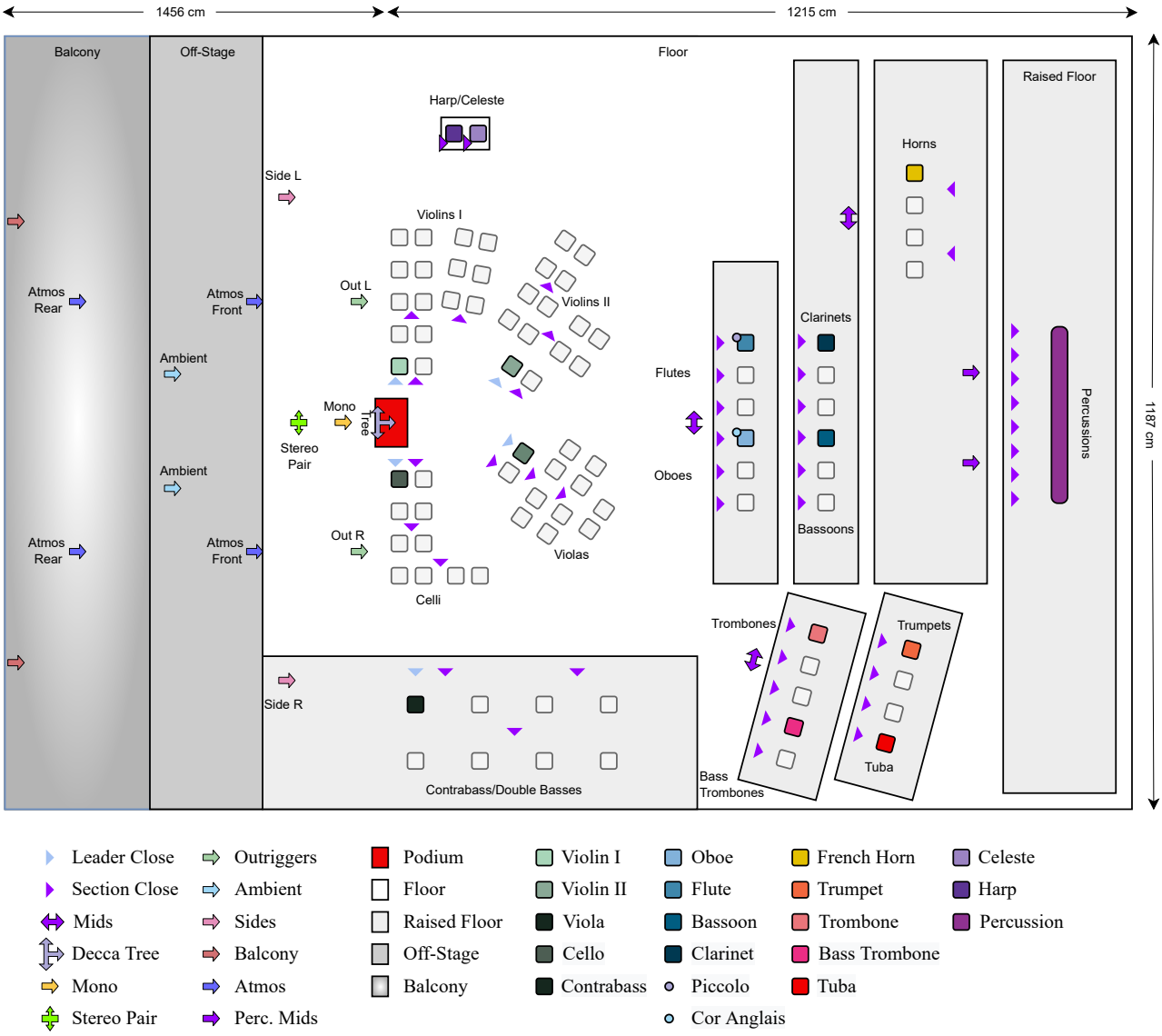

Figure 1. Recording configuration for the Spitfire Audio BBC Symphony Orchestra sample library depicting the placement of individual microphones, performers. The scale of the image is not uniform as the off-stage elements are placed further than they appear.

