fine-tune it for vocal melody transcription. The resulting model achieves state-of-the-art performance across all metrics and exhibits strong robustness in detecting note offsets, which is considered to be the most challenging aspect of the task [22].

Readers can refer to the open-sourced implementation [1] and configurations [2] for Mel-RoFormer and its variants.

2. RELATED WORK

To address the capabilities of DNN models that pay attention to modeling the frequency dimension for MIR tasks, SpecTNT is one of early successful attempts. Central to SpecTNT's architecture is the TNT block, where two Transformer encoders are strategically arranged to model along both frequency and time axes. A novel concept introduced by SpecTNT is the Frequency Class Token (FCT), which serves to bridge of the two Transformers, enabling the interchangeability of embeddings across both axes within the TNT block. However, SpecTNT relies on CNNs at the front-end, and the FCT is obtained through aggregating features from the frequency sequence, potentially leading to information loss. Nonetheless, SpecTNT has showcased remarkable performance across various MIR tasks such as beat tracking [14], chord recognition [5], structure segmentation [10], and vocal melody estimation [5]. Its successor, Perceiver TF [8], is designed to enhance efficiency while demonstrating outstanding performance in multitrack instrument/vocal transcription tasks.

Moving on to MSS, a key MIR task that has significantly benefited from DNNs, approaches typically span frequency-domain and time-domain methodologies. The benchmark MUSDB18 dataset [23] offers 4-stem sources including vocals, bass, drums, and others, adhering to the definition established by the 2015 Signal Separation Evaluation Campaign (SiSEC) [24]. Frequency-domain approaches rely on spectrogram-based representations as input, leveraging models such as fully connected neural networks [25], CNNs [26–28], and RNNs [29] to achieve separation. Conversely, time-domain approaches such as Wave-U-Net [30], ConvTasNet [31], and Demucs [32] construct their DNNs directly on waveform inputs. Recently, Hybrid Transformer Demucs (HTDemucs) [33] has proposed a novel approach, utilizing a cross-domain Transformer to amalgamate both frequency- and time-domain models, showcasing promising potential in this field. However, none of the mentioned approaches employ Transformers to model the inter-context of frequency and time as two separate sequences.

The output of vocal melody transcription is a sequence of non-overlapping notes, each comprising onset and offset times along with a pitch key, assuming the melody is monophonic. Due to limited training data availability, only a few studies have focused on transcribing note-level outputs from polyphonic music audio, underscoring the significance of pre-training [34] or semi-supervised [35] techniques. Recently, Wang et al. released a human-annotated dataset comprising 500 Chinese songs [36], along with a baseline CNN-based model. Donahue et al. [34] propose leveraging pre-trained representations from Jukebox [37] to enhance melody transcription, primarily focusing on lead instruments such as synthesizers, guitars, piano, and vocals. They curate a dataset of 50 hours of melody transcriptions sourced from crowdsourced annotations. However, clarity regarding the quality and identification of vocal melody annotations within songs remains lacking. In [35], a teacher-student training scheme is introduced to leverage pseudo labels derived from fundamental frequency (F0) estimations of vocals. On the other hand, [38] presents a system that necessitates a vocal separation as a front-end. In our approach, we employ vocal separation as a pre-trained model for fine-tuning a specialized model, which can be more efficient and task-optimal.

3. MODEL

Figure 1 illustrates the diagram of Mel-RoFormer, consisting of three major modules: Mel-band Projection, RoFormer blocks, and Embedding Projection. Subsequent subsections will delve into the Mel-band Projection and Embedding Projection modules, while readers can find further details on the RoFormer blocks in [11], where they are referred to as "RoPE Transformer blocks."

Mel-RoFormer takes the input of a complex spectrogram X with dimensions $(C \times F \times T)$, where C, F, and T denote the number of channels, frequency bins, and time steps, respectively. This frequency-time representation X is typically obtained via a short-time Fourier transform (STFT), encompassing both real and imaginary parts. In stereo mode, C is defined as $2 \times 2 = 4$, reflecting the presence of two channels for real and imaginary spectrograms. The output of Mel-RoFormer is denoted by Y, with dimensions $(Z \times T)$, where Z and T represent the number of output features and time steps, respectively. One can treat Y as the feature matrix over time. Depending on the downstream tasks, appropriate values for Z can be set, and Y can be rearranged accordingly, as detailed in Section 4.

