Queen Mary University of London
School of Electronic Engineering and Computer Science

TIME-DOMAIN MUSIC SOURCE SEPARATION FOR CHOIRS AND ENSEMBLES

SAURJYA SARKAR

Ph.D. Thesis

Submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

This work was supported by the AI & Music CDT (EP/S022694/1)

March 2024

Saurjya Sarkar: *Time-domain Music Source Separation for Choirs and Ensembles* , Submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Artificial Intelligence and Music,

SUPERVISORS:

Mark Sandler

Emmanouil Benetos

EXAMINERS:

Slim Essid

Mona Jaber

LOCATION:

London, United Kingdom

DECLARATION

I, Saurjya Sarkar, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that Queen Mary University of London has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

The copyright of this thesis rests with the author and no quotation from it or information derived from it may be published without the prior written consent of the author.

London, United Kingdom, March 2024

Saurjya Sarkar

Dedicated to the loving memory of Janaranjan Sarkar.

1938 – 2022

ABSTRACT

Music source separation is the task of separating musical sources from an audio mixture. It has various direct applications including automatic karaoke generation, enhancing musical recordings, and 3D-audio upmixing; but also has implications for other downstream music information retrieval tasks such as multi-instrument transcription. However, the majority of research has focused on fixed stem separation of vocals, drums, and bass stems. While such models have highlighted capabilities of source separation using deep learning, their implications are limited to very few use cases. Such models are unable to separate most other instruments due to insufficient training data. Moreover, class-based separation inherently limits the applicability of such models to be unable to separate *monotimbral* mixtures.

This thesis focuses on separating musical sources without requiring timbral distinction among the sources. Preliminary attempts focus on the separation of vocal harmonies from choral ensembles using time-domain models with permutation invariant training. The method performs well but fails to generalise across datasets mainly due to a lack of sizeable clean training data. Recognising the challenge of obtaining sizeable, bleed-free data for ensemble recordings, a new high-quality synthesised dataset "EnsembleSet" is presented which was used to train a monotimbral ensemble separation model for string ensembles. Moreover, training a model using permutation invariant training is found to be capable of separate mixtures of identical, distinct, and unseen timbres as well. Although models trained on EnsembleSet can separate mixtures from unseen real-world datasets, performance drops are observed for out-of-domain test data. Subsequently improving cross-dataset performance using fine-tuning is explored for time-domain and complex-domain separation models. Further investigation into the performance of these models with different training strategies and different musical contexts is investigated to achieve a better understanding of the behaviour of these timbre-agnostic separation models. The techniques developed in this work are currently being utilised in the industry for vocal harmony separation and also lay the groundwork for future exploration toward universal source separation based on monophonic sound event separation.

Happiness is not a potato.

— SAURJYA SARKAR

ACKNOWLEDGEMENTS

Embarking on my eccentric journey into Music Research, a decade-long endeavour culminating in this thesis, has many people to thank for, over the span of a decade.

Firstly, I would like to thank Vivek, who introduced me to the guitar and told me "it's really not that difficult you know". Subsequently, joining BITS Pilani led me to meet Shruti, Sachin, Manickam, Pushkar and Prasanna who inspired me to pick up music and audition for the Music Club. Not only did I learn all I know as a performer there, but rigging up all those DIY gigs was my first introduction to digital audio effects and music production. I am deeply indebted to all my peers at Music Club BITS Pilani for the joy of learning and playing music together that motivated me to pursue a career in audio and music. I would also like to thank Prof. Kaushar Vaidya, without her unwavering faith in me and providing me with my first research opportunity in Taiwan, none of the things that followed might have happened.

This is the phase where C4DM comes into my life. I extend my gratitude to Prof. Josh Reiss for inviting me as a visitor and mentoring my undergraduate thesis. His willingness to invest time in an unknown undergrad from India was nothing short of mind-blowing and laid the foundation for my journey in audio research.

This is where I'd like to take the opportunity to thank Prof. Mark Sandler, my primary supervisor. His unwavering faith in my capability and intuition has been instrumental in shaping the trajectory of my PhD. I am equally indebted to Dr. Emmanouil Benetos, my second supervisor. The sheer amount

of time he committed to support me, his constant availability combined with his academic rigour and attention to detail helped me grow the most in this PhD. I could not have asked for a better supervisor duo.

I would also like to thank Alvaro Bort, our research manager for his immense commitment to support students. Especially in my case being an international student, he always took it upon himself to try and resolve my issues with funding and other administrative rigmaroles without ever having to ask twice. Thanks to Prof. Simon Dixon for being the architect behind the AIM CDT, which provided us with the opportunity to have a community of PhD students in AI and Music.

Speaking of community, I would like to thank my colleagues at C4DM who became my family. Pedro, Max, Ellie, Drew, Remi, Selim, Mary, Emir, Jack, Shubhr and all the others. You made London feel like home and life could've been a lot worse without you guys.

Ma, Baba, and Dadabhai. Thank you for everything.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

IRM Ideal Ratio Mask

cIRM Complex Ideal Ratio Mask

cRM Complex Ratio Mask

MFCC Mel-frequency Cepstral Coefficients

PIT Permutation Invariant Training

ICA Independent Component Analysis

NMF Non-negative Matrix Factorisation

CNN Convolutional Neural Network

RNN Recurrent Neural Network

Bi-RNN Bidirectional RNN

DNN Deep Neural Network

MWF Multichannel Wiener Filtering

STFT Short Time Fourier Transform

FFT Fast Fourier Transform

ReLU Rectified Linear Unit

LSTM Long Short Term Memory

BLSTM Bidirectional LSTM

TCN Temporal Convolutional Network

TasNet Time-domain Audio Separation Network

SDR Source to Distortion Ratio

SIR Source to Interference Ratio

SAR Source to Artifact Ratio

SI-SDR Scale-invariant Source to Distortion Ratio

URMP University of Rochester Multi-modal Music Performance

RoPE Rotary Position Embedding

GLU Gated Linear Unit

DPRNN Dual-path Recurrent Neural Network

DPTNet Dual-path Transformer Network

DSD100 Demixing Secrets Database

SiSEC Signal Separation and Evaluation Campaign

SOTA state-of-the-art

BSS Blind Source Separation

FIR Finite Impulse Response

MIDI Music Instrument Digital Interface

RWC Real World Computing

BBC British Broadcasting Corporation

BBCSO BBC Symphony Orchestra

LMD Lakh MIDI Dataset

SLakh Synthesised LMD

EQ Equalisation

VRAM Video RAM

GPU Graphics Processor Unit

MIR Music Information Retrieval

HPC High-performance Compute Cluster

LaSAFTNet Latent Source Attentive Frequency Transformation Network

DCCRNet Deep Complex Convolutional Recurrent Neural Network

BC Bach Chorales Dataset

BQ Barbershop Quartet Dataset

BCBQ Bach Chorales and Barbershop Quartet Dataset

C-U-Net Score-conditioned U-Net

C-Wave-U-Net Score-conditioned Wave-U-Net

ES EnsembleSet

cv Cross-validation

MSI Multi-task Source Informed Separation

SATB Soprano, Alto, Tenor and Bass

DCUNet Phase-aware Deep Complex U-Net

SR Sampling Rate

HO Harmonic Overlap

Fo Fundamental Frequency

X+O Pitch Crossovers + Pitch Overlaps

O-lap Pitch Overlap

Part I

CONTEXT

Placing Ensemble Separation in the context of existing research.

1

INTRODUCTION

This thesis investigates the use of deep-learning-based time-domain source separation models applied for the purpose of separating musical sources typically found in chamber ensembles, namely choirs, string, wind and brass instruments. This is a relatively unexplored task, especially in the presented approach which does not utilise any form of source-identity priors while training these models. This work relies on significant advances made in deep-learning-based source separation techniques while distinguishing itself from the approaches typically utilised in music source separation.

This chapter outlines the motivations and aims for the work presented in this thesis in Section 1.1, followed by the structure of this thesis in Section 1.2, subsequently list the major contributions of this thesis in Section 1.3 and finally lists the associated publications of this work in Section 1.4.

1.1 MOTIVATION, HYPOTHESIS AND RESEARCH QUESTIONS

Music source separation is the task of separating distinct musical sources from an audio mixture. While ideally a solution should be able to separate any given instrument from a mixture, there are two main challenges. Firstly, there are not enough bleed-free multitrack audio data available for a vast majority of instruments. Moreover, current approaches rely on either separating a given set of instrument classes or using a query-based separation approach, which limits their applicability to only being able to separate sources of distinct classes. The majority of prior research focuses on the instrument

class-based separation of the most omnipresent instrument stems in popular music i.e. drums, bass and vocals.

However, this formulation has some limitations. If a mixture consists of multiple instances of the same instrument, they cannot be distinguished using this method and will be considered as a single stem. Restricting music source separation to a class-based separation task would deem it impossible to train a model to separate a mixture of the same instruments. While separating choral mixtures based on vocal registers as class labels has been explored in other works, it has seen limited success. Current class-based source separation methods that rely on timbral features also suffer when a non-target instrument sounds similar to one of the target classes (for example lead guitars bleeding into vocal stems). Another limitation of their approach is that a model is applicable only for separating the specific set of instruments it has been trained for, which results in a lack of separation solutions for instruments that are underrepresented in current music separation datasets.

On the other hand, in the domain of speech separation, models are capable of separating very similar-sounding sources, namely mixtures of multiple speakers. Recent state-of-the-art speech separation results perform very well for 2 speaker mixtures (Luo and Mesgarani, 2019; Subakan et al., 2021). Moreover, these models are capable of simultaneously separating mixtures of same-gendered and different-gendered speakers. This is achieved by permutation invariant training objectives, where instead of training the models with specific output channels for distinct source classes, the model is trained to separate the sources regardless of which output channel a source gets assigned to.

However, in speech mixtures, there are few inherent properties that are different in the context of musical mixtures. Firstly, the sources are statistically uncorrelated, which is not the case in music. Musical sources usually play in the same key creating significant harmonic overlap, and are also often temporally synchronised. Secondly, speech mixtures typically consist of distinct

speakers who have distinguishable vocal features, whereas musical mixtures may include identical instruments playing in harmony or the same vocalist singing multiple harmonies in a recording.

The primary questions this thesis aims to address are:

- Are deep learning models trained with permutation invariant training capable of separating timbrally indistinguishable sources?
- Can harmonically correlated sources with high spectral overlap be separated using time-domain deep learning methods?
- Since bleed-free recordings for ensembles are very difficult to obtain, is it possible to train music source separation models that perform reliably on real-world chamber ensemble recordings using synthetic data?

While these uncertainties might deter the application of permutation invariant training to music source separation, the possibility of being able to separate mixtures of similar-sounding instruments motivates this research. However, on further research, it was discovered that training a model in a permutation invariant fashion not only enables the separation of mixtures of identical sources but also enables the separation of unseen sources. This thesis documents the experiments related to these findings which suggest that it is possible to train timbre-agnostic separation models, with the constraint that the sources are expected to be monophonic.

1.2 THESIS STRUCTURE

This section provides a structural skeleton of this thesis to guide the reader through its contents.

Chapter 2 presents an overview of relevant source separation research. It begins with the formal definition of the source separation problem and its

various sub-tasks. It then presents the reader with a brief overview of classical approaches to solve the source separation problem. Subsequently, an introduction to the various deep-learning based approaches that have been successful across different source separation tasks in recent years is presented. Finally, a survey of related works that tackle the ensemble separation problem is presented.

Chapter 3 presents *EnsembleSet*, a high-quality synthesised dataset consisting of instrument renders of chamber ensemble music. This chapter elaborates on the necessity of this dataset in the context of the research undertaken in this thesis and subsequently details the process of generating this dataset. It provides a detailed description of the contents of the dataset and can be used as a guide for users of this dataset.

Chapter 4 presents the proposed method using time-domain source separation architectures with permutation invariant training for musical ensembles. Since these architectures were originally designed for speech separation, the modifications and optimisation required to adapt them to work effectively for high-sample rate musical mixtures are presented here.

Chapter 5 presents the ensemble separation experiments using the proposed method. Firstly, the applicability of these models is tested for choral music separation using existing datasets. Since these experiments were conducted on a small dataset, the results were not generalisable. Subsequently, experiments using EnsembleSet for chamber ensemble separation are presented, which present the first cross-dataset generalisable ensemble separation results. Finally, domain adaptation experiments are presented on ensemble separation models pre-trained on *EnsembleSet*, which are subsequently fine-tuned to real-world datasets.

Chapter 6 presents various analysis methods to understand the performance of the presented ensemble separation models under different musical contexts. A novel measure to quantify the musical complexity of an ensemble mixture is presented, and its impact on separation performance is

investigated. The impact of training models with musically incoherent mixtures is also evaluated. The timbre-agnosticism of these models is explored in an instrument-wise analysis of performance. Finally, the impact of pitch overlaps amongst sources in the mixture is identified as detrimental to the performance of these separation models and specific failure modes caused by these are explored.

Chapter 7 summarises the findings of this thesis. The limitations and advantages of the proposed method are presented with an outlook on future improvements possible in source separation based on the insights from this thesis.

1.3 CONTRIBUTIONS

The contributions of this thesis can be categorised into three major parts:

A new dataset for chamber ensemble separation

- In Chapter 3, a new high-quality synthesised dataset for chamber ensemble music is presented. This is the largest dataset of bleed-free high quality chamber music currently available.
- The large size of this dataset, in combination with the multi-mic renders used as data augmentation enabled training the first generalisable source separation model for chamber ensembles.
- Unlike previous synthesised datasets, models trained on this larger synthesised dataset perform better than training on smaller real-world datasets.

A novel approach to music source separation

- In Section 5.3, permutation invariant training of time-domain deep learning models is shown to be effective at separating choral music

mixtures. These models outperform other class-based separation approaches.

- In Section 5.4, it is shown that the proposed models are capable of separating mixtures of identical sources.
- In Section 5.4 it is also shown that time-domain deep learning models with permutation invariant training are capable of separating sources of a variety of different timbres.
- In Section 5.5 it is observed that pre-training our proposed models on chamber ensemble instrument mixtures from EnsembleSet and fine-tuning them with vocal harmony separation data results in improved vocal harmony separation as compared to training on vocal data alone. This implies that the model is able to learn features from chamber ensemble mixtures that are useful for separating vocal harmony mixtures as well.
- In Section 6.4, it is shown that time-domain source separation models trained on EnsembleSet behave in a timbre-agnostic fashion, and are capable of separating mixtures of rare/unseen instruments as well. Moreover, the separation quality of these instruments does not seem to be correlated to their representation/distribution in the training dataset.

A better understanding of how TasNets work

- In Chapter 4 the optimisation process for adapting TasNets to work at high sampling rates is presented. The impact of different training and network hyperparameters are studied in the context of VRAM limitations.
- In Chapter 4 a performance study of the proposed model for ensemble separation tasks for 2, 3 and 4 sources is presented.
- In Section 6.2 a novel measure for quantifying the harmonic complexity of an input mixture is proposed. In Section 6.2.3 the performance

correlation of the proposed model and this harmonic overlap metric is measured. A moderate negative correlation was found between the separation performance of our proposed model and the harmonic overlap of a given input mixture.

- In Section 6.5, it is shown that pitch overlaps, crossovers and unisons in a given input mixture significantly deteriorate the separation quality achieved using our proposed model.
- *Source confusion* as a failure mode of our proposed model is identified, where the source-channel assignment in the model output is found to be inconsistent for input mixtures where the pitch trajectories of the sources have crossovers.
- *Unisons* are identified as another failure mode of our proposed method, where if the sources present in the input mixture are in unison, the model fails to separate them.
- Based on these findings, it leads to the understanding that TasNet based models trained with PIT are possibly separating sources based on the onsets and pitch trajectories of constituent sources and not on the basis of timbral characteristics of the sources.

1.4 ASSOCIATED PUBLICATIONS

Portions of the work described in this thesis have been presented at peer-reviewed international conferences and online repositories, as follows:

Peer-reviewed conference papers:

[i] S. Sarkar, E. Benetos, and M. Sandler, "Vocal Harmony Separation using Time-domain Neural Networks" in *Proceedings of INTERSPEECH 2021*, 2021

[ii] S. Sarkar, E. Benetos, and M. Sandler, “EnsembleSet: a new high quality synthesised dataset for chamber ensemble separation”, in *Proceedings of the 23rd Int. Society for Music Information Retrieval Conf., Bengaluru, India*, 2022.

[iii] S. Sarkar, L. Thorpe, E. Benetos, and M. Sandler, "Leveraging synthetic data for improving chamber ensemble separation", in *2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), New Paltz, NY, USA, 2023,*. (**Best Student Paper Award**)

Extended abstracts:

[iv] S. Sarkar, E. Benetos, and M. Sandler, "Music Source Separation in the Wild" in *DMRN+14: Digital Music Research Network Workshop Proceedings*, 2019

[v] S. Sarkar, E. Benetos, and M. Sandler, "Choral Music Separation using Time-domain Neural Networks" in *DMRN+15: Digital Music Research Network Workshop Proceedings*, 2020

[vi] S. Sarkar, E. Benetos, and M. Sandler, "Monotimbral Ensemble Separation using Permutation Invariant Training" in *Proceedings of the MDX Workshop, ISMIR*, 2021

Online repositories:

[vii] S. Sarkar, "MedleyDB dataloader for generating monotimbral mixtures", uploaded to GitHub at "https://github.com/saurjya/asteroid/data/medleydb_dataset.py", 2021

[viii] S. Sarkar, "Vocal Harmony Separation - audio examples", available at "http://c4dm.eecs.qmul.ac.uk/ChoralSep/", 2021

[ix] S. Sarkar, E. Benetos, and M. Sandler, "EnsembleSet", dataset uploaded to *Zenodo* at ”https://zenodo.org/record/6519024", 2022

[x] S. Sarkar, E. Benetos, and M. Sandler, "EnsembleSet with MIDI annotations", dataset uploaded to *Zenodo* at ”https://zenodo.org/record/7327175", 2022

[xi] S. Sarkar, "Music Ensemble Separation using EnsembleSet", uploaded to GitHub as a fork of the "Asteroid" Project at "https://github.com/saurjya/asteroid", 2022

[xii] S. Sarkar, "Music Ensemble Separation - audio examples", available at "http://c4dm.eecs.qmul.ac.uk/EnsembleSet/", 2023

It should be noted that for all the above-mentioned works, the author is the main contributor under the supervision of Dr. Emmanouil Benetos and Prof. Mark Sandler. In the case of [iii], the author was assisted by Louise Thorpe in preparing the instrument-wise analysis plots, of which Figure 17 and Figure 18 are presented in this thesis.

2

BACKGROUND

In this chapter, a brief overview of how source separation models evolved over recent years is presented and structured in a fashion where these methods are categorised based on their working principles. Due to the vast scope of this research field, some of the sections that are not directly related to the work presented in this thesis are described in limited detail.

Section 2.1 introduces the main source separation tasks of speech separation, music separation and speech enhancement. Section 2.2 describes the various reference-based evaluation metrics that are used for the task of source separation, which are also used in this thesis. Section 2.3 - Section 2.7 detail the evolution of source separation architectures across all source separation tasks.

Section 2.3 presents preliminary approaches in the literature that utilise spatial filtering to identify sources from multichannel mixtures. These methods rely on specific constraints in the spatial distribution of sources in the mix or the distribution of microphones to be able to extract the sources from them, which inevitably restrict the applicability of such models.

Section 2.4 discusses the initial methods introduced to be able to extract sources from a mixture based on features identifying independent sources present in a spectrogram which are not restricted by similar constraints and used preliminary machine learning methods to identify these sources from a spectrogram. However, due to their data-driven approach and the limitations of simple machine learning tools, these methods do not produce generalisable results.

Section 2.5 presents the methods based on magnitude spectrogram masking introduced after the advent of deep learning which is data-driven and produces the first generalisable source separation tools which were subsequently presented as publicly available tools for music demixing. While these methods work well, they were still limited by a lack of accurate phase estimation resulting in an upper limit of how well they could perform.

Section 2.6 presents deep-learning based solutions that work directly on time-domain audio signals in order to bypass the phase estimation problem which has shown exceptional performance in the task of speech separation. In particular, the methods described in Section 2.6.2 elaborate on the evolution of the particular methods that are used in this thesis.

Section 2.7 presents an alternative to time-domain processing as a solution to overcome the phase estimation problem, by predicting the complex-domain spectrogram of the target sources. These methods have shown exceptional performance in music source separation and speech enhancement, both of which problems rely heavily on the capability of these models to be able to model the timbral characteristics of the target sources.

Section 2.8 - Section 2.10 present a few alternative training paradigms for deep-learning based source separation solutions. Section 2.8 introduces the training method that allows models to be trained in a class-agnostic fashion which has been used successfully in conjunction with the methods presented in Section 2.6.2 to achieve generalisable results for the task of speech separation. This is also the training objective that is used in this thesis in order to achieve instrument-class agnostic ensemble separation. Section 2.9 presents a brief mention of methods that use adversarial training for the tasks of music separation as an alternative to using reference-based objective functions. Section 2.10 provides an introduction to unique methods that have been proposed to solve the universal source separation task with the expectation of being able to separate any sound source either in an unsupervised fashion or in a user query-based separation tool.

Section 2.11 introduces the datasets that are typically used in the context of music separation, which also includes the descriptions of the evaluation datasets used in this thesis.

Section 2.12 presents the evolution of music source separation as a task through the lens of the various public source separation challenges/campaigns. It presents the evolution of architectures described in Section 2.3 - Section 2.7 in the context of these SiSEC challenges and the concurrent evolution of the datasets presented Section 2.11.

Finally, Section 2.13 provides an overview of other works in the field of choral and ensemble separation that have been presented by other researchers recently, which have evolved in parallel to the work presented in this thesis. Finally, Section 2.14 includes an overview and perspectives on the overall research trends in source separation.

2.1 TASKS IN SOURCE SEPARATION

While all the models described later in this chapter have been presented as generic source separation architectures, they are typically designed to tackle one of three tasks: Speech separation, Speech Enhancement or Music Source Separation. While speech separation deals with separating mixtures of concurrent speakers in a label-agnostic fashion, speech enhancement and music separation are more similar as the models are trained to recognise and separate the target speech/musical sources based on modelling the timbre of the target source and extracting it from a mixture based on the timbre of the desired source class.

2.1.1 *Speech Separation*

Speech separation is the task of separating a mixture containing multiple speakers in a class-agnostic fashion. In this task, the models are not trained using a class-based regression fashion, but instead either solved using permutation invariant training (described in Section 2.8), or deep clustering (Hershey et al., 2016).

Here a multi-speaker mixture can be defined as a mixture of K speakers as:

$$s_{mixture}(t) = \sum_{k=1}^{K} s_k(t) \tag{1}$$

where each s_k is the speech signal of the k^{th} speaker. These separation models are expected to be able to distinguish between speech from different speakers without any prior about the timbral characteristics of the constituent speakers.

2.1.2 *Speech Enhancement*

Speech enhancement is defined as the task of improving the perceptual quality and intelligibility of noisy speech by performing denoising and dereverberation. The typical problem formulation provides a mixture/recording of a single speaker in the presence of noise as $s_{mixture}(t) \in \mathbb{R}^{1\times T}$ of duration T samples. This mixture can be decomposed into 2 stems $s_{speech}(t), s_{noise}(t) \in \mathbb{R}^{1\times T}$ as:

$$s_{mixture}(t) = s_{speech}(t) + s_{noise}(t) \tag{2}$$

where the model is expected to be able to extract the speech from the mixture without the noise.

This may even be further extended to include dereverberation of the speech and subsequently remove the noise. In this task reverberation is applied by convolving the clean speech signal with a room impulse response (s_{RIR}) which may be synthetic or real. This formulation can be expressed as:

$$s_{mixture}(t) = s_{RIR}(t) \circledast s_{speech}(t) + s_{noise}(t) \tag{3}$$

In this problem, often the input SNR of the noisy speech mixture can have a direct impact on the difficulty of the enhancement task, thus often models will report results for different levels of input noise, while some models are in fact also specifically designed to tackle high input-SNR scenarios.

Similar to the music demixing task, the models trained for speech enhancement typically learn the timbral distribution of speech to be able to extract it from noisy mixtures. However, unlike music separation where the models are trained to learn the timbral features of other target classes and separate them simultaneously, speech enhancement often is trained to predict only the speech signal and not the noise signal. This is due to the large variation present in noise signals which does not result in performance improvement if the model is trained to also learn the timbral characteristics of the noise signal.

2.1.3 *Music Source Separation*

Music source separation or music demixing is the task of splitting a mixed and mastered song into its constituent instrument stems. This task has typically focused on separating the vocals, bass and drum stems. The most basic form of music separation as described by Hershey et al. (2016) can be de-

fined as: a music mixture $s_{mixture}(t) \in \mathbb{R}^{1\times T}$ of duration T samples can be decomposed into 2 stems $s_{vocals}(t), s_{accompaniment}(t) \in \mathbb{R}^{1\times T}$ as:

$$s_{mixture}(t) = s_{vocals}(t) + s_{accompaniment}(t) \tag{4}$$

Here $s_{vocals}(t), s_{accompaniment}(t)$ are the isolated vocal and remaining accompaniment music signals respectively. The typical music separation task that is currently considered as the mainstream task as presented in recent music separation challenges such as Mitsufuji et al. (2022) is the task of decomposing mixtures into 4 stems:

$$s_{mixture}(t) = s_{vocals}(t) + s_{bass}(t) + s_{drums}(t) + s_{others}(t) \tag{5}$$

Here the $s_{accompaniment}(t)$ signal from Equation 4 is further decomposed into its constituent drums and bass instrument stems as $s_{drums}(t)$ and $s_{bass}(t)$ respectively. The mixtures described in Equation 4 and Equation 5 are of mixed and mastered pop songs where each of the constituent stems are composite stems which can be further decomposed as Equation 6:

$$s_{accompaniment}(t) = s_{bass}(t) + s_{drums}(t) + s_{others}(t) \tag{6}$$

The deep learning solutions trained to solve this problem formulation work in a class-based regression approach. These deep learning models typically are trained with a large variety of examples for each of the target classes, and subsequently, the model learns to separate the target sources from the mixture based on learning the timbral distribution for each class.

2.2 EVALUATION METRICS

BSS-Eval: The Blind Source Separation Evaluation (BSS Eval) toolkit (Vincent, Gribonval, and Févotte, 2006) decomposes the error between the target

source and the extracted source into a target distortion component reflecting spatial or filtering errors, an artefacts component pertaining to artificial noise, and an interference component associated to the bleeding in of unwanted sources. The salience of these components is quantified using three energy ratios: source Image-to-Spatial distortion Ratio (ISR), Source-to-Artifacts Ratio (SAR) and Source-to-Interference Ratio (SIR). A fourth metric, the Source-to-Distortion Ratio (SDR), measures the global quality (all impairments combined). It assumes the decomposition of a separated source $\hat{s}_{separated}$ into:

$$\hat{s}_{separated} = \boldsymbol{s}_{\text{target}} + e_{\text{interf}} + e_{\text{noise}} + e_{\text{artif}} \tag{7}$$

where, e_{interf}, e_{noise}, e_{artif} are respectively the interference, noise and artifact error terms.

$$\begin{aligned} \text{SDR} &:= 10\log_{10} \frac{\left\| s_{\text{target}} \right\|^2}{\left\| e_{\text{interf}} + e_{\text{noise}} + e_{\text{artif}} \right\|^2} \\ &= 10\log_{10} \frac{\left\| s_{\text{target}} \right\|^2}{\left\| \hat{s}_{\text{separated}} - s_{\text{target}} \right\|^2} \end{aligned} \tag{8}$$

$$\text{SIR} := 10\log_{10} \frac{\left\| s_{\text{target}} \right\|^2}{\left\| e_{\text{interf}} \right\|^2} \tag{9}$$

$$\text{SNR} := 10\log_{10} \frac{\left\| s_{\text{target}} + e_{\text{interf}} \right\|^2}{\left\| e_{\text{noise}} \right\|^2} \tag{10}$$

$$\text{SAR} := 10\log_{10} \frac{\left\| s_{\text{target}} + e_{\text{interf}} + e_{\text{noise}} \right\|^2}{\left\| e_{\text{artif}} \right\|^2} \tag{11}$$

SI-SDR: The $BSS - Eval$ metrics make a critical assumption that time-invariant filters are considered allowed deformations to the target/reference

signals. This justifies the transformations made in the sources by convolving short FIR filters, for example, applying the room impulse response to create a reverberated image. This however leads to a major problem, because the space of signals achievable by convolving the source signal with any short FIR filter is extremely large and includes perceptually widely different signals from the spatial image. Roux et al. (2019) proposes a new Scale-Invariant Signal-to-Distortion Ratio that tries to address this problem by ensuring the residual/error/noise in the signal to be orthogonal to the source/target by rescaling (by factor β) the reference while comparing to the estimate.

$$\text{SI} - \text{SDR} := 10 \log_{10} \frac{\left\| s_{\text{target}} \right\|^2}{\left\| \beta \hat{s}_{separated} - s_{\text{target}} \right\|^2} \text{ for } \beta \text{ s.t. } s_{target} \perp (\beta \hat{s}_{separated} - s_{target}) \tag{12}$$

PEASS: A perceptually-motivated adaptation of the BSS-Eval toolkit is PEASS (Emiya et al., 2010) (Perceptual Evaluation method for Audio Source Separation), which estimates the three distortion (target distortion, interference, artefacts) from auditory representations of the reference and extracted sources, which are then input to the PEMO-Q auditory model to measure their salience. Emiya et al. (2011) presents an extension where a neural-network trained on human data combines the resulting component-wise salience features into four objective predictors: Target-related Perceptual Score (TPS), Artifacts-related Perceptual Score (APS), Interference-related Perceptual Score (IPS) and Overall Perceptual Score (OPS).

Neural Network based Reference-less Evaluation: In experimental situations, the reference sources are usually available for use in evaluating the performance of a certain source separation approach. However, for practical applications of source separation, the mixtures are available but the reference source is not available. Most objective evaluation metrics rely on having the reference source signal available and, thus are rendered useless to evaluate

such scenarios. Moreover, some of these higher-level metrics can often be calculated at utterance level and not frame level. In Grais et al. (2019), Neural networks using RNNs are used to predict such metrics from Emiya et al. (2011) at the frame level.

2.3 SPATIALLY-INFORMED SOURCE SEPARATION

Spatial information regarding individual sources in multichannel mixtures can be utilised to separate sources, where they are non-identically mixed to each channel. In the case of music, audio objects are routinely located at different *panning positions* in the left-right stereo plane which has been exploited by techniques like DUET (Yilmaz, Rickard, and Yılmaz, 2004) and ADRESS (Barry, Lawlor, and Coyle, 2004) for estimating a time-frequency mask for each source that has a distinct position in the left-right stereo plane. While both the previously mentioned methods use binary masking which inherently introduces various artefacts (since sources are not sparse/non-overlapping, further discussed in Section 2.5), PROJET (Fitzgerald, Liutkus, and Badeau, 2016) utilises spatial projection hypothesis to allow soft-masking while being computationally less intensive than multichannel Wiener filtering (Liutkus and Badeau, 2015).

While these methods were some of the earliest ways of attempting to separate sources, beamforming is not well suited for music separation for two reasons. Firstly, the number of sources in a musical mixture is typically far greater in number than the number of recordings/channels available. Secondly, musical mixtures are typically not mixed in a physically constrained fashion, i.e. the sources are panned artificially and often consist of many stereo effects. Although beamforming is still quite relevant and used heavily in speech separation (Luo et al., 2020), such methods typically rely on capturing mixtures with fixed-geometry microphone arrays which are not viable for mixed music.

2.4 TIME-FREQUENCY MASKING USING MACHINE LEARNING

The foundation of the modern approaches used for source separation was laid down prior to the advent of deep learning using neural networks. While the methods used were theoretically quite similar to the modern approaches, the lack of sizeable datasets and hardware-accelerated computation limited the ability of the following methods to produce generalisable source separation methods.

Independent Component Analysis (ICA) assumes that the individual source components in an unknown mixture have the property of mutual statistical independence, and this property is exploited to algorithmically identify the latent sources. The joint probability density function is equal to the product of marginal densities if individual source components are statistically independent. The observed mixtures are expressed as products of a mixing matrix and vectors of statistically independent signals. Sources are separated by finding an inverse of the mixing matrix using only the observed mixture and assuming statistical independence of the sources. While these methods were applied to speech-related separation tasks, they are ill-suited for the task of music separation since the fundamental assumption for statistical independence between musical sources may be invalid.

Nonnegative matrix factorisation (NMF) attempts to represent the features of the sources via sets of basis functions and their respective activation coefficients (Weninger et al., 2014). In both NMF and ICA, constituent sources are modelled using determined magnitude or power spectrum elements, which by definition are non-negative as seen in real sound sources. Unlike ICA which utilises statistical independence, the NMF algorithm works by minimising the reconstruction error. Time-frequency energy distribution of sound sources is usually sparse, which means that most frequency bins and most frames are usually inactive. Sparse spectrograms often have a unique decomposition into nonnegative components, correlating to individ-

ual sound sources in the mixture. Durrieu, David, and Richard (2011) proposed an NMF-based multi-channel source separation model which achieved state-of-the-art for the source-image estimation task on multichannel music dataset in SiSEC 2011 (Araki et al., 2012).