3.1.2 Mutopia

The Mutopia project [9] is a publicly sourced and manually verified free content sheet music library. The sheet music scores are manually annotated from old scores that are now public domain, and digitally archived using the lilypond format which can be converted to MusicXML and MIDI. This library has a large collection of string ensembles, of which we utilise 71 pieces (5h 5m 35s)[2] comprising a variety of chamber ensembles primarily composed of string quartets but also including other instruments such as Trumpet, Horn, Oboe, Clarinet, Flute and Bassoon. Although all the lilypond files come with their standard MIDI conversions, we utilise the lilypond to MusicXML conversion python library [33], to preserve the articulation information present in the lilypond files. For the files which we are able to convert to MusicXML, we import them to Dorico, where these articulations are translated to keyswitches (described in Table 2) and rendered to MIDI format which can then be utilised by the BBCSO plugin when rendered on Reaper [32].

3.1.3 Data Cleaning

Many of the scores used to render our dataset contain instruments that are absent (e.g., piano, vocals) in our sample library. Since the intent of EnsembleSet is to generate realistic renders of instruments performing and being recorded in the same physical space, we chose to remove the incompatible instruments as rendering them using other plugins will not be consistent. While converting Mutopia based files using the lilypond to MusicXML conversion tool, many files resulted in erroneous MusicXML files. For the corrupted conversions, we used the MIDI files directly from the database and were unable to preserve articulation for those pieces. For these pieces we use the same

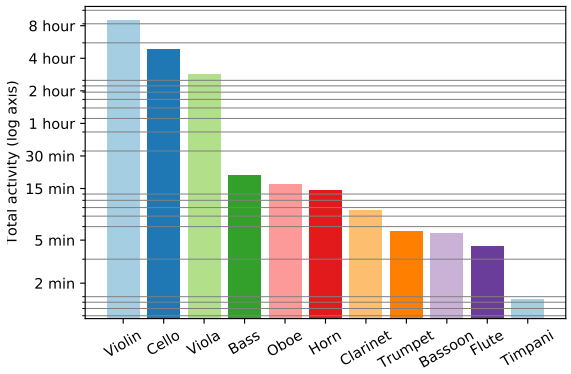

Figure 2. Instrument wise activity duration in EnsembleSet

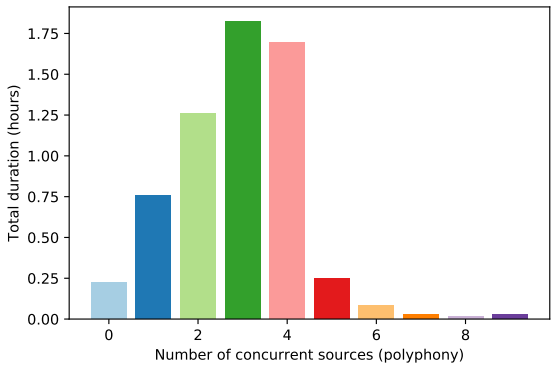

Figure 3. Polyphony distribution across EnsembleSet

articulation generation pipeline as the RWC sourced files. For some other files where the errors were minor (incorrect timing and track assignments), we used their corresponding MIDI files from the library to manually inspect and correct these MusicXML files.

3.2 BBC Symphony Orchestra Sample Library

This library was developed in partnership between BBC Studios and Spitfire Audio, by capturing a full orchestra as sections as well as individual section leaders. Each instrument was recorded for each note in a variety of articulation modes using multiple microphones placed at different positions in the room. For shorter notes, multiple iterations were recorded which are rendered in a round-robin fashion to simulate microtiming variations of real performers. The sample library was recorded in the same fashion as a film score would be recorded in a studio with a multi-microphone setup that enables the capture of each performer from different perspectives in the room. This is what allows us to simulate high quality recordings of chamber ensemble pieces from digital music scores, rendered using individual section leaders.

No.	Render Name	Type	# Mics	Pan
1	Mono	Bidirectional	1	Mono
2	Leader	Unidirectional	1	Stage Pan
3	Decca Tree	Omnidirectional	3	Stereo
4	Outriggers	Omnidirectional	2	Stereo
5	Ambient	Omnidirectional	2	Stereo
6	Balcony	Omnidirectional	2	Stereo
7	Stereo Pair	Coles 4038	2	Stereo
8	Mids	Omnidirectional	2	Stereo
9	Sides	Omnidirectional	2	Stereo
10	Atmos Front	Omnidirectional	2	Stereo
11	Atmos Rear	Omnidirectional	2	Stereo
12	Close	Unidirectional	1	Stage Pan
13	Close Wide	Unidirectional	1	Mono
14	Spill String	Unidirectional	15	Stage Pan
15	Spill Brass	Unidirectional	11	Stage Pan
16	Spill Woodwind	Unidirectional	12	Stage Pan
17	Spill Percussion	Unidirectional	10	Stage Pan
18	Spill Full	Unidirectional	48	Stage Pan
19	Mix 1	Mix	12	Stage Pan
20	Mix 2	Mix + FX	12	Stage Pan

Table 1. List of available renders in EnsembleSet. It must be noted that the Leader microphone is only available for string instruments.