[1] https://github.com/lucidrains/BS-RoFormer

[2] https://github.com/ZFTurbo/Music-Source-Separation-Training

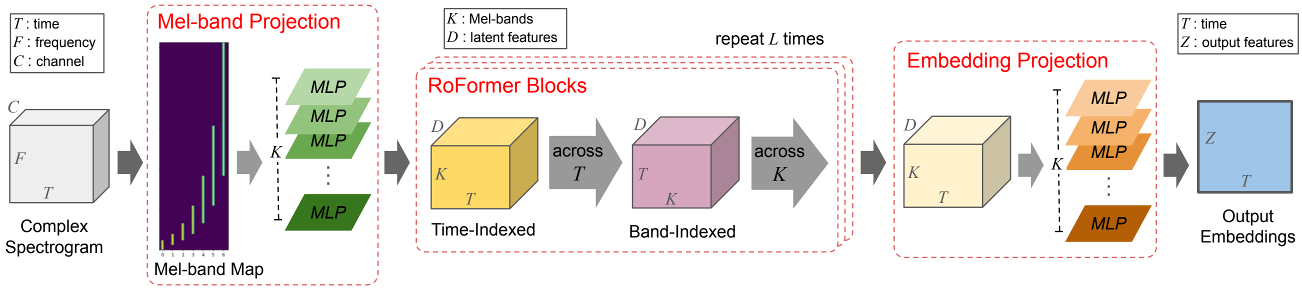

Figure 1. The diagram of Mel-RoFormer, which is consist of three major modules: Mel-band Projection, RoFormer Blocks, and Embedding Projection. The input is a Complex Spectrogram, and the output is an Embedding tensor, which can be rearranged into the desired shape.

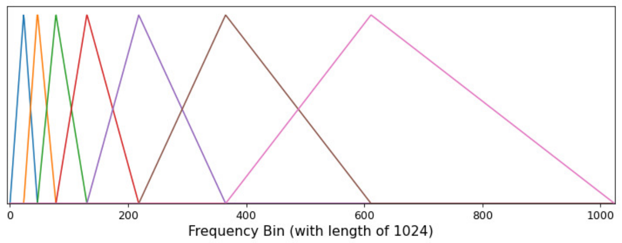

Figure 2. Illustration of Mel filter-bank with 7 bands. In this example, the length of frequency bins is 1024. Here, the frequency bins from 1 to 46 are encompassed by the 0-th Mel-band (i.e. $\mathcal{F}_0$), those from 24 to 77 are encompassed by the 1-th Mel-band (i.e. $\mathcal{F}_1$), and so forth.

3.1 Mel-band Projection Module

The Mel-band Projection module comprises a frequency-to-Mel-band mapping and a set of K multi-layer perceptrons (MLPs). Each subband of X is denoted as X_k with dimensions $(C \times |\mathcal{F}_k| \times T)$, where $\mathcal{F}_k \in \{0, 1, \ldots, F-1\}$ represents the indices of frequency bins for the k-th subband, and F is the length of frequency bins.

The frequency-to-Mel-band mapping, represented as $\{\mathcal{F}_k\}_{k=0}^{K-1}$, stems from the Mel filter-bank, as depicted in Fig. 2, where triangular-shaped filters are centered at different Mel frequencies on the Mel-scale. The indices of non-zero values of a filter correspond to the frequency bins of the respective Mel-band. Mathematically, the Mel-scale follows a quasi-logarithmic function of acoustic frequency, ensuring that perceptually similar pitch intervals (e.g., octaves) possess equal width across the entire audible range. The width of a Mel-band (i.e., $|\mathcal{F}_k|$) is twice the distance between its center and the center of the preceding Mel-band. Consequently, the latter half of a Mel-band overlaps with its subsequent Mel-band, and so forth, until reaching the final Mel-band. In contrast, due to the band-split design in BS-RoFormer [11], the frequency ranges of different subbands do not overlap. The Mel-band Map depicted in Fig. 1 exemplifies the frequency-to-Mel-band mapping, illustrating a binary relationship between 1024 frequency bins and 7 Mel-bands.

The input X_k is rearranged into a shape of $(C|F_k| \times T)$ for the MLP layer. The k-th MLP, denoted as Λ_k, comprises an RMSNorm layer [39] followed by a linear layer. The linear layer transforms from $C|F_k|$ dimensions to D dimensions, where D is the number of latent features. The resulting outputs $\{\Lambda_k(X_k)\}_{k=0}^{K-1}$ are stacked to form a shape of $(D \times K \times T)$, serving as input to the subsequent RoFormer blocks. The Mel-band Projection module can be conceptualized as a learnable Mel filter-bank, with the MLP layer functioning as the mechanism to learn the filters. This grants the model greater flexibility in determining the optimal shapes for different filters, without being confined to predefined filter designs such as triangle-shaped ones (see Fig. 2).