Informed Source Separation covers a variety of methods where both the sources and mixtures are known at a certain encoding stage. In this scenario, certain additional information apart from the mixture observation is available to aid the separation task. This side-information may be in the form of a specific source model (Ozerov, Vincent, and Bimbot, 2012; Durrieu, David, and Richard, 2011) or be provided by a user (Smaragdis and Mysore, 2009) or by a partial transcription of the musical sources. In some cases, side information regarding the sources is encoded into the mixture file (Ozerov et al., 2013). Other methods include side-information embedded into the mixture imperceptibly using fingerprinting techniques (Liutkus et al., 2012).

2.5 TIME-FREQUENCY MASKING USING DEEP LEARNING

Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have found immense success in the field of image processing and natural language processing respectively, thus being able to leverage architectures that have proven to be useful in these domains for use in audio-related tasks has resulted in reasonable success. Such networks have achieved great performance in various audio-related tasks:

- Sound Event Detection (Cakir et al., 2017; Adavanne, Pertila, and Virtanen, 2017)
- Audio Scene Classification (Xu et al., 2018; Hershey et al., 2017)
- Automatic Music Transcription (Sigtia, Benetos, and Dixon, 2016; Cong et al., 2018)

- Onset Detection (Schlüter and Böck, 2013)
- Automatic Speech Recognition (Graves, Mohamed, and Hinton, 2013; Chiu et al., 2018)
- Speech Enhancement (Luo and Mesgarani, 2019; Pascual, Bonafonte, and Serrà, 2017)
- Music Source Separation (Uhlich et al., 2017; Jansson et al., 2017)

For the purpose of source separation, these DNNs can be used to predict dynamic time-frequency masks for each source, such that these masks when applied on the input mixture will filter out a specific source from the mixture (Grais et al., 2016). For time-frequency masking-based source separation methods, the separation is usually accomplished by filtering the mixture using one of three filtering methods: binary masking, soft-masking and multichannel Wiener filtering (MWF) which also incorporates techniques from spatial beamforming (Liutkus and Badeau, 2015).

2.5.1 *Fully-connected Feed-forward Network*

The first method to utilise deep learning for music source separation was introduced by Uhlich, Giron, and Mitsufuji (2015). The proposed method converts the input mixture waveform into a series of STFT frames with a rectangular window. For each frame, a number of preceding and succeeding (C) windows are used to generate the input vector. This concatenated vector (of length $2C + 1$) including additional contextual information is presented to a series of fully connected layers followed by a ReLU activation function.

The training of this experiment was done in a sequential fashion where additional layers were added, initialised and trained sequentially. This method was used to train models to separate instruments in an instrument class-based regression method. Two piano trios were used as an evaluation set, and

the proposed method including Wiener filtering marginally outperformed the based separation method using mel spectrogram by Spiertz and Gnann (2009).

2.5.2 *U-Net*

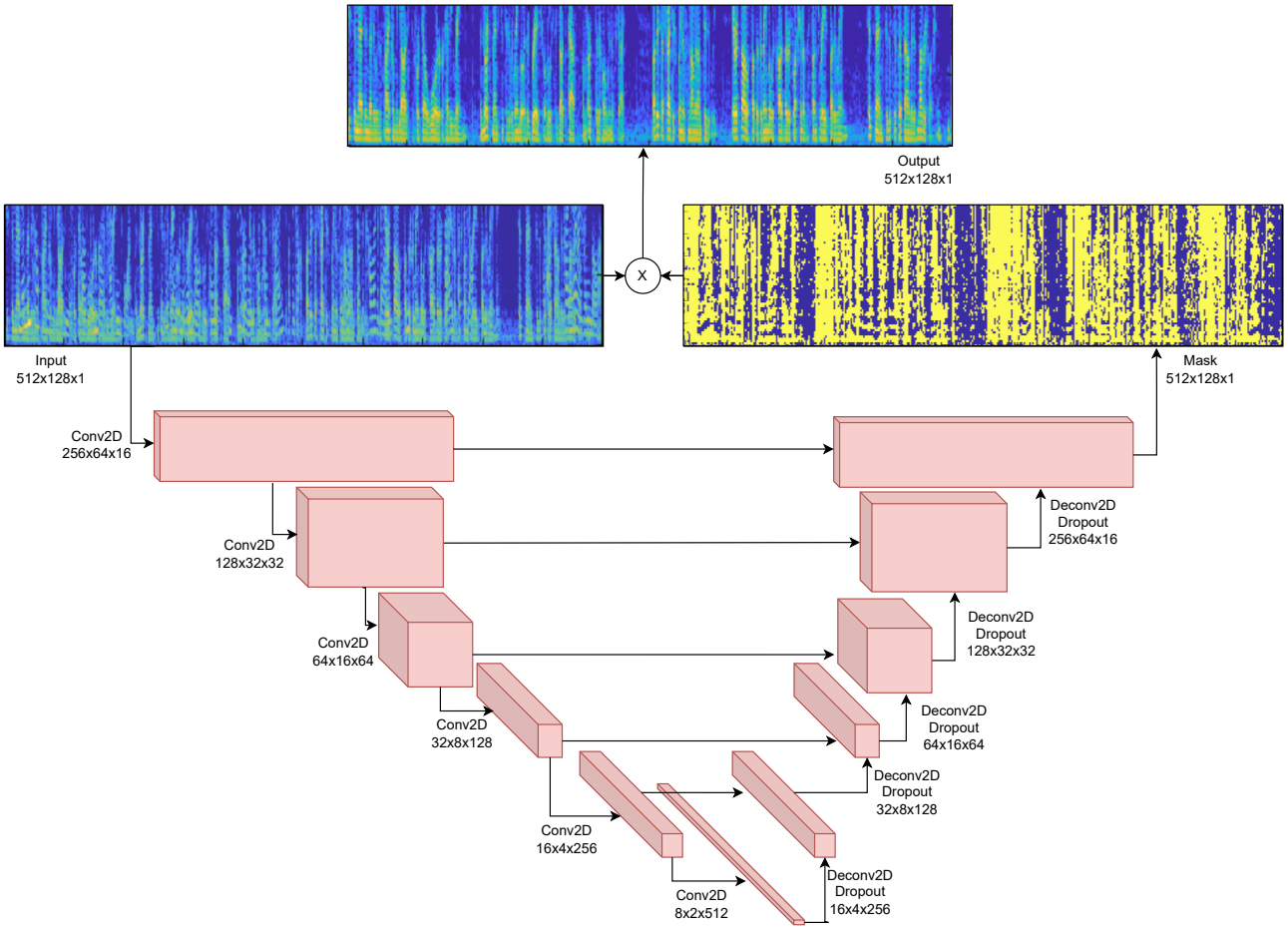

Figure 1: Magnitude spectrogram masking based U-Net singing voice separation architecture by Jansson et al. (2017).

The U-net architecture (see Figure 1) was introduced for the purpose of biomedical image segmentation in Ronneberger, Fischer, and Brox (2015). Jansson et al. (2017) adapts this architecture for singing voice separation by predicting a soft-mask to be multiplied with the time-frequency representation of the mixture to obtain the isolated sources. U-nets originally designed

for image reconstruction could suffer from small pixel shifts without significant deterioration in output quality. However, in the context of music separation, the mask predicted must be perfectly aligned to the input mixture. This prompted the authors to add skip-connections between each corresponding encoder and decoder layer. Two separate networks are trained to predict the vocal and instrumental masks. This architecture was subsequently presented as an open-source implementation in 2019 including additional instrument stems in Spleeter (Hennequin et al., 2019). The pre-trained model they released has been spun off into various commercial applications of source separation including iZotope RX 8[1], Virtual DJ[2], Acoustica[3] and Spectral layers[4].

The network architecture is largely leveraged from the generator architecture presented by Isola et al. (2017). In the given implementation, every encoder layer consists of strided 2D convolution with stride of 2 and kernel size 5×5, batch normalisation and leaky ReLU. The decoder layers mirror the encoder layers by using strided deconvolution (transposed convolution) and plain ReLU with skip connections from corresponding encoder layers. The datasets used for both the original paper and Spleeter were not publicly available.

2.5.3 *Open-unmix*

Improving on their previous work (described in Section 2.5.1), Uhlich et al. (2017) revolutionised music source separation as it was the first solution that showed generalisable results. This stride was a culmination of increased training data, a more standardised problem formulation which can be attributed to SiSEC 2016 (see Section 2.12) and improvements in deep learning architecture and training methods.

1 https://www.izotope.com/en/shop/rx-8-standard.html
2 https://www.virtualdj.com/stems/
3 https://acondigital.com/products/acoustica-audio-editor/
4 https://new.steinberg.net/spectralayers/

It improves on their previous work by training an additional deep learning network, which is based on a stack of bi-directional LSTMs which process each spectrogram frame sequentially. This in addition to the previously proposed dense model utilising $2C + 1$ frames was used simultaneously for inference, by combining their predictions with a fine-tuned blending ratio. Additionally, this work introduced the random mixing augmentation method and the use of multichannel Wiener filtering which greatly improved this model's performance. An open-source implementation of this model on pytorch was subsequently released by Stöter et al. (2019).

2.5.4 *DenseNet*

The core idea of DenseNets proposed by Huang et al. (2017) is to have all the outputs of preceding convolutional layers in a network be concatenated and presented as the input to the subsequent convolutional layer. This was leveraged for the task of audio source separation as it was considered that subsequent layers in the model would progressively be able to utilise information from previous layers to be able to reconstruct the source spectrogram for subsequent masking/separation. The proposed method for **Multi-scale Multi-band DenseNet** (Takahashi and Mitsufuji, 2017) combined the principle of sequential downsampling and upsampling with skip connections as introduced by the U-Net, but here with additional feature connections between layers. In addition, the method proposes to use distinct DenseNet stacks for different frequency bands of the input spectrogram. This was eventually extended to **D3Net** (Takahashi and Mitsufuji, 2020) which introduced a multi-dilated convolutional layer with multiple dilation factors within the same layer. This was able to perform comparably to Spleeter and Open-unmix with higher average performance on vocal separation.

2.6 TIME-DOMAIN SOURCE SEPARATION

Even though the described solutions in Section 2.5 are considered as the baseline for music source separation and do produce very impressive results, their performance is theoretically bounded by the imperfections introduced due to the lack of phase information of the extracted sources. Thus their maximum performance is always limited by the ideal ratio mask (IRM) (Vincent, Gribonval, and Plumbley, 2007). Another fundamental drawback of time-frequency masking approaches to source separation is that STFT is a generic signal transformation which is not optimised to encapsulate the most relevant features for a given instrument separation task Luo and Mesgarani (2019). Moreover, successful separation in the time-frequency domain requires a high-resolution frequency representation, which requires STFT calculation with long temporal windows. This inherently is a trade-off with the temporal accuracy of source separation, moreover increases the temporal averaging of features which is determined by the window size (typically 46 ms for music separation for a window size of 2048 samples at 44.1 kHz sampling rate).

Initial studies have explored the feasibility of time-domain separation using ICA (Choi et al., 2005) and NMF (Yoshii et al., 2013), which struggled to generalise and their performance was not comparable to their respective spectrogram-based counterparts. Few recent works have explored the use of deep learning for source separation in the time domain (Stoller, Ewert, and Dixon, 2018b; Luo and Mesgarani, 2019; Shi et al., 2019). The common approach in all these models is to replace the STFT step for feature extraction with a data-driven representation that is jointly optimised with an end-to-end training paradigm Luo and Mesgarani (2019).

2.6.1 *Time-domain U-Nets*

2.6.1.1 *Wave-U-Net*

Wave-U-Net (Stoller, Ewert, and Dixon, 2018b) is based on adapting the U-net architecture (see Section 2.5.2) to 1-D convolutions to process audio samples in the time domain. The network consists of a contracting half and an expanding half consisting of downsampling and upsampling blocks respectively, with skip connections between each corresponding level with same dimensionality. The downsampling structure computes increasingly higher-level features at longer time scales resulting in multi-scale features used for predicting the separated sources. Additionally, the implementation is able to improve on the challenges faced in Jansson et al. (2017) by incorporating linear interpolation during upsampling instead of strided convolution to avoid artefacts. The implementation is also able to utilise stereo inputs for separation and observe improvements.

2.6.1.2 *Demucs*

The **Demucs** architecture (Défossez et al., 2019) is similar to the Wave-U-Net (Stoller, Ewert, and Dixon, 2018b) architecture with variations in the encoder and decoder layer structures and an additional BLSTM layer at the end of the bottleneck between the encoder and decoder layers. While in Wave-U-Net the encoder and decoder layers are 1-D convolutional layers which downsample and upsample the latent space successively, the encoder and decoder layers in Demucs additionally use a Gated Linear Unit (GLU) after the 1-D convolution layer in each encoder and decoder layer. Similar to Wave-U-Net there are skip connections between each corresponding encoder and decoder layer. Unlike Wave-U-Net, this architecture uses transposed convolution for upsampling and downsampling, instead of using linear interpolation.

This was eventually extended to **Hybrid Demucs** by Défossez (2021) which incorporates the use of both spectrogram and raw audio as input and has a spectral and a temporal branch with shared layers allowing the model to learn features in both representations. The outputs of the spectral and temporal branches are then summed to generate the separated output. This model was the winner of the Music Demixing Challenge 2021 (Mitsufuji et al., 2021).

2.6.2 *TasNets*

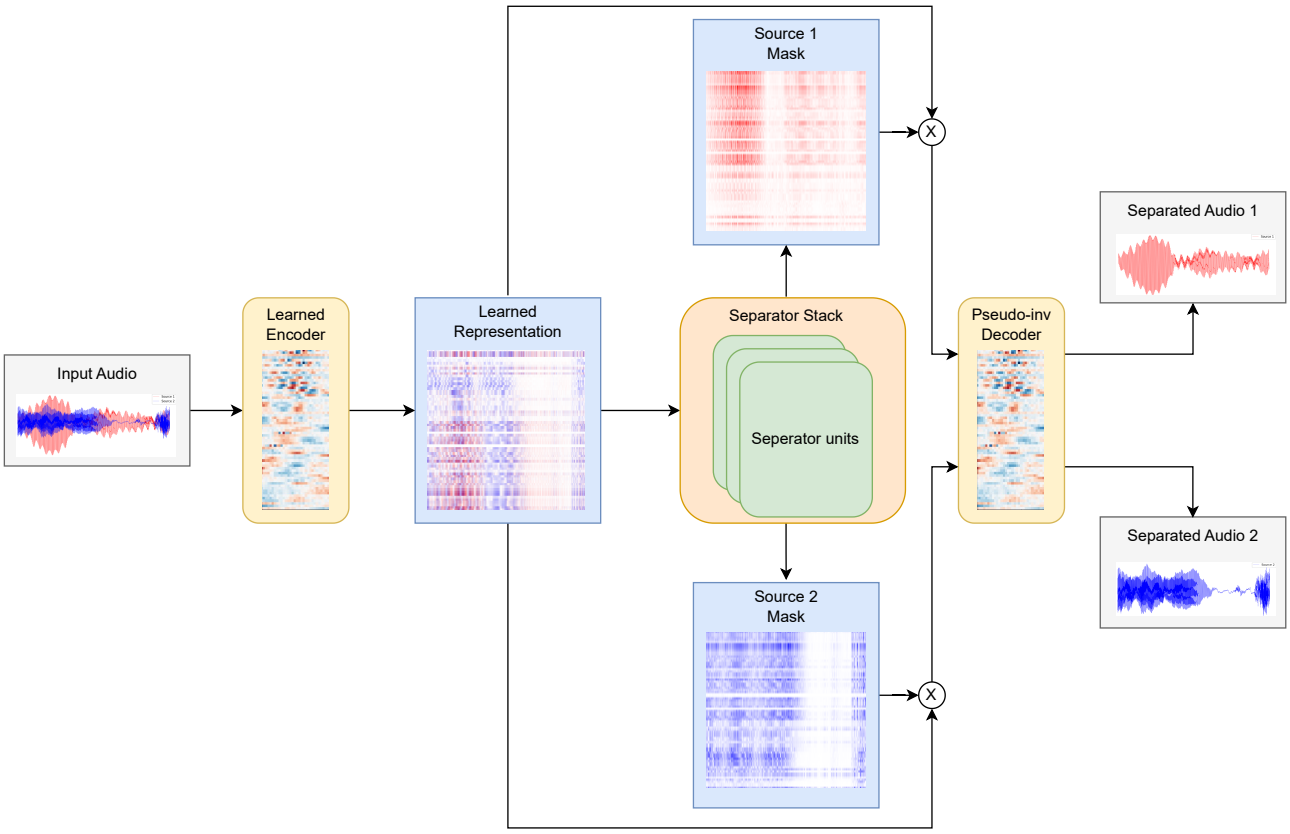

Figure 2: General separation pipeline for learnable filterbank (TasNet) based audio source separation models.

Recent advances in speech separation have shown great success in end-to-end methods that transform the time domain signal to a non-negative real-

valued, learned latent space to perform separation on. Such learned transformations inherently capture the phase information within the latent representation and thus do not require any explicit phase reconstruction, which has been a long-standing challenge for STFT-based source separation algorithms. These models usually comprise three parts (see Figure 2): an encoder for estimating the mixture weight, a separation module to predict the mask on this latent space of mixture weights and a decoder for reconstructing the source waveform from the masked mixture weight representation. Luo and Mesgarani (2018) introduced this method of end-to-end source separation for multi-speaker speech mixtures, in conjunction with utilising Permutation Invariant Training (PIT) (Yu et al., 2017). Subsequently, other works (Shi et al., 2019; Luo, Chen, and Yoshioka, 2020) Zeghidour and Grangier (2020) further explored new separation/masking modules and have since then produced great leaps in separation performance for speech and universal audio separation Wisdom et al. (2020) (described in Section 2.10).

2.6.2.1 *LSTM-TasNet*

The **TasNet** by Luo and Mesgarani (2018) attempts to bypass the phase prediction problem of T-F masking methods by directly extracting features from the raw audio as a time-series by learning an encoder based on a 1-D convolution layer which transforms the audio to a 2-D latent representation on which the masking is performed and subsequently inverted back to the time-domain audio by a pseudo-inverse 1-D convolution layer to transform the masked 2-D representation back to 1-D audio. Since this method operates using a 1-D convolution layer, unlike the STFT it can operate on arbitrarily short filter lengths, thus allowing the model to mask at a much higher temporal resolution than STFT domain masking as the latter are limited by the window size of the STFT which is typically much larger than the filter length used for the 1-D convolution filterbank encoder. It applies a series of LSTMs to then learn the temporal relationships and predict a mask, however due

to the short length of the 1-D filterbanks, the latent representation is much longer than the STFT based representation which makes it challenging for the LSTMs to incorporate information across long sequences.

2.6.2.2 *Conv-TasNet*

Conv-TasNet by Luo and Mesgarani (2019) is a fully-convolutional time-domain audio source separation network for multi-speaker speech separation. ConvTasNet improves on the original TasNet by using a temporal convolutional network (TCN) to calculate the masks instead of a stack of bidirectional-LSTM blocks. This model is both significantly smaller than typical STFT based models and also was able to surpass ideal time frequency masking methods for separation in the frequency domain. This model is used in this thesis for experiments is described in Section 5.3.

However, unlike the original TasNet, due to the nature of the dilated convolution based TCN stack, the receptive field of the separator stack is limited. The separator stack architecture/parameters have to be modified if the length of learned the latent representation changes due to increase in sampling rate, input segment length or reducing the encoder filterbank hop size. This becomes a challenging network parameter optimisation task due it's implications on the GPU VRAM consumed by the model during training. Further details regarding the implementation can be found in Section 4.3.1.1 and hyperparameter optimisation of this model is presented in Section 4.3.4.

2.6.2.3 *DPRNN*

Dual-path recurrent neural network (DPRNN) Luo, Chen, and Yoshioka (2020) comprises of replacing the separator architecture of TasNet with a 2 level hierarchical (local-global) RNN structure (see Figure 9) allowing effective modelling of long sequences without increasing optimisation difficulty drastically. The structure splits the input segments into smaller chunks and

interleaves two RNNs, one at intra-chunk level and one at inter-chunk level. The intra-chunk RNN first processes the smaller segments independently and subsequently the inter-chunk RNN aggregates the information across all the chunks allowing complete utterance level inference. Due to this, unlike the TCN masking network in Conv-TasNet which has a limited receptive field, DPRNN is able to fully utlise information across the input audio segment and achieve superior performance with a smaller model size. This model is used in this thesis for experiments described in Section 5.3. Further details regarding the implementation can be found in Section 4.3.1.2 and hyperparameter optimisation of this model is presented in Section 4.3.1.2.

2.6.2.4 *DPTNet*

Dual-path Transformer Network (DPTNet) Chen, Mao, and Liu (2020) is based on a similar 2 level RNN based separator architecture as introduced in DPRNN. It replaces the BLSTM layers in the separator architecture with 2 modified transformer attention heads (originally introduced by Vaswani et al. (2017)) which are able to handle temporal relationships for long sequences more effectively. They achieve the state-of-the-art speech separation performance with a small model size due to using the transformer encoder network instead of the BLSTM in DPRNN. This model is utilised as the baseline for all experiments described in Section 5.3, Section 5.4 and Section 5.5. Further details regarding the implementation can be found in Section 4.3.1.2 and hyperparameter optimisation of this model is presented in Section 4.3.4.

2.6.2.5 *SepFormer*

SepFormer by Subakan et al. (2021) (extended by Subakan et al. (2023)) improves on DPTNet by removing the recurrent unit in DPTNet's feed-forward network within the modified transformer unit with a simple feedforward layer (shown in Figure 3), which makes training much more parallelisable

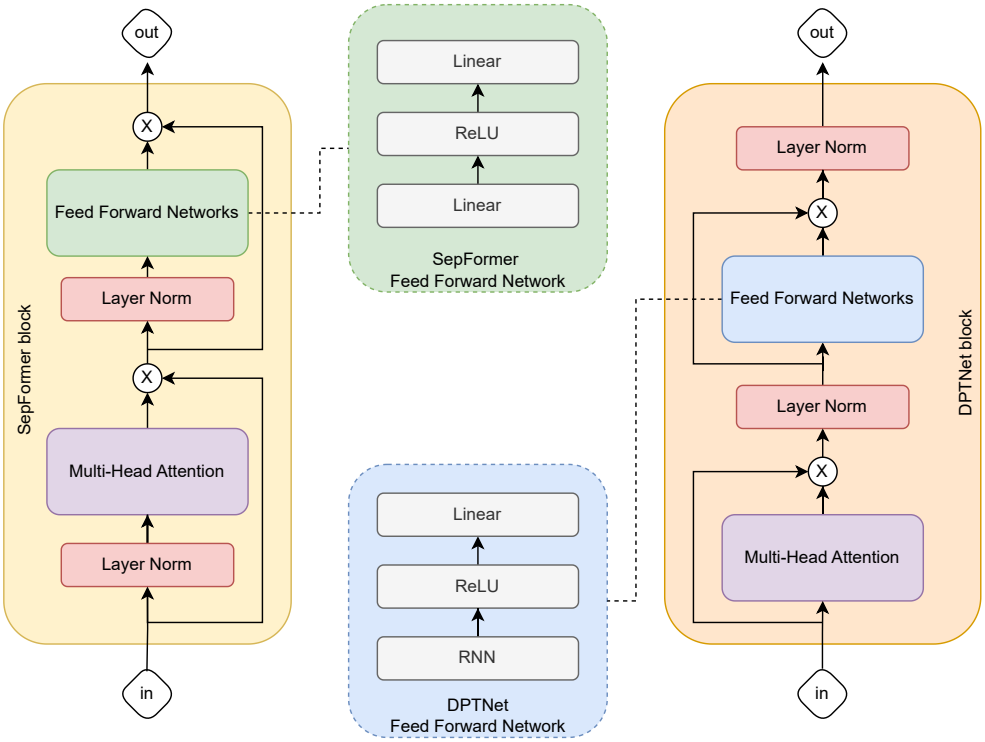

Figure 3: Comparison of multi-head attention-based transformer blocks of DPTNet and Sepformer.

and reduces the computational requirements for both training and inference. Thus its performance is at par with DPTNet but with reduced resource utilisation. This model may be considered the best-performing TasNet architecture as of this work.

2.7 COMPLEX-DOMAIN SOURCE SEPARATION

Another approach to surpassing the glass ceiling introduced by the lack of accurate phase estimation is to actually attempt to estimate the complex domain spectrogram of the target sources. Estimating the phase of constituents of a mixture has been considered a significant challenge, thus preliminary works (as described in Section 2.5) only focussed on the estimation of the magnitude spectrogram as the training target. While a method for estimating phase by Griffin and Lim (1984) has been known, it involves significant

computational expense. The methods discussed in this section propose estimating the complex spectrogram as the output representation to bypass this problem. While a large variety of complex-domain separation methods have been introduced in recent years, they can largely be categorised into two groups based on whether they utilise real-valued operations or complex-valued operations.

2.7.1 *Real-valued Systems*

This subsection describes source separation models capable of handling complex-valued spectrograms, but rely on using only real-valued tensors within the model.

2.7.1.1 *Discretised phase classification*

PhaseNet by Takahashi et al. (2018) proposes to treat the phase estimation as a classification problem by discretising the phase space and designs a DNN that predicts the phase for the target spectrogram independent of the magnitude prediction. This is motivated by the fact that predicting the phase as a regression problem is challenging due to the wrapping around of phase to its principal value that lies within a typical range of $(-\pi, \pi]$, which results in the phase spectrogram having no clear structure. Thus casting the problem as a classification problem in a discretised space results in an improvement in SDR (defined in Section 2.2) performance and perceptual quality for both Speech Enhancement and Music Source Separation tasks compared to the use of mixture phase for signal reconstruction.

A similar idea by Le Roux et al. (2019) **Phasebook** uses a similar discrete phase classification technique. Instead of treating the phase classification as an independent problem to magnitude regression, it also treats magnitude

estimation as a quantised classification technique and trains a DNN to predict both simultaneously.

2.7.1.2 *MLP-based Complex Masking*

The ideal ratio mask (IRM) which is the theoretical upper limit for deep-learning methods presented in Section 2.5 is designed to accurately estimate the magnitude response of a target from a mixture, while assuming the mixture phase for the separated audio. Williamson, Wang, and Wang (2016) defined the complex Ideal Ratio Mask (cIRM) and trained a DNN to simultaneously predict the real and imaginary components of the target spectrogram, which resulted in improved perceptual quality for speech separation. The cIRM can be defined as below, where $S \in \mathbb{C}$ represents the separated signal's complex spectrogram, $M \in \mathbb{C}$ represents the predicted complex-ratio mask and $Y \in \mathbb{C}$ represents the input mixture's complex spectrogram.

$$\begin{aligned} S_r + iS_i &= (M_r + iM_i) \cdot (Y_r + iY_i) \\ &= (M_r Y_r - M_i Y_i) + i\,(M_r Y_i + M_i Y_r) \end{aligned} \tag{13}$$

Complex numbers are represented here in the cartesian coordinate representation such that given $X \in \mathbb{C}$, it can be represented as $X = X_r + iX_i$, where X_r represents the real component of X and iX_i represents the imaginary component of X.

The DNN used in Williamson, Wang, and Wang (2016) was a simple MLP with 3 hidden layers. The model takes as input a set of crafted features such as MFCCs, cochleagram response and some high-level features extracted from a 64-channel gammatone filterbank. For the output layer, the MLP predicts 2 vectors, one for the real mask, and the other for the imaginary mask. This method does not involve any complex operations within the network architecture. This was eventually extended by Tan and Wang (2019) treating

the MLP layers as an encoder-decoder and inserting LSTM layers between these MLP layers while the input and output of the model was the mixture complex spectrogram and the cRM (complex Ratio Mask).

Deep-ResUNet by Kong et al. (2021) aims to achieve a cIRM-like mask without solely relying on the estimated real and imaginary masks due to the challenging unbounded nature of these masks. Instead, the model only takes the magnitude spectrogram as an input to a very deep U-Net ($>$ 100 layers) and simultaneously predicts the magnitude spectrogram and a magnitude mask and the real and imaginary masks for the separated output. These 2 real and imaginary mask predictions are then used in conjunction with the input phase to predict the output phase spectrogram, and the magnitude spectrogram is generated by combining the magnitude mask and the magnitude spectrogram prediction. The work also compares their performance with respect to DCUNet. It shows that the significantly deeper UNet structure significantly improves the separation performance for the music demixing task, however at a significantly higher computational cost.

Band-split RNN by Luo and Yu (2023) operates on a similar paradigm which involves predicting real and imaginary masks to be applied to the complex mixture spectrogram directly. However, it also introduces a feature extraction module that takes the complex spectrogram of the mixture $X \in \mathbb{C}^{F \times T}$ as input and splits it into K subband spectrograms $B_i \in \mathbb{C}^{G_i \times T}, i = 1, ..., K$. The real and imaginary parts of the sub-band spectrograms are then concatenated and passed through a fully connected layer to generate the latent representation. This latent representation is then processed similarly to the masking stage of the dual-path RNN (as described in Section 2.6.2.3). However, it must be noted that the separator stack and subsequently the model size (50M parameters) for Band-split RNN is significantly larger than the DPRNN stack as it uses up to twice as many repeat layers, and 4 times as many hidden units.

Band-Split RoPE Transformer by Lu et al. (2023) extends the band-split RNN architecture to a DPTNet-like separator stack (described in Section 2.6.2.4). It is the most recent state-of-the-art for music separation and the winner for the Music Demixing track for the Sound Demixing Challenge 2023 (Fabbro et al., 2023). However, it must be noted that this was achieved with an extremely compute-intensive model with more than 93.4 million parameters (10x the parameter count of experiments presented in this work with DPTNet). This is attributed to it using 12 repeat layers (2x of our implementation) and 8 heads for each transformer (2x of our implementation). This increased complexity required 16 Nvidia A100-80GB GPUs for 4 weeks to train the model (38x of our experiments).

2.7.2 *Complex-domain Neural Networks*

While all the previously mentioned techniques rely solely on real-valued neural networks to deal with complex spectrograms for source separation, few attempts have been made to enable neural networks to be able to handle complex numbers and operate in the complex domain directly. The first reported attempt for true complex domain source separation was by Lee et al. (2017) which reported limited success as network architectures were still primitive MLPs at the time. The methods mentioned below are the only works that have shown reasonable success at complex-domain source separation using complex-domain neural networks.

2.7.2.1 *DCUNet*

Phase-aware Deep Complex U-Net by Choi et al. (2019) modifies the original U-Net based separation solution (as described in Section 2.5.2) by replacing all convolution blocks with complex convolution blocks. The model is also modified to predict a cRM instead of a magnitude mask. A crucial difference

introduced by complex ratio masking, is that the real and imaginary masks predicted are not bounded, i.e. the mask values may be greater than zero. This is due to the fact that the energy within the real and imaginary parts would not be conserved but the RMS of the real and imaginary parts would. Due to this, the authors propose to use the polar representation to predict the cRM such that the magnitude mask output range for the network can be bounded. They restrict the magnitude part of the predicted cRM to be bounded between $(0, 1]$ by using a hyperbolic tangent non-linearity and the subsequent phase mask is extracted by dividing the predicted output by its magnitude. This model is used as the complex-domain baseline for the experiments described in Section 5.5.

2.7.2.2 *DCCRN*

Deep complex convolution recurrent network (DCCRN) (Hu et al., 2020) utilises similar complex convolution layers as DCUNet but instead of a direct regression U-Net based approach, it uses LSTMs to model temporal context instead of stacked dilated convolutions. This results in significantly lower $(1/6^{th})$ trainable parameters without any performance compromise.