3.2.1 Microphone Renders

The BBCSO sample library provides an entire portfolio of recording stems/microphones that a mix engineer would rely on while producing orchestral scores including traditional mixing setups like decca tree, outriggers, ambient microphones, far balcony mics, side mics and more modern setups including Atmos front and back mics placed at a height in the room. The placement of the individual microphones and each performer is shown in Figure 1. These recorded samples not only preserve the timbral changes that occur for a source recorded at various positions based on their distance, microphone type and directionality, but also preserves the phase shifts that occur across these different microphones placed at different distances w.r.t. the sources. The recorded samples are rendered without any time-correction, which implies that different mic renders would have different time delays and phase shifts based on the distance between the source and the mic. The renders also realistically reflect the timing adjustments performers for different sections make based on their instruments dynamics and position on stage.

It must be noted that in EnsembleSet, the positions for the close microphones are unique for each instrument, but the remaining room microphones are common across all instruments. Thus to simulate a realistic microphone bleed scenario, one can simply render each source at any given room microphone and downmix the resulting instrument stems. On the other hand, downmixing the close microphones would simulate the more typical scenario of music separation from a mixed song. Further details about individual microphone/mix setups can be found in Table 1.

3.2.2 Mixes

Apart from the individual microphone stems, the plugin also provides two professionally mixed stems. Mix 1 is a

Switch	Strings	Horn & Trumpet	Flute & Clarinet	Oboe & Bassoon
C-1	Legato	Legato	Legato	Legato
C#-1	Long	Long	Long	Long
D-1	Long Con Sordino	Staccatissimo	Trill Major 2nd	Trill Major 2nd
D#-1	Long Flautando	Marcato	Trill Minor 2nd	Trill Minor 2nd
E-1	Spiccato	Long Cuivre	Staccatissimo	Staccatissimo
F-1	Staccato	Long Sforzando	Tenuto	Tenuto
F#-1	Pizzicato	Long Flutter	Marcato	Marcato
G-1	Col Legno	Multi-tongue	Long Flutter	Multi-tongue
G#-1	Tremolo	Trill Major 2nd	Multi-tongue	-
A-1	Trill Major 2nd	Trill Minor 2nd	-	-
A#-1	Trill Minor 2nd	Long (muted)	-	-
B-1	Long Sul Tasto	Staccatissimo (muted)	-	-
C0	Long Harmonics	Marcato (muted)	-	-
C#0	Short Harmonics	-	-	-
D0	Bartok Pizzicato	-	-	-
D#0	Marcato	-	-	-

Table 2. List of keyswitch-articulation mappings for different instruments.

general starting point for a mix engineer with a good balance of the commonly used microphones like Decca Tree, Outriggers, Ambient, Balcony, Mids and Close mics. Mix 2 provides a more intense sound with some added compression, EQ and reverb. These stems are ideal to simulate the typical music separation scenario as the mixes provided present a good simulation of an unmastered and a mastered mix for an orchestral ensemble.

3.3 Articulation Automation

The BBCSO plugin allows rendering each note in a variety of articulations that are particular to each instrument. We use Dorico which in case of importing scores as MusicXML files, is capable of mapping articulations from MusicXML to keyswitches in the -1 octave in MIDI. Alternatively if articulations are unavailable, as is the case for importing scores as MIDI files, Dorico automatically selects either staccato or long articulation based on individual note lengths with a crossover at 187.5ms (16th note at 80bpm). The list of keyswitches and articulation mappings for each of the instruments available in EnsembleSet is shown in Table 2.

3.4 Dataset Contents

EnsembleSet contains a total of 6 hours and 9 minutes of multi-instrument, multi-mic data and is available on Zenodo [3] . The resulting total active duration of each instrument in EnsembleSet can be seen in Figure 2. The dataset presented is focused around string ensembles, and each of the 80 tracks presented in the dataset contains at least one string instrument, while the majority of pieces comprise string quartets. EnsembleSet also contains other woodwind and brass instruments, although their distribution is rather sparse. The overall polyphony distribution across the dataset is shown in Figure 3. Each song is also paired with its accompanying MIDI file which was used to generate the renders, and also contains the articulation information. Our implemented data preprocessing (described in section 4.2), data augmentation pipeline and other metadata related to the tracks such as song title, author, instrumentation and audio examples are available online [4] .