3.2 RoFormer Blocks

The RoForMer blocks consist of a stack of L interleaved RoPE Transformer encoders [40]. The interleaved sequence modeling processes the data across time (T) and subband (K) dimensions alternately. In BS-RoFormer [11], the authors observed that Rotary Position Encoding (RoPE) played a crucial role in enhancing Transformer performance compared to using traditional absolute position encoding. It is suggested that RoPE aids in preserving the positional information within the sequence, making it invariant to repetitive processes of rearrangement.

Specifically, the data is first rearranged into a time-indexed shape of $(DK \times T)$, allowing modeling across time. Subsequently, it is rearranged into a band-indexed shape of $(DT \times K)$, facilitating modeling across subbands. The former step treats the data as a time sequence, while the latter treats it as a subband sequence. By repeating this process, information from different time steps and subbands becomes interchangeable, thereby enhancing the model's ability to generalize.

3.3 Embedding Projection Module

The Embedding Projection module plays a crucial role in generating suitable embeddings necessary for various downstream tasks. It has been observed that utilizing MLPs can lead to more effective mask estimation compared to using plain linear layers in source separation tasks [41]. Our preliminary investigation also suggests that removing this module can result in unstable training.

This module comprises K individual MLPs, denoted as Φ_k, each containing an RMSNorm layer, a linear layer followed by a Tanh activation, and another linear layer followed by a gated linear unit (GLU) layer [42]. The first linear layer transforms from D to $4D$ dimensions, while the subsequent linear layer with GLU transforms from $4D$ to a desired length Z_k. Here, Z_k specifies the number of output features for Φ_k based on its specific purpose, as elaborated in Section 4. All the MLP outputs are concatenated along the feature dimension, resulting in a final output shape of $(Z \times T)$, where $Z = \sum_k Z_k$.

4. DOWNSTREAM TASKS

This section details the application of Mel-RoFormer tailored for vocal separation and vocal melody transcription.

4.1 Vocal Separation

For vocal separation, the Embedding Projection estimates the mask for the complex spectrogram. Each MLP Φ_k is designed to align its output shape with that of the corresponding input Mel-band complex spectrogram, with Z_k set as $C|\mathcal{F}_k|$. The output Y is rearranged into

$$\left[\hat{\Phi}_0, \hat{\Phi}_1, \ldots, \hat{\Phi}_{(K-1)}\right], \tag{1}$$

with dimensions $(C \times \hat{Z} \times T)$, where $\hat{\Phi}_k$ with a shape of $(|\mathcal{F}_k| \times T)$ is the output corresponding to the k-th Mel-band, and $\hat{Z} = \sum_k |\mathcal{F}_k|$. Because adjacent Mel-bands overlap, the estimated mask values of the overlapping frequency bins are averaged:

$$\hat{M}[c, f, t] = \frac{1}{S_f} \sum_k \hat{\Phi}_k[c, f, t], \tag{2}$$

where $\hat{M}$ represents the estimated mask, c, f, and t are the indices of channel, frequency bin, and time step, respectively, and S_f is the count of the overlapping frequency bins. The estimated mask $\hat{M}$ has the same shape $(C \times F \times T)$ as that of the input X, encompassing both the real and imaginary parts of the complex spectrogram.

We utilize complex Ideal Ratio Masks (cIRMs) [43] as our optimization goal for the vocal separation model. The estimated mask $\hat{M}$ derived from Embedding Projection can serve as the cIRMs. The separated complex spectrogram $\hat{Y}$ is obtained by element-wise multiplication of the cIRM with the input complex spectrogram: $\hat{Y} = \hat{M} \odot X$. Subsequently, an inverse STFT (iSTFT) is applied to $\hat{Y}$ to reconstruct the separated signal $\hat{y}$ in the time-domain.

Let ψ denote the target time-domain signal, and $\Psi^{(w,r)}$ denote the corresponding complex Spectrogram using a window size w and time-resolution r for STFT. We employ the mean absolute error (MAE) loss to train the cIRMs $\hat{M}$. Specifically, the objective loss encompasses both the time-domain MAE and the multi-resolution complex spectrogram MAE [44]:

$$\mathcal{L} = ||\psi - \hat{y}|| + \sum_{w \in W, r \in R} ||\Psi^{(w,r)} - \hat{Y}^{(w,r)}||, \tag{3}$$

where the configurations for multi-resolution STFTs cover 5 window sizes with $W = \{4096, 2048, 1024, 512, 256\}$, and 2 resolutions with $R = \{100, 300\}$ frames per second.