2.8 PERMUTATION INVARIANT SOURCE SEPARATION

Permutation invariant training Yu et al. (2017) solves the problem of source-label ambiguity in source separation. Label ambiguity occurs in source separation tasks with mixtures containing multiple sources of the same type/-class (for example speech mixtures, singing choirs, string quartets). Although one may be able to decompose such mixtures into classes like male/female for speech, or soprano/alto/tenor/bass for music, such classifications are often not consistent and limits model training to very specific mixture combinations. Using permutation invariant training, a model is able to separate mul-

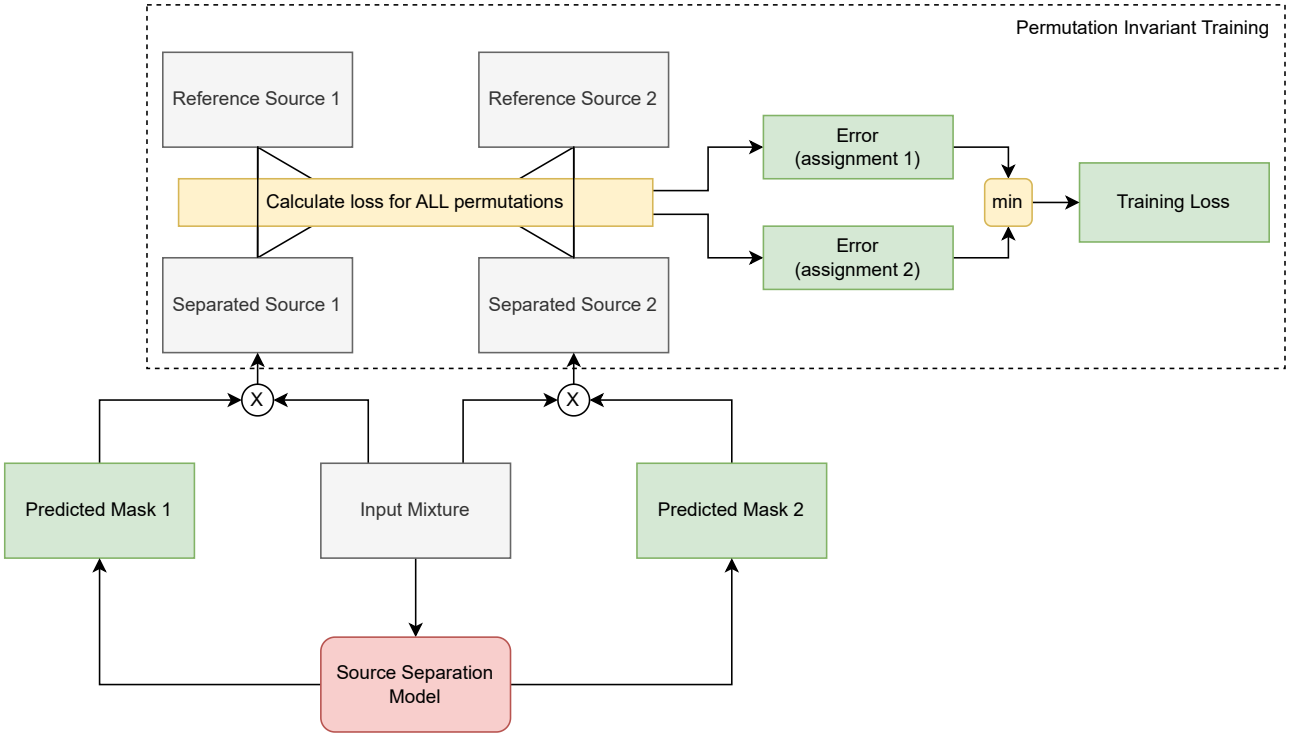

Figure 4: Loss calculation for a mixture of 2 sources using permutation invariant training.

tiple instances of similar sources. This method presents the reference sources to the model as a set instead of an ordered list. For each loss calculation, permutation invariant training computes the loss for each target-channel permutation possible for the given segment, and considers the minimum loss calculated across all such permutations (depicted in Figure 4). Given a loss-function L comparing the difference between the predicted instrument signal $\hat{s}_i$ and target instrument signal s_i, the training objective $\mathcal{L}$ can be defined as Equation 14, where $\{\hat{\mathbf{s}}_i(t)\}_\pi$ denotes a given permutation order of the separated outputs from all possible permutations for output-target assignments Π_I.

$$\mathcal{L} = \arg\min_{\pi \in \Pi_I} \mathrm{L}\left(\{\hat{\mathbf{s}}_i(t)\}_\pi, \{\mathbf{s}_i(t)\}\right) \tag{14}$$

This has helped recent TasNet-based approaches in speech separation (discussed in Section 2.6.2) to make significant improvements compared to the previous state-of-the-art approaches based on deep clustering (Hershey et al., 2016).

Due to the nature of the PIT training criteria, the output-target assignment may change across processing frames. This implies that for longer sequences, if the output of one channel is concatenated across frames, the resulting audio may have different sources present across different frames. This may be improved by applying a speaker/source tracking algorithm across frames and fixing source mismatches by swapping misaligned frames across channels as a post-processing step.

2.8.0.1 *Variable Sources*

Permutation invariant training works by comparing all possible permutations for a given number of sources. The knowledge of the number of sources present in the mixture is crucial as it determines the number of output channels and thus the permutations possible. Thus, separate models need to be trained for any given number of sources in the input mixture. Knowing the number of sources in the mixture is often a challenge as the number of sources present at any point may vary across a given input segment. Moreover, there should be an advantage in sharing knowledge across models designed to separate different numbers of sources since the separation task across different levels of polyphony is the same.

OR-PIT: Takahashi et al. (2019) suggests using one-and-rest permutation invariant training where the model is trained to separate only one source from the mixture at a time. Such a model may be applied recursively to separate any number of sources present in the mixture. The loss function also reduces the complexity of calculating all different permutations at each

epoch, since only N permutations are possible while selecting 1 from N.

A2PIT: Luo and Mesgarani (2020) introduces Auxiliary Autoencoding Permutation Invariant Training (A2PIT) where a model is trained for a maximum of sources that can be separated by it. For use cases where the mixture contains a lesser number of sources than present in the mixture, the remaining output channels are trained to reconstruct the original mixtures (like an autoencoder). Their experiments prove that the performance of the separation improves in this case, as compared to training the model to predict silences for the surplus output channels.

2.9 GAN BASED SOURCE SEPARATION

In generative adversarial networks (GANs) (Goodfellow et al., 2014) a generator network is trained to produce samples from a given target distribution. To train such a generator network, a discriminator network is employed to distinguish between “real" samples from the actual distribution and “fake" samples generated using the network. The generator is trained in such a manner that it consistently takes feedback from the discriminator and tries to optimise its output in order to fool the discriminator. For source separation, such architectures have been employed with generators based on the usual source separation algorithms, which can be either on TF domain or time domain (Pascual, Bonafonte, and Serra, 2017; Fan, Lai, and Jang, 2018; Stoller, Ewert, and Dixon, 2018a). In an adversarial training approach for source separation, the discriminator acts like an intelligent loss function. Lately, parallels have been drawn between discriminator architectures and neural network based evaluation metrics (Fu, Liao, and Tsao, 2020). The discriminator might take only the separated source vector as the input, or in some implementations, it

uses both the source and mixture vectors as input (Stoller, Ewert, and Dixon, 2018a; Fan, Lai, and Jang, 2018).

2.10 UNIVERSAL SOUND SEPARATION

Universal sound separation unifies all the previous tasks and further extends it by removing all known priors about a mixture. It is the task of separating all constituent sounds of a mixture regardless of the number or the type of these sound sources. While the definition of the task is very broad, different approaches have been presented which aim to tackle this challenge with some success and inherent limitations based on their approach, of which some are discussed in this section.

Mix Invariant Training (MixIT) by Wisdom et al. (2020) presents an unsupervised training method that enables universal source separation by extending permutation invariant training. It relaxes the permutation invariant training criteria by allowing the reference and output target assignment to be a sum of a subset of the predicted outputs. Using this, the model can be trained by presenting a mixture of mixtures as well as mixtures of isolated sources, which enables it to be trained in an unsupervised or semi-supervised fashion without requiring all training examples to be clean/isolated sound sources.

Hyperbolic Audio Source Separation by Petermann et al. (2023) proposes to extend typical time-frequency masking based separation methods by additionally computing a hyperbolic embedding for each time-frequency bin of the mixture spectrogram. The embedding learnt for each time-frequency bin in the mixture is classified using a hyperbolic soft-max layer which assigns it to every hierarchical source class present in the bin (example: *horns* $\subset$ *brass section* $\subset$ *music*). The hierarchical structure of class labels results in enabling the separation of sources with unseen or ambiguous labels. Such a source will likely be classified as one of the higher-level class labels and thus can

be separated, as long as the interfering sources do not fall within the same class.

Separate anything you describe by Liu et al. (2023) presents a language-queried universal source separation method. The method consists of a language-query encoder (QueryNet) and uses it to condition a time-frequency masking based source separation model (SeparationNet). The QueryNet takes the natural language query and reduces it to a latent representation of fixed length and dimensionality using the large-scale contrastive language-image pre-trained model (CLIP) (Raffel, 2016) and large-scale contrastive language-audio pre-trained model (CLAP) (Elizalde et al., 2023). The CLIP encoder learns to map text embeddings to the same space as visual embeddings, which is used by the authors to train the model on large-scale unlabelled audio-visual data by only using the visual embeddings. They also utilise CLAP which maintains a similar latent representation as CLIP, which provides better time-aligned text-audio context. However, CLAP alone is limited by the diversity and scale of its training data due to lack of large labelled text-audio training datasets. The SeparationNet used in this work is based on the deep-ResUNet by Kong et al. (2021).

.

2.11 MULTI-TRACK MUSIC DATASETS

Most modern solutions to audio source separation are based on deep-learning, which requires bleed-free(isolated) source signals to train source separation algorithms. This section provides an overview of various publicly available datasets comprising bleed-free multitrack recordings of musical sources.

2.11.1 *Music Demixing Datasets*

DSD100: Demixing Secrets Database (DSD100) was released as a part of the MUS task for SiSEC 2016 (Liutkus et al., 2017). It contains four semi-professionally produced stereo stems per track: vocals, drums, bass and others which when summed up provide a realistic mix. This dataset was generated using multitracks from the 'Mixing Secrets' Free Multitrack Download Library. Along with the stems+mixture data, Python and MATLAB toolboxes were also provided for streamlining the data pipeline.

MUSDB18: The MUSDB18 corpus (Rafii et al., 2017) was released for the musical source separation (MUS) challenge from SiSEC 2018 (Stöter, Liutkus, and Ito, 2018) and is effectively an extension of the DSD100 dataset. The authors utilised additional multitracks from MedleyDB and other material provided by Native Instruments. This dataset contains a total of 150 songs with an overall duration of 10 hours. The design of the dataset is identical to DSD100 in terms of the processing and representation of stereo stems.

2.11.2 *Multi-track Music Datasets:*

MedleyDB: MedleyDB (Bittner et al., 2014) is a dataset of royalty-free multitrack music recordings with instrument activity annotations. The advantage of MedleyDB over other multitrack datasets is the availability of both RAW and STEM files and an accompanying YAML metadata file that contains the hierarchical structure of the STEM and RAW tracks and other information regarding the multitrack. The metadata consists of song-level metadata like: artist, title, composer and bleed while the stem-level metadata contains the instrument type.

Slakh: The Synthesised Lakh dataset(SLakh) by Manilow et al. (2019) was published for music source separation research and is made up of high-

quality rendering of instrumental mixtures. It uses the Lakh MIDI dataset (LMD) (Raffel, 2016) to provide professional virtual instrument renders for corresponding stems. The mixing procedure of this dataset follows using preset effects on each patch (randomly assigned per instrument class) and then normalising loudness across tracks using the algorithm defined by ITU-R BS1770-4 (BS, 1770) with uniform gains on each track. However, models trained on this dataset have not resulted in generalisable performance on real datasets. This could be due to the automated nature of their MIDI score collection process and the randomised selection of rendering parameters. This results in poor realism and diversity of their rendered instrument tracks. Wrongly assigned instrument labels/program numbers in MIDI result in poor-quality audio renders.

URMP: The URMP dataset (Li et al., 2018) is a multi-modal multi-track dataset comprising audio-visual recordings of 44 chamber ensemble pieces, ranging from duets to quintets. Unlike most other multi-track datasets of chamber ensembles, this dataset takes particular care to ensure that the individual instrument recordings do not contain bleed. In order to achieve this, each instrument was recorded in a separate take, subsequently, each of these recordings was downmixed together with the other instruments with reverb. Because this dataset was recorded for chamber music but with individual takes for each performer, achieving perfect synchronisation and timbral coherence across the performers proved to be challenging during the creation of this dataset. The recordings for the individual parts were conducted in an anechoic sound booth. The microphone used for these recordings was an Audio Technica AT2020 condenser microphone. The recordings were then manually time-aligned, edited and remixed to be well-synchronised with other performers.

TRIOS: The TRIOS dataset (Fritsch, 2012) is a set of 5 short recordings of chamber ensemble trios. The dataset consists of multi-track recordings where each performer was recorded in a separate take, thus resulting in bleed-free

recordings. This dataset is also used for evaluation in some experiments in this thesis.

2.11.3 *Choral Music Datasets*

Creating a bleed-free multitrack dataset of choral singing poses technical challenges due to the simultaneous singing of multiple performers in a typical choir setting. While recording individual singers within the ensemble in most datasets has been accomplished using highly directional microphones to minimise leakage from other singers, this approach is not flawless and still yields some degree of bleed.

The Choral Singing Dataset (CSD) (Cuesta et al., 2018) is an openly accessible multitrack dataset showcasing Western choral music. It encompasses recordings of three songs in the SATB format, each performed in a distinct language (Catalan, Spanish, and Latin). These performances feature a choir comprising 16 singers, organised into four per section. Individual sections of the choir were independently recorded, utilising microphones to capture each singer's voice distinctly. Fo trajectories and section-wise MIDI notes are provided for each song. The total audio duration is approximately 7 minutes, categorising it as a relatively small dataset. These recordings exhibit some leakage from adjacent singers within the same section, thus making them unsuitable for the work presented in this thesis.

The **Dagstuhl ChoirSet (DCS)** (Rosenzweig et al., 2020) is a dataset featuring ensemble singing recordings of two songs in Latin and Bulgarian. Additionally, the dataset incorporates a series of vocal exercises encompassing scales, cadences, and intonation exercises. Recordings were made using combinations of handheld dynamic microphones, headset microphones, throat microphones, and a stereo pair, capturing the performances of 13 singers grouped into uneven SATB sections. All singers were recorded simultane-

ously, resulting in high leakage in individual tracks. The total audio duration spans approximately 55 minutes.

The **Bach Chorales and Barbershop Quartet Dataset (BCBQ)** is a commercially available multitrack dataset employed in the experiments detailed in (Schramm, Benetos, et al., 2017). Comprising 26 Bach Chorales and 22 Barbershop Quartets performed by an SATB quartet, with one singer per part, the total audio duration of BCBQ is approximately 104 minutes. Each singer in the quartet was meticulously recorded in a professional setup, ensuring the absence of inter-singer leakage in the recordings. BCBQ includes individual audio tracks for each singer, as well as the combined mixture of the four voices. While the Bach Chorales songs involve 2 female and 2 male singers, the Barbershop Quartets comprise of all 4 male singers. This dataset was used for the choral separation experiments described in this thesis and is accessible at http://pgmusic.com. Due to the commercial nature of this dataset, other works on choral music separation have not used this dataset and only rely on the previously mentioned datasets which contain bleed, thus both their performance is vastly limited by the same and their evaluation metrics presented also may not be directly comparable as their reference tracks would include bleed.

MedleyVox by Jeon et al. (2023) is an evaluation dataset for vocal ensembles, constructed from the MedleyDB dataset. This dataset is created based on the manual annotations released in this work, which allows selecting tracks from MedleyDB which have harmonised singing content without bleed.

jaCappella by Nakamura et al. (2023) is a dataset of 35 vocal ensemble pieces of Japanese children's songs consisting of 6 vocal parts (SATB + lead vocals + vocal percussion). This is a bleed-free dataset of 34 minutes including music from various genres, although excluding choral music. This data was released after the experiments related to choral music separation presented in this thesis were conducted, and thus are not included in this thesis.

2.12 PUBLIC EVALUATION CAMPAIGNS FOR MUSIC SEPARATION

The Signal Separation Evaluation Campaign (SiSEC) (Vincent, Araki, and Bofill, 2009) started in 2008 as a source separation challenge which was held every 1-2 years, primarily focussed on speech and music separation. In this section, the evolution of music separation research is presented in the context of the music separation tasks that have been featured as a part of public evaluation campaigns through the years. Subsequently, the parallel evolution of multi-track music datasets is also presented.

SiSEC 2008: The original problem formulation of the SiSEC challenge described the source separation problem as a 4 step problem, with different entrants presenting solutions for each of the 4 sub-tasks mentioned below:

1. Source Counting
2. Mixing System Estimation
3. Source Signal Estimation
4. Source Spatial Image Estimation

The presented solutions for these sub-tasks were then evaluated in the context of datasets consisting of different mixture scenarios. Although the "Professionally produced music recordings" dataset was introduced as D4 in SiSEC 2008 (Vincent, Araki, and Bofill, 2009), it was only presented as a small test dataset of 2 recordings and was not seen as a standalone task in itself.

SiSEC2010 (Araki et al., 2010) expanded the music separation dataset to 5 full-length recordings as a test set. SiSEC 2010 also saw introduction of a larger variety of test datasets (increased to 7 from 4), however this resulted in dilution of number of solutions received for each of these datasets. This prompted the discussion of considering the different test datasets as distinct tasks for future SiSEC challenges.

In **SiSEC 2011** (Araki et al., 2012), the professionally produced music mixtures dataset (D3), was tied specifically to the task T3 - Source Spatial Image Estimation. This was the first time the music separation task was formalised as a separate task, and saw unexpectedly high participation with an increased interest in NMF-based techniques (described in Section 2.4).

SiSEC 2013 (Ono et al., 2013) expanded the professionally mixed music dataset and included 20 stereo music pieces (8 pieces for training and 12 pieces for testing respectively). The 8 pieces in the training set are presented as full-length stereo mixes of music sources, while the evaluation set consisted to 20-second excerpts of mixed music. This challenge also saw surprisingly high participation in the music separation task. Moreover, there was a strong correlation observed between the performance of these systems on the full-length training set and the 20-sec excerpt test set, indicating stable performance of these music separation solutions.

SiSEC 2015 (Ono et al., 2015) saw the first significant jump in separation quality for music separation with a deep-learning based solution by Uhlich, Giron, and Mitsufuji (2015) (described in Section 2.5.1). This was in part supported by the new dataset Mixing Secret Database (MSD100) released for the music separation task, which consists of a 100 songs of various styles. This dataset introduced a new formalism for the music separation task (as described in Equation 5) where each song consists of 4 instrument stems: drums, bass, vocals and "other accompaniments". Similar to previous SiSECs, the music separation task attracted the most number of participants.

SiSEC 2016 (Liutkus et al., 2017) the MSD100 dataset was enhanced and the Demixing Secrets Database (DSD100) was introduced. It was a revision of the previous dataset, now including professionally mastered versions for the drum, bass and vocal stems (as compared to unmastered instrument stems/recordings in MSD100) such that the linear sum of these stems resulted in more realistic mastered musical mixtures. This challenge again saw overwhelming participation for the music separation task. This challenge for the

first time reported that supervised systems consistently outperform blind separation systems, and also reported deep-learning based separation methods with data augmentation techniques involving random track mixing (as described in Section 2.5.3) to result in improved generalisability and state-of-the-art performance.

SiSEC 2018: (Stöter, Liutkus, and Ito, 2018) introduced the MUSDB18 dataset (Rafii et al., 2017). In this challenge, it was observed that all solutions based on deep-learning with additional training data consistently outperform all other systems. It is also reported in this challenge that very different deep-learning based solutions perform comparably and that the performance difference seems to largely be associated with the use of larger training datasets. The best-performing architecture for vocal separation (independent of training data) in this challenge was found to be the Multi-dilated DenseNet by Takahashi and Mitsufuji (2017) (described in Section 2.5.4).

Music Demixing Challenge 2021: The music separation task was eventually separated from the other speech challenges into its own task as the Music Demixing Challenge in 2021 Mitsufuji et al. (2021), which focussed on vocals, bass and drum stem separation as the main goal. This challenge introduced MUSDB18-HQ (Rafii et al., 2019) as a new standardised training dataset. It also addressed concerns of overfitting on the training dataset by introducing an unseen test dataset MDXDB21. The best performing method for this challenge (without use of additional training data) was reported to be Hybrid-Demucs (described in Section 2.6.1.2). This challenge also introduced the definition of stem-averaged SDR_{song} (defined in Equation 15) which may be used as an objective measure to compare different deep-learning architectures trained on the standardised training dataset MUSDB18-HQ and tested on MDXDB21.

$$\text{SDR}_{\text{Song}} = \frac{1}{4}\left(\text{SDR}_{\text{Bass}} + \text{SDR}_{\text{Drums}} + \text{SDR}_{\text{Other}} + \text{SDR}_{\text{Vocals}}\right) \tag{15}$$

System	SDR_{Song}	SDR_{Bass}	SDR_{Drums}	SDR_{Other}	SDR_{Vocals}
Open-unmix (2016)	5.18	5.40	5.71	3.56	6.07
D3Net (2020)	5.80	5.74	6.18	4.30	6.97
Hybrid Demucs (2021)	5.81	6.48	6.44	3.96	6.37
Band-split RNN (2022)	6.142	5.628	6.534	4.425	7.983
Band-split RoPE* (2023)	9.965	11.153	10.269	7.075	11.363
IRM+MWF	9.78	9.39	9.59	8.84	11.30

Table 1: Comparison of state-of-the-art music separation architectures over the years tested on the MDXDB21 dataset. All models were trained on MUSDB18-HQ, except Band-split RoPE* which was trained on a larger private dataset. The theoretical upper-limit for magnitude spectrogram masking-based solutions with multi-channel Wiener filtering on this test dataset is also presented as IRM+MWF.

Sound Demixing Challenge 2023: This was eventually further extended to the Sound Demixing Challenge Fabbro et al. (2023) to include other tasks such as Cinematic Demixing. The winner of the Music Demixing challenge was the Band-split RoPE Transformer architecture by Lu et al. (2023) (described in Section 2.7). However, it must be noted that the training data limitation was removed for the main leaderboard in the Music Demixing Challenge 2023. Results for state-of-the-art models across the years which have been benchmarked on the MDXDB21 test dataset are presented in Table 1. While the numbers are indicative of the capabilities of these architectures, it is noteworthy that the significant performance jump between Band-split RNN and Band-split RoPE is largely due to the increased training data size used for the latter. It's also interesting to note that the parameter count of the newer solutions is significantly larger (details in Section 2.7). However, they do not report significant performance improvement unless the size of the training dataset is also scaled accordingly.

2.13 ENSEMBLE SEPARATION

The work presented in this thesis involves the separation of multiple musical sources performing in harmony, including choirs and chamber ensembles.

While this thesis approaches it in a class-agnostic fashion using a permutation-invariant objective function, other works have approached this challenge in different ways. This section outlines these alternative approaches.

2.13.1 *Choral Music Separation*

Choral singing is a form of ensemble vocalisation, which is seen in diverse musical cultures worldwide. Termed vocal ensembles, these musical performances involve multiple singers performing concurrently, often organised into sections based on vocal range, collectively forming what is commonly known as a choir. The prevalent Western choral configuration is the Soprano, Alto, Tenor, and Bass (SATB) arrangement, encompassing four distinct sections. The prevalent structure in choral singing involves utilising distinct male and female vocal ranges. Female singers adept at high pitches typically contribute to the soprano and alto sections, while male singers lend their voices to the tenor and bass parts. Soprano singers occupy the 260–880 Hz vocal range, while those in the alto section cover the 190–660 Hz range. The tenor and bass voices, associated with lower ranges, span 145 Hz–440 Hz and 90–290 Hz, respectively. In the context of an SATB choir, the distribution of these parts may involve four individual singers, each responsible for one section, resulting in a quartet arrangement.

Gover and Depalle (2019) is the first work that investigated the task of choral music separation. They attempted to overcome the challenges associated with the lack of clean datasets by generating a synthesised choral singing dataset. They subsequently approach the separation task as a class-based regression task using Wave-U-Net (see Section 2.6.1.1). Using this approach, they report good results on separation of 2 source mixtures of the lowest and the highest registers (Soprano and Bass), however separating 4 source mixtures performed poorly. Moreover, since these models were trained on synthesised data, they failed to produce results on real-world

recordings of singers. This could be attributed to the nature of their synthesised vocal dataset with limited realism. They subsequently present a score-conditioned separation method by concatenating the score along with the extracted audio features to the Wave-U-Net architecture, which produced some cross-dataset generalisability. This method heavily relies on well-aligned score information to produce some level of Fo-based filtering.

Petermann et al. (2020) uses the U-Net (as described in Section 2.5.2) as a baseline for a class-based SATB choir separation and subsequently propose a Fo-conditioned separation method, that provides the Fo contours for each of the sections along with the mixture audio as input. The Fo-contours are presented to the U-Net bottleneck via FiLM conditioning (Perez et al., 2018). The work uses the Choral Singing Dataset (CSD) and a proprietary Spanish choral singing dataset for these experiments. The data used in this work consists of more than one singer per section SATB choirs (not a quartet) and contains bleed.

In a study by Chandna et al. (2022), a survey of modern deep learning techniques for source separation in the context of choral music is presented. They find that models designed for music source separation are more apt for the task compared to those intended for speech source separation. It is important to note that the speech separation models were tested with a class-based regression objective, and not with a permutation invariant objective which they were originally designed for. They also find that waveform-based models demonstrate comparable efficacy to models employing intermediate representations such as spectrograms. In order to tackle the problem of bleed present in choral music datasets, and subsequently bleed present in class-based SATB separation methods, they propose a new method for separating SATB choirs and then resynthesising solo singing voices. The synthesis is based on extracted dynamics, pitch, and linguistic information derived from the four-part separation.

Jeon et al. (2023) work also utilised the class-agnostic separation method proposed in this thesis and highlighted the challenges associated with unison separation. The models trained in this work were trained using random mixing of 400 hours of solo singing voice and speech data from 13 different datasets. They segregated the vocal separation task into 3 unique tasks of unison singing, duet and lead vs. backing vocal separation and reported the applicability of PIT and class-based separation for each of these tasks. Crucially, this work highlights that by using large-scale datasets, models are able to perform vocal harmony separation without the need of musically coherent training mixtures. This work also highlights the channel swap problem across processing frames for PIT-based models and subsequently deals with it as a wav2vec (Baevski et al., 2020) based singer identification based post-processing method. It must be noted that the channel swap problem discussed in this work is distinct from the observations reported in this thesis in Section 6.6.2, where channel swaps within the same processing frame are also observed.

2.13.2 *Chamber Ensemble Separation*

While no other work has explored the separation of mixtures of identical instruments using deep learning prior to this thesis, Lin et al. (2021) have tackled the separation of separation of harmonised sources. This work proposes a zero-shot adaptation method and presents a multi-task separation, transcription and synthesis model that enables score-informed separation of harmonised chamber ensemble sources from the URMP dataset. Their work utilises the spectrogram masking-based U-Net (described in Section 2.5.2) in conjunction with a QueryNet branch comprising of 2 CNN blocks that allows conditioning the model with the F0 of the target source. It also uses the Griffin-Lim (Griffin and Lim, 1984) method to estimate the target phase, which is known to be computationally expensive.

2.14 DISCUSSION

The majority of the existing literature and proposed methods for performing music source separation rely on the assumption that sources are defined by their timbral characteristics. Hence models are trained to learn the timbral characteristics of the target source in order to separate it. Even in speech separation, the original deep-clustering based separation methods aim to extract contrastive embeddings for various time-frequency regions in the mixture spectrogram, which are likely to be able to distinguish speakers based on their timbral characteristics. Meanwhile, the permutation invariant training method in combination with TasNets does not explicitly constrain the model to learn timbral characteristics by only focusing on optimising the separation error.

However, prior to the work presented in this thesis, it was unknown whether TasNets trained with PIT were somehow overfitting to specific acoustic/timbral characteristics due to poor cross-dataset generalisability when trained on the WSJ-0 dataset (Cosentino et al., 2020). TasNets are able to perform exceptionally well and reach very high separation quality (up to 25 dB SDRi) while maintaining relatively small network sizes (less than 10M parameters). Meanwhile, state-of-the-art music separation architectures after U-Net and open-unmix increased in model capacity very rapidly (exceeding 100M parameters) in order to achieve improved separation. Moreover, TasNets had never been utilised successfully and did not perform comparably for the music separation task (Défossez et al., 2019). This may suggest that TasNets with PIT may fundamentally learn differently as compared to speech enhancement and music separation models which are expected to be able to model the timbral characteristics of their target sources.

However, there was little understanding of what distinguishing features are the TasNets learning when trained with PIT, due to their end-to-end nature. Initial understanding suggested that they are able to generate a sparser

latent representation as compared to the STFT on which masking is more suitable and their higher temporal resolution of separation is the primary contributing factor to their improved performance. Although these explanations should also suggest TasNets should perform exceptionally well for music separation, since both these factors should also work well for the music separation task. This was not observed in experiments by Défossez et al. (2019). The explanation for this was assumed that higher sampling rates used in music separation make TasNets unsuitable for the task as their receptive field/sequence length scales poorly as the sampling rate goes up. However, this was not proven and is also difficult to experiment with as TasNets typically require significantly higher VRAM consumption to train, even though their network sizes are relatively small.

In this thesis, the applicability of TasNets to music separation at high sampling rates, while presenting a music separation task that fits the permutation invariant objective is tested in Chapter 4. In order to explore whether TasNets do overfit to timbral/acoustic cues, a large synthesised dataset is created and presented in Chapter 3. Chapter 5 explores the applicability of TasNets to various music separation tasks, where it is observed that TasNets not only are able to generalise to unseen datasets/acoustic scenarios when trained on the synthetic dataset presented in Chapter 3, but are also able to generalise well across a broad range of instrument timbres. Finally, Chapter 6 presents a series of evaluation and analysis experiments to identify what musical scenarios the TasNets find challenging to separate, which leads us to a better understanding of how TasNets trained with PIT are able to separate sources in a timbre-agnostic fashion.

Part II

THE SETUP

Setting up the building blocks to enable ensemble separation.

3

ENSEMBLESET: A NEW SYNTHESISED DATASET OF CHAMBER ENSEMBLES

3.1 INTRODUCTION

Music source separation research has made great advances in recent years, especially towards the problem of separating vocals, drums, and bass stems from mastered songs. The advances in this field can be directly attributed to the availability of large-scale multitrack research datasets for these mentioned stems. Tasks such as separating similar-sounding sources from an ensemble recording have seen limited research due to the lack of sizeable, bleed-free multitrack datasets. While specific sub-tasks in the speech-domain like speech denoising, multi-speaker separation and dereverberation have been thoroughly explored, music separation research has largely been focused on the demixing challenge (Mitsufuji et al., 2021) aided by the popular MUSDB dataset (Rafii et al., 2019). The demixing challenge is targeted at solving the problem of separating vocals, bass, and drums from mixed and mastered pop songs. This has greatly benefited the field by demonstrating that source separation is indeed possible at a commercial scale with state-of-the-art deep learning-based architectures. Unfortunately, this also has resulted in the research towards this specific task to dwarf other problems that would also fall under the umbrella of music source separation, to the extent that music source separation has become synonymous with the task of separating vocal, drums and bass stems from mastered songs.

Training supervised source separation models typically require datasets that provide clean target sources as a reference for the deep learning models to learn from. While the majority of popular music can be recorded in separate takes for different performers, with a reference metronome or a backing track, ensembles are usually recorded together in the same take. This is due to the fact that ensemble performers rely on being able to hear each other during the performance to be able to synchronise perfectly. This raises the problem that the majority of stems available from recording projects of monotimbral ensembles contain bleed[1] from non-target sources (Rosenzweig et al., 2020; Bittner et al., 2014). This becomes problematic for training models for source separation due to the lack of clean ground truth as a target result for the model (see Section 2.13). This lack of clean and sizeable datasets for ensembles has affected the amount of research seen in this domain.

To overcome the challenge of bleed-free real recorded datasets for ensembles, a novel dataset “EnsembleSet" is presented, which utilises a highly realistic orchestral sample library by Spitfire Audio called "BBC Symphony Orchestra" (BBCSO) (SpitfireAudio, 2019). This sample library was used to render digital chamber ensemble scores from MIDI and MusicXML format to 18 unique multi-mic recordings and 2 professional mixes. For this work, the RWC Classical Music Database (Goto et al., 2002) and Mutopia (Praetzel, 2000) were used to source the chamber ensemble MIDI and MusicXML (converted from lilypond) scores. It must be noted that MIDI data are not ideal for capturing string, wind, and brass instrument scores as they do not encapsulate articulation information. On the other hand, lilypond scores contain minimal dynamics (velocity) information, which is essential for realistic rendering using virtual instruments. In order to address these challenges, expression maps from Dorico (Steinberg, 2016) were utilised, which is a scorewriter software that enables automatic selection of articulation mode for each note in the piece.

1 Sound picked up by a microphone from a source other than that which is intended.

Section 3.2 describes the motivation and design considerations behind the creation of the dataset "EnsembleSet" (Sarkar, Benetos, and Sandler, 2022) for chamber ensemble separation. Section 3.3 a description of the chosen BBC Symphony Orchestra library is provided, with details of the library's recording conditions and creative intent. Section 3.4 describes the sources and the process of curation of the digital music scores used for generating this dataset. Section 3.5 provides details about the data synthesis process and the resulting audio renders. Section 3.6 gives an overview of the musical content present in the final dataset. Section 3.7 and Section 3.8 provide a discussion regarding the utility of this dataset and its potential future applications.

3.2 MOTIVATION AND DESIGN CONSIDERATIONS

Existing synthesised datasets for source separation, such as Slakh2100 (Manilow et al., 2019) are generated in an automated fashion, where both the source MIDI data is scraped procedurally (Raffel, 2016) and subsequently rendered in an automated fashion. While such datasets do benefit from their larger size, models trained using such datasets suffer from poor cross-dataset generalisability, at least in the context of music source separation as reported in Manilow et al. (2019). This can be attributed to a few factors listed below.

- Using low-quality synthesizers results in very low timbral variation which often results in large deep-learning models overfitting to specific characteristics of the synthesizer.
- The automated scraping of MIDI data results in incorrect program numbers for various instruments, which is difficult to detect and/or correct as these datasets (LMD Raffel (2016)) don't have paired audio data to be able to detect such errors.

- MIDI is not well suited to capture most non-percussive musical instruments. Although almost all synthesizers do work with a piano-like interface enabling the use of MIDI to render any instrument, these synthesizers require careful control of additional parameters based on the musical context to be able to render realistic audio.

- Typically the only additional information that can be consistently provided in MIDI data is velocity[2], which was observed during our data exploration to be very poor for most MIDI datasets for non-piano instruments. RWC Classical Music Database (Goto et al., 2002) was one of the only sources found to have carefully mapped dynamics as note velocities.