3.5 Limitations

While we have tried our best to make the synthesised recordings sound as realistic as possible, the achieved quality was still limited by the amount of information available in the source MIDI/lilypond files. All of the 9 tracks sourced from the RWC database have very good dynamics and realistic tempo variations in the renders, but since the source data was MIDI, the articulations are limited to long, staccato and pizzicato. For the 71 songs sourced from Mutopia, we were able to render from MusicXML for 30 of them, thus these are the only songs that are able to map to all possible articulations present in the source sheet music. For the remaining 41 tracks which were rendered from MIDI, the articulations are similarly limited to long, staccato and pizzicato. For all the songs sourced from Mutopia, the dynamics mapping available was limited due to limitations of the source lilypond format, thus resulting in each note having only one of 3 levels of velocity. While the instrument names in the renders have been standardised across the dataset, the accompanying MIDI files provided with each of the renders do not have standardised track names as they were preserved from the original track names from the source MIDI/Lilypond files.

4. EXPERIMENTS

To exhibit the value of our synthesised dataset, we use EnsembleSet to train a source separation model that is able to separate any chamber ensemble duet as explored in [30]. While we are training our model exclusively on our generated data, we evaluate on real-world data from the URMP dataset [20]. We make use of the multi-mic renders that are available in EnsembleSet as a form of data augmentation by randomising the mix/mic(s) presented to the

[3] https://zenodo.org/record/6519024

[4] http://c4dm.eecs.qmul.ac.uk/EnsembleSet/

Model	Train	Eval	SDR	SI-SDR
MSI [30]	URMP	URMP	+6.33 dB	-
DPTNet	URMP	ES	+6.29 dB	+4.37 dB
DPTNet	ES	URMP	+11.37 dB	+9.06 dB
DPTNet	ES	ES	+14.17 dB	+12.87 dB

Table 3. 2-source Chamber Ensemble Separation results.

model at each epoch. In addition, we use other augmentations including pitch shift and gain modulation to help the model generalise better to unseen source/microphone configurations. We utilise the same architecture as presented in [11] which is based on [10], modified to accommodate for 44.1kHz sample rate audio.

4.1 Model

We utilise the Dual-path Transformer (DPTNet) [10] based architecture using PIT [27] and modify the filterbank, scheduler and other network parameters to accommodate input segments at a sampling rate of 44.1kHz. Our model takes 2.97 second input frames (131072 samples) with 8 repeating separator units. We define the 1-D encoder filterbank to have a filter length of 32 samples with a hop size of 4 samples which resulted in best results in our experiments. Utilising a PIT loss for monotimbral ensembles is particularly well suited, as this enables our model to be able to separate any two monotimbral instruments regardless of their class.

4.2 Data

We train the model using all possible combinations of chamber ensemble duets playing simultaneously from EnsembleSet (ES) amounting to about 53 hours of data. To achieve this we implemented a novel dataloader that measures instrument activity confidence for each instrument track and identifies pairs of instrument segments where both the sources have some overlapping activity in all possible combinations (for eg: a string quartet piece for 2 source separation can be used as 6 different pairs of string duets). We used the URMP dataset (URMP) [20] to generate real examples for cross-validation and testing in a similar fashion resulting in 4.5 hours of 2 source mixtures. We utilise torch-audiomentations [34,35] for data augmentation such as gain modulation, channel swap and pitch-shifting by up to +/- 2 semitones. We also use the multi-mic renders of each instrument track as data augmentation by randomly choosing one of the 20 renders for each instrument for each iteration. It must also be noted that we maintain temporal and harmonic integrity of the mixtures through all the data augmentations. This is unlike the typical music separation data augmentation pipeline where the constituent parts of the mixtures are randomised across different songs at every epoch during training [36].

4.3 Training

We train the models for 100 epochs with early stopping patience of 10 epochs. We start with a learning rate of $5 \times e^{-3}$ with a scheduler that halves the learning rate if the validation loss does not improve for 3 epochs. We train the models on 4 x NVIDIA A100 GPUs using a distributed data parallel back-end. Each epoch in our experiments took 40 minutes with a batch size of 1 per GPU.