4.2 Vocal Melody Transcription

Instead of starting training from scratch, we utilize a pre-trained vocal separation Mel-RoFormer and fine-tune it for vocal melody transcription. Given that the Embedding Projection module in the pre-trained vocal separation model functions as a mask estimator, it might inherently possess biases towards signal-level semantics. Therefore, we opt to replace the pre-trained Embedding Projection with a newly initialized one, with a modification specifically on the output dimension. To ensure equal contribution from each Mel-band to the feature dimension, we set a uniform value of 64 for all Z_k's. The resulting embeddings take the shape of $(64K \times T)$, with a $64K$-dimensional feature vector for each time step.

We adopt the "onsets and frames" approach [45], employing two frame-wise predictors: an onset predictor and a frame predictor, both of which receive embeddings from Mel-RoFormer. The onset predictor identifies the onset event of a pitched note, while the frame predictor determines the continuation of a pitched note. This design facilitates a post-processing method where the initiation of a new note is determined only if the onset predictor indicates the start of a pitch, and simultaneously, the frame predictor confirms the presence of an onset for that pitch within the succeeding frames. The onset and frame predictors operate at a time-resolution of 50 frames per second. In cases where the Mel-RoFormer embeddings do not match this time-resolution, we employ 1-D adaptive average pooling over the time dimension.

We employ an MLP layer for the onset predictor, consisting of a linear layer followed by a Rectified Linear Unit (ReLU), dropout with a rate of 0.5, and another linear layer. The linear layer has 512 hidden channels, and the output dimension is set to 60, representing 60 supported pitches. For the frame predictor, a single linear layer is utilized, outputting 61 pitch classes, with one indicating non-pitch. Binary cross-entropy losses of the two predictors are summed as the final loss to train the entire model. The thresholds for the onset and frame predictors are set at 0.45 and 0.25, respectively.

5. EXPERIMENT

Our experiments cover vocal separation and vocal melody transcription. In the vocal separation evaluation, we train and test two types of models: the first on 44.1kHz stereo audio recordings, and the second on 24kHz mono audio recordings. Next, we utilize the model trained on 24kHz mono audio recordings as the pre-trained model for fine-tuning in the vocal melody transcription evaluation.

5.1 Datasets

Table 1 overviews the datasets used in this study. We use four public datasets for evaluation. The data splitting ad-

Dataset	Task	Songs	Split
MUSDB18HQ [23]	sepa	150	train 100, test 50
MoisesDB [46]	sepa	240	train 200, val 40
MIR-ST500 [36]	trans	500	train 330, val 37, test 98
POP909 [47]	trans	909	train 750, val 50, test 109
In-House	sepa	1533	train 1433, val 100

Table 1. Summary of the datasets used in this study. Abbreviations: ‘sepa’: vocal separation, ‘trans’: vocal melody transcription, ’val’: validation.

Model	Vocals	# Param
HDemucs [52]†	8.04	-
Sparse HT Demucs [33]†	9.37	-
BSRNN [7]†	10.01	-
TFC-TDF-UNet-V3 [53]†	9.59	-
BS-RoFormer ⓐ	11.49	93.4M
Mel-RoFormer ⓐ	12.08	105M
BS-RoFormer ⓑ†	12.82	93.4M
Mel-RoFormer ⓑ†	**13.29**	105M
Tested with resampled 24kHz mono audio		
BS-RoFormer (24k-small) ⓒ	10.56	8.0M
Mel-RoFormer (24k-small) ⓒ	11.01	9.1M
BS-RoFormer (24k-large) ⓒ	12.19	48.4M
Mel-RoFormer (24k-large) ⓒ	**12.69**	50.7M

Symbol † indicates models trained with extra data.
Symbols ⓐ, ⓑ, and ⓒ indicate the three evaluation scenarios.

Table 2. Result (in SDR) on MUSDB18HQ test set.

heres to the official guidelines of each dataset, except for POP909, where songs with IDs ranging from 801 to 909 are reserved for testing. The ‘Split’ column of Table 1 indicates the numbers of songs allocated for training, validation, and testing. All data for separation tasks are stereo recordings with a sampling rate of 44.1kHz, and stem-level recordings are pre-mixed into four stems: vocals, bass, drums, and other. The audio of transcription data was resampled to mono with a 24kHz sampling rate to follow the conventional setting [38]. Although access to some songs in MIR-ST500 was restricted, our test set, comprising 98 songs, closely resembles the original setting of 100 songs.