- Chamber ensemble instruments have many different nuanced playing styles such as staccato, legato and tremolo. which are extremely challenging to render using MIDI data alone. The only information available in MIDI is the velocity and the distinction between strings and pizzicato strings as separate program numbers.

The goal with this dataset was to be able to use the high-fidelity sample library based BBCSO plugin to be able to create a highly realistic synthesised dataset that is able to produce generalisable deep learning models. The motivation for a synthesised dataset in this case was not only with the intent to create a sizeable dataset, but also because finding clean stems from real-world chamber music recordings is very difficult.

BBCSO was chosen because of the large timbral variety available in the plugin, where every note can be rendered in various articulation styles and multiple mics. For example: in the legato articulation mode, every possible note transition is recorded as a unique sample. Moreover, every note is recorded with up to 5 unique takes, and during rendering one of the 5 takes is cho-

2 Velocity indicates how hard the key was struck when the note was played, which usually corresponds to the note's loudness

sen in a round-robin fashion. This enables additional variety in the rendered dataset which could be useful to train deep learning models.

We chose to focus our dataset to maximize the capabilities of the sample library in the context of permutation invariant training. While orchestral scores had a much higher number of concurrent sources on average, each individual source in an orchestra often plays in a polyphonic fashion [3] which would make it unsuitable for training deep learning models in a permutation-invariant fashion. While the dataset could eventually be extended to include orchestral scores as well, it was consciously chosen to be restricted chamber ensembles, where each part is played by a single performer. Thus each part is rendered as an individual performer/section leader (for eg: Violin 1 leader, instead of Violins 1 section) instead of rendering an entire section with multiple players.

As the data rendering pipeline was unable to render certain instruments (such as Piano) that are commonplace in chamber ensembles, such instruments were filtered from the source symbolic data such that every piece would have at least 2 sources that can be rendered using BBCSO, and at least one of the sources should be a bowed instrument. While the process could have also included rendering pieces that had 2 non-bowed instruments playing together as well, at the time of rendering the dataset it was unknown that monophonic separation would work in a timbre-agnostic fashion (discussed in Section 5.4), thus the dataset was focussed on bowed instruments, primarily consisting of string quartets. Although it must be noted that there were only 2 pieces available in Mutopia that could have been included without the aforementioned restriction, this choice did not affect the dataset size significantly.

3 different instruments within a single section play distinct notes instead of playing in unison

3.3 HOW WAS THE BBCSO LIBRARY CREATED?

This section presents key takeaways from an interview conducted with Jake Jackson, Recording Engineer at Maida Vale studios (who was also the recording engineer for the Spitfire Audio BBCSO sample library) which is presented in Section B.1.

The main intent for Spitfire Audio with the creation of the BBCSO library was to record a sample library in the same setup that is used for professional recording of film/game scores, such that it would have all the microphones typically used in such projects. They also provide additional renders such as spill mics and distant balcony and Atmos mics in order to futureproof the library as growingly most audio projects are rendered/upmixed to immersive/3D audio compatible formats. It includes all traditional room capture mic configuration such as decca tree, ambience, and also more dry and closer perspective recording scenarios that use close mics for each performer and section. The idea was to make available all possible perspectives to a performer as a mix engineer might want. All the different mics accurately represent the sonic image of a source from the given recording position, including the time it takes for the sound from the source to arrive at each mic without any correction. Performers in an orchestra actually play in a fashion such that the sound from each source arrives at the conductor at the same time, so performers at the back of the orchestra would actually anticipate the beat and play slightly before the beat to compensate for the time it'd take for the sound to travel to the conductor. Thus the time alignment of the instrument midi onsets and actual sample time is aligned in a fashion such that the sound from each source at the Mono mic (placed right above the head of the conductor) arrives at the same time.

3.4 COLLECTING DIGITAL MUSIC SCORES

The first step to generating the dataset was to obtain digital music scores that could be rendered using BBCSO. One of the fundamental principles of this project was to maintain the realism of the synthesised dataset, thus choosing clean and high-quality data sources was essential. As a starting point Kern scores (Sapp, 2005) and RWC Classical music database (Goto et al., 2002) were explored as sources for high-quality digital scores.

During the exploration, kern scores were found nonideal as all note events were annotated with a constant velocity value, and the articulation information was not preserved when converted to MIDI. Subsequently, RWC Classical Music database was investigated as a source, which was ideal as it had very accurate velocity and time annotations for each note event. While these MIDI scores lacked articulation information, they did have separate MIDI tracks for pizzicato strings which was useful.

On further exploration, it was observed that some of the scores present in RWC were for orchestral pieces while some were for chamber ensemble pieces. The orchestral pieces consisted of polyphonic scores per section which would require it to be split and rendered as individual sources, and also opened up our first challenge i.e. which articulation should be used for as default playback? Legato would be ideal but it only works for monophonic scores, as it assumes that the performer is playing a sequence of single notes.

Subsequently, polyphonic sections were split into multiple monophonic tracks. This resulted in the fundamental/lowest monophonic track being consistently active but often with discontinuous melodies (if a lower harmony appears). While the higher harmonic tracks were mostly inactive with sparse and seemingly random melodic content. Using a method to identify the main melody from these polyphonic sections (and ignoring the harmonisation) could be useful, but that was determined to be a fairly challenging

problem and beyond the scope of this work. Thus it was decided to restrict the dataset to chamber ensemble scores. This resulted in the number of usable tracks from RWC being only 9 pieces.

The first version of this dataset used a few of the orchestra scores and was presented at ISMIR 2021 at the sound demixing challenge workshop where the dataset was commended for its realism. Based on feedback received during this demo, Mutopia was suggested as a digital score source and potentially Lilypond as a source format as it would preserve articulation information. This expanded the scope of the dataset to include a much larger library of ensemble scores through Mutopia (140 pieces). However, some of the scores present in the Mutopia collection were orchestral pieces, which had to be excluded due to the presence of polyphonic sections in their instrument scores.

Knowing the limitations of PIT, it was decided to focus on chamber ensemble scores exclusively. Including both Mutopia and RWC as sources, 81 scores (6 hours and 9 minutes duration) were collected which was sizeable enough for a useful source separation training dataset. Another issue with orchestra scores (apart from the polyphonic sections) was the poorly annotated lilypond scores, many of which resulted in errors during conversion to MIDI (66 out of 68 orchestral lilypond scores from Mutopia threw conversion errors) and the effort required for cleaning these was much higher. Thus it was decided to focus on rendering only the chamber ensemble pieces and the 78 orchestra scores (68 from Mutopia, 10 from RWC) were set aside as potential future work that could be released as OrchestraSet.

3.4.1 *RWC Classical Music Database*

The RWC Classical Music Database (Goto et al., 2002) consists of 50 public-domain classical pieces performed by musicians and then manually tran-

scribed to MIDI with high-quality tempo and velocity mapping. Since the database only provides the final mix of these performances, its applications are limited especially in the context of source separation. A subset of these pieces that contain chamber ensembles were chosen as they can be rendered using our method. The 9 selected pieces (1h 3m 34s)[4] consist of 4 string quartets, 2 clarinet quintets, 2 piano trios, and 1 piano quintet. It must be noted that for the piano trios and quintets, only the string instrument parts were rendered. Since MIDI files lack articulation information, the MIDI files were augmented using Dorico[5] to automatically add articulation modes in the MIDI scores using keyswitches, which were then subsequently rendered as multi-tracks on Reaper (Cockos, 2006).

3.4.2 *Mutopia*

The Mutopia Project (Praetzel, 2000) 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 71 pieces were chosen (5h 5m 35s)[2] including 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 LilyPond (2016), to preserve the articulation information present in the lilypond files. For the files which successfully converted to MusicXML, we import them to Dorico, where these articulations are translated to keyswitches (described in Table 3) and rendered to MIDI format which can then be utilised by the BBCSO plugin when rendered on Reaper (Cockos, 2006).

4 Rendered duration in dataset.
5 https://www.steinberg.net/dorico/

3.4.3 *Data Cleaning*

Many of the scores used to render this dataset contained instruments that were 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, the source MIDI files made available in the database were used as is and thus were unable to preserve articulation for those pieces. For these pieces, the same articulation generation pipeline as the RWC sourced files was used. For some other files where the errors were minor (incorrect timing and track assignments), their corresponding MIDI files from the source database were used to manually inspect and correct these MusicXML conversions.

3.5 DATA GENERATION

3.5.1 *BBC Symphony Orchestra*

The BBCSO 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

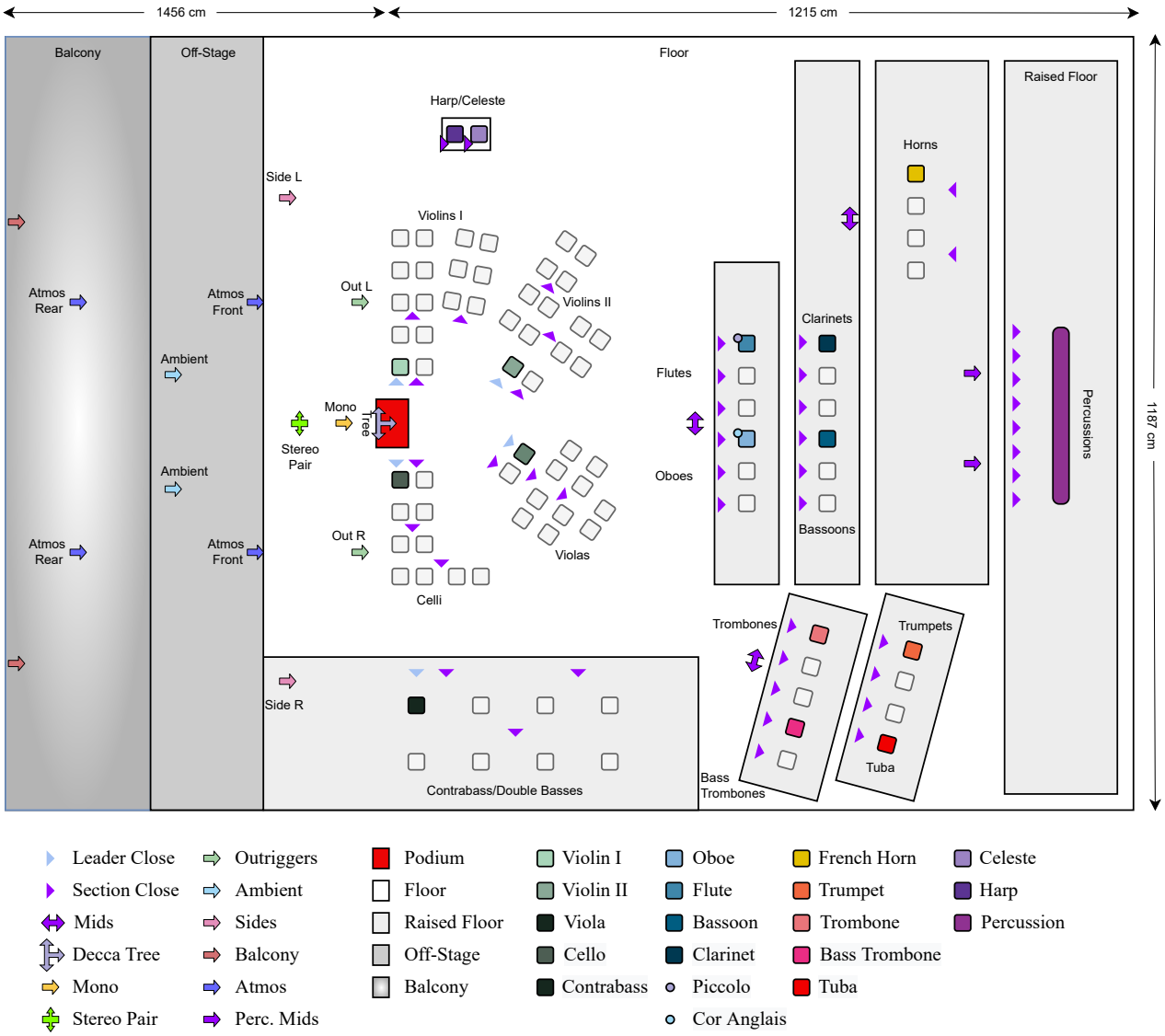

Figure 5: Recording configuration for the Spitfire Audio BBC Symphony Orchestra sample library depicting the placement of individual microphones and performers.

that enables the capture of each performer from different perspectives in the room. This allows us to simulate high quality recordings of chamber ensemble pieces from digital music scores, rendered using individual section leaders.

3.5.2 *Microphone Renders*

- **Mono**: An old-fashioned bi-directional (figure of 8) microphone positioned behind the conductor's head for a close-to-realistic mono pickup. The renders are upmixed to stereo based on the angle of performer w.r.t. the conductor. This render can be downmixed to mono without any phase artifacts.

- **Leader**: A condenser microphone placed close to the leader of each of the string sections at instrument height. The renders are upmixed to stereo based on the angle of performer w.r.t. the conductor. This render can be downmixed to mono without any phase artifacts.

- **Decca Tree**: Three omnidirectional microphones are placed in the traditional Decca Tree arrangement, high above the conductor's head. The three microphones are panned hard left, centre and hard right in the stereo render.

- **Outriggers**: Two omnidirectional microphones placed midway between the orchestra at the same line and height as the Decca Tree. The two microphones are panned hard left and hard right on the stereo render.

- **Ambient**: Two omnidirectional microphones placed towards the rear of the room, higher than the outriggers. The two microphones are panned hard left and hard right on the stereo render.

- **Balcony**: Two omnidirectional microphones placed at the very rear of the hall, high up in the balcony. The two microphones are panned hard left and hard right on the stereo render.

- **Stereo Pair**: Two Coles 4038 microphones placed in a stereo arrangement close to the musicians at head height. The two microphones are panned hard left and hard right on the stereo render.

- **Mids**: A stereo pair placed above the brass, woodwind and percussion sections. These are used as a mid pickup between the Close and Tree microphones. The two microphones are panned hard left and hard right on the stereo render.

- **Sides**: Two omnidirectional microphones placed at the very edge of the orchestra, in the same line as Decca Tree and Outriggers. The two microphones are panned hard left and hard right on the stereo render.

- **Atmos Front**: Two omnidirectional microphones placed high above the orchestra in the front. The two microphones are panned hard left and hard right on the stereo render.

- **Atmos Rear**: Two omnidirectional microphones placed high above the orchestra in the rear. The two microphones are panned hard left and hard right on the stereo render.

- **Close**: The standard close microphones which are unique for each section placed close to the performers. This render is panned as per the position of the performer on stage w.r.t. the conductor. (3 microphones per string section, each of the 3 microphone signals are downmixed to stereo based on the angle of each of the microphones w.r.t. the conductor, for eg: violins section 1 close mics would be panned roughly -75 degrees, -72 degrees and -70 degrees w.r.t. conductor)

- **Close Wide**: The standard close microphones which are unique for each section placed close to the performers. Unlike the default panning of this render for whole sections which uses multiple mics panned wide apart, this render is panned center in our case as we only render the leaders of each section. (each mic of section panned hard left, center and hard right)

- **Spill String**: These are all the close microphones from the Violin 1, Violin 2, Viola, Cello, and Bass sections. These mics can be used to simulate the bleed of any of the other sections being picked up from

the strings section mics. This render is a downmix of 15 mics where each of the microphone signals is panned based on their position w.r.t. the conductor.

- **Spill Brass**: These are all the close microphones from the Horn, Trumpet, Tuba, Trombone, and Bass Trombone sections. These mics can be used to simulate the bleed of any of the other sections being picked up from the brass section mics. This render is a downmix of 11 mics where each of the microphone signals is panned based on their position w.r.t. the conductor.

- **Spill Woodwind**: These are all the close microphones from the Clarinet, Bassoon, Flute, and Oboe sections. These mics can be used to simulate the bleed of any of the other sections being picked up from the woodwind section mics. This render is a downmix of 12 mics where each of the microphone signals is panned based on their position w.r.t. the conductor.

- **Spill Percussion**: These are all the close microphones from the Percussion sections. These mics can be used to simulate the bleed of any of the other sections being picked up from the Percussion section mics. This render is a downmix of 10 mics where each of the microphone signals is panned based on their position w.r.t. the conductor.

- **Spill Full**: These are all the close microphones from all the instrument sections. These mics can be used to simulate how any instrument is being picked up by the sum of all the close mics on stage. This render is a downmix of 48 mics where each of the microphone signals is panned based on their position w.r.t. the conductor.

Summarised details about individual microphone/mix setups can be found in Table 2.

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

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 3: List of keyswitch-articulation mappings for different instruments.

3.5.3 *Mixes*

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

3.5.4 *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 articula-

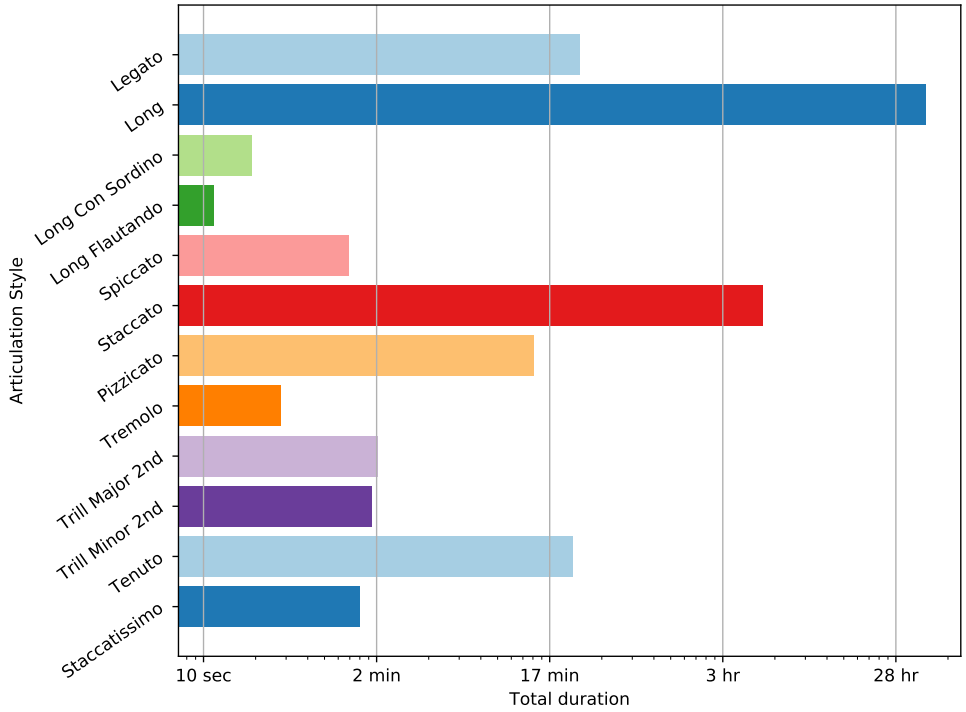

Figure 6: Articulation distribution across EnsembleSet

tions 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 resulting distribution across different articulation styles is shown in Figure 6. The list of keyswitches and articulation mappings for each of the instruments available in EnsembleSet is shown in Table 3.

3.6 DATASET CONTENTS

EnsembleSet contains a total of 498.5 hours of unique audio renders across all instruments and mixes. The dataset contains a total of 6 hours and 9 minutes of multi-instrument, multi-mic data and is available on Zenodo[6]. The resulting total active duration of each instrument in EnsembleSet can be

6 https://zenodo.org/record/6519024

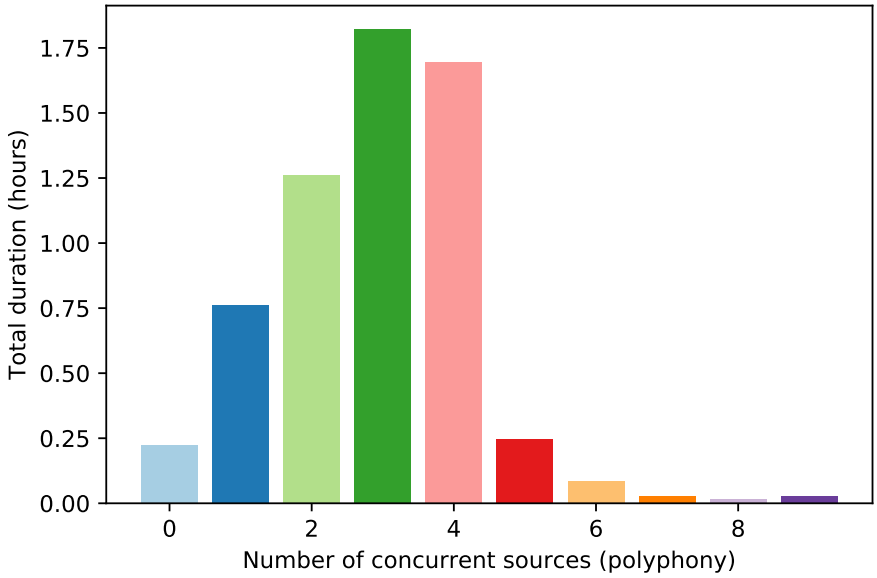

Figure 7: Polyphony distribution across EnsembleSet

seen in Figure 8. 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 7. Each song is also paired with its accompanying MIDI file which was used to generate the renders, which also contains the articulation information.

3.7 DISCUSSION

A large focus of this dataset was to generate large amounts of realistic chamber ensemble mixtures for training ensemble separation models using permutation invariant training. The assumption (based on experiments presented in Section 6.3) behind maximising the size of the dataset was that random

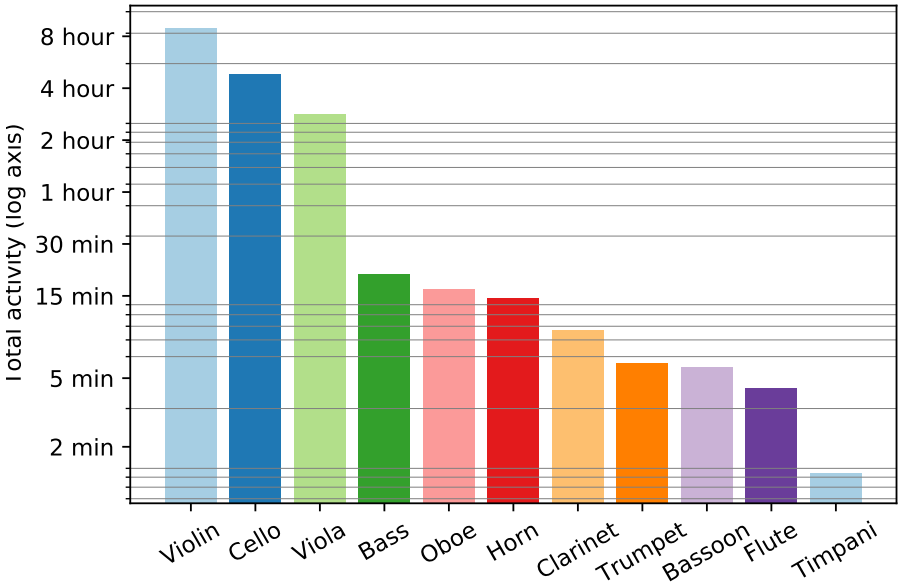

Figure 8: Instrument wise activity duration in EnsembleSet

track mixing[7] does not work for ensemble separation. On subsequent exploration after the sizeable dataset was created, it was observed that random mixing does in fact work for ensemble separation with PIT as well, although it does not improve performance when average metrics are compared (discussed in Section 6.3.1). This raises the possibility of training high-quality ensemble separation models by randomly mixing solo instrument performance recordings, which bypasses the bleed in ground truth issue while still being able to train on real-world recordings. While this could definitely improve the generalisability of ensemble separation models for a variety of timbres and recording conditions, Chapter 6 shows that the large variance in our model's performance is observed due to specific scenarios such as synchronised onsets and pitch crossovers with staccato observed in real mixtures, which would be absent in randomised mixtures of solo performance. It would be interesting to explore the impact of models trained on real but randomised mixtures on the aforementioned scenarios since such scenarios are indeed present in EnsembleSet. Potentially it may be found valuable to

7 musically incoherent mixtures using instrument stems from different pieces of music.

train models with both the presented synthesised synchronised dataset and real-world randomised mixtures from solo performances combined. Due to the large size of this dataset including the multi-mic renders, the use of audio effects as data augmentation was restricted to be applied on the fly during model training. However, it must be noted that using realistic audio effects using VST plugins could improve the real-world generalisability of models trained using this dataset.

3.8 POTENTIAL APPLICATIONS

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 (Wu, Chen, and Su, 2020), instrument recognition (Garcia et al., 2021), score-informed source separation (Garcia et al., 2021), microphone simulation (Mathur et al., 2019), and automatic mixing (Reiss, 2011).

4

MUSIC SOURCE SEPARATION USING TASNETS

4.1 INTRODUCTION

Time-domain source separation models based on TasNets (Luo and Mesgarani, 2018) are distinct from STFT-based source separation models by allowing the model to learn a 1-D convolutional filter to transform 1-dimensional time-domain audio to a 2-dimensional representation without using the Fourier transform. Previous methods reliant on magnitude-spectrogram-based masking were effectively working under a theoretical performance limit determined by the lack of accurate phase estimation. This limit is known as the Ideal-Ratio Mask, which represents the upper-bound of phase-agnostic magnitude masking based methods by measuring the SNR of an estimated signal assuming perfect magnitude spectrogram masking without accurate phase estimation. These time-domain methods can be broadly categorized into 2 groups: based on learnable 1-D encoder/decoder filterbanks introduced by Luo and Mesgarani (2018) or based on U-Net architectures, first used in Stoller, Ewert, and Dixon (2018b) and subsequently improved on by Défossez et al. (2019). More details about the details of these methods can be found in Section 2.6. This work utilizes the TasNet (Luo, Chen, and Mesgarani, 2018) based architectures Conv-TasNet (Luo and Mesgarani, 2019), DPRNN (Luo, Chen, and Yoshioka, 2020) and DPTNet (Chen, Mao, and Liu, 2020) based on their proven success with permutation invariant training (Yu et al., 2017) for speech separation.

Contributions of this chapter include a survey of TasNet-based networks applied to high-fidelity music signals at 44.1 kHz which were originally designed for speech separation at 8 kHz. The impact of different model structures and parameters is presented in the context of available GPU resources for training such models. The network parameter optimisation explored in this chapter acts as the baseline architecture for the experiments presented in Chapter 5.

In this chapter, Section 4.2 first formulates the ensemble separation problem which enables use of PIT for music source separation. Section 4.3 discusses the intricacies of using TasNets for music source separation, establishing the challenges associated with limited hardware capabilities and balancing various aspects of the architecture to make TasNets work at 44.1 kHz for music separation effectively. Section 4.4 explores the performance of TasNets with PIT for varying number of sources and discusses the utility of such models for source separation considering the quality that is achieved with different sized ensembles.

4.2 LEVERAGING PIT FOR CHAMBER ENSEMBLES

The experiments in this thesis focus on the separation of chamber ensemble mixtures from monaural recordings. Each of the stems that are extracted in the music demixing task (as described in Section 2.1.3) can be further decomposed to obtain individual monophonic sources, for example, the vocals stem $s_{vocals}(t)$ can be decomposed to individual vocalists $s_{v_n}(t)$ as Equation 16 where each decomposed signal is a monophonic signal.

$$s_{vocals}(t) = \sum_{v=v_1}^{V} s_v(t) \tag{16}$$

In other forms of music, especially in the context of typical chamber music, the mixture can be composed of multiple monophonic instruments and additional polyphonic (e.g.: piano, harpsichord) and percussive instruments (e.g.: timpani, snare) as:

$$s_{ensemble}(t) = \sum_{i=i_1}^{I} s_i(t) + s_{polyphonic}(t) + s_{percussion}(t) \tag{17}$$

While the individual monophonic instruments $s_i(t)$ can be categorised based on instrument type (such as violin, viola, cello, flute, trumpet, etc.) and class (such as string section, woodwind section, brass section), considering such instruments in a label agnostic fashion allows training source separation models in a permutation invariant fashion with the constraint that each source is monophonic. In this thesis, the decomposition problem is defined as Equation 18, where given a mixture consisting of N monophonic sources, the model is trained to separate them into the constituent monophonic parts $s_i(t)$ for $i \in I$. In this formulation, mixtures are only considered to contain monophonic sources, thus chamber ensemble mixtures with percussion or polyphonic instruments are not considered. This task can then be solved in a permutation-invariant fashion (see Section 2.8) as introduced by Yu et al. (2017).

$$s_{mixture}(t) = \sum_{i=i_1}^{I} s_i(t) \tag{18}$$

4.3 MAKING TASNETS WORK AT 44.1 KHZ

A disadvantage of learnable free-filterbanks as encoder/decoder pair with respect to the FFT is the poor scalability of the 1-D convolution encoder layer for higher sampling rates. In the case of FFT, the hop size and window size

of the FFT can be scaled according to the sampling rate of the signal without a significant increase in computational complexity[1]. On the other hand, TasNets (described in Section 2.6.2) introduce a tradeoff between temporal resolution and temporal context when choosing the filterbank hop and stride length parameters. Lower hop lengths improve the temporal resolution of the system, at the expense of longer sequences/limited receptive field for the separation stack and higher VRAM consumption for training. The implications of these tradeoffs vary based on architecture, which are explored in the subsequent sections.

Section 4.3.1 first presents the various TasNet architectures used in this Thesis, and introduces their various network parameters and design considerations. Section 4.3.3 describes the dependence of these network parameters based on memory capacity restrictions for different GPUs, especially in the context of dealing with high-sampling rate audio. Section 4.3.4 introduces the various network and training parameters, establishing the trade-offs between memory consumption, training speed, model capacity and separation performance. The experiments conducted during the optimisation process are described in Section 4.3.5 and Section 4.3.6 and the data augmentation techniques used for these experiments which are also included as the baseline in the remaining experiments in this thesis are described in Section 4.3.7.

4.3.1 *Architectures*

The experiments described in this thesis primarily rely on using two TasNet-based architectures (described in Section 2.6.2 and shown in Figure 2): ConvTasNet and DPTNet. Dual-path RNN (Luo, Chen, and Yoshioka, 2020) was also explored but DPTNet was a similar architecture released soon after DPRNN which showed improved performance compared thus subsequent

1 Due to the FFT algorithm reducing the computational complexity of the discrete Fourier transform to $O(N \log(N))$.

experiments relied on DPTNet. While SepFormer by Subakan et al. (2021) reports improved performance and efficiency as compared to DPTNet by removing the recurrent operations present in DPTNet, an open-source implementation of SepFormer was not available until 2023 thus the experiments presented in this thesis were limited to DPTNet. However, it can be assumed that SepFormer-based experiments would show similar performance trends as reported in this thesis.

4.3.1.1 *ConvTasNet*

The ConvTasNet architecture by Luo and Mesgarani (2019) improved upon the original TasNet architecture by Luo and Mesgarani (2018) by keeping the 1-D convolution-based encoder/decoder layer but replaced the LSTM-based separation stack with a TCN-based separation stack. The original LSTM-based separation stack limited the capability of the model with shorter filterbank length/hop sizes as reducing the filter hop size significantly increases the length of the latent representation as given by Equation 19, where L is the length of the latent representation generated by the encoder determined by Equation 19 where T is the duration of the training samples, F_s is sampling rate and L_f is the length of the filters in the encoder.

$$L = \frac{2 \times T \times F_s}{L_f} \tag{19}$$

To overcome the challenges associated with training LSTMs for long sequences, the authors Luo and Mesgarani (2019) suggest replacing the LSTM stack with a dilated-TCN stack, where the dilation increases exponentially until 2^{x-1} for the final x^{th} layer of the TCN stack. While the TCN stack can consistently extract temporal relationships within the receptive field of the TCN-stack, this receptive field reduces as the sampling rate of the signal increases, requiring either increasing the hop size of the filterbank or increasing

the number of layers of the stack, thus increasing model size and computational complexity.