5. RESULTS

We present our baseline results based on the experiments described above and compare it to previous experiments conducted for a similar task as described in [30]. The results from [30] are based on a zero-shot learning + multi-task source informed (MSI) separation model designed to tackle the limitation of a very small training dataset. We compare our model's cross-dataset evaluation performance between the URMP Dataset [20] and EnsembleSet (ES) with the experiments from [30] as shown in Table 4.1. We find that our model trained on URMP and tested on ES reports similar separation quality as the MSI experiments from [30], although the test sets were not identical. The same model trained on ES and tested on URMP reports an improvement of 5dB in separation quality.

6. CONCLUSION

In this paper we introduced a new dataset constructed using digitised chamber ensemble scores and a professional orchestral sample library to address the lack of multitrack chamber ensemble datasets. We described our data generation process and data augmentation methods to enable generalisable deep learning solutions using the same. We provided a baseline for the task of separating 2 monotimbral instruments playing simultaneously and are able to show that models trained exclusively on our synthesised dataset are able to generalise to real world data for the same task. This outcome emphasises the strong dependence of the performance of deep learning on training regimes, in particular the quality of the training dataset.

The presented dataset not only contains high quality multi-microphone renders of various instruments, but is also accompanied by the MIDI files that were utilised for generating this dataset. This paired data can be utilised for various tasks including multi-instrument transcription [12], instrument recognition [13], score-informed source separation [13], microphone simulation [15], and automatic mixing [16].

While PIT is well suited for monotimbral ensemble separation as it can separate any 2 sources regardless of their instrument class, it is limited by polyphony where a model only works for mixtures with a fixed number of sources. In the future we intend to explore source conditioned separation models which would enable separating any particular source from a mixture. Although the efficacy of such solutions in the case of mixtures with multiple instances of same instruments has to be tested. In our current work we perform the separation on single channel audio, but we would like to extend our model to be capable of handling multi-channel audio input and utilise spatial information implicitly during separation.

7. ACKNOWLEDGMENTS

Special thanks to Jake Jackson, Michael Krause, Dave Foster, and Mary Pilataki-Manika for insightful discussions and advice on generating this dataset. S. Sarkar is a research student at the UKRI Centre for Doctoral Training in Artificial Intelligence and Music, supported jointly by UK Research and Innovation [grant number EP/S022694/1] and Queen Mary University of London. This work was supported by a Turing Fellowship for E. Benetos under the EPSRC grant EP/N510129/1. The computations described in this research were performed using the Baskerville Tier 2 HPC service (https://www.baskerville.ac.uk/), funded by the EPSRC and UKRI through the World Class Labs scheme (EP/T022221/1) and the Digital Research Infrastructure programme (EP/W032244/1) and is operated by Advanced Research Computing at the University of Birmingham.

8. REFERENCES

[1] E. Vincent, T. Virtanen, and S. Gannot, *Audio source separation and speech enhancement*. John Wiley & Sons, 2018.

[2] Y. Mitsufuji, G. Fabbro, S. Uhlich, and F.-R. Stöter, "Music demixing challenge at ISMIR 2021," *arXiv e-prints*, pp. arXiv–2108, 2021.

[3] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimilakis, and R. Bittner, "MUSDB18-HQ - an uncompressed version of MUSDB18," Dec. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3338373

[4] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 31–35.

[5] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep neural networks for single-channel multi-talker speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 10, pp. 1670–1679, 2015.

[6] S. Sarkar, E. Benetos, and M. Sandler, "EnsembleSet," May 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6519024

[7] SpitfireAudio, "User Manual BBC Symphony Orchestra Professional," 2019. [Online]. Available: https://d1t3zg51rvnesz.cloudfront.net/p/files/product-manuals/4126/1648649726/BBCSOPro_Manual_v2.0.pdf

[8] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Popular, classical and jazz music databases." in *Proceedings of the 2nd International Society for Music Information Retrieval Conference (ISMIR)*, vol. 2, 2002, pp. 287–288.

[9] E. Praetzel, "Mutopia project: Free sheet music for everyone," 2000. [Online]. Available: https://www.mutopiaproject.org/index.html

[10] J. Chen, Q. Mao, and D. Liu, "Dual-path transformer network: Direct context-aware modeling for end-to-end monaural speech separation," *arXiv preprint arXiv:2007.13975*, 2020.

[11] S. Sarkar, E. Benetos, and M. Sandler, "Vocal Harmony Separation Using Time-Domain Neural Networks," in *Proc. Interspeech 2021*, 2021, pp. 3515–3519.

[12] Y.-T. Wu, B. Chen, and L. Su, "Multi-instrument automatic music transcription with self-attention-based instance segmentation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2796–2809, 2020.