5.2 Configuration for Vocal Separation

To obtain the frequency-to-Mel-band mapping, we employ the Mel filter-bank implementation in librosa [48], which emulates the behavior of the function in MATLAB Auditory Toolbox [49]. By using `librosa.filters.mel`, we acquire the mapping matrix comprising a triangle filter for each Mel-band. Subsequently, we binarize this matrix by setting all non-zero values to 1, thereby discarding the triangle filters. This process yields the Mel-band Map depicted in Figure 1.

We follow the method outlined in [11] for performing random remixing data augmentation. This strategy involves cross-dataset stem-level combination, resulting in a significantly larger number of examples than the original size of the datasets combined.

Tree evaluation scenarios are considered: ⓐ *musdb18-only*: train a 44.1kHz stereo model only on the MUSDB18-HQ training set; ⓑ *all-data*: all additional data, including MUSDB18HQ, MoisesDB, and In-House, are used to train a 44.1kHz stereo model; ⓒ *musdb18+moisesdb*: MUSDB18HQ and MoisesDB are resampled to train a 24kHz mono model, serving as the pre-trained model for fine-tuning for the melody transcription task.

Our main baseline is BS-RoFormer [11]. For scenarios ⓐ and ⓑ, we set the parameters as follows: T=800 (8-second chunk), K=60, D=384, L=12, and a window size of 2048 and a hop size of 441 for STFT. In scenario ⓒ, two models are trained: *24k-small* and *24k-large*. The small model uses T=300 (6-second chunk), K=32, D=128, L=12; while the large model uses T=300 (6-second chunk), K=32, D=256, L=24. Both models adopt a window size of 1024 and a hop size of 480 for STFT. All the separation models use the “overlap & average” deframing method [11] with a hop of half a chunk. These above mentioned settings remain consistent between Mel-RoFormer and BS-RoFormer.

For training, we utilize the AdamW optimizer [50] with a learning rate (LR) of 0.0005. The LR is reduced by 10% every 40k steps. To optimize GPU memory usage, we employ flash-attention [51] and mixed precision. Specifically, the STFT and iSTFT modules use FP32, while all other components use FP16. Regarding hardware configurations, we employ different setups for each scenario. In scenario ⓐ, 8 Nvidia A-100-80GB GPUs with batch_size=64 are used, and the training stopped at 400K steps (~40 days). For scenario ⓑ, 16 Nvidia A100-80GB GPUs with batch_size=128 are utilized, and the training halted at 1M steps (~93 days). In scenario ⓒ, 16 Nvidia V100-32GB GPUs with batch_size=96 are used, and the training stopped at 500K steps (~31 days).

The reason to use a large number of training steps is driven by the continuous improvement observed in the model’s performance, coupled with the absence of overfitting. This can be attributed to two main factors: the effect of the random remixing augmentation and the inherent capability of the model itself. These factors contribute to the model’s ability to continuously learn and adapt to the training data, resulting in sustained performance improvements without encountering overfitting issues.

5.3 Result for Vocal Separation

Table 2 presents the results, with the signal-to-distortion ratio (SDR) values [54] computed by `museval` [55] as the evaluation metric. The median SDR across the median SDRs over all 1-second chunks of each test song is reported, following prior conventions. Several representative existing models are included for comparison.

From the result, we see that Mel-RoFormer achieve state-of-the-art performance. It is evident that Mel-band Projection significantly enhances vocal separation performance, leading to a consistent improvement over BS-

Model	#Param	COn	COnP	COnPOff
Efficient-b1 [36]	-	.754	.666	.458
JDC$_{\text{note}}$ [35]	-	.762	.697	.422
A-VST [56]	-	.783	.707	.538
Perceiver TF [8]	-	-	.777	-
MERT [15] ⓓ	324M	.775	.751	.530
SpecTNT [5] ⓓ	8.4M	.801	.778	.550
Mel-RoF-small ⓓ	14.5M	.807	.786	.609
Mel-RoF-large ⓓ	64.6M	**.819**	**.798**	**.625**
Mel-RoF-small ⓕ	14.5M	.780	.765	.574
Mel-RoF-large ⓕ	64.6M	.790	.776	.594

Symbols ⓓ, ⓔ, and ⓕ indicate three evaluation scenarios.

Table 3. Model comparison on MIR-ST500 test set.