4.3.1.2 *Dual-path RNN/Transformer*

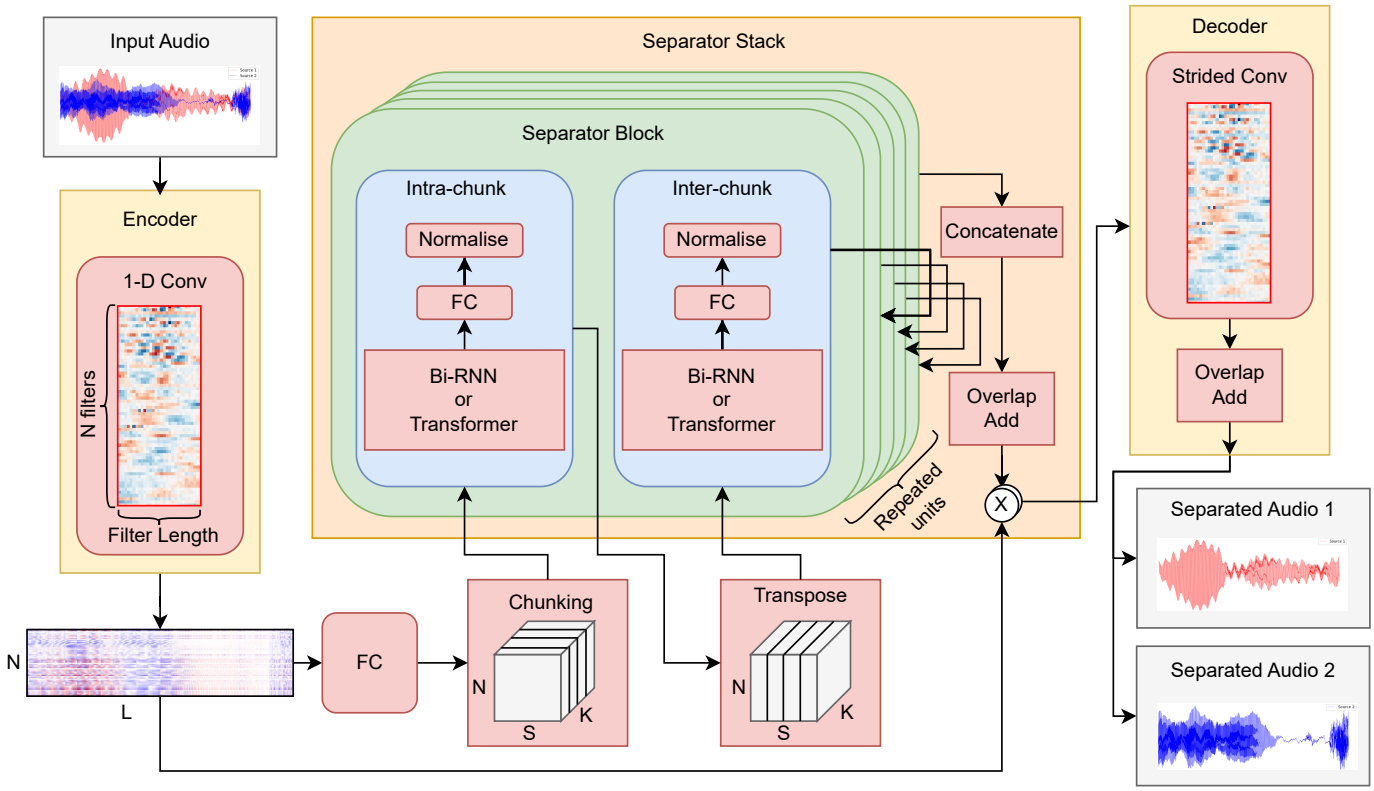

Figure 9: Dual-path processing based architecture for DPRNN and DPTNet audio source separation models.

An alternate approach to solve the vanishing gradients problem for long sequences generated by the encoder/decoder filterbank was proposed by Luo, Chen, and Yoshioka (2020) to convert the 2-D representation into a 3-D representation by chunking the feature sequence into segments and then stacking them (as shown in Figure 9). Once this has been done, the latent features can be processed in an *intra-chunk* and *inter-chunk* fashion by having different RNN heads operating on short-term and long-term features. This was subsequently improved upon by Chen, Mao, and Liu (2020) by replacing

the RNN with a modified transformer encoder architecture with a better capacity to capture temporal relationships than bi-LSTMs.

For a given input sequence of length L, the chunking is performed such that the length of the intra-chunk segment (S) and inter-chunk segment (K) are evenly matched to maximize the effectiveness of each of the *intra-chunk* and *inter-chunk* RNN such that $S \approx K$, which results in Equation 20. This results in the latent representation of the input sequence, and thus the computational complexity of the separation stack to scale by $O(\sqrt{L})$ instead of $O(L)$.

$$K = \sqrt{2L} = \sqrt{2 \times \frac{2 \times T \times F_s}{L_f}} \tag{20}$$

This technique enables scaling TasNets to work effectively at higher sampling rates (for reference: the computational complexity of FFT scales at $O(N \log(N))$.

4.3.2 *Experimental Setup*

For the presented optimisation experiments, models were trained to separate 4 part vocal harmony mixtures from the BCBQ datasets (see Section 2.11.3). The songs present in the combined dataset were into 3 groups for training, validation, and testing roughly in ratio 8:1:1 (since song lengths vary), making sure that the test and cross-validation sets consist of songs (and not just segments) that are unseen in the training set. Though the audio mixtures are identical across the different test scenarios, the frame length used for evaluating the models was the same as their training configuration. The data pre-processing involves activity detection on the monophonic vocal audio files and identifying frames where all constituent sources are concurrently

active for at least 10% of the frame length for 4 and 3 source mixtures, and 40% for 2 source mixtures. This was applied to prevent the model from being trained on frames where less than the desired number of sources were active. The models were trained using SI-SDR as the loss function with a permutation invariant objective (see Section 5.3.3 for details). Models were trained for 100 epochs with 0.0001 as the starting learning rate with a scheduler that halves the learning rate if the validation loss doesn't improve for more than 3 epochs and an early stopping condition with the patience of 5 epochs.

4.3.3 *Hardware Limitations*

TasNets typically require very high GPU VRAM during training as compared to similar capacity FFT-based methods. This is largely due to the small hop size of the encoder/decoder filterbank resulting in a significantly higher number of decisions and backpropagations per second. Such models have predominantly been trained on the highest grade GPUs typically available in high-performance compute clusters (HPCs) such as Tesla P100, V100, and most recently A100s. While TasNets have been trained on consumer-grade GPUs such as NVIDIA RTX 2080s and 3090s for speech separation/enhancement applications operating at 8 kHz, other reported uses of TasNets at 44.1 kHz (as reported by Défossez et al. (2019)) were also trained on Tesla V100s (which were the highest VRAM GPUs available at the time).

Preliminary experiments using Conv-TasNet attempted on NVIDIA RTX 2080s (8GB VRAM) failed to converge due to models being limited to less than 3 repeat units with 20 sample filter lengths resulting in a very low receptive field w.r.t. input segment length of the separation stack at a batch size of 1. Subsequent experiments moved to use 4 x Tesla V100s (16GB) with distributed-data-parallel backend resulting in the first convergent Conv-TasNet model for vocal harmony separation by using parallel processing to increase the effective batch size to 4. Experiments with both data-parallel

and distributed-data-parallel packages resulted in a similar performance, but distributed-data-parallel processing was chosen due to the flexible scalability of the network and training parameters independent of the number of GPUs available during training.

Subsequently, experiments with DPRNN and DPTNet showed robust scalability across various input lengths without the receptive field constraints of ConvTasNet. Experiments showed DPTNet and DPRNN to have very similar resource consumption both in terms of VRAM and speed with DPTNet performing marginally better than DPRNN due to improved sequence modeling, thus DPTNet was used as the baseline for all subsequent experiments. Table 4 lists the results of various experiments with DPTNet compared with the best-performing ConvTasNet model with similar resource consumption using 8 x NVIDIA V100 (16GB) GPUs.

With the introduction of NVIDIA A100s in the late 2020s, per GPU VRAM increased to 80GB which enabled exploring TasNets at the highest time resolution of 2 sample filter lengths. However, model performance is not only dependent on the capability of the model but also on data size, diversity, and training time. Training TasNets at the highest temporal resolution comes at the expense of the total number of training iterations that can be achieved within the limits of GPU run time. In the experiments described in this chapter, we focus on the choral separation experiments described in Section 5.3, as the train and test data for the task are of limited diversity, thus can be used as a control experiment for optimizing the performance limits of a model without the consideration of cross dataset generalisability. For subsequent experiments described in Section 5.4 and Section 5.5, models with lesser resource consumption were used as better cross-dataset performance was achieved by reducing per iteration training time and allowing the model to train for longer with more diverse data.

4.3.4 *Network Optimisation*

4-source choral separation based on the Bach Chorales and Barbershop Quartet datasets was used for the optimization experiments presented in this section. The hyperparameter configurations presented in Table 4 and Table 5 used 8 x NVIDIA V100 GPUs with 16GB VRAM, while the configurations presented in Table 6 used 4 x NVIDIA A100 GPUs with 80GB VRAM which were not available during the previous experiments. The configurations presented in Table 4 compare models that are capable of fully utilising the available 16GB VRAM. Experiments presented in Table 5 show all different filterbank and input segment length configurations possible on 16GB VRAM GPUs without changing the model capacity. Experiments presented in Table 6 utilise 4 NVIDIA A100 GPUs with 40GB VRAM, and present configurations that maximise memory utilisation and network capacity while preserving small filterbank lengths. A result for 20GB VRAM consumption is also presented in Table 6 as it allows having a batch size of 2 per GPU which doubles the training speed. This was particularly useful in the context of time limitations enforced on shared GPU clusters, where faster training speed enabled the use of larger datasets such as EnsembleSet for experiments presented in Section 5.4 and Section 5.5.

The performance trends observed (shown in Table 4) were similar to the observations presented by Luo and Mesgarani (2019) with certain differences that may be attributed to the memory, data, and hardware limitations associated with high sampling rate processing using TasNets. The general trends observed were:

- **Number of filters**: Increasing the number of filters present in the 1-D encoder/decoder convolutional filterbank improves performance, but results in diminishing returns for more than 64 filters. Increasing the number of filters does not result in a significant impact on resource consumption. This may be attributed to the fully connected layer bot-

tleneck between the encoder and the separation stack, which results in increasing the number of filters in the filterbank not affecting the computational complexity of training the separation stack.

- **Filter Length**: Although in the original architecture, the filter length of the 1-D encoder/decoder convolutional filterbank is always considered to be twice the filter stride, it was observed that increasing filter length to be 4 times than the filter stride resulted in performance improvement without increasing computation time/memory. This may be attributed to the increase in the local context available and improved waveshape modeling at each convolution step without reducing the temporal resolution of predictions. While increasing the filter length may suggest averaging of dynamics in the audio signal over the length of the filter, given the time scales at which these filters operate the averaging occurs over very short time periods (1.5 milliseconds for 64 sample filter length @ 44100 Hz), where the signal composition may be considered quasi-stationary.

- **Filter stride**: The stride length of the 1-D convolutional encoder/decoder filterbank was one of the parameters with the most significant impact on performance and resource consumption. Reducing the stride length inversely increases the VRAM consumption of the model per training batch, as it increases the length of the latent representation as per Equation 19. Simultaneously it increases the separation performance of the model as the temporal resolution of the masked representation increases. Changing this parameter does not impact model size or capacity, only impacts training time and VRAM consumption.

- **Segment duration**: The segment duration linearly increases the VRAM consumption of the model per batch item during training. This is also due to its linear impact on the sequence length of the latent representation as given by Equation 19. Increasing the input segment duration improves model performance, especially for DPTNet as the temporal

Table 4: Results for different models running on 16GB V100s

Model	Stride	Time (sec)	Repeat	SI-SDRi	SDRi
ConvTasNet	10	5	6	10.71 dB	11.85 dB
DPTNet	10	5	6	**10.91 dB**	**12.14 dB**
DPTNet	8	10	2	9.30 dB	10.57 dB
DPTNet	4	5	2	10.2 dB	11.4 dB
DPTNet	2	2	2	10.05 dB	11.2 dB
DPTNet	1	2	2	10.53 dB	11.63 dB

context available to the model improves. For ConvTasNet the improvement in performance with increased input segment duration is limited by the receptive field of the TCN stack. Changing this parameter does not impact model size or capacity, only impacts training time and VRAM consumption.

- **Repeat units**: The number of repeat units directly impacts the separation performance of the model and the model size. Increasing the number of repeat units linearly increases the model's VRAM consumption during training. Increasing the model capacity not only improves the ability of the model to separate sources more effectively, but it also improves the model's capacity to generalize better although that is also dependent on the diversity of training data presented to the model.

4.3.5 *Temporal Context vs. Temporal Resolution*

The length of the 2-D representation generated by the 1-D convolution filterbank is dependent on both the input segment duration and the filterbank hop size (1-D convolution stride length). Increasing the input segment duration provides the model with more temporal context which typically results in a performance improvement, while also increasing the length of the generated 2-D latent representation. Reducing the stride length of the filter-

Table 5: Results for different temporal contexts and filterbank lengths

Model	VRAM	Stride	Time (sec)	Repeat	SI-SDRi	SDRi
DPTNet	16GB	8	10	2	9.30 dB	10.57 dB
DPTNet	16GB	4	5	2	10.2 dB	11.4 dB
DPTNet	16GB	2	2	2	10.05 dB	11.2 dB
DPTNet	16GB	1	2	2	**10.53 dB**	**11.63 dB**

bank increases the temporal resolution of the masking which results in improved separation performance, but this too increases the length of the latent representation. As the length of the 2-D latent representation increases, the VRAM consumed per batch item increases, thus both these parameters need to be balanced for optimal performance. It's also worth noting that while increasing the input segment length does not affect inference speed/resource consumption, reducing the filterbank hop size does reduce the speed at inference. Table 5 reports the results of experiments conducted to study the trade-off between sequence length and filterbank size. It must be noted that only 2 repeat units were used for these experiments, which is non-ideal for separation performance as it significantly reduces the model capacity. It is observed that increasing the temporal context beyond 5 seconds does not result in any performance improvement, but reducing the temporal context below 5 seconds does result in a performance drop.

4.3.6 *Model capacity vs. Temporal Resolution*

Further experiments were undertaken to optimise temporal context and resolution while simultaneously increasing the model capacity. Table 6 reports results of models trained with different combinations of the three mentioned parameters. It is observed that increasing the number of repeated separator units in the separator stack has the most significant impact on separation performance, followed by filterbank stride. Since it was observed that increasing

Table 6: Results for different model capacities and filterbank lengths

Model	VRAM	Stride	Time (sec)	Repeat	SI-SDRi	SDRi
DPTNet	32GB	3	5	6	12.21 dB	13.16 dB
DPTNet	32GB	1	2	6	11.05 dB	12.31 dB
DPTNet	32GB	2	5	4	**11.49 dB**	**12.47 dB**
DPTNet	32GB	1	2	3	10.65 dB	11.85 dB
DPTNet	20GB	4	2.97	8	10.58 dB	12.04 dB

temporal context beyond 5 seconds did not improve performance, experiments conducted here use shorter input segment durations. With resource consumption and total training time in consideration, the final parameter settings chosen for the majority of the experiments mentioned in Chapter 5 utilized the final configuration with 20GB per item memory consumption. This allowed using a batch size of 4 per GPU with 80GB NVIDIA A100s which resulted in quicker and more stable training of models. The duration of input segments was chosen to be 2.97 seconds (131072 samples at 44.1 kHz sampling rate) as data augmentations applied on the fly are significantly sped up when utilizing input segments of 2^n samples.

4.3.7 *Data Augmentation*

GPU-based data augmentations were applied on the fly using the torch-audiomentations (Jordal, 2021) library. The order in which the targets are presented during training was randomized at every iteration to ensure effective permutation invariant training. Each source is randomly pitch-shifted by up to 2 semitones in either direction with a 50% likelihood, independently. Similarly, a randomized gain is applied to each source in a range of -15 dB to +5 dB with a 50% likelihood. Experiments conducted without these augmentations failed to converge well and reported very poor results for unseen songs even with the same singers present in training and test data.

Table 7: Results of 2, 3 and 4 source DPTNet based separation models trained and tested on Choral Mixtures.

Metric	2-source	3-source	4-source
SDR_{in}	0.1949	-3.0593	-4.8173
SIR_{in}	0.1949	-3.0593	-4.8173
$SI\text{-}SDR_{in}$	0.0063	-3.3568	-5.2214
SDR_{out}	17.5583	12.6603	7.2267
SIR_{out}	25.6133	19.2468	11.6555
SAR_{out}	18.4744	13.9668	9.8893
$SI\text{-}SDR_{out}$	16.8424	11.6020	5.3622
ΔSDR	17.3634	15.7196	12.0440
ΔSIR	25.4184	22.3061	16.4729
ΔSI-SDR	16.8361	14.9588	10.5836

4.4 HOW MANY SOURCES CAN YOU SEPARATE?

TasNets with PIT can be scaled to any number of sources without additional computational overhead. Separation models with identical architectures (with additional output nodes for models with more sources) are compared for separating 2, 3, and 4 source mixtures. The Bach Chorales and Barbershop Quartet datasets (described in Section 2.11.3) are used for these experiments with the same training, validation, and test split as used in the experiments in Section 4.3.5 and Section 4.3.6.

Table 7 shows input, output, and improvement metrics for DPTNet models trained for separating 2, 3, and 4 source mixtures of choral music. These models are trained with the same set of singers present in training and testing, but the songs used in testing are unseen during training. The performance reported here is a good representation of the performance difference of the given model for each of 2, 3 and 4 source mixtures.

The first observation is the comparison of the output SDR values, where a 5 dB performance degradation is observed as the number of sources to be

separated is increased. This drop is caused by a combination of two factors. Firstly, the input SDR (equivalent to SNR at input) of a given source in a mixture of sources with equal loudness decreases as the number of sources in the mixture increases since the sum of the energy of interfering sources will increase as the number of interfering sources increases. Secondly, a reduction in the ability of the model to extract the source from the mixture is observed, which is reflected in the ΔSDR and ΔSI-SDR metrics. This implies that as the input SNR of the target source in the mixture reduces, the ability of the model to separate the source reduces.

This reduction in separation capability is reflected both in the capability of the model to remove the interfering sources and the noise that is introduced by the model during separation. While the reduction in SIR improvement might not have a significant negative impact on the perceptual value of the separated output, the significant increase in output SAR for mixtures of more than 3 sources has a significant impact on the perceptual quality of the separated output, as artifacts are strongly undesirable.

These observations help us understand the current state-of-the-art and the applicability of source separation research in the real world. While the overarching desire for an ensemble separation model is to be able to extract any given performer/section from a mixed chamber ensemble/orchestral recording, the current capability of masking-based separation is limited by the loudness of the target source in the input mixture. While source separation of larger ensembles may have value for other downstream MIR tasks such as transcription, using the separated audio stem from high-polyphony mixtures or high input-SNR mixtures is as of yet not achievable.

4.5 CONCLUSION

This chapter presented an approach for handling music source separation in a different format that is applicable to chamber music. It presented a problem formulation that lies at the intersection of the traditional music separation and speech separation problems and utilized vocal harmony separation as a task to test the proposed formulation. The results presented in this chapter only use average metrics across the entire test set, but the variation in performance across different examples was quite high. Although the overall variance/standard deviation for these experiments is not presented in this chapter, Section 6.2.3 and Section 6.3 show the performance distribution including confidence intervals for the 4 source vocal harmony separation task.

Experiments with different levels of polyphony were conducted to assess the performance potential of TasNet models for vocal harmony separation, which prior art suggested to be unsuitable for music separation. Results for 2 and 3 source separation presented minimal artifacts, while higher polyphony resulted in significantly lower performance with sonic artifacts.

However, experiments presented in this chapter relied on a small dataset of 90 minutes with limited diversity, thus the models fail to generalise to unseen singers/datasets. Using the architecture optimizations described in this chapter, the following Chapter 5 discusses the applicability of such models for vocal harmony separation in further detail in Section 5.3, extending it to other monophonic chamber ensemble instruments in Section 5.4 and also addresses the problem of poor cross-dataset generalisability in Section 5.5.

Part III

ENSEMBLE SEPARATION

5

ENSEMBLE SEPARATION USING PERMUTATION INVARIANCE

5.1 INTRODUCTION

Audio source separation aims to extract individual sound sources from a digital audio mixture. Based on the constituents of the input mixture and the target output, the problem definition can be further refined to specific audio separation tasks like speech separation, speech enhancement, and music source separation (Vincent, Virtanen, and Gannot, 2018). While specific sub-tasks in the speech domain like speech denoising, multi-speaker separation, and dereverberation have been explored, music separation research has largely been focused on the demixing challenge (Mitsufuji et al., 2021) aided by the popular MUSDB dataset (Rafii et al., 2019). The demixing challenge is targeted at solving the problem of the separation of vocals, bass, and drums from mixed and mastered pop songs. This has greatly benefited the field by demonstrating that source separation is indeed possible at a commercial scale with state-of-the-art deep learning based architectures. The music demixing challenge has shown successful separation of instruments with distinct spectro-temporal cues like vocals, drums, and bass.

In this chapter, a different area in music source separation is explored with a focus on the separation of chamber ensembles, where the target sources are harmonized have very high spectral overlap, and often contain multiple sources with identical or very similar timbres. Separating ensembles is an inherently challenging task as they combine challenging aspects of both

speech and music separation. In chamber ensembles, the sources are found to occupy similar frequency ranges, thus having label ambiguity similar to the speech separation task (Hershey et al., 2016; Weng et al., 2015) due to multiple sources belonging to the same instrument family. Meanwhile, they are temporally and harmonically correlated (similar to sources in music separation task) which makes the separation problem more challenging due to their musical structure which further increases their spectral overlap.

The problem of label ambiguity is also present in the multi-speaker speech separation scenario, where training a model with class-based target-channel assignment is difficult. Permutation invariant training was introduced as a solution for this task by Yu et al. (2017) to enable speaker-independent speech separation. Instead of solving the separation problem as a class-based regression task, permutation invariant training allows a model to be trained to minimize the separation error only, by not fixing a target-channel assignment but allowing the model to assign any target to any output channel.

In this chapter, Section 5.2 introduces the motivation and utility of permutation invariant training in the context of music separation. Section 5.3 presents preliminary experiments utilizing PIT for choral music separation. While good results were achieved, the experiments suffered from lack of sizeable training data, which resulted in poor cross-dataset performance. To resolve this, the dataset presented in Chapter 3 was used in Section 5.4 to train models that are able to separate mixtures of string instruments. Using this larger dataset cross-dataset performance was improved, and it was also observed that PIT based models could be extended to a variety of instrument timbres. This was further explored in Section 5.5 which explores the timbre-agnostic separation of monophonic instruments with further improvements on real world performance using fine-tuning strategies.

5.2 PERMUTATION INVARIANT TRAINING FOR ENSEMBLES

5.2.1 *Motivation*

Typical approaches to source separation assume a class-based regression approach where different output channels of a deep learning model are expected to consistently be able to estimate the desired sound class from a mixture. This approach has the limitation that different sources present in a mixture should belong to distinct classes. While defining the classes as instrument types in music works well, in speech the class-based approach quickly fell out of favour to training models in a class-agnostic fashion using PIT (described in further detail in Section 4.2). The primary modification introduced by PIT was to make the model class-label assignment invariant, which is obtained by calculating the best source-channel assignment (from all possible permutations) and then minimizing the error for the given assignment. In the context of speech separation, this enables the model to learn separation based on acoustic cues that are speaker and language-invariant. With the same motivation, PIT is used to enable the separation of sources based on acoustic cues that are instrument/timbre invariant. This could enable the model to be able to separate unseen instrument types which have been a long-standing problem for music separation, especially due to limited data availability for rarer instruments. This also enables our model to be able to separate multiple instances of the same instrument, which has never been achieved in music separation before.

5.2.2 *Problem Definition*

Models presented in this work were trained to separate mixtures of a given number of monophonic musical sources regardless of the type of instrumen-

t/source, unlike other music source separation tasks. Using a PIT objective (described in Section 2.8) should enable a model to learn features that enable it to separate sources from any given mixture regardless of its constituent instrument timbres. This approach also enables the model to be able to separate mixtures of identical instruments (eg: 2 violins), similar sounding instruments (eg: violin+viola, or 2 singers), and also unseen instruments/sources. Another advantage of using PIT is related to the amount of training data, where $\binom{N}{2}$ training examples can be generated from a piece with N concurrent sources which greatly improves the total number of unique and musically coherent training examples.

There are a few drawbacks to this problem formulation as well. Firstly, there are some monophonic instruments (such as violins) where there are rare instances of a performer playing multiple notes at an instance, in which case the model is confounded since it expects the sources to be monophonic. The second drawback is that due to the nature of PIT, each model is constrained to the number of instruments present in the mixture to be the number of output nodes of the model. Moreover, given that instrument assignment is variable, it is not only unknown at inference which instrument is present on which channel but also long-term consistency for instrument-channel assignment is a concern as well. These issues are investigated in detail in Chapter 6.

5.2.3 *Models*

Popular approaches to perform music source separation have typically relied on magnitude spectrogram masking based methods (Stöter et al., 2019; Hennequin et al., 2019), where the spectrogram of the mixture is provided as the input to the model which subsequently predicts a mask that is applied on the mixture spectrogram to suppress all non-target sources. Although magnitude spectrogram-based methods have consistently reported state-of-the-art

results in music separation (Takahashi and Mitsufuji, 2020), the lack of accurate phase estimation still proves to be the Achilles heel of this task.

Time-domain source separation models have surpassed this threshold in the domain of speech separation, due to their ability to encapsulate phase information in the learnt filterbanks which removes the requirement of phase reconstruction (Heitkaemper et al., 2020). Since Conv-TasNet (Luo and Mesgarani, 2019), further developments on time-domain source separation (Luo, Chen, and Yoshioka, 2020; Chen, Mao, and Liu, 2020; Zeghidour and Grangier, 2020) methods have pushed speech separation performance far beyond soft-masking based approaches for single-channel 2 and 3 speaker separation tasks. Although music separation has seen some success with time-domain approaches (Stoller, Ewert, and Dixon, 2018b; Défossez et al., 2019), these approaches introduce time-domain processing in a direct regression fashion to model musical sources, whereas popular speech separation models work with a different encoder-masker-decoder philosophy with a significantly smaller number of model parameters.

More recently, complex-domain spectrogram models such as DCUNet (Choi et al., 2019) have shown great success in source separation and can perform competitively to time-domain source separation models with much lesser computational expense. This is mainly due to the difference in hop size between spectrogram-based and time-domain models. Time domain models typically operate with really small hop sizes ranging between 1-16 samples whereas spectrograms operate with hop sizes of 128-1024 samples, resulting in a much lesser number of operations per second of audio. The majority of experiments presented in this work use time-domain models but comparisons with a complex-domain separation model (Choi et al., 2019) are also presented. Additional details about the implementations of the used models are elaborated in Section 4.3.1.

In the presented experiments, ConvTasNet (described in Section 4.3.1.1) was used for the initial experiments described in Section 5.3. The network pa-

rameters[1] were modified to accommodate for the higher sampling rate data in this use case within the available GPU resources (NVIDIA RTX 2080). Subsequently, the experiments were moved to larger high-performance compute clusters (HPCs) which had GPUs with larger memory (NVIDIA Tesla V100s) which enabled the utilization of more advanced network architectures. Subsequently, experiments with DPRNN architecture (Luo, Chen, and Yoshioka, 2020) were explored which bypassed the receptive field limitation introduced by the TCN stack in ConvTasNet (Luo and Mesgarani, 2019). Around the same time DPTNet (Chen, Mao, and Liu, 2020) was released which presented improvements over DPRNN by adding a transformer encoder with multi-head attention in conjunction with the dual-path processing-based chunking framework. Thus most of the subsequent experiments in Section 5.4 primarily report results based on DPTNet. In Section 5.5, DPTNet (Chen, Mao, and Liu, 2020) is used as a baseline for end-to-end free-filterbank based source separation, and comparisons with the complex-domain model DCUNet (Choi et al., 2019) were presented as a complex-domain spectrogram-based separation baseline. Some other complex domain separation models were also experimented with, namely DCCRNet (Hu et al., 2020) and LaSAFTNet (Choi et al., 2021). DCCRNet models failed to converge on the given training pipeline with the multi-mic augmentation, likely due to the model being originally designed for source enhancement and dereverberation. LaSAFTNet-based experiments did converge but were not directly comparable to the other baselines, as LaSAFTNet is a source label conditioned separation model which alters the problem definition as it cannot accommodate mixtures of identical instruments and cannot be trained in a permutation invariant fashion. Details regarding the implementation and optimization of the TasNet models are described in Chapter 4.

1 Complete model parameters and audio examples for each model can be found at: http://c4dm.eecs.qmul.ac.uk/ChoralSep/.

5.3 CHORAL ENSEMBLE SEPARATION

5.3.1 *Introduction*

Choral music consists of a group of singers typically singing the same lyrics but in different vocal styles and notes creating a polyphonic harmony. These different vocalists are usually categorized into 4 parts by their singing style and vocal registers (Shewan, 1979). These classes are often also used to identify parts of other musical ensembles such as brass sections. Such musical ensembles consisting of sources with similar timbres can be defined as monotimbral ensembles. Polyphonic vocal recordings are an inherently challenging source separation task due to the melodic structure of the vocal parts and the unique timbre of its constituents.

Two recent works (Petermann et al., 2020; Gover and Depalle, 2019) explore score-informed choral separation utilizing conditioned U-Net (Jansson et al., 2017) and Wave-U-Net (Stoller, Ewert, and Dixon, 2018b) architectures. While both models show reasonable success, it is difficult to compare the performance of the two since (Petermann et al., 2020) is trained and evaluated on real data with bleed, and (Gover and Depalle, 2019) utilizes synthesized vocal choirs. The presented time-domain source separation models in Chapter 4 outperform the non-informed separation baselines presented in (Petermann et al., 2020; Gover and Depalle, 2019). Moreover, the presented model performs better than even the score-informed models presented in (Petermann et al., 2020; Gover and Depalle, 2019).

Unlike current deep-learning based choral separation models where the training objective is to separate constituent sources based on their class, models are trained using a permutation invariant objective in these experiments. Using this state-of-the-art results were achieved for choral music separation. In this section, a time-domain neural network architecture re-purposed from

speech separation research was used to separate *a capella* mixtures. For these experiments, four-part (soprano, alto, tenor and bass) *a capella* recordings of Bach Chorales and Barbershop Quartets were used to train and test the applicability of such models for vocal harmony separation.

Two different encoder-masker-decoder type TasNet based architectures were adapted for vocal harmony separation. The task of Vocal Harmony Separation fits in a unique space between music and speech separation, where challenging aspects of both tasks are combined. Sources present in choral mixtures are often very similar with weak distinction between them thus allowing the possibility of training them using permutation invariant training (Kolbæk et al., 2017) like speech separation models. Meanwhile, unlike speech separation, the sources present in these mixtures are highly correlated and synchronized to each other as they sing the same lyrics with unique harmonizations. This poses a unique problem where there is minimal timbral distinction between the sources and high temporal synchronization and frequency overlap due to their musical structure. The implications of these constraints on the training methods of these models are explored in Section 6.2. While randomized mixing is a well-established data augmentation technique used in music source separation (Uhlich et al., 2017), the implications of randomized mixing for choral separation experiments presented here are explored in Section 6.3.1.

5.3.2 *Data*

There are very few clean datasets available for choral music, where isolated ground truth for each source is present. This is especially challenging as compared to other forms of music since choral singers typically perform together and are rarely recorded in isolation (Ihalainen, 2008). It is known that choral singers tend to perform much better when the entire choir performs together in a physical space (Fischinger, Frieler, and Louhivuori, 2015),

i.e. each singer can monitor themselves and the rest of the choir with every participant making minor adjustments during performance (Dai and Dixon, 2017). This makes it very difficult to record each individual singer without any bleed from the other sources. There is one publicly available dataset that consists of 3 choral pieces performed by 16 singers (Cuesta et al., 2018), but the recordings are not clean as all the sources are recorded simultaneously. This causes the non-target sources to bleed into each of the recordings, resulting in a noisy ground truth.

For the presented experiments, two datasets of *a capella* recordings without bleed from (Schramm, Benetos, et al., 2017) were used, 26 songs from Bach Chorales (BC) and 22 songs from Barbershop Quartets (BQ). The two datasets combined have a total of 104 minutes of 4 parts: Soprano, Alto, Tenor, and Bass (SATB) recordings, where BC contains 2 male (tenor and bass) and 2 female (Soprano and alto) vocalists, and BQ contains all 4 male vocalists. The songs present in the combined dataset were split into 3 groups for training, validation, and testing roughly in ratio 8:1:1 (since song lengths vary), making sure that the test and cross-validation sets consist of songs (and not just segments) that are unseen in the training set.

5.3.3 *Training*

The models based on Conv-TasNet and DPTNet were trained for 200 epochs with early stopping given a patience of 30 epochs. SI-SDR by Roux et al. (2019) (see Section 2.2 for more details) was used as the loss function as shown in Equation 12 where $\bar{x}$ is the predicted source and x is the target source. SI-SDR was used in a class-agnostic/permutation invariant (described in Section 2.8) fashion where the pair-wise SI-SDR for each predicted source w.r.t. each target source was computed, and then the prediction-target assignments for the lowest cumulative SI-SDR was chosen for backpropagating the loss through the network.

For the Conv-TasNet based model, the learning rate was initialized to $5e^{-3}$ with a scheduler that halves the learning rate if the validation loss (cross-validation set of 2 unseen songs) does not improve for 3 consecutive epochs. For the DPTNet-based model, the linear warmup followed by exponential decay scheduler was used as presented in the original paper (Chen, Mao, and Liu, 2020).

5.3.4 *Results*

Table 8 reports the performance of these models was evaluated on 4 unseen songs (9 minutes in total) from the Bach Chorales dataset (track names: [*BC032,BC049,BC057*]) and Barbershop Quartet dataset (track name: [*BQ041*]). It was observed that although the training and test sets are similar (due to similar singing style and limited variety of vocalists), these models perform better than other non-informed separation models based on U-Net and Wave-U-Net (Petermann et al., 2020) which were trained on a dataset of similar duration and limited diversity (Cuesta et al., 2018). This non-informed model performs at par with the state-of-the-art score-informed separation models Conditioned U-Net (Petermann et al., 2020) and Conditioned Wave-U-Net (Gover and Depalle, 2019). It must be noted that results from both (Petermann et al., 2020; Gover and Depalle, 2019) were reported on different datasets than the ones used in our experiments, thus it is difficult to make conclusive performance comparisons on the reported results. Moreover, the models trained were definitely overfitting to the limited data available and did not perform well on unseen data. This was the motivation to eventually generate EnsembleSet as described in Chapter 3 which enabled the experiments described in Section 5.4 and Section 5.5.