[13] H. F. Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging hierarchical structures for few-shot musical instrument recognition," *arXiv preprint arXiv:2107.07029*, 2021.

[14] S. Ewert, B. Pardo, M. Muller, and M. D. Plumbley, "Score-informed source separation for musical audio recordings: An overview," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 116–124, 2014.

[15] A. Mathur, A. Isopoussu, F. Kawsar, N. Berthouze, and N. D. Lane, "Mic2mic: using cycle-consistent generative adversarial networks to overcome microphone variability in speech systems," in *Proceedings of the 18th international conference on information processing in sensor networks*, 2019, pp. 169–180.

[16] J. D. Reiss, "Intelligent systems for mixing multichannel audio," in *2011 17th International Conference on Digital Signal Processing (DSP)*, 2011, pp. 1–6.

[17] F.-R. Stöter, A. Liutkus, and N. Ito, "The 2018 signal separation evaluation campaign," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2018, pp. 293–305.

[18] S. Rosenzweig, H. Cuesta, C. Weiß, F. Scherbaum, E. Gómez, and M. Müller, "Dagstuhl choirset: A multitrack dataset for mir research on choral singing," *Transactions of the International Society for Music Information Retrieval*, vol. 3, no. 1, 2020.

[19] 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 *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, vol. 14, 2014, pp. 155–160.

[20] B. Li, X. Liu, K. Dinesh, Z. Duan, and G. Sharma, "Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 522–535, 2018.

[21] C. Böhm, D. Ackermann, and S. Weinzierl, "A multi-channel anechoic orchestra recording of beethoven's symphony no. 8 op. 93," *Journal of the Audio Engineering Society*, vol. 68, no. 12, pp. 977–984, 2021.

[22] Z. Duan and B. Pardo, "Soundprism: An online system for score-informed source separation of music audio," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6, pp. 1205–1215, 2011.

[23] J. Fritsch and M. D. Plumbley, "Score informed audio source separation using constrained nonnegative matrix factorization and score synthesis," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 888–891.

[24] D. Petermann, P. Chandna, H. Cuesta, J. Bonada, and E. Gómez Gutiérrez, "Deep learning based source separation applied to choir ensembles," in *Proceedings of the 21st International Society for Music Information Retrieval Conference (ISMIR)*, 2020.

[25] M. Gover and P. Depalle, "Score-informed source separation of choral music," Ph.D. dissertation, McGill University, 2019.

[26] P. Chandna, H. Cuesta, D. Petermann, and E. Gómez, "A Deep-Learning Based Framework for Source Separation, Analysis, and Synthesis of Choral Ensembles," *Frontiers in Signal Processing*, vol. 2, 2022. [Online]. Available: https://www.frontiersin.org/article/10.3389/frsip.2022.808594

[27] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 241–245.

[28] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 696–700.

[29] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 46–50.

[30] L. Lin, Q. Kong, J. Jiang, and G. Xia, "A unified model for zero-shot music source separation, transcription and synthesis," in *Proceedings of 21st International Conference on Music Information Retrieval, ISMIR*, 2021.

[31] Steinberg, "Dorico: Music notation software | steinberg," 2016. [Online]. Available: https://www.steinberg.net/dorico/

[32] Cockos, "Reaper: Digital audio workstation," 2006. [Online]. Available: https://www.reaper.fm/

[33] LilyPond, "python-ly: Python library containing various python modules to parse, manipulate or create documents in lilypond format," 2016. [Online]. Available: https://github.com/frescobaldi/python-ly

[34] I. Jordal, "torch-audiomentations: Audio data augmentation in pytorch," 2021. [Online]. Available: https://github.com/asteroid-team/torch-audiomentations

[35] M. Pariente, S. Cornell, J. Cosentino, S. Sivasankaran, E. Tzinis, J. Heitkaemper, M. Olvera, F.-R. Stöter, M. Hu, J. M. Martín-Doñas *et al.*, "Asteroid: the pytorch-based audio source separation toolkit for researchers," *arXiv preprint arXiv:2005.04132*, 2020.

[36] 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 *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 261–265.