Model	COn	COnP	COnPOff
MERT [15] ⓔ	.745	.697	.315
SpecTNT [5] ⓔ	.797	.775	.371
Mel-RoF-small ⓔ	.831	.805	.398
Mel-RoF-large ⓔ	**.869**	**.842**	.486
Mel-RoF-small ⓕ	.833	.808	.405
Mel-RoF-large ⓕ	.864	.839	**.494**

Table 4. Model comparison on POP909 test set. Evaluated with a time tolerance of 80 ms.

RoFormer, with an average gain of 0.5 dB across all scenarios. This showcases the effectiveness of the Mel-band mapping scheme in capturing human voices. Qualitative analysis indicates that Mel-RoFormer produces smoother vocal sounds with more consistent loudness. On the other hand, the 24kHz mono model also performs admirably, which bodes well for downstream tasks like vocal melody transcription, as they do not necessitate high-quality audio with high sampling rates. Furthermore, the smaller model with 9.1M parameters achieves over 11 dB, demonstrating its potential for resource-constrained environments.

5.4 Configuration for Vocal Melody Transcription

Three evaluation scenarios are studied: ⓓ trained on MIR-ST500; ⓔ trained on POP909, and ⓕ trained on a combination of MIR-ST500 and POP909. With T=300 and K=32 for Mel-RoFormer, the resulting embedding matrix for the onset and frame predictors has a shape of (2048×300), representing a 6-second input with a frame rate of 50Hz. We also consider a variant that is trained from scratch without a pre-trained model.

For fine-tuning, we use the AdamW optimizer with a LR of 0.001 for the onset and frame predictors and 0.0001 for the Mel-RoFormer module. The LR is reduced by 10% with a patience of 15 epochs, where an epoch comprises 100 steps. The best model is selected based on validation performance for testing.

For baselines, we implement SpecTNT [5] and MERT [15]. SpecTNT features 5-layer TNT blocks and is trained with the random mixing augmentation method outlined in [8]. For MERT, we use the pre-trained weights of "MERT-v1-330M," where the model accepts 5-second input with a 24kHz audio sampling rate and produces embeddings with a shape of (1024×375). All mentioned models are trained or fine-tuned using 8 Nvidia V100-32GB GPUs.

Evaluation metrics include the F-measures of Correct Onset (COn), Correct Onset and Pitch (COnP), and Correct Onset, Pitch, and Offset (COnPOff). These metrics are computed using `mir_eval` [57], with a pitch tolerance of 50 cents and a time tolerance of 50 ms. For POP909, we adjust the time tolerance to 80ms due to the less precise nature of note onsets and offsets in its labeling method [47].

5.5 Result for Vocal Melody Transcription

Tables 3 and 4 display the results on MIR-ST500 and POP909, respectively, featuring a comparison with various existing models. It is worth noting that fine-tuning a pre-trained model typically converges in fewer than 15K steps, while training from scratch requires significantly more steps (e.g., 50k steps) and yields significantly inferior performance compared to fine-tuned models.

Several key observations emerge from the results. Firstly, our proposed model (Mel-RoF-large and Mel-RoF-small) achieves state-of-the-art performance across all metrics on both MIR-ST500 and POP909, showcasing its effectiveness. Particularly noteworthy is the substantial performance improvement in COnPOff over the baselines (e.g., a 7.5 percentage point increase in Mel-RoF-large compared to SpecTNT), highlighting the robustness of Mel-RoFormer in accurately determining full notes, including onset, pitch, and offset. This can be attributed to its capability to extract clean singing voices, thereby minimizing the influence of irrelevant instruments. Comparing our models to MERT underscores the superiority of Mel-RoFormer, owing to its architectural design and pre-training with a relevant task. This emphasizes the importance of explicitly modeling frequency with Transformers and suggests that the separation task can be a valuable objective when training a foundation model.

Cross-comparing MIR-ST500 and POP909, we note that annotations are relatively more consistent in MIR-ST500. In contrast, POP909 exhibits errors primarily at note offsets, along with global time shifts in several songs. Consequently, we accept a larger time tolerance of 80 ms in the evaluation scenario. Particularly, in scenario ⓕ, training with combined datasets improves test performance on POP909 but degrades that of MIR-ST500, consistent with our observations about labeling quality.

6. CONCLUSION

We have presented the Mel-RoFormer model, which integrates the Mel-band Projection scheme to enhance its ability to model musical signals effectively. Our experiments have shown highly promising results in vocal separation and vocal melody transcription. These findings suggest the potential of Mel-RoFormer as a foundation model for various other MIR tasks, including chord recognition and multi-instrument transcription [58].