Table 8: Results for 4-source Choral Music Separation w.r.t. other works in literature. It must be noted that both (Petermann et al., 2020; Gover and Depalle, 2019) use different datasets to train and evaluate their models and thus are not directly comparable.

Model	**SIR**	**SAR**	**SDR**
ConvTasNet	+12.23 dB	+9.27 dB	+7.52 dB
DPTNet	**+14.42 dB**	**+10.25 dB**	**+8.61 dB**
U-Net (Petermann et al., 2020)	+9.30 dB	+5.69 dB	-
Wave-U-Net (Petermann et al., 2020)	+7.07 dB	+5.54 dB	-
Wave-U-Net (Gover and Depalle, 2019)	-	-	+5.4 dB
C-U-Net (Petermann et al., 2020)	+12.08 dB	+7.21 dB	-
C-Wave-U-Net (Gover and Depalle, 2019)	-	-	+8.1 dB

5.4 MONOTIMBRAL ENSEMBLE SEPARATION

5.4.1 *Introduction*

In this section, a different area in music source separation is explored, with a focus on the separation of chamber ensembles. In these chamber ensembles, the target sources are harmonized and have very high spectral overlap but are not as temporally synchronized as choral music. The music demixing challenge has shown successful separation of instruments with distinct spectro-temporal cues like vocals, drums, and bass (Stöter, Liutkus, and Ito, 2018; Mitsufuji et al., 2021). The chamber ensemble separation problem has two significant differences from the aforementioned task. Firstly, if the sources in the mixtures are similar sounding (e.g., a mixture of a strings section), it results in high spectral energy overlap. This is further compounded by the fact that such sources often play in a very synchronized fashion while harmonizing with each other. Secondly, often in such mixtures, there may be multiple sources of the same instrument family present (such as a string ensemble). Not only are the distinctions between timbres of instruments of the same family often very similar and difficult to distinguish, but such monotim-

bral ensembles may also contain multiple instances of the same instrument, which makes it an unsuitable problem to be solved using *class-based* separation methods (Hershey et al., 2016). The task of separating such mixtures with constituent sources suffering from label ambiguity and high timbral similarity can be called *monotimbral ensemble separation*.

There are some tasks that fit the definition of monotimbral separation that have been explored recently. One is vocal harmony separation (Petermann et al., 2020; Gover and Depalle, 2019; Sarkar, Benetos, and Sandler, 2021; Chandna et al., 2022) which was discussed in Section 5.3. The only work that aims to tackle the exact task mentioned in this section was a zero-shot learning framework for simultaneous separation, transcription, and synthesis of 2 source chamber ensemble mixtures from the URMP dataset (Lin et al., 2021), where the task of separating mixtures of string instruments has been tackled.

5.4.2 *Data*

EnsembleSet (Sarkar, Benetos, and Sandler, 2022), a multi-track chamber ensemble music dataset (described in Chapter 3) was used for training. The dataset's 18 unique multi-mic recordings and 2 professional mixes were used as data augmentation to avoid overfitting models to the synthesized dataset. The total length of unique monophonic training data amounted to 498 hours including the multi-microphone augmentations. The string quartet track RM-C021 is excluded from the training and validation data and used as test data to present as a baseline for the same dataset performance. The training and validation data was generated by randomly choosing 90% of the remaining dataset as training set and 10% as validation set.

Since the models are trained on synthetic data, real-world recordings from the URMP dataset and TRIOS dataset (described in Section 2.11) are used

for testing these models cross-dataset performance and real world generalisability. 10% of tracks from URMP dataset were used as test data, and the remaining dataset was split 9:1 to generated the train and validation data for experiments presented in Table 10 and Table 11. For experiments involving the TRIOS dataset, the entire dataset excluding the piano stems was used as the test dataset.

5.4.3 *Training*

The Dual-path Transformer (DPTNet) based architecture using PIT was modified by altering the filterbank, scheduler, and other network parameters to accommodate input segments at a sampling rate of 44.1kHz (described in Section 4.3.1.2). The model takes 2.97 second input frames (131072 samples) with 8 repeating separator units. Choosing a filter length of 32 samples with a hop size of 4 samples for the encoder-decoder filterbank resulted in the best results in these experiments. Utilizing a PIT loss for monotimbral ensembles is particularly well suited, as this enables our model to be able to separate any two monophonic instruments regardless of their instrument label.

The model was trained using all valid combinations of chamber ensemble duets playing simultaneously from EnsembleSet (ES) amounting to about 53 hours of data. To achieve this a novel dataloader was implemented 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).

The models were trained for 100 epochs with an early stopping patience of 10 epochs. The learning rate is initialized to $5 \times e^{-3}$ with a scheduler that halves the learning rate if the validation loss does not improve for 3 epochs. The models were trained on 4 x NVIDIA A100 GPUs using a distributed

Metric	All instruments	Strings only
SI-SDR	9.059	9.213
SDR	11.368	11.211
SIR	17.507	14.361
SAR	17.598	16.617

Table 9: Comparing performance of DPTNet trained using all chamber ensembles and only string ensembles from EnsembleSet, tested on string ensembles from URMP dataset

data parallel back-end. Each epoch in the experiments took 40 minutes with a batch size of 1 per GPU.

5.4.4 *Beyond Monotimbral*

Initial experiments using EnsembleSet were focussed on string ensembles, where only mixtures of string instruments were presented during training and were tested exclusively on mixtures of string instruments from the URMP Datasets. Based on the prior art, it was unknown whether TasNet-based models with free-filterbanks were able to successfully model sources of different timbral characteristics/waveshapes, as prior experiments with Music Source Separation and ConvTasNet in (Défossez et al., 2019) reported significant audio artefacts. However, it was observed that including non-string instruments in the training set did not harm separation performance as seen in Table 9. Although initially suspected to be due to the very limited amount of wind and brass instrument data present in EnsembleSet ($\leq$ 30 minutes), this is explored further in Section 6.4.

Subsequently, EnsembleSet was used to train a source separation model that is able to separate any chamber ensemble duet, regardless of instrument family type (see Table 11). While the model was trained exclusively on a synthesised dataset, its performance was evaluated on real-world data from the URMP dataset (Li et al., 2018). The multi-mic renders that are available

in EnsembleSet were used as a form of data augmentation by randomizing the mix/mic(s) presented to the model at each epoch. In addition, other augmentations including pitch shift and gain modulation were used to help the model generalize better to unseen source/microphone configurations.

5.4.5 *Separating Real-world Mixtures*

Data augmentation is essential for cross-dataset generalisability, especially given that the models were trained exclusively on synthesised data. Prior work utilizing synthesised datasets such as (Manilow et al., 2019) has consistently struggled to generalise well to unseen real-world data. To tackle this challenge, torch-audiomentations (Jordal, 2021; Pariente et al., 2020) was used for data augmentations such as gain modulation, channel swap, and pitch-shifting by up to +/- 2 semitones. It must also be noted that the temporal and harmonic integrity of the mixtures was maintained through all the data augmentations. This is unlike the typical music separation data augmentation pipeline where the constituent parts of the mixtures are randomized across different songs at every epoch during training (Uhlich et al., 2017). The URMP dataset (Li et al., 2018) was used to generate real examples for cross-validation and testing in a similar fashion to our pipeline with EnsembleSet resulting in 4.5 hours of 2 source mixtures.

5.4.5.1 *Choice of Mic/Mix Renders from EnsembleSet*

Given the wide variety of microphone placements and mix configurations available in EnsembleSet, initially, the mono and Close microphones were chosen for the experiments as these were the only single mic renders available in the dataset. With the assumption that these renders would result in the most similar characteristics compared to the URMP dataset, the experiments used EnsembleSet duet mixtures for training and cross-validation. Ini-

Test Data	ESCV	ES+URMPCV	URMPCV
ES	13.24 dB	12.56 dB	14.21 dB
TRIOS	14.43 dB	13.62 dB	14.54 dB
URMP	9.29 dB	8.45 dB	9.15 dB

Table 10: SI-SDR performance of monotimbral ensemble separation models trained on EnsembleSet with different validation datasets, tested on EnsembleSet, TRIOS and URMP datasets.

tial experiments using single-mic data for training and cross-validation from EnsembleSet performed poorly (SI-SDR $<$ 3dB) when tested on the URMP dataset. These trends were observed across various architectures.

Subsequently, the multi-mic renders of each instrument track were used as data augmentation by randomly choosing one of the 20 renders for each instrument for each training and cross-validation step. It was observed that choosing different renders for training and cross-validation resulted in the training not converging well and negatively affected performance even for the same dataset test scenarios. It was also observed that using multi-mic renders greatly improves cross-dataset performance for DPTNet. This is elaborated further with analysis across various models in Section 5.5.6.

5.4.5.2 *Choice of cross-validation data*

Initial experiments using single-mic renders from EnsembleSet also experimented with different cross-validation data splits between EnsembleSet and URMP with the goal of achieving better cross-dataset performance. It was observed that the choice of different cross-validation sets did have varying impacts on performance when tested across different datasets. Experiments on the TRIOS dataset are included as an independent control test where the test data is independent of the training and validation data. Table 10 shows that mixing data from EnsembleSet and URMP consistently reduced performance for all test scenarios. This could be attributed to the significant differences in acoustics and processing between URMP and EnsembleSet.

5.4.6 *Results*

The presented baseline results based on the experiments described above are compared to previous work conducted for a similar task as described in (Lin et al., 2021). The results from Lin et al. (2021) 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. The model's cross-dataset evaluation performance is compared between the URMP Dataset and EnsembleSet (ES) with the experiments from Lin et al. (2021) (presented in Table 11).

The DPTNet model trained on URMP and tested on ES performs the worst, which is expected since URMP is a very small dataset. This model is the only model that performs poorer than the prior-art (MSI). This is expected as the MSI model is trained and tested on the same dataset. It must be noted that while both the MSI experiments and our experiments use the URMP dataset as test set, the test sets are not identical, since our test set also includes mixtures of the same instruments. Moreover, the MSI experiments perform score-informed separation.

The other results simultaneously highlight the capability of the presented DPTNet architecture, and the EnsembleSet dataset. It is seen that the model trained and tested on URMP shows the highest SI-SDR, which is understandable since URMP dataset is so small, the model probably overfits. The model that was both trained and tested on ES also reports a very high SI-SDR, likely due to overfitting. While the slightly lower average SI-SDR may be attributed to the larger test dataset size of ES, the difference is too small to make any significant conclusions regarding the diversity of EnsembleSet and URMP. However, both these results may be indicative of the upper limit of SI-SDR performance achievable by the DPTNet architecture.

Model	Train	Eval	SDR	SI-SDR
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
DPTNet	URMP	URMP	+14.69 dB	+13.24dB
MSI (Lin et al., 2021)	URMP	URMP	+6.33 dB	-

Table 11: 2-source Chamber Ensemble Separation results.

The DPTNet model trained on ES and tested on URMP shows significantly higher performance as compared to the MSI experiments. This exhibits the value of the synthesised dataset and shows that PIT-based DPTNet models trained exclusively on synthetic data from ES are able to produce generalisable results on unseen real-world mixtures. It also exhibits the inherent capability of PIT-based DPTNet models, as it significantly outperforms the score-informed MSI model even when its trained only on synthetic data, while the MSI model was trained and tested on the same dataset.

5.5 DOMAIN ADAPTATION FOR IMPROVING ENSEMBLE SEPARATION

5.5.1 *Introduction*

In this section, the experiments presented are based on separating mixtures that contain any pair of monophonic instruments. The observations from Section 5.4 show that DPTNet trained with PIT is able to generalise across a range of instrument timbres. However, a drop in performance was observed when models are trained and tested on different datasets. In this section, the experiments are focussed on further exploration of what makes these models generalise across timbres. Moreover, how far the performance of these models can these models be pushed both for unseen datasets and unseen tasks (like vocal separation) using domain adaptation is explored.

A similar data augmentation method as Section 5.4, enabled by the multi-microphone renders available in EnsembleSet is used and its impact on cross-dataset generalisability for time-domain and complex-domain separation models is evaluated. A pre-training strategy using synthetic data from EnsembleSet followed by fine-tuning on real-world dataset URMP is presented for improving cross-dataset performance, which enables complex domain model DCUNet to perform comparably to time-domain models. These complex domain models were unable to separate instrument mixtures from unseen datasets in the previous experiments, which in Section 5.5.6 is shown to be due to the subtle effect of the data augmentation on spectrograms, as compared to raw/time-domain data. The impact of pre-training using EnsembleSet for both same-domain tasks (chamber ensemble instruments from URMP dataset) and cross-domain tasks (harmonized vocals from Bach Chorales and Barbershop Quartet datasets) is explored.

It is found that fine-tuning using very limited amounts of target domain data (from URMP) reports a 5.5 dB SI-SDR improvement compared to training on ES alone. It is also found that pre-training on ES improves DPTNets SI-SDR performance on URMP dataset by 1.6 dB as compared to training on URMP alone. Meanwhile, using almost half of the training data from BCBQ (used experiments presented in Section 5.3) as fine-tuning data for domain adaptation, DPTNet models show almost 12 dB SI-SDR performance improvement on choral separation as compared to training on ES only. Moreover, pre-training models on ES show an improvement of 1.1 dB SI-SDR than training on BCBQ alone.

5.5.2 *Data*

The models presented here are pre-trained on synthetic data from EnsembleSet (described in Chapter 3). To provide a baseline result to compare the domain adaptation/finetuning experiments, the training and validation data

used in these experiments for pre-training are the same as the training data used in experiments presented in Section 5.4.

Since the models are pre-trained on synthetic data, real-world recordings from the URMP dataset (described in Section 2.11) are used for the fine-tuning experiments. For the fine-tuning experiments, only one piece (String Quintet K515) of 3 min 45 sec duration was used for fine-tuning related training and validation while the remaining dataset was used for testing the performance of these fine-tuned models.

To study the transferability of features learned from chamber ensemble instruments to vocals, which have significantly different dynamics and modulations compared to bowed and wind instruments, the Bach Chorales and Barbershop Quartets (BCBQ) datasets (see Section 2.11.3) are used for domain-adaptation experiments. For these experiments, half of the duration of the dataset was used for training and validation for domain adaptation, while the remaining half of the dataset was used for testing the domain-adapted models. It must be noted that the amount of training data from BCBQ used in these experiments is half of what is used in the experiments described in Section 5.3.

5.5.3 *Training*

Two different baselines were chosen for the experiments, one for time-domain end-to-end separation (DPTNet (Chen, Mao, and Liu, 2020)), and one for complex domain separation (DCUNet (Choi et al., 2019)), both of which have shown comparable results for speech separation using PIT.

The DPTNet models (9.9M parameters) as described in Section 2.6.2 are used for experiments with URMP and Choral Music, respectively. The models are trained at 44.1 kHz except for experiments related to vocal harmony mixtures, where the models are pre-trained at 22.05 kHz as the test dataset

is bandlimited and training high-sample rate models with bandlimited data has been reported to introduce noise in the separated output from the model (Sarkar, Benetos, and Sandler, 2021).

DCUNet (7.7M parameters) builds upon the original U-Net by introducing a phase-aware complex-valued masking framework (see Section 2.7 for more details). The asteroid (Pariente et al., 2020) implementation of this model with PIT is used as a baseline to compare and contrast with TasNet-based experiments.

The models are trained with synchronized pairs of musical instruments. The data pre-processing involves activity detection on the source monophonic instrument audio files and identifying frames of 2.97 seconds (131072 samples at 44.1 kHz) where both instruments are concurrently active for at least 40% of the frame. The train-validation split is generated by randomly choosing 10% of the training frames presented to the dataloader as validation set. The input mixtures are generated by linearly downmixing the augmented versions of our reference sources.

All of the models are trained at 44.1 kHz, except the experiments associated to vocal harmony separation which are trained and evaluated at 22.05 kHz. SI-SDR is used as the loss function with PIT. The DPTNet models are trained for 100 epochs with early stopping patience of 10 epochs. The learning rate is initiated at $5 \times e^{-3}$ with a scheduler that halves the learning rate if the validation loss does not improve for 3 epochs. The models are trained on 4 x NVIDIA A100 GPUs with a batch size of 3 per GPU using a distributed data parallel backend.

5.5.4 *Data Augmentation*

Data augmentation is applied on-the-fly using torch-audiomentations (Jordal, 2021) and is applied across all the experiments, except using multi-mic ren-

ders from EnsembleSet. Gain modulations are applied with a 50% random chance to each of the sources in a mixture separately in the range of +5dB to −15dB, pitch shift by up to ±2 semitones, followed by channel swaps for the reference targets.

The experiments using EnsembleSet for training have the opportunity to use the 20 unique microphone and mix configurations that are presented in the dataset for each source. Since most of these renders are stereo and utilise multiple microphones, they are downmixed to mono for the experiments. This exposes the models to a good variety of recording and microphone configurations which could improve the models' cross-dataset generalizability. To enable this, the dataloader selects one of the 20 available renders at random at each training iteration and the data augmentation described before is applied subsequently.

5.5.5 *Fine-tuning/Pre-training*

To enable the model to adapt to unseen acoustics and instruments, it is proposed to use EnsembleSet with multi-mic augmentation as a pre-training step before finetuning/re-training the model on limited test-domain data for improved performance. For the pre-training stage, the same training configuration as the EnsembleSet baseline experiments is used, where the train and validation sets are generated with a random split and the model is trained for 100 epochs with an early stopping patience of 5 epochs. Subsequently, the learned model weights are used as the initial weights for re-training on target domain data with a low learning rate of $1 \times e^{-6}$. For URMP cross-dataset performance experiments, a single song from URMP dataset was used, and 10 songs each from BC and BQ datasets as the fine-tuning data while using the remaining tracks as test data. No layers were frozen during this fine-tuning stage due to the nature of joint optimization of the free-filterbank and the separator stack in DPTNet.

5.5.6 *Impact of Microphone Augmentation*

The DPTNet and DCUNet models were trained on EnsembleSet with and without multi-mic augmentation and tested on EnsembleSet and URMP. Table 12 shows the results of the models trained on EnsembleSet alone with and without multi-render data augmentation and tested on a separate test set from EnsembleSet and real-world data from URMP. It was observed that both models suffer from overfitting and poor cross-dataset performance when tested on URMP data when trained without using multi-mic augmentation (MicAug). However, a significant improvement in cross-dataset performance was observed when using multi-mic augmentation only on DPTNet, while DCUNet results do not show any noticeable difference. The performance drop observed on evaluation on EnsembleSet for DPTNet models trained with MicAug is because the same "Close" microphone configuration was used for training and testing the without MicAug scenario. This shows how the DPTNet model can overfit to a given microphone configuration.

The lack of improvement for DCUNet models could potentially arise from the fact that the multi-mic renders for a given audio segment would result in a much more drastic change in the signal when observed in the raw/time-domain representation which only affects DPTNet. On the other hand, the complex-domain spectrogram representation would show changes based on microphone characteristics and phase differences, which are represented as relative differences between the real and imaginary parts of the complex spectrogram. It is however unclear, if the microphone augmentation fails to improve complex-domain separation due to the nature of these relative differences in the real and imaginary representation of the complex domain being difficult to capture for the DCUNet model's encoder, or because DCUNet models are inherently biased towards learning magnitude spectral features (which don't show significant difference across augmentations) due to the bounded masking method which relies on estimating the magnitude mask

as the bounded target and subsequently extracting the phase from the predicted bounded magnitude mask.

Table 12: Output SI-SDR for 2-source Chamber Ensemble Separation models trained on EnsembleSet with and without multi-mic augmentation (MicAug), tested on EnsembleSet (ES) and URMP.

Model	Sample Rate	MicAug	ES	URMP
DPTNet	22.05 kHz	✓	+13.67 dB	+9.39 dB
DPTNet	22.05 kHz	✗	+18.39 dB	+5.74 dB
DPTNet	44.1 kHz	✓	+13.24 dB	+9.23 dB
DPTNet	44.1 kHz	✗	+18.84 dB	+3.54 dB
DCUNet	44.1 kHz	✓	+14.43 dB	+4.49 dB
DCUNet	44.1 kHz	✗	+14.43 dB	+4.65 dB

5.5.7 *Cross-dataset Performance*

Table 13 shows the models' evaluation results on cross-domain real-world datasets after pre-training on EnsembleSet with and without fine-tuning. The experiments demonstrate that pre-training using EnsembleSet leads to better separation results for both chamber ensemble separation (tested on URMP dataset) and vocal harmony separation (tested on BCBQ dataset). Interestingly, even though choral singing is significantly different from chamber ensemble instruments, pre-training on chamber ensembles provides a +1.08 dB performance improvement over training on harmonised vocal data alone. While the SI-SDR achieved for vocal harmony separation is higher, it must be noted that the vocal harmony separation experiments were run at 22.05 kHz (instead of 44.1 kHz for other experiments) due to the band-limited nature of the BCBQ datasets as noted in (Sarkar, Benetos, and Sandler, 2021).

Table 13: SI-SDR for 2-source Ensemble Separation models trained on EnsembleSet with fine-tuning on respective test datasets.

Model	SR	Train	Test	FT	SI-SDR
DPTNet	22.05 kHz	ES	BCBQ	✗	4.99 dB
DPTNet	22.05 kHz	ES	BCBQ	✓	17.92 dB
DPTNet	22.05 kHz	BCBQ	BCBQ	✗	16.84 dB
DPTNet	44.1 kHz	ES	URMP	✗	9.23 dB
DPTNet	44.1 kHz	ES	URMP	✓	14.87 dB
DPTNet	44.1 kHz	URMP	URMP	✗	13.25 dB
DCUNet	44.1 kHz	ES	URMP	✗	4.49 dB
DCUNet	44.1 kHz	ES	URMP	✓	12.71 dB
DCUNet	44.1 kHz	URMP	URMP	✗	10.61 dB

5.6 DISCUSSION

In this chapter, a completely new problem of ensemble separation was tackled and explored in a novel fashion using PIT. The observations presented in this chapter shed light on the critical importance of sizeable, clean and diverse training data which in fact in most of the presented experiments has a larger impact than the choice of the deep-learning separation model used. It also highlights the interdependence of data, models and training paradigms, and the fact that they cannot be evaluated independently. This observation is clearly highlighted in the experiments related to multi-mic data augmentation vs. generalisability (in Section 5.5.6) as its impact depended on the model used and also varied based on the diversity of training data and the relative domain shift in the test data.

Section 5.3 first investigates use of PIT to solve the choral separation problem. While using TasNets with PIT did perform very well, and arguably outperformed other class-based methods. Direct comparisons were difficult as no standard sizeable training data was available and all results reported in this work and in the prior art were all trained and tested on very small

datasets. With the introduction of EnsembleSet, Section 5.4 tackles the ensemble separation problem with cross-dataset performance in mind and showed reasonable success in separating string ensembles. Results from Section 5.4 suggested that models could in fact be timbre-agnostic, which was further explored in Section 5.5. The results presented in this chapter only use average SI-SDR and SDR, but the variation in performance across different examples was quite high. Although the overall variance/standard deviation for these experiments is not presented in this chapter, they are presented in an instrument-wise fashion in Section 6.4 and across different musical contexts in Section 6.5 for chamber ensemble separation.

The experiments presented in this chapter were the first in literature (Sarkar, Benetos, and Sandler, 2021; Sarkar, Benetos, and Sandler, 2022) which showed that deep learning models can in fact be trained in a permutation-invariant fashion to be able to separate mixtures of monophonic instruments with both high-timbral similarity and diversity simultaneously without a performance tradeoff. This would be the first reported results suggesting source separation models might be able to operate in a timbre-agnostic fashion which is significantly divergent from the current narrative presented in source separation literature. This not only enables solving new source separation tasks such as separating mixtures of the same instruments, but also opens up new ways of training models for multiple instruments which can be useful for instruments with limited data. In Chapter 6, this possibility is explored in further detail and presents deep analyses of various instrument mixture scenarios and their impact on the performance of source separation models trained with PIT.

Although the comparisons presented in Table 8 and Table 11 show that PIT based methods seem to outperform other class-based separation methods, the metrics presented are not directly comparable. Presenting a non-PIT baseline result using DPTNet in these experiments could shed light the specific impact of PIT. However, using a class-based training objective inherently

requires changing the data and problem formulation as class-based methods are only applicable for a given pre-defined instrument configuration. This could be achieved by testing a DPTNet model on specific instrument pair configuration, however, that drastically reduces the amount and diversity of training data that is available. Due to these reasons, such a baseline was not included in the ensemble separation results presented in this chapter.

6

DEEPER INSIGHTS INTO ENSEMBLE SEPARATION

6.1 INTRODUCTION

The experiments presented in previous chapters explore the applicability of source separation models for a novel task in the context of music separation. The previous chapters presented the typical metrics used in source separation, SDR, SIR, SAR, and SI-SDR (see Section 2.2 for definitions) as evaluation metrics which compared the performance of the various models presented. While the global average of such metrics is generally a good indicator of the performance of these models, they are susceptible to many biases introduced due to training data, training strategy, and evaluation data. Understanding the real-world applicability of the proposed methods requires a careful analysis of the success and failure modes of these models, especially given that the training used is synthesised and the test data is of limited diversity.

In this chapter, a deeper analysis of the performance of source separation models presented in previous chapters is presented. First, a new measure for the harmonic complexity of an input mixture is presented in Section 6.2. The goal of this measure was to quantify the harmonic overlap present in an input mixture and then this measure is compared with the separation performance of the PIT-based separation models presented in Section 6.2.3. Various methods of computing this score are suggested based on relevance to known harmonic relationships between intervals, which are subsequently tested against different training scenarios.

Preliminary experiments based on choral separation in Section 6.3.1 suggest that random mixing during training deteriorates the separation performance in the context of choral separation. This is contrary to observations reported in the music separation task by Uhlich et al. (2017). Subsequently, this harmonic overlap metric is used along with the traditional source separation metrics such as SDR and SI-SDR to investigate the impact of random track mixing as a data augmentation method in Section 6.3.2. The presented Harmonic Overlap score shows a moderate negative correlation with separation performance however it does not present a consistent correlation when compared with the impact of random mixing. To obtain a clearer understanding of the impact of random mixing, experiments based on a significantly larger dataset EnsembleSet are presented in Section 6.3.3, which shows that the negative impact of random mixing is larger as the amount of training data is reduced which places the observations presented in Section 6.3.1 in better context.

Although the common understanding from the music separation task based on instrument classes would suggest timbral similarity to be a key confounding factor for source separation, results described in Section 6.4 suggest that models trained for experiments from Section 5.4 and Section 5.5 work in an instrument/timbre agnostic fashion, both in the context of the amount of training data available for different timbres and also in the context of separation performance.

Section 6.5 analyses the performance of these models based on different musical scenarios and where pitch overlaps and crossovers are found to have a significant impact. Section 6.6 present a few of the worst-performing examples from the URMP test set, which show a significant number belonging to scenarios with pitch overlaps and crossovers. These examples are found to show very different performance characteristics which are then analysed in the context of the entire test dataset in Section 6.7

Key contributions of this chapter are highlighted below:

- A novel measure called Harmonic Overlap that measures the musical complexity of a mixture is presented. A moderate negative correlation (Pearson correlation coefficient: -0.334) is observed between output SI-SDR and Harmonic Overlap score for 4-part vocal harmony separation experiments from Section 5.3.

- Random track mixing as data augmentation for training ensemble separation models is found to negatively impact performance for smaller training datasets, but at larger dataset sizes the negative impact diminishes.

- Instrument-agnostic performance of chamber ensemble separation models presented in Section 5.4 and Section 5.5 are investigated. It is found that PIT based DPTNet models indeed perform in a timbre-agnostic fashion. It is also observed that separation performance seems to be correlated with pitch trajectories of the constituent sources and is independent of the timbre of the constituent sources.

- Two failure modes are identified for DPTNet based models trained with PIT: *unisons* and *pitch crossovers*. While the models fail to separate sources playing in unison, it is observed that examples with pitch crossovers are still separated well but suffer from *source confusion*, where the instrument-output channel assignment is swapped at the crossover point.

6.2 HARMONIC OVERLAP

Traditional training strategies for source separation using deep learning have advocated for randomised mixing of the constituent parts of a musical mixture as a form of data augmentation (Uhlich et al., 2017). While this has proven to be successful for the traditional music separation task of vocal, drums, and bass stem separation, its applicability in the context of ensem-

ble separation was unknown. One side-effect of using randomised mixtures during training is that the model's exposure to harmonically correlated and synchronised sources is limited. This may be especially impactful for chamber ensemble separation models as the sources in these mixtures have a significantly higher overlap of partials as compared to the stem separation task where vocals, drums, and bass have very minimal harmonic overlap. The experiments presented in this section seek to investigate the correlation between harmonic overlap and separation performance.

6.2.1 *Harmonic Overlap Score*

A novel measure for calculating the harmonic overlap for any two given monophonic sources is presented. It is based on calculating the number of coinciding partials observed in the first 16 overtones for a given pair of notes being played by two sources. This also correlates well with the perceived resonance for any given interval (pair of notes), where the strongest resonances are seen for octave intervals, followed by perfect fifths, perfect fourths, major thirds, and so on. This measure is particularly apt for monophonic sources in an ensemble where such instruments perform together with the intention of blending well with each other to create a coherent sonic texture. The harmonic overlap metric is designed to provide a measure of coherence for such ensembles by calculating pair-wise harmonic overlaps normalised by their duration of activity.

Given a set of N sources x_i for $i \in \{1, 2, ..., N\}$, we utilise the pYIN pitch detection algorithm from Mauch and Dixon (2014) to estimate their pitches F_i^0. We then compute the first 16 overtones P_i^j for $j \in \{1, 2, ..., 16\}$ and quantise them to a 20 cent log-frequency scale for each of the sources as per Equation 21. We convert the obtained set of harmonic pitches to a binary vector $B_{i,k}$ as per Equation 22 for $k \in \{1, 2, 3, ...\}$ where k is the 20 cent quantized note number ($k/5$ gives us the MIDI note number equivalent for a given har-

monic). We subsequently get the harmonic overlap for a given time frame by counting the total number of overlaps per frame for each pair of sources in a mixture as per Equation 23. We then aggregate the pair-wise scores over the entire input segment and normalise the score by dividing by the overall pair-wise activity duration, i.e. for each pair we calculate the total number of frames where both sources were active and divide the aggregated score by that value.

$$P_i^j = \left\lceil 60 \times log_2(\frac{j \times F_i^o}{440}) \right\rceil + 345 \tag{21}$$

$$B_{i,k} = \begin{cases} 1, & \text{for } k \in P_i^j \\ 0, & \text{for } k \notin P_i^j \end{cases} \quad j \in \{1, 2, ..., 16\} \tag{22}$$

$$\textit{Harmonic Overlap} := \sum_{i \neq j}^{N} B_{i,k} \cdot B_{j,k}^T \tag{23}$$

6.2.2 *Implementation*

The harmonic series of a given note does not perfectly overlap with the A440 pitch series for a 12-note equal tempered scale, as seen in Table 14. While computing the harmonic series of a note, only the even-powered harmonics (octaves) coincide perfectly with the equal-tempered tuning chart while most other harmonics typically have small errors. Thus to compute the proposed harmonic overlap score, a minimum pitch resolution needs to be determined to obtain a meaningful score. Figure 10 presents the harmonic overlap score for different musical intervals considering different levels of frequency res-

Overtone	Frequency (Hz)	Closest Pitch	Frequency (Hz)	Error (Hz)
1	880	A5	880	0
2	1320	E6	1318.51	1.49
3	1760	A6	1760	0
4	2200	C#7/Db7	2217.47	17.47
5	2640	E7	2636.99	3.01
6	3080	G7	3135.96	55.96
7	3520	A7	3520	0
8	3960	B7	3951.1	8.9
9	4400	C#8/Db8	4434.94	34.94
10	4840	D8	4698.62	141.38
11	5280	E8	5273.98	3.02
12	5720	F8	5587.71	132.29
13	6160	G8	6271.87	111.87
14	6600	G#8/Ab8	6644.86	44.86
15	7040	A8	7040	0

Table 14: Frequencies and Closest Pitches for the Overtones of A440

olution. These are compared to the empirical understanding of interval relationships, such that octaves are the strongest overlaps, followed by perfect fifths, fourths, and major thirds while dissonant intervals such as diminished fifths should have the lowest overlaps. Comparing the different pitch resolutions results in the choice of 5 bins per semitone to be the proposed measure to generate scores that correlate well with our understanding of pitch interval relationships.