END-TO-END LYRICS TRANSCRIPTION INFORMED BY PITCH AND ONSET ESTIMATION

Tengyu Deng[1] **Eita Nakamura**[1] **Kazuyoshi Yoshii**[1,2]
[1] Graduate School of Informatics, Kyoto University, Japan
[2] PRESTO, Japan Science and Technology Agency (JST), Japan
deng@sap.ist.i.kyoto-u.ac.jp, {eita.nakamura, yoshii}@i.kyoto-u.ac.jp

ABSTRACT

This paper presents an automatic lyrics transcription (ALT) method for music recordings that leverages the framewise semitone-level sung pitches estimated in a multi-task learning framework. Compared to automatic speech recognition (ASR), ALT is challenging due to the insufficiency of training data and the variation and contamination of acoustic features caused by singing expressions and accompaniment sounds. The domain adaptation approach has thus recently been taken for updating an ASR model pre-trained from sufficient speech data. In the naive application of the end-to-end approach to ALT, the internal audio-to-lyrics alignment often fails due to the time-stretching nature of singing features. To stabilize the alignment, we make use of the semi-synchronous relationships between notes and characters. Specifically, a convolutional recurrent neural network (CRNN) is used for estimating the semitone-level pitches with note onset times while eliminating the intra- and inter-note pitch variations. This estimate helps an end-to-end ALT model based on connectionist temporal classification (CTC) learn correct audio-to-character alignment and mapping, where the ALT model is trained jointly with the pitch and onset estimation model. The experimental results show the usefulness of the pitch and onset information in ALT.

1. INTRODUCTION

Automatic lyrics transcription (ALT) refers to a task that aims to estimate the sung texts from music recordings, typically under the presence of accompaniment sounds. Since music composition and sharing have become very popular among non-professional people (e.g., YouTube), the number of non-annotated music data without lyrics transcriptions has been increasing rapidly. ALT has thus gained a lot of attention from the music information retrieval (MIR) community because of its usefulness in karaoke subtitle generation and text-based indexing.

Considering the similarity between ALT and automatic speech recognition (ASR), most studies on ALT have attempted to use ASR techniques, with some modifications if necessary. The hybrid approach based on a hidden Markov model (HMM) enhanced by a deep neural network (DNN), for example, has been used, where only the acoustic model was optimized for ALT [1]. Another way of ALT is to take the end-to-end approach that directly learns a sequence-to-sequence (audio-to-text) mapping. While the connectionist temporal classification (CTC) [2] and/or the attention mechanism [3] have widely been used for ASR [4, 5], the CTC has mainly been used for ALT [6, 7], in conjunction with the attention mechanism [8]. One reason is that the CTC considers only the monotonic audio-to-text alignment, which is relatively easier to infer from a limited amount of training data.

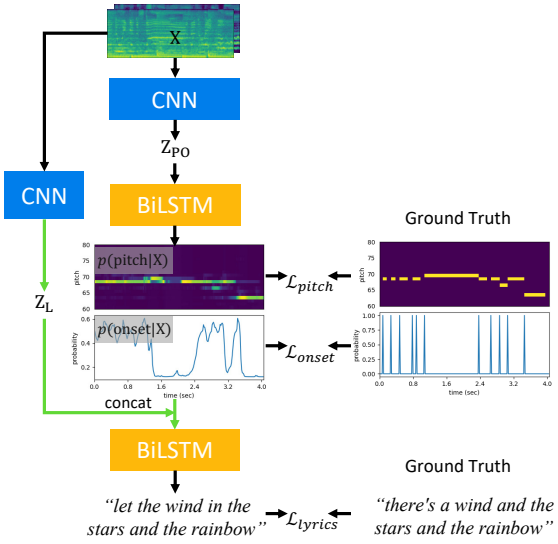

Figure 1. The proposed lyrics transcription method that estimates the pitches and onsets of singing voice and then uses them for character-level lyrics transcription.

The audio-to-text alignment plays a key role in end-to-end ALT and still remains a challenging problem. Firstly, the acoustic characteristics of singing voice vary over time in various ways according to the underlying sung notes and singing expressions. While the phones of speech tend to have particular durations and pitches specific to the speaker, those of singing voice may have time-stretched durations and semi-stepwisely time-varying pitches determined by the singer and the score. Secondly, the acoustic features of singing voice are contaminated by accompaniment sounds

or distorted by singing voice separation. It is, however, difficult to collect as training data a sufficient amount of music recordings with aligned lyrics annotations that cover a wide variety of recording conditions and singing styles.

To solve these problems, we make effective use of the framewise pitches and onset times of sung notes as acoustic features exclusive for ALT. In musical scores, lyrics are typically described synchronously with notes, i.e., words or characters tend to coincide with notes. This implies that the note onset information can be used for guiding an end-to-end ALT model to efficiently find the correct audio-to-lyrics alignment from a limited amount of training data. Multi-conditional training with the note pitch information is also expected to make the ALT model robust against the pitch variations.