7. REFERENCES

[1] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *ICASSP*, 2014, pp. 6964–6968.

[2] R. M. Bittner, J. J. Bosch, D. Rubinstein, G. Meseguer-Brocal, and S. Ewert, "A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation," in *ICASSP*, 2022, pp. 781–785.

[3] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *ISMIR*, 2016.

[4] M. Won, S. Chun, O. Nieto, and X. Serrc, "Data-driven harmonic filters for audio representation learning," in *ICASSP*, 2020, pp. 536–540.

[5] W.-T. Lu, J.-C. Wang, M. Won, K. Choi, and X. Song, "SpecTNT: A time-frequency transformer for music audio," in *ISMIR*, 2021.

[6] A. Zadeh, T. Ma, S. Poria, and L.-P. Morency, "Wildmix dataset and spectro-temporal transformer model for monoaural audio source separation," *arXiv preprint arXiv:1911.09783*, 2019.

[7] Y. Luo and J. Yu, "Music Source Separation With Band-Split RNN," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 31, pp. 1893–1901, 2023.

[8] W.-T. Lu, J.-C. Wang, and Y.-N. Hung, "Multitrack music transcription with a time-frequency perceiver," in *ICASSP*, 2023.

[9] K. Toyama, T. Akama, Y. Ikemiya, Y. Takida, W.-H. Liao, and Y. Mitsufuji, "Automatic piano transcription with hierarchical frequency-time transformer," in *ISMIR*, 2023.

[10] J.-C. Wang, Y.-N. Hung, and J. B. Smith, "To catch a chorus, verse, intro, or anything else: Analyzing a song with structural functions," in *ICASSP*, 2022, pp. 416–420.

[11] W.-T. Lu, J.-C. Wang, Q. Kong, and Y.-N. Hung, "Music source separation with Band-Split RoPE Transformer," *arXiv preprint arXiv:2309.02612*, 2023.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, vol. 30, 2017.

[13] M. Won, K. Choi, and X. Serra, "Semi-supervised music tagging transformer," in *ISMIR*, 2021, pp. 769–776.

[14] Y.-N. Hung, J.-C. Wang, X. Song, W.-T. Lu, and M. Won, "Modeling beats and downbeats with a time-frequency transformer," in *ICASSP*, 2022, pp. 401–405.

[15] L. Yizhi, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, C. Xiao, C. Lin, A. Ragni, E. Benetos *et al.*, "Mert: Acoustic music understanding model with large-scale self-supervised training," in *ICLR*, 2023.

[16] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 26, no. 8, pp. 1307–1335, 2018.

[17] 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*, 2022.

[18] G. Fabbro, S. Uhlich, C. Lai, W. Choi, M. Martinez-Ramirez, W. Liao, I. Gadelha, G. Ramos, E. Hsu, H. Rodrigues *et al.*, "The sound demixing challenge 2023–music demixing track," *arXiv preprint arXiv:2308.06979*, 2023.

[19] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," in *INTERSPEECH*, 2020.

[20] S. S. Stevens, J. Volkmann, and E. B. Newman, "A scale for the measurement of the psychological magnitude pitch," *The journal of the acoustical society of america*, vol. 8, no. 3, pp. 185–190, 1937.

[21] L. R. Rabiner, R. W. Schafer *et al.*, "Introduction to digital speech processing," *Foundations and Trends in Signal Processing*, vol. 1, no. 1–2, pp. 1–194, 2007.

[22] E. Molina, A. M. Barbancho-Perez, L. J. Tardon-Garcia, I. Barbancho-Perez *et al.*, "Evaluation framework for automatic singing transcription," in *ISMIR*, 2014.

[23] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "The MUSDB18 corpus for music separation," Dec. 2017.

[24] A. Liutkus, F.-R. Stöter, Z. Rafii, D. Kitamura, B. Rivet, N. Ito, N. Ono, and J. Fontecave, "The 2016 signal separation evaluation campaign," in *13th International Conference on Latent Variable Analysis and Signal Separation*, 2017, pp. 323–332.

[25] E. M. Grais, M. U. Sen, and H. Erdogan, "Deep neural networks for single channel source separation," in *ICASSP*, 2014, pp. 3734–3738.

[26] P. Chandna, M. Miron, J. Janer, and E. Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *Latent Variable Analysis and Signal Separation (LVA/ICA)*, 2017, pp. 258–266.

[27] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang, "Decoupling magnitude and phase estimation with deep resunet for music source separation," in *ISMIR*, 2021.