6.2.3 *Harmonic Overlap vs. Performance*

Using the proposed Harmonic Overlap metric, a moderate negative correlation (*Pearson correlation coefficient*: -0.334) between the harmonic complexity of the audio mixture and the separation performance achieved is found (for the model described in Section 5.3), which is shown in Figure 11. This shows

Note	Interval	Bins per semitone (resolution)							
	w.r.t. C3	2	3	4	5	6	8	10	12
C3	0	16	16	16	16	16	16	16	16
C#3	1	2	1	1	0	0	0	0	0
D3	2	2	2	1	1	1	1	1	1
D#3	3	3	2	1	0	0	0	0	0
E3	4	4	4	1	1	0	0	0	0
F3	5	4	4	4	4	4	4	4	4
F#3	6	0	0	3	0	0	0	0	0
G3	7	5	5	4	5	5	5	5	4
G#3	8	3	3	0	1	0	0	0	0
A3	9	3	3	1	0	0	0	0	0
A#3	10	2	2	1	1	1	1	1	1
B3	11	2	1	1	0	0	0	0	0
C4	12	8	8	8	8	8	8	8	8
C#4	13	0	0	0	0	0	0	0	0
D4	14	1	1	1	1	1	1	1	1
D#4	15	1	1	0	0	0	0	0	0
E4	16	3	3	1	0	0	0	0	0
F4	17	2	2	2	2	2	2	2	2
F#4	18	0	0	1	0	0	0	0	0
G4	19	5	5	4	5	5	5	5	4
G#4	20	1	1	0	0	0	0	0	0
A4	21	1	1	0	0	0	0	0	0
A#4	22	0	0	0	0	0	0	0	0
B4	23	1	1	1	0	0	0	0	0
C5	24	4	4	4	4	4	4	4	4

Figure 10: Harmonic overlap scores for all intervals up to 2 octaves for various pitch resolutions. The rows are highlighted based on intervals that should have higher or lower overlap scores based on our understanding of interval relationships. Cells highlighted in red represent hypotheses where the score does not correlate well to perceptual interval relationships.

that mixtures with stronger resonant intervals are more difficult to separate, thus musical structure does impact separation negatively. This agrees with our perceptual ability to distinguish harmonies being sung as the Harmonic Overlap score ranks resonant intervals much higher than dissonant intervals.

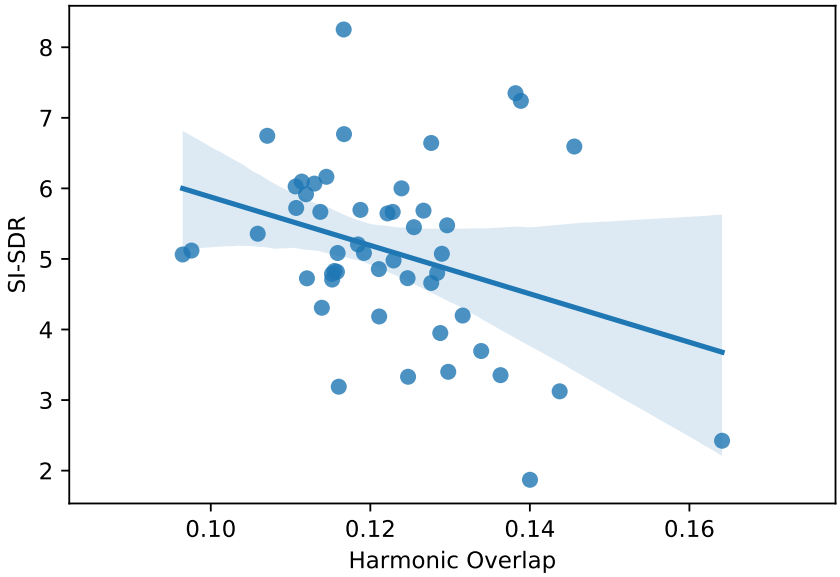

Figure 11: Linear fit with 95% confidence interval of Harmonic Overlap score for test audio mixtures vs. output SI-SDR achieved with ConvTasNet model.

6.3 RANDOM MIXING

Random mixing is a commonly used data augmentation method which was introduced for Music Source Separation by Uhlich et al. (2017) and subsequently also utilised by Zeghidour and Grangier (2020) and Défossez et al. (2019) where they find that mixing segments from different musical pieces to generate new training examples for training improves separation performance. Since finding clean multi-tracks for highly polyphonic ensembles like choral music is difficult, random mixing would enable us to generate data with any amount of polyphony by mixing monophonic singing tracks from various songs.

We systematically study the impact on the model's performance of randomly mixing vocal parts from different songs during training. We randomly choose a number of data samples from the training set and shuffle their constituent parts to generate a training set with a desired percentage of randomisation.

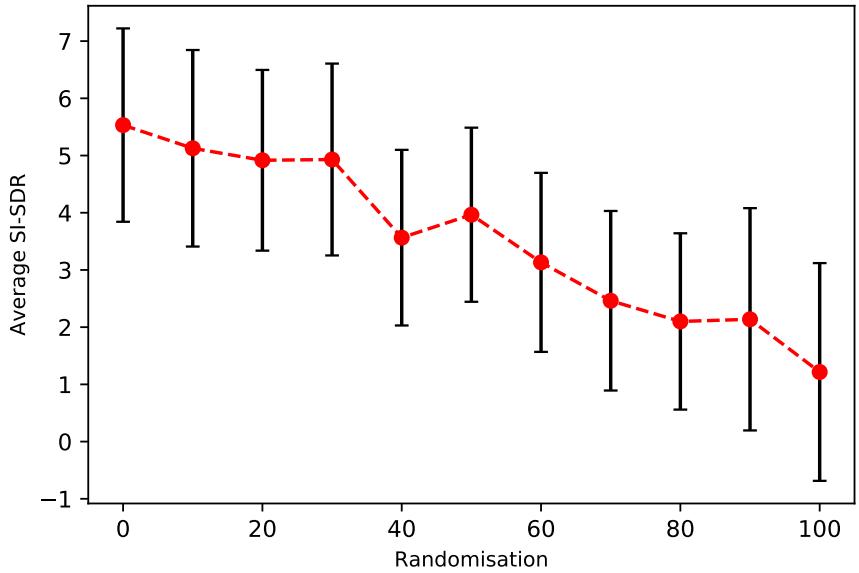

Figure 12: Average output SI-SDR achieved by different ConvTasNet-based models trained with varying proportions of randomised (musically incoherent) and synchronised mixtures.

6.3.1 *Random Mixing vs. Performance*

Models described in Section 5.3 were trained with and without randomisation to test the impact of using randomised training data to train choral separation models with PIT. These experiments show that using random mixing as a data augmentation method results in models failing to converge and/or generalise to unseen test data. Using synchronised training data generated good results. To study the impact of randomised mixing, models with different blends of randomised and synchronised audio pairs from the BCBQ dataset were used to train choral separation models.

In Figure 12 we see that the model performance monotonically decreases as the number of randomised mixtures in the training data is increased. We see a 4.32 dB decrease in average SI-SDR improvement between a model trained on synchronised mixtures vs. randomised mixtures. It is noteworthy that our randomisation process preserves the SATB choral structure, i.e. the

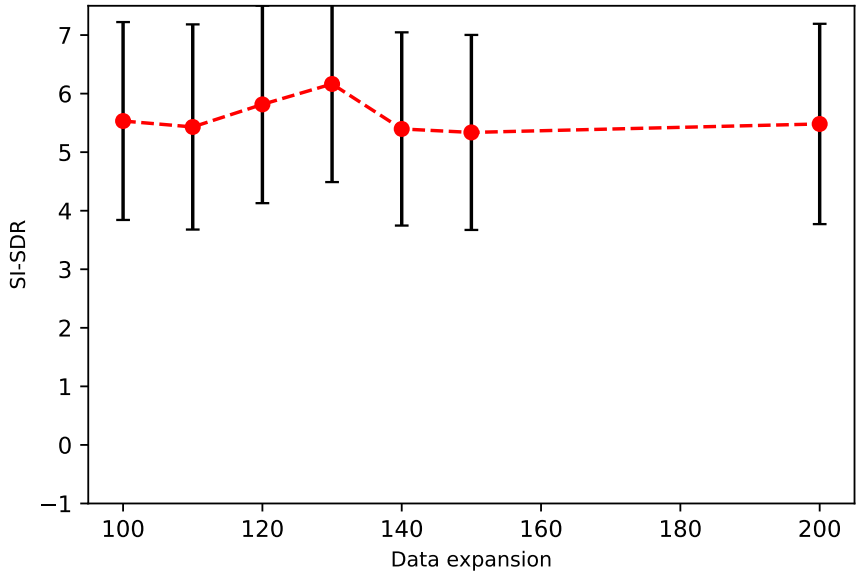

Figure 13: Average output SI-SDR achieved by different ConvTasNet-based models trained with the full amount of synchronised (musically coherent) data with an additional amount of randomised (musically incoherent) mixtures. The X-axis values represent the total dataset size as a percentage of original dataset size.

four sources consist of one source each from soprano, alto, tenor and bass registers.

We also carry out experiments where we increase the overall dataset size by adding new mixtures of generated by downmixing randomly selected solo choral singing segments. Figure 13 shows that increasing the dataset size does not improve separation performance. Models trained on expanded datasets with $10 - 100\%$ additional training samples of randomised mixtures show an average performance difference of $+0.07 \pm 0.32$ dB ΔSI-SDR w.r.t. our baseline model.

6.3.2 *Random Mixing vs. Harmonic Overlap Performance*

Mixing random tracks during training would suggest that the model is presented with mostly harmonically uncorrelated data. This would suggest that models trained on such data could find harmonically correlated content more

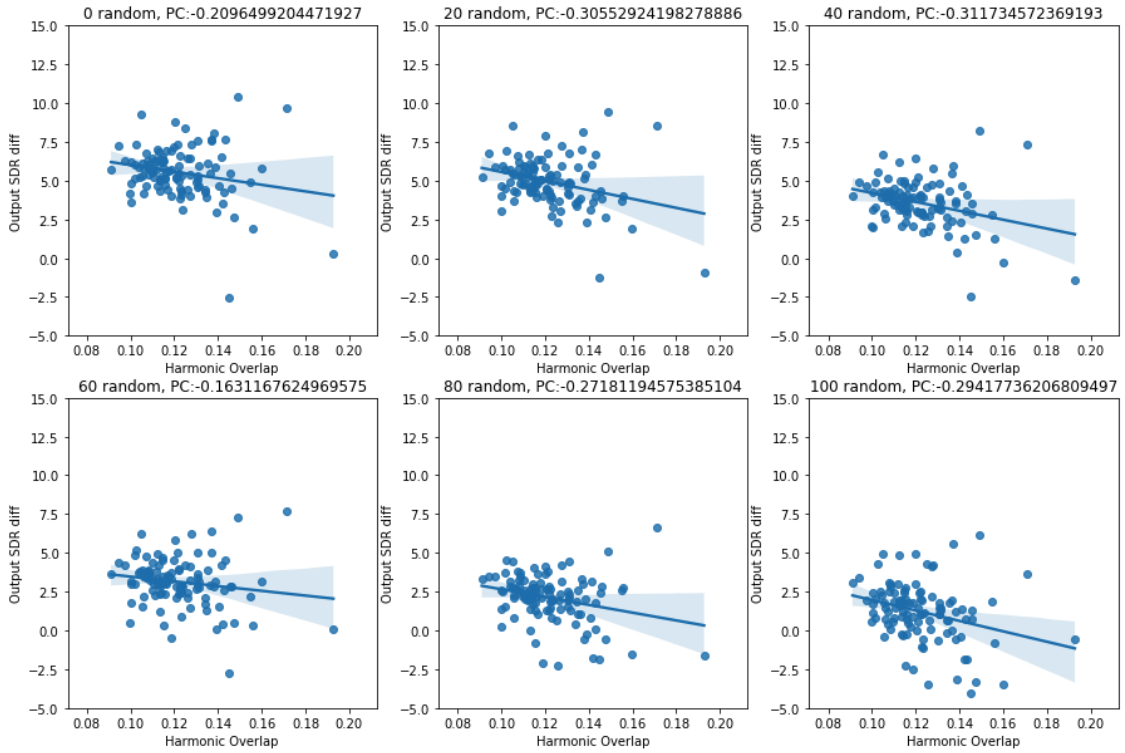

Figure 14: Linear fit with 95% confidence interval of Harmonic Overlap score for test audio mixtures vs. output SI-SDR achieved with ConvTasNet models trained with various proportions of randomised data.

difficult to separate. To test this, similar experiments to the ones described in Section 6.2.3 were conducted for DPTNet models trained with different balances of randomised data, and their performance for various test examples was compared w.r.t. each test example's harmonic overlap score. Figure 14 presents 6 such models trained with different balances of randomised and synchronised data and the Pearson correlation coefficient for Harmonic Overlap vs. Output SI-SDR is presented. While all models show a mild negative correlation between Output SI-SDR and the Harmonic Overlap score of the presented mixture, the correlation is not seen to be affected consistently across the models.

Figure 15 shows that most models trained with randomised data show a marginally more negative correlation between performance and harmonic

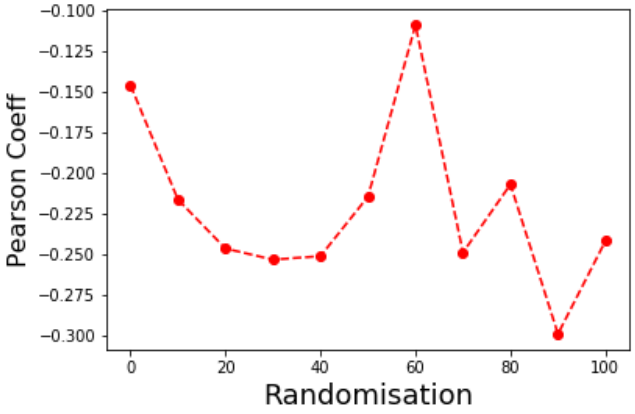

Figure 15: Pearson correlation coefficient of Harmonic Overlap score for test audio mixtures vs. output SI-SDR achieved with ConvTasNet models trained with various balances of randomised data. The X-axis value represents the percentage of total training data samples that were randomised.

overlap scores for models trained with more randomised data, with the exception of one model.

This may also explain why randomisation of mixtures during training affects the overall model performance as randomising mixtures significantly reduces the average harmonic overlap in the mixtures presented during training.

6.3.3 *Random Mixing vs. Dataset Size*

Experiments on the BCBQ datasets suggested that using randomised mixtures significantly deteriorates the performance of choral separation models which was reported in Sarkar, Benetos, and Sandler (2021). Subsequently Jeon et al. (2023) used randomised mixing of solo vocal performances to train vocal harmony separation models which reported comparable performance to the results reported in Sarkar, Benetos, and Sandler (2021). While the BCBQ dataset used in Sarkar, Benetos, and Sandler (2021) was only 90 minutes long, the solo singing dataset used in Jeon et al. (2023) was 400 hours long. Thus experiments were conducted to test the impact of randomisation

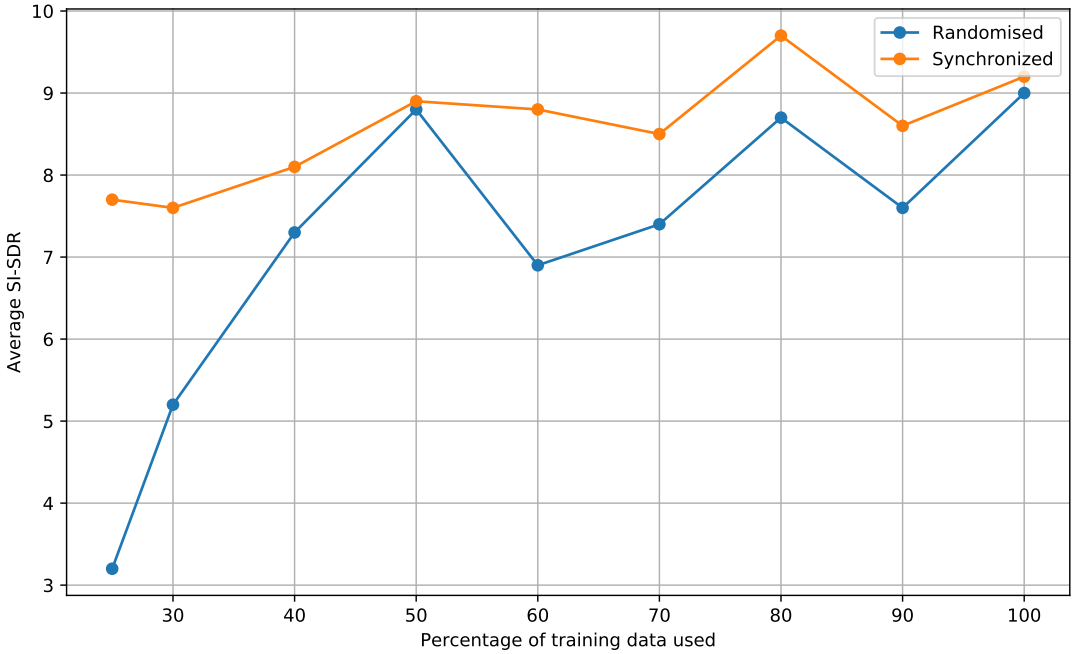

Figure 16: Average output SI-SDR achieved by DPT models trained with a reduced amount of data from EnsembleSet presented in a synchronised (musically coherent) and randomised (musically incoherent) fashion when tested on URMP Data.

as the amount and diversity of training data is scaled up. EnsembleSet was utilised to run experiments comparing models trained with randomised and synchronised audio data, where fractional amounts of the total dataset were used to train models to separate 2 source chamber ensemble mixtures.

Figure 16 shows the results of these experiments. It is observed that at larger dataset sizes we do not see a significant drop in performance between models trained on randomised vs. synchronised data. Meanwhile, a sharp relative performance drop is observed as the amount of training data is reduced. It is also notable that the models trained on randomised data never outperformed models trained on synchronised data.

However, the impact of increased training data size and diversity is much greater than using synchronised training data. Especially in the context of

ensemble music where finding clean stems is extremely challenging, using randomised mixtures of solo instrument recordings can be valuable.

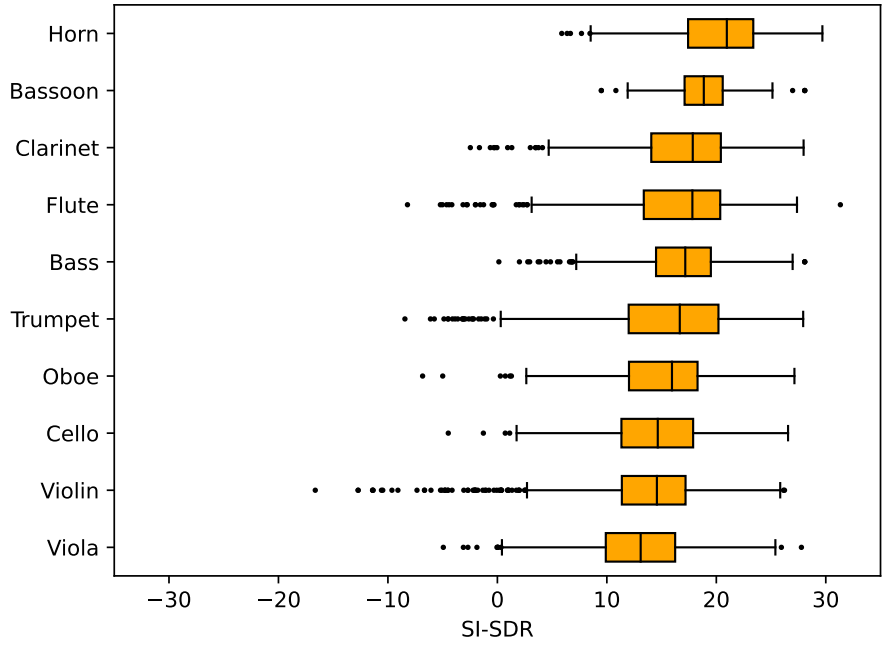

Figure 17: Instrument-wise median output SI-SDR of DPTNet trained on Ensemble-Set with fine-tuning using a single URMP string quartet example tested on 2 source mixtures from URMP dataset.

6.4 INSTRUMENT-AGNOSTIC PERFORMANCE

As mentioned in Section 4.2, models trained using PIT are trained in an instrument-agnostic fashion, where the model is expected to separate any given mixture regardless of the instrument types present in the mixture. While the models are trained in an instrument-agnostic fashion, their performance may be affected by instrument or timbre especially given that cross-dataset evaluation often results in performance drop. This cross-dataset performance drop may be attributed to either altered recording conditions or unseen instruments/timbres. In this section, an instrument-wise performance analysis of models trained in Section 5.5 is presented.

EnsembleSet contains 24 hours of strings and 1 hour of all other instruments (see Section 3.6), in Figure 17 we do not see any correlation between instrument training data size and instrument separation performance. In fact, we see that rarer instruments in EnsembleSet on average perform better than the most dominant instruments which are violin, viola, and cello. To test the proposed model's timbre-agnostic behaviour, a DPTNet model (training details described in Section 5.4) is trained with multi-render augmentation, excluding French horn training examples (see Figure 18). The average SI-SDR on URMP data was +8.8 dB, slightly lower (-0.49 dB) than the baseline. However, we found similar instrument-wise SI-SDR performance across different instruments, including the French Horn, which is an unseen instrument for the model. This suggests that models trained with PIT can separate unseen instruments.

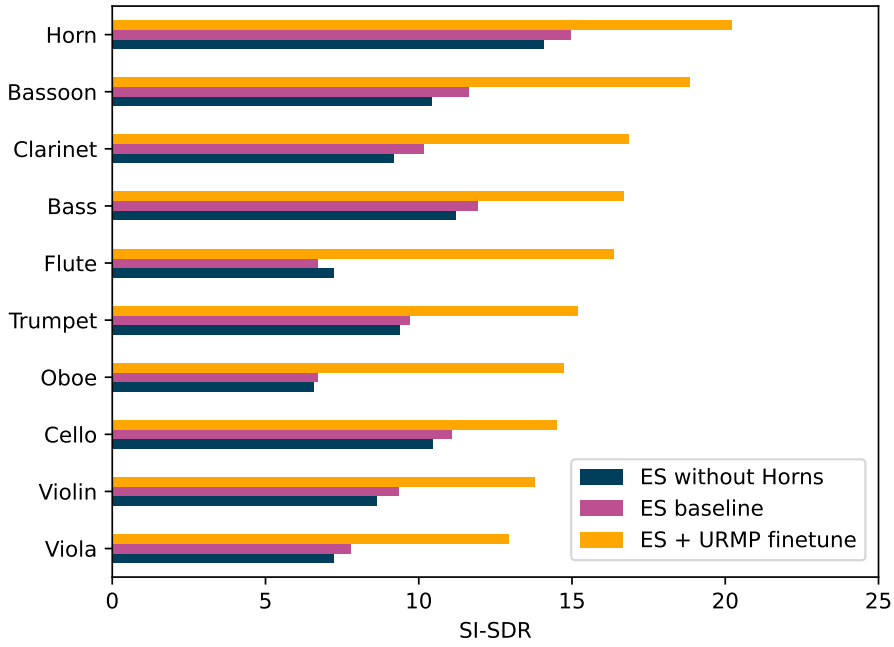

Figure 18: Average performance of DPTNet models tested on 2 source URMP mixtures.

The scores presented in Figure 18 only show average performance for each instrument class, without any information on the interferer instrument type. It could be possible that the separation performance of the unseen instrument

is influenced by the interferer being a well-known instrument. To investigate this, Figure 19 presents instrument pair-wise average SI-SDR, which shows that unseen/rare instruments performing well are unrelated to the interferer instrument being a well-known instrument.

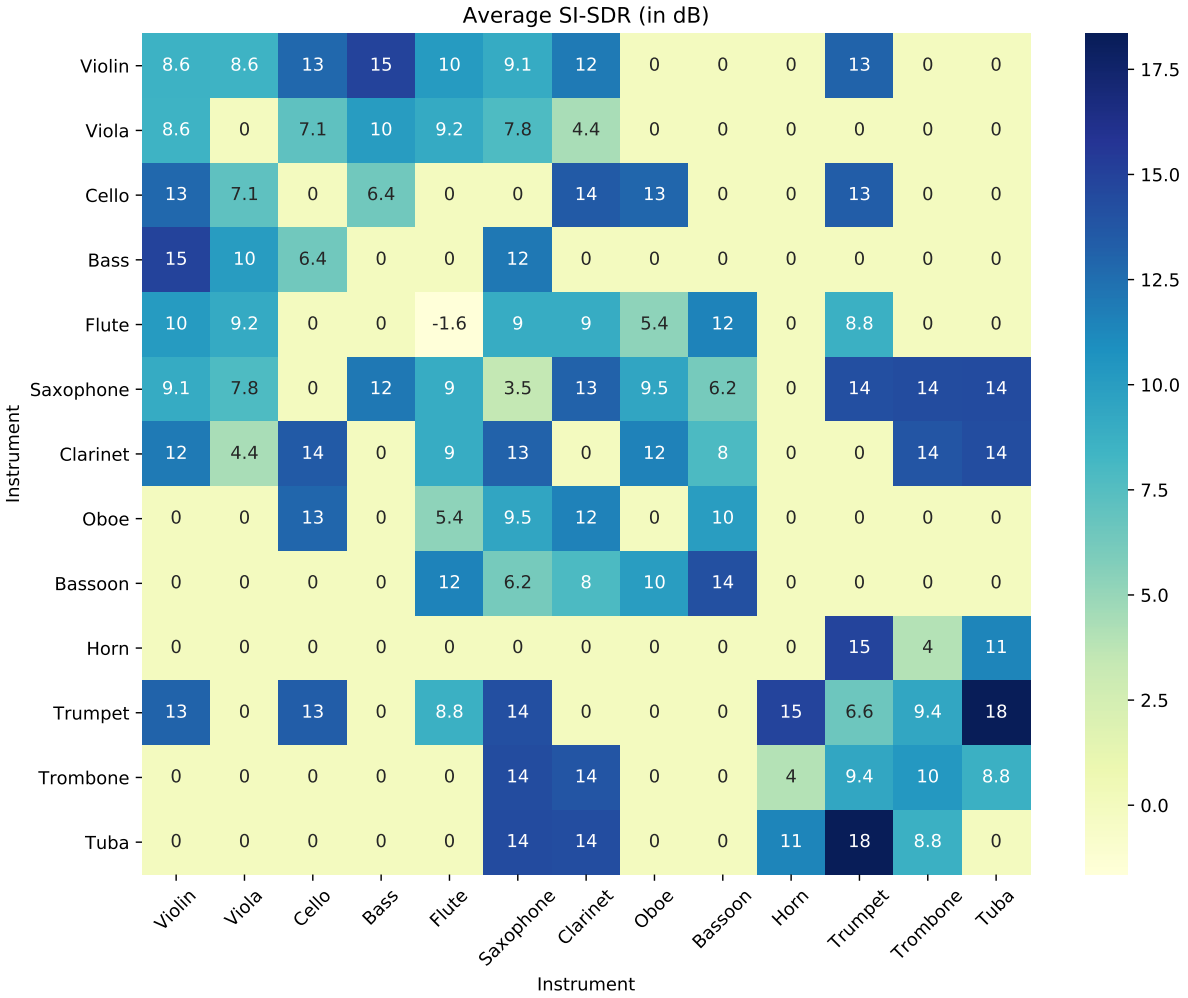

Figure 19: Instrument pairwise average output SI-SDR for 2 source DPTNet based ensemble separation model trained on EnsembleSet and evaluated on URMP.

Instrument pair-wise performance analysis also shows that instruments with similar registers typically perform poorer than instruments with distinct registers. The saxophone is an unseen instrument to the model both during training and finetuning and performs significantly better when presented as a mixture with a different instrument, but performs quite poorly

for monotimbral mixtures of the saxophone. While this could be interpreted as mixtures of the same instruments may perform poorly, it must also be kept in mind that instrument timbres are not the only factor that may affect separation performance. It may also be related to the fact that monotimbral mixtures may have a higher likelihood of having melodies containing pitch overlaps and crossovers as compared to mixtures of instruments of different registers.

6.5 MUSICAL CONTEXT VS. SEPARATION PERFORMANCE

In this section, we explore how different musical scenarios impact the performance of ensemble separation models by analysing results from the experiments described in Section 5.5. Figure 20 and Figure 21 show the performance of DPTNet-based model with fine-tuning for different mixture types based on the musical context present in the mixture. In Figure 20 the URMP

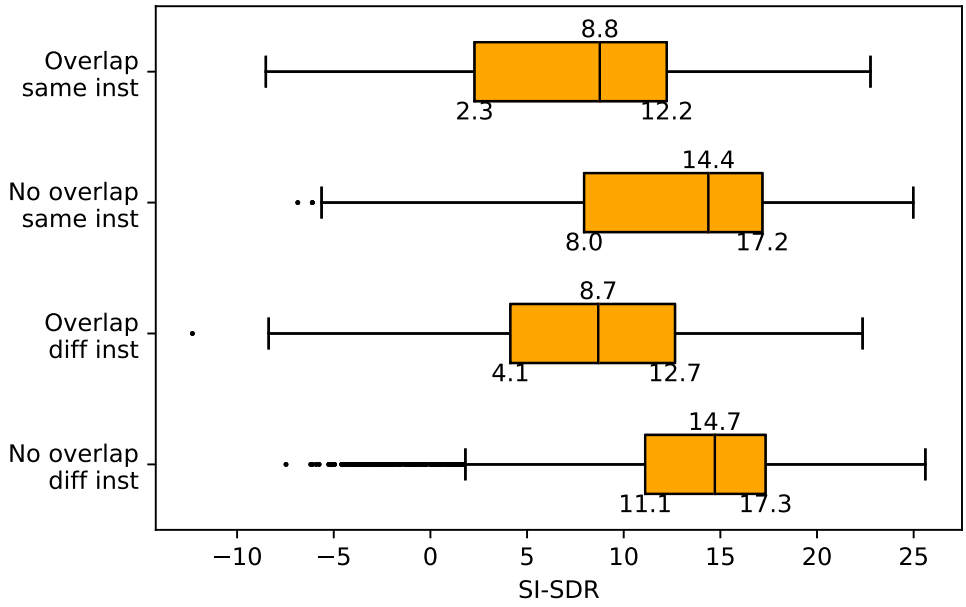

Figure 20: 2-source separation performance w.r.t. pitch overlap of DPTNet trained on EnsembleSet with fine-tuning on URMP.

test data is divided into 4 categories: same instruments vs. different instruments, mixtures with pitch overlaps, and without pitch overlaps. Pitch over-

laps are detected using pYIN (Mauch and Dixon, 2014) on each instrument's ground truth audio tracks. An example is classified to observe a pitch overlap if at any point during the mixture, the two sources are less than less than half a semitone apart.

It is observed that examples with pitch overlap perform significantly worse ($\approx$ 6 db) than examples without pitch overlaps across all examples. It is also found that mixtures of the same instruments and different instruments do not have a significant performance difference to obtain any insights. However, mixtures with no overlap and different instruments do contain a significantly higher number of outliers, arguably due to most examples belonging to this category and other factors contributing to separation performance that are not captured here. Similarly, in Figure 21 the URMP test data is divided

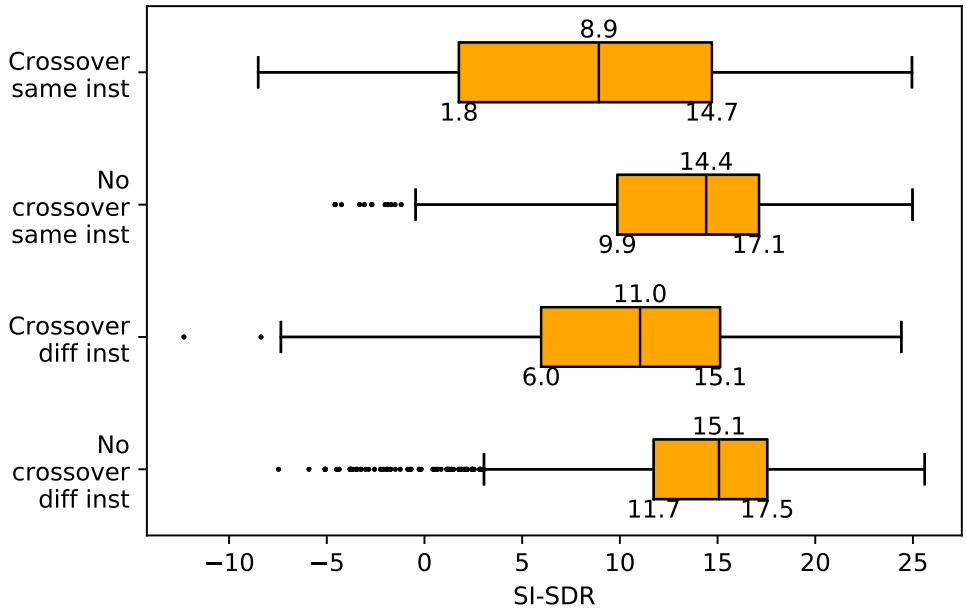

Figure 21: 2-source separation performance w.r.t. pitch crossovers of DPTNet trained on EnsembleSet with fine-tuning on URMP.

into 4 categories, but this time comparing pitch crossovers present in the mixture. To detect pitch crossovers, we use pYIN and then detect if one of the sources is either always higher or always lower than the other, and if not then such mixtures are classified as mixtures with crossovers. It is observed that examples with crossovers on average perform poorer than examples

without crossovers. It is also observed that the performance drop caused by crossovers is in fact larger in the case of mixtures with the same instruments ($\approx$ 5.5 dB) than mixtures with distinct instruments ($\approx$ 4.1 db).