In this paper, we propose an ALT method based on a cascading multi-task architecture that estimates the pitches and onset times of sung notes and then transcribes the lyrics at the character level in a pitch- and onset-conditioned manner (Fig. 1). More specifically, a convolutional recurrent neural network (CRNN) is used for jointly estimating the semitone-level pitches and onset times while eliminating the intra- and inter-note pitch variations (e.g., vibrato and glissando). Informed by this estimation, another CRNN is then used with CTC for learning audio-to-character alignment and mapping. These two CRNNs are trained jointly with backpropagation such that the sum of the pitch, onset, lyrics estimation losses is minimized. To mitigate the data insufficiency problem, we also use a domain adaptation method that fine-tunes a baseline ASR model pre-trained from huge speech data.

The main contribution of this paper lies in the first attempt for joint lyrics and pitch transcription towards comprehensive singing voice analysis. We experimentally show that the score information exclusive to music can be used effectively for finding audio-to-text alignment in end-to-end ALT with insufficient training data.

2. RELATED WORK

This section reviews related work on automatic lyrics transcription (ALT) and singing voice transcription (SVT).

2.1 Automatic Lyrics Transcription

ALT for music recordings under the presence of accompaniment sounds is still a difficult task due to the contamination of acoustic features [6–8]. A standard way of mitigating the adverse effect of accompaniment sounds is to take the two-step approach that performs singing voice separation [9] and lyrics transcription in this order. Although the remarkable improvement has been made in terms of the pure signal processing performance typically measured by the signal-to-distortion ratio (SDR) [10], the separated singing voice, however, is hard to transcribe, *i.e.*, the word error rate (WER) might be low, because an ALT model trained with clean isolated singing voice would suffer from the distortion of acoustic features and the mismatch between the training and test conditions. Although even ALT for clean singing voice [1] has much room for performance improvement due to the considerable variation of acoustic features caused by singing expressions, working directly on music recordings without singing voice separation could achieve better performance of ALT [7].

Inspired by the great success of the end-to-end approach to ASR, several attempts have been made for directly learning the mapping between non-aligned input and output sequences (audio and lyrics) of different lengths [6–8]. At the heart of the end-to-end learning is the audio-to-lyrics alignment based on the connectionist temporal classification (CTC) [2] and/or the attention mechanism [3]. The CTC-based approach estimates the posterior probabilities of labels (*e.g.*, words and characters) at the frame level and aims to maximize the total score obtained by efficiently accumulating the posterior probabilities of all possible *monotonic* alignment paths between the estimated and ground-truth label sequences. The attention-based approach is more powerful in that it can consider non-monotonic alignment and is thus useful for a wider variety of tasks (*e.g.*, machine translation). In general, however, the latter needs a larger amount of training data for finding the correct alignment and is thus hard to apply solely to ALT.

Some studies on audio-to-lyrics alignment have reported the effectiveness of using pitch information [11, 12]. Considering the semi-synchronous relationships between notes and characters, joint estimation of the pitches, onset times, and lyrics of singing voice would help an end-to-end model find the correct audio-to-lyrics alignment.

2.2 Singing Voice Transcription

The ultimate goal of SVT, a special case of automatic music transcription (AMT), is to estimate a human-readable vocal score underlying a given music recording. Towards this goal, a lot of efforts have been made for estimating the continuous fundamental frequencies (F0s) or discrete semitone-level pitches of singing voice at the frame or note level [13,14]. In particular, frame-level F0 estimation (*a.k.a.* melody extraction) has conventionally been considered as a subtask of SVT and intensively studied thanks to the great advance of supervised deep learning [15, 16]. There is, however, a big gap between melody extraction and genuine SVT because accurate scores are hard to obtain just by quantizing continuous F0 contours, typically at the tatum level, due to the large fluctuations and smooth transitions of F0s (*e.g.*, vibrato or glissando).

The latest study on SVT attempted to directly estimate a note sequence with discrete pitches and score positions [17]. This method is based on a hierarchical hidden semi-Markov model (HHSMM) that represents the generative process of observed singing spectra from a latent sequence of notes whose pitches and onset positions are assumed to follow a key-dependent Markov model and a metrical Markov model, respectively. The emission model was implemented with a CRNN that is pretrained to estimate the pitches of singing voice at the frame level from music spectra such that the intra- and inter-note F0 variations are eliminated. This technique forms the basis of the pitch and onset estimators used in our study.