[28] A. Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep U-Net convolutional networks," in *ISMIR*, 2017.

[29] S. Uhlich, M. Porcu, F. Giron, M. Enenkl, T. Kemp, N. Takahashi, and Y. Mitsufuji, "Improving music source separation based on deep neural networks through data augmentation and network blending," in *ICASSP*, 2017, pp. 261–265.

[30] D. Stoller, S. Ewert, and S. Dixon, "Wave-U-Net: A multi-scale neural network for end-to-end audio source separation," *arXiv preprint arXiv:1806.03185*, 2018.

[31] Y. Luo and N. Mesgarani, "Conv-TasNet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019.

[32] A. Défossez, N. Usunier, L. Bottou, and F. Bach, "Music source separation in the waveform domain," *arXiv preprint arXiv:1911.13254*, 2019.

[33] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation," in *ICASSP*, 2023.

[34] C. Donahue, J. Thickstun, and P. Liang, "Melody transcription via generative pre-training," in *ISMIR*, 2022.

[35] S. Kum, J. Lee, K. L. Kim, T. Kim, and J. Nam, "Pseudo-label transfer from frame-level to note-level in a teacher-student framework for singing transcription from polyphonic music," in *ICASSP*, 2022.

[36] J.-Y. Wang and J.-S. R. Jang, "On the preparation and validation of a large-scale dataset of singing transcription," in *ICASSP*, 2021, pp. 276–280.

[37] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020.

[38] J.-Y. Hsu and L. Su, "Vocano: A note transcription framework for singing voice in polyphonic music." in *ISMIR*, 2021, pp. 293–300.

[39] B. Zhang and R. Sennrich, "Root mean square layer normalization," *NeurIPS*, vol. 32, 2019.

[40] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *arXiv preprint arXiv:2104.09864*, 2021.

[41] K. Li, X. Hu, and Y. Luo, "On the use of deep mask estimation module for neural source separation systems," *INTERSPEECH*, 2022.

[42] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *ICML*, 2017, pp. 933–941.

[43] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018.

[44] E. Gusó, J. Pons, S. Pascual, and J. Serrà, "On loss functions and evaluation metrics for music source separation," in *ICASSP*, 2022, pp. 306–310.

[45] C. Hawthorne, E. Elsen, J. Song, A. Roberts, I. Simon, C. Raffel, J. Engel, S. Oore, and D. Eck, "Onsets and frames: Dual-objective piano transcription," in *ISMIR*, 2018, pp. 50–57.

[46] I. Pereira, F. Araújo, F. Korzeniowski, and R. Vogl, "Moisesdb: A dataset for source separation beyond 4-stems," in *ISMIR*, 2023.

[47] Z. Wang, K. Chen, J. Jiang, Y. Zhang, M. Xu, S. Dai, X. Gu, and G. Xia, "Pop909: A pop-song dataset for music arrangement generation," in *ISMIR*, 2020.

[48] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, 2015, pp. 18–25.

[49] M. Slaney, "Auditory toolbox," *Interval Research Corporation, Tech. Rep*, vol. 10, no. 1998, p. 1194, 1998.

[50] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2019.

[51] T. Dao, "Flashattention-2: Faster attention with better parallelism and work partitioning," *arXiv preprint arXiv:2307.08691*, 2023.

[52] A. Défossez, "Hybrid spectrogram and waveform source separation," *arXiv preprint arXiv:2111.03600*, 2021.

[53] M. Kim and J. H. Lee, "Sound demixing challenge 2023–music demixing track technical report," *arXiv preprint arXiv:2306.09382*, 2023.

[54] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio Speech Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[55] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *LVA/ICA*, 2018, pp. 293–305.

[56] X. Gu, W. Zeng, J. Zhang, L. Ou, and Y. Wang, "Deep audio-visual singing voice transcription based on self-supervised learning models," *arXiv preprint arXiv:2304.12082*, 2023.

[57] C. Raffel, B. McFee, E. J. Humphrey, J. Salamon, O. Nieto, D. Liang, D. P. Ellis, and C. C. Raffel, "Mir_eval: A transparent implementation of common mir metrics." in *ISMIR*, vol. 10, 2014.

[58] K. W. Cheuk, K. Choi, Q. Kong, B. Li, M. Won, J.-C. Wang, Y.-N. Hung, and D. Herremans, "Jointist: Simultaneous improvement of multi-instrument transcription and music source separation via joint training," *arXiv preprint arXiv:2302.00286*, 2023.