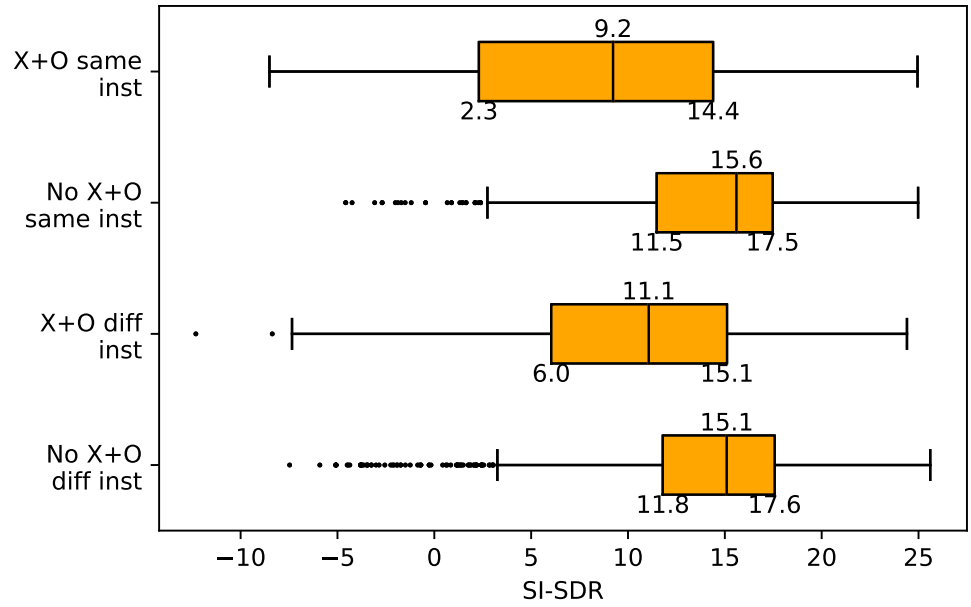

Figure 22: 2-source separation performance w.r.t. pitch crossovers and overlaps of DPTNet trained on EnsembleSet with fine-tuning on URMP.

Combining both these categories, Figure 22 shows the impact of these two musical contexts on separation performance. It is observed that crossovers and overlaps contribute to a 6.4 dB performance drop to mixtures of the same instruments as compared to a smaller drop of 4 dB when comparing mixtures of different instruments. This may suggest that timbral differences may still play a role in helping the model separate instruments, but evidently, the pitch content of the mixtures is far more impactful to the separation performance.

While pitch crossovers and pitch overlaps both significantly deteriorate the performance of the presented separation models, further exploration of how these specific scenarios affect the separation quality is explored in Section 6.6.

6.6 CASE-STUDIES

In order to explore the failure modes of the presented separation models, the worst-performing examples from the test data with less than 0 dB output SI-SDR were aggregated. Amongst them, the 10 most occurring instrument-pairs were filtered (as shown in Figure 29) and 36 of them were inspected at random. 17 of them were found to contain unisons, 12 of them had pitch crossovers, the remaining 7 did not exhibit any distinguishable characteristic. The two dominant classifications of unisons and pitch crossovers had consistent performance characteristics which are discussed in Section 6.6.1 and Section 6.6.2. The remaining examples did not exhibit any distinguishable characteristics. Of the remaining 7 examples 3 exhibited bleed while 4 had significantly loud artefacts although the cause for these failures could not be identified.

6.6.1 *Unison*

All the examples showing unison exhibit bleed present in both channels. Figure 23 shows a mixture of two violins playing in unison. While there are timbral and microtiming differences between the two violins, listening to this example reveals that the separation fails and both sources are audible in both estimates. While Figure 23 is an example of both sources playing in unison for the entire segment, Figure 24 is an example where the two sources play distinct note for the first half and play in unison for the second half. Listening to this example reveals that even though the first half of the estimates show some separation, the second half where both sources play in unison separation fails completely again. Moreover, the separated part of the excerpt has audibly higher artefacts, resulting in a SAR of 8.1 dB. This behaviour is consistently seen in the remaining examples of pitch overlaps with SARs ranging between 7-10 dB, where even momentary pitch overlaps

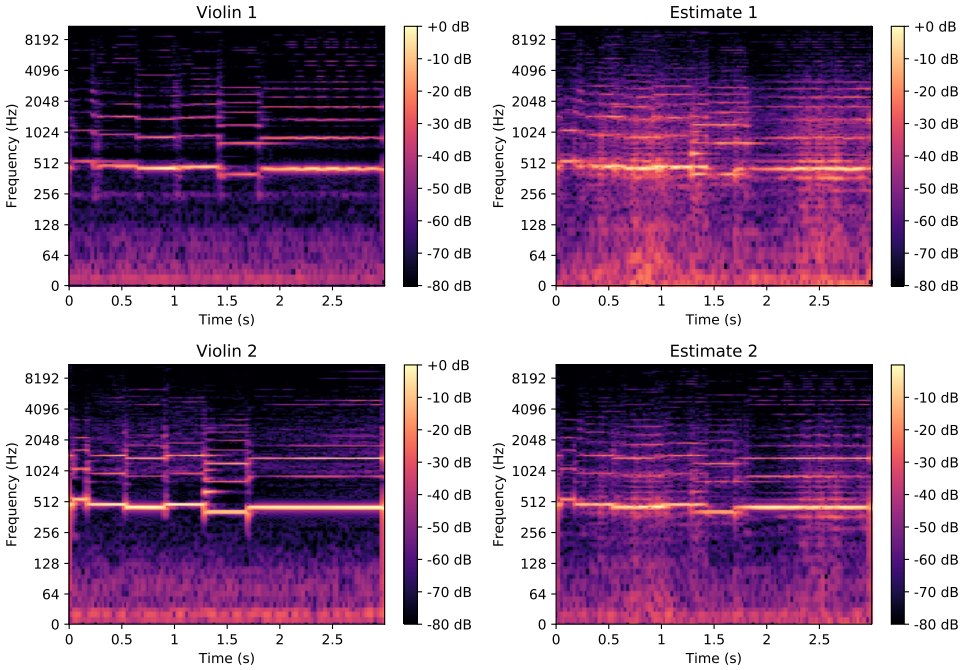

Figure 23: Example of complete unison between two Violins observed in URMP test data.

between the 2 sources seem to affect the separation quality of the entire audio segment.

6.6.2 *Source Confusion*

Analysis of examples of pitch crossovers reveals that the models are indeed able to successfully separate these sources, however, the sources are swapped across channels at the crossover point as seen in Figure 25. We define this phenomenon as *source confusion*, where the instruments within an output segment are swapped. Examining the SIR and SAR of these mixtures reveal that while the SIR of these examples are similarly low as observed in unisons ($\approx 0dB$), the SARs of the crossover examples are very high ($\approx +18dB$).

It is also observed that one of the output channels consistently outputs the higher melody and the other outputs the lower melody. This is also observed

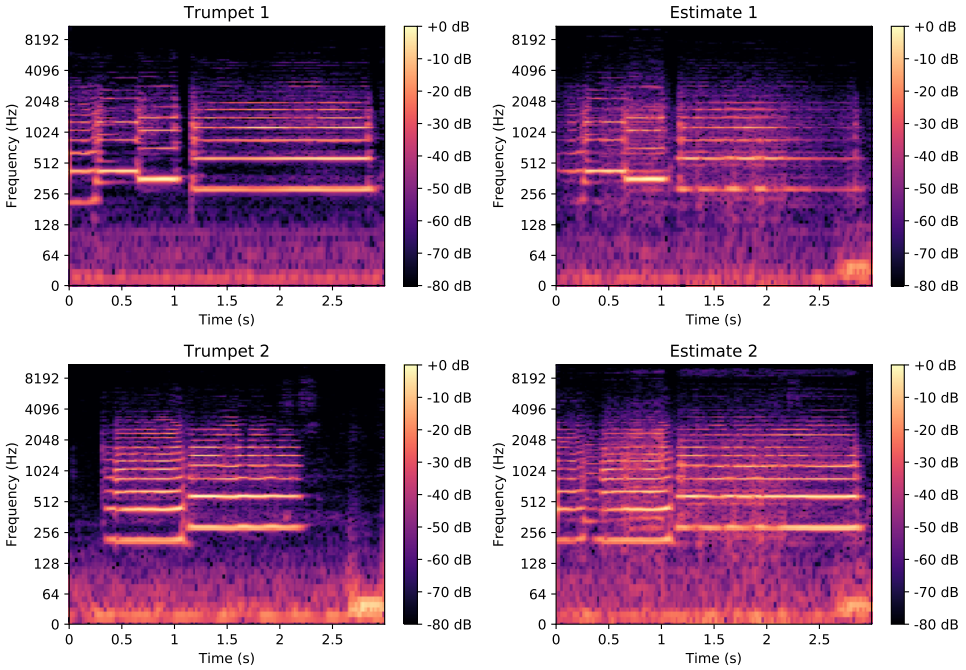

Figure 24: Example of partial unison observed between two Trumpets in URMP test data.

as a general trend across all examples, where the lower register instruments are typically present in the second channel in our experiments. It is however unclear whether these channel swaps also occur in the case of pitch crossovers between different instruments.

6.7 PERFORMANCE INSIGHTS

Table 15 presents a comparative analysis of average SI-SDR, SIR and SAR values for scenarios of pitch crossovers and pitch crossovers for mixtures of same instruments and different instruments. While typically both crossovers and overlaps negatively affect separation performance across all mixtures, the relative difference of impact of these scenarios on monotimbral vs. polytimbral mixtures provides some insights.

Metric	Inst	O-lap	Δ	No O-lap	Cross	Δ	No Cross
SI-SDR	Same	8.8 dB	-5.6 dB	14.4 dB	8.9 dB	-5.5 dB	14.4 dB
	Δ	-0.1 dB	-0.4 dB	0.3 dB	2.1 dB	1.4 dB	0.7 dB
	Diff	8.7 dB	-6 dB	14.7 dB	11 dB	-4.1 dB	15.1 dB
SAR	Same	15.1 dB	-3.3 dB	18.4 dB	14.8 dB	-4 dB	18.8 dB
	Δ	-0.8 dB	-1 dB	0.2 dB	1.3 dB	1.4 dB	-0.1 dB
	Diff	14.3 dB	-4.3 dB	18.6 dB	16.1 dB	-2.6 dB	18.7 dB
SIR	Same	18.5 dB	-2.1 dB	20.6 dB	15.4 dB	-7.3 dB	22.7 dB
	Δ	-3.3 dB	-4.2 dB	0.9 dB	2.4 dB	3.2 dB	-0.8 dB
	Diff	15.2 dB	-6.3 dB	21.5 dB	17.8 dB	-4.1 dB	21.9 dB

Table 15: Comparative study of average SI-SDR, SAR and SIR values for mixtures of same and different instruments in different musical contexts across all test examples from URMP dataset.

6.7.1 *Unison*

Pitch overlaps have a greater impact on all metrics for polytimbral mixtures as compared to monotimbral mixtures. Although the impact on SIR is more profound than the impact on SAR. This behaviour is observed in some case studies as unisons of different instruments result in the unison of both sources is considered as a single source and the higher partials of the brighter source are attempted to be separated as the second source.

It is understandable how the separation of a unison of distinct timbres may be more challenging as this appears to be a timbre decomposition problem in the case of polytimbral mixtures. In the case of monotimbral mixtures playing in unison, it is an ill-posed problem as effectively the model is expected to perform as a chorus removal tool, which in the case studies is observed to result in the model not being able to separate the sources at all. This observation is well aligned with the relative performance difference between unison and duet vocal separation reported by Jeon et al. (2023).

6.7.2 *Source Confusion*

Pitch crossovers have a greater impact on all metrics for monotimbral mixtures as compared to polytimbral mixtures. This may suggest that some polytimbral mixtures with pitch crossovers may not suffer from source confusion, while examples of pitch crossovers for monotimbral mixtures are more likely to result in source confusion.

This poses a fundamental question of whether this kind of error can be classified as a false negative. In the case of mixtures of identical timbres, the channel swap between output sources when pitches crossover may be seen as a valid separation result. Whereas in the case of crossovers with distinct timbres, if the instrument present in the separated output channel changes within a segment it should be seen as an incorrect separation result. Moreover, the bias observed towards output channels favouring lower or higher melodies poses a problem of instrument swap across segments within a larger recording.

6.8 DISCUSSION

In this chapter, an in-depth analysis of the performance of PIT-based models for ensemble separation was presented. The presented analysis methods exhibit that the performance of the model is independent of the type of instrument present in the mixture regardless of the training data. However, the strongest factor determining separation performance is the musical context of the mixture.

The Harmonic Overlap score presented a moderate negative correlation with the performance of the model. However, the correlation between the harmonic overlap and performance was not observed to be very strong. Other factors such as average pitch distance also presented similar weak correla-

tions (presented in Section A.1) with respect to separation performance. One explanation for these observations could be that these metrics might only be a part of the picture, amongst a larger set of conditions that determine the separation difficulty of a mixture/capability of the presented models to separate a mixture.

While interval relationships between the pitch content of the sources did not provide significant insights, analysing the mixtures in the context of pitch overlaps and crossovers presented clear correlations. It was found that both pitch overlaps and crossovers significantly affect the separation performance of the model, however, the manner in which they affect performance was found to be very different. In the case of pitch overlaps, it was found that unisons are inherently an ill-posed separation problem thus the poor separation performance observed is understandable and expected. However, it was found that pitch crossovers do not necessarily result in poor separation performance. Figure 26 presents an example of a polytimbral mixture with a large pitch discontinuity for one of the sources including a crossover with respect to the other source. In this example, the model is able to maintain source consistency and source confusion is not observed. This implies that the model may not suffer from source confusion in scenarios where the sources do not suffer from label ambiguity. It also shows that the model can maintain source consistency across large jumps in pitch trajectories.

This raises fundamental questions of what is the performance expectation of a source separation model. In the case of unisons, a timbre decomposition problem is a non-defined problem, as a given source timbre may be decomposed in multiple non-deterministic ways. Meanwhile, in the case of source confusion, while polytimbral mixtures suffering from source confusion could indeed be a problem, source confusion in monotimbral mixtures is a case where the metrics may diverge from the perceptual quality of separation. As in the case of monotimbral mixtures with pitch crossovers, a human may not

be able to identify which melody is expected from each source, especially without a larger musical context.

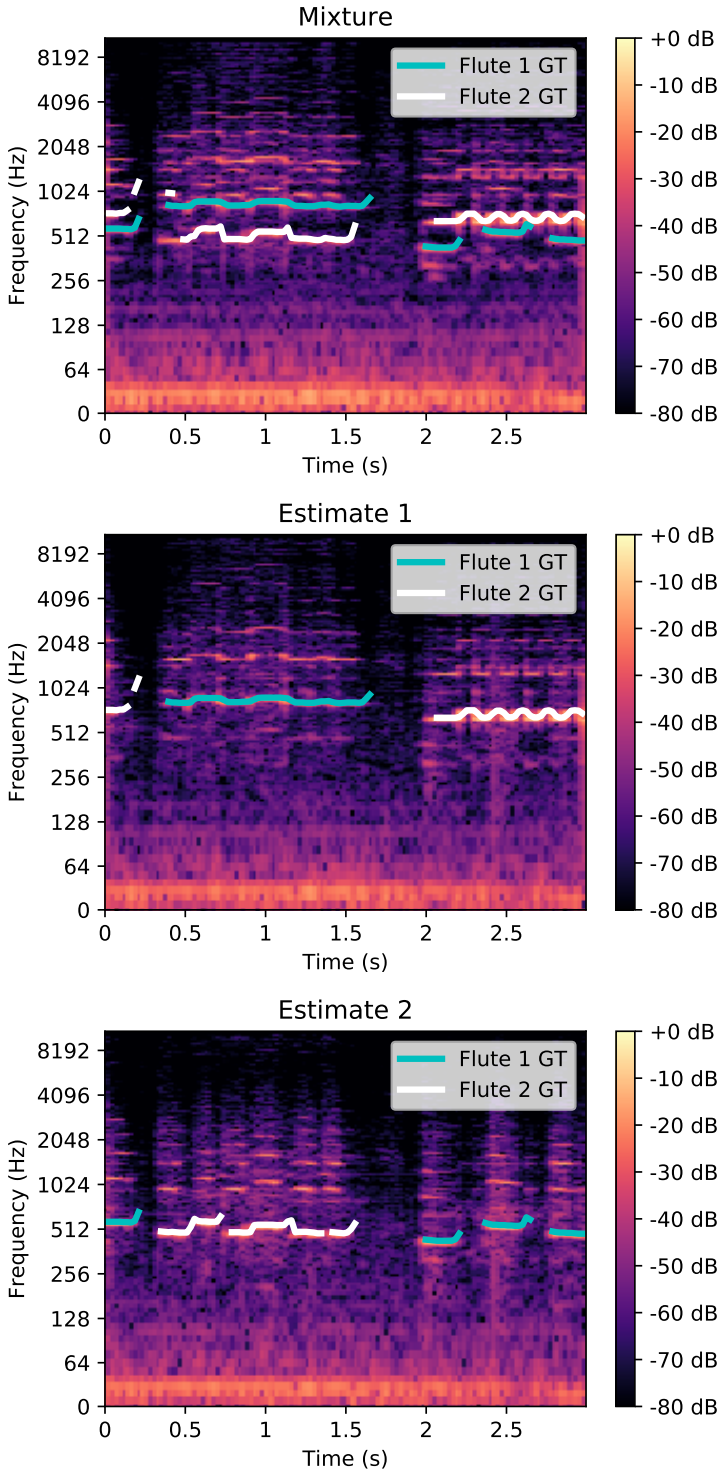

Figure 25: Handpicked example for pitch crossover observed in URMP test data. It can be observed that the model is able to separate the two sources, except that the separated sources are swapped across channels at sections with pitch crossovers preceded by silence.

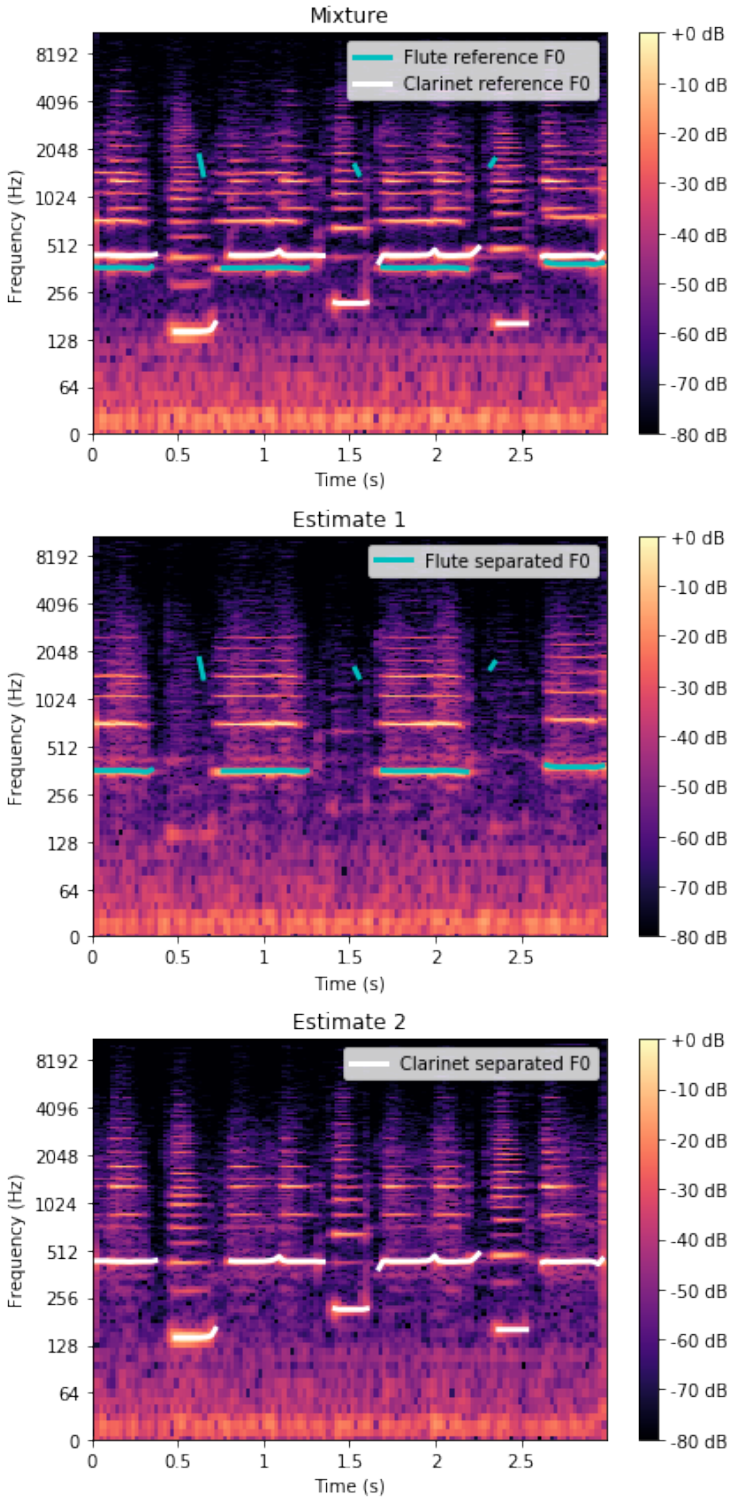

Figure 26: Handpicked example for a polytimbral pitch crossover observed in URMP test data. It can be observed that the model can separate the two sources effectively regardless of their pitches crossing over and the sources having pitch jumps up to 17 semitones.

Part IV

THE FUTURE IS EXCITING

Source separation is not a solved problem.

7

CONCLUSIONS AND PERSPECTIVES

7.1 SUMMARY

This thesis proposed a novel approach to music source separation using time-domain deep learning models with permutation invariant training for monophonic sources. While this method had proven successful for speech separation (Luo and Mesgarani, 2019), its applicability to music was unexplored prior to this work. To explore the applicability of permutation invariant training for musical mixtures, a bleed-free high-quality dataset of chamber ensemble instruments *EnsembleSet* was released. Using this dataset, it is shown that time-domain deep learning models trained on musical mixtures of monophonic sources with PIT are capable of separating musical sources in a class-agnostic fashion. This solved two main unsolved problems of class-based separation methods. Firstly, it enables the separation of similar/identical sources in a mixture without relying on timbral differences between sources. Secondly, it also enables the separation of rare/unseen instruments in a mixture which the model might not have been trained on. The majority of the work presented in this thesis has been presented in international peer-reviewed conferences as shown in Section 1.4. In this chapter, the main contributions of this thesis are summarised and perspectives on future work are presented.

7.1.1 *Impact of EnsembleSet*

In Chapter 3, a novel dataset of high-quality synthesised chamber ensemble music was presented. A significant drawback of synthesised audio datasets for training source separation models had been their poor generalisability to real-world test data. To overcome this challenge, high-quality digital music scores for chamber ensembles were collected and a data rendering pipeline that preserves articulation information from these digital scores was presented. This was used in conjunction with a high-quality sample library *Spitfire Audio BBC Symphony Orchestra Library* (BBCSO) with the capability of rendering a large number of these articulation modes for each instrument to generate realistic bleed-free renders of chamber ensemble instruments from classical musical pieces that are either public domain or under a creative commons license, enabling their use in research. This articulation-preserving audio rendering pipeline, in conjunction with the ability of BBCSO to render 20 unique microphone/mix configurations for each instrument enables the models trained in Section 5.4 and Section 5.5 to generalise well to real-world chamber ensemble mixtures from the TRIOS and URMP datasets (details in Section 2.11).

The presented dataset is now available on Zenodo and has had over 1000 views and has been downloaded 76 times at the time of this thesis. This dataset is currently also being used in the upcoming task of the Cadenza challenge (Dabike et al., 2023) to improve classical music listening experience for listeners with hearing loss.

7.1.2 *TasNets with PIT for Monotimbral Separation*

In Section 5.2 an alternative definition to the music separation task is presented that enables the use of permutation invariant training to separate mix-

tures of monotimbral musical sources in a class-agnostic fashion. Using this problem definition, Section 5.3 presents experiments using bleed-free choral singing data from the Bach Chorales and Barbershop Quartet datasets (details in Section 2.11.3) that perform class-agnostic separation of multi-singer mixtures using PIT. The optimisation process to adapt TasNet-based architectures to work at high sampling rates was presented in Chapter 4. Based on these optimisations, experiments in Section 5.3 show that class-agnostic separation of 4-part choral music reports the highest SDR scores reported in the prior art. While these results do exhibit the suitability of the proposed method, the experiments were trained and tested on a very small dataset and cross-dataset generalisability was yet to be achieved.

Using the dataset *EnsembleSet* presented in Chapter 3 as training data, models that can separate monotimbral mixtures of two instruments of the same instrument family (strings) were presented in Section 5.4. These experiments performed well when tested on unseen real-world datasets TRIOS and URMP. These experiments also showed that including instruments from other instrument families (brass and woodwind) did not deteriorate performance. It was also observed that models trained with PIT were in fact able to separate instruments from underrepresented and unseen instrument classes as well. This timbre-agnostic performance is further explored in Section 6.4, where it is found that the model does not show a performance correlation between the distribution of instruments in the training set and the test set. It was found that these models sometimes perform significantly better on unseen instruments.

While models trained on EnsembleSet did show cross-dataset generalisability, a performance drop was observed when these models were tested on the URMP dataset. In Section 5.5, experiments with models pre-trained on EnsembleSet and fine-tuned with target domain data are presented. It is shown that for same-domain tasks (chamber ensemble instruments), fine-tuning with a very limited amount of data (3min 45sec) shows significant

performance improvement (+5.5 dB SI-SDR improvement for URMP dataset). Domain adaptation for a cross-domain task (choral music separation) was also experimented with, where fine-tuning the pre-trained model using choral singing data showed an average of 17.82 dB SI-SDR performance (+1 dB higher than training on choral data alone).

7.1.3 *How do TasNets actually work?*

The experiments presented in Chapter 5 exhibit that TasNets trained with PIT are able to separate mixtures of monophonic musical sources in a class-agnostic fashion. In Section 6.7, further analysis of these models' performance for ensemble separation was explored. While the average performance metrics of these models were quite high (up to +18 dB average SI-SDR), the performance variation across different test examples was quite large. Analysis of this variation across instruments (shown in Section 6.4) did not show any correlation between the distribution of instrument timbres in training data and test data. Thus musical complexity (in Section 6.2) and context (in Section 6.5) is explored as a method to investigate the performance of these models.

A novel measure for quantifying the harmonic complexity of an input mixture is presented in Section 6.2. Analysing the results from experiments presented in Section 5.3 using this harmonic overlap score, it is found that these models have a moderate negative correlation with the harmonic overlap score of the input mixture. While a moderate negative correlation was observed, it was apparent that there were other factors that also affected the models' performance.

Subsequently, pitch overlaps, unisons and crossovers were investigated for their impact on the separation performance of models presented in Section 5.5. It was observed that ensemble mixtures with pitch overlaps showed

$\approx$ 6 dB performance drop as compared to mixtures without overlap. It was also observed that ensemble mixtures of distinct instruments and mixtures of identical instruments did not show significant performance differences. Further analysis revealed that mixtures with pitch crossovers and unisons perform 4-6 dB worse on average than the rest.

The worst performing examples were manually examined in Section 6.6. On inspecting these, $\approx$ 80% of the worst performing examples were found to contain pitch crossovers and unisons. Even though the output SI-SDR of both these cases was <0 dB, it is shown in Section 6.6.2 that examples with pitch crossovers show successful separation, however, the source-channel assignment is swapped at the crossover points. Meanwhile, examples with unisons result in bleed in the separated output which is shown in Section 6.6.1.

These results suggest that TasNet based models trained with PIT seem to separate sources based on pitch tracking. It is observed that the separation performance deteriorates significantly when the pitches of the individual sources overlap. Moreover, it was also observed that these models always present the higher-pitched source in one of the channels, whereas the other channel is always assigned to the lower-pitched source, regardless of source timbre. Analysing the performance distribution across the instruments (presented in Section 6.4) also shows that instruments with shorter attack times/onsets (brass instruments) seem to perform better than instruments with longer attack times/onsets (string instruments). All these observations combined suggest that TasNets trained with PIT may be learning to detect instrument onsets and then follow their pitch trajectories, and in fact, distinguish between instruments based on these characteristics.

7.2 LIMITATIONS AND OPPORTUNITIES

Experiments in Section 6.3 show that while training models on musically incoherent mixtures does deteriorate the performance of the presented models, in Section 6.3.3 it is seen that this effect diminishes and becomes negligible as the size of the training data is increased. This reduces the usefulness of EnsembleSet, as it is not crucial to have bleed-free stems of musically coherent instrument mixtures. However, it opens up the possibility of using solo performance data of monophonic sources (which is widely available) to generate random mixtures for training PIT-based models which makes the presented method more impactful. This was also utilised in the work by Jeon et al. (2023), where solo speech data was used to augment vocal harmony separation training data.

The experiments presented in Section 5.4 show that DPTNet models trained on EnsembleSet are capable of separating chamber ensemble mixtures from unseen real-world recordings. However, the performance variance of these models (as shown in Section 6.7) is very high. Many examples show performance <3 dB output SI-SDR, where perceptually the separation may be unsatisfactory. While the scenarios where these models fail have been explored, how to tackle these challenging scenarios has not been explored in this work.

It is observed in Section 6.6.2 that pitch crossovers between the sources in a mixture cause *source confusion*. In case of monotimbral mixtures, this phenomenon may not affect the perceived performance of a separation model, since the separation quality achieved is good. However, in the case of mixtures of sources with distinguishable timbres (especially in the case of vocal harmony mixtures), source confusion is not desirable. This may however be treated as a source tracking problem, which has been explored in the continuous speech separation (Chen et al., 2020).

The analysis presented in Section 6.6.1 shows that separation fails completely when the sources in the mixture perform in unison. Jeon et al. (2023) also reported that unison separation is significantly more challenging than the separation of harmonised vocals. This is still an unsolved problem. However, such a problem may even be considered as a unique problem altogether, as decomposing a mixture of sources performing in unison may also be considered as a timbre-disentanglement problem. While timbre-disentanglement should be considered as a source separation problem as well, however the expectation of what a model is expected to learn/achieve is very different. Moreover, timbre-disentanglement might even appear as a many-solution problem, as there is no ideal solution available for this task.

Perceptual evaluation through user studies was not undertaken in this study. Initially, perceptual studies were not considered in this research as aggregating a singular mean observation score from users and comparing it with SDR/SI-SDR metrics for the same model was deemed unlikely to yield meaningful insights when averaged across the extensive test set, comprising over 8500 examples. Additionally, the experiments exhibited substantial variance in objective metrics.

However, subsequent identification of failure modes in experiments (as illustrated in Section 6.6) revealed discrepancies between objective metrics and perceived separation quality, particularly in pitch crossover scenarios. Recognizing this discrepancy, it appears worthwhile to conduct a user study specifically focused on the perceptual quality of separation for these specific failure modes related to unisons and pitch crossovers. Regrettably, these findings emerged late in the Ph.D. timeline (August 2023) and are consequently deemed as prospective avenues for future research.

While Section 5.5 shows that the performance for real-world examples can be improved with fine-tuning, it is still not understood why models trained on EnsembleSet show such a significant drop in performance when tested on URMP, which was not observed when tested on mixtures from TRIOS. While

it may be due to the fact that the mixtures obtained from TRIOS consist of fewer examples of instruments with similar pitch ranges, the evidence is not conclusive. It may also be due to the synthesised nature of EnsembleSet, and the limited diversity of onsets observed in the synthesised dataset.

Although the impact of fine-tuning is reported in Section 5.5, it is not understood how exactly the model benefits from the fine-tuning. Analysis of the impact of fine-tuning on the encoder filterbanks presented in **??** did not provide any insights. This is especially difficult to interpret as the model was fine-tuned without freezing any weights, and since the 1-D convolutional encoder filterbank is followed by a fully connected layer, the differences observed in the filterbank after fine-tuning are hard to interpret. While the cross-domain performance of fine-tuning experiments for choral separation suggests that the model does learn features from EnsembleSet that improve choral separation, it is unclear what these features are.

Another major limitation of the proposed method of PIT based deep-learning solutions is that the models are trained for specific numbers of concurrent sources. It is a known phenomenon in speech separation that when the number of sources present in the mixture is lesser or more than the number of sources the model is trained for, the model performance significantly deteriorates (Luo and Mesgarani, 2020). While this is not reported in this thesis, our experiments also observed the same. However, this also suggests that existing solutions for speech separation of a variable number of sources described in Section 2.8.0.1 should also be useful for this task.

A significant advantage of the proposed method for using time-domain source separation is its ability to not only separate unseen instruments, but the potential separation quality achievable is very high (up to +25 dB SI-SDR) with a significantly smaller model capacity than the state-of-the-art music separation models such as Res-U-Net, Band-split RNN and RoPE transformer (described in Section 2.7). While class-based music separation models' performance seemed to be strongly correlated to both training data size

and model size, TasNet+PIT-based solutions may offer an alternative high-quality solution to specific music separation scenarios with much smaller model sizes.

7.3 FUTURE PERSPECTIVES

This thesis proposes an alternate formulation to the music source separation problem which is capable of solving a few of the challenges that are not solvable using class-based music source separation methods. While the presented method has significant limitations to its range of applicability, it may be used in conjunction with class-based source separation models to provide a full-scale music source separation solution.

A noteworthy observation from this work is the necessity for sources to be monophonic to achieve timbre-agnostic separation. This in conjunction with the impact of overlapping pitch trajectories of sources provides key insights into the strengths and weaknesses of permutation invariant training. Exploring the use of permutation invariant training for universal sound separation by explicitly requiring the sources in the training data to be monophonic may result in significant performance improvement.

Based on the experiments from Section 5.5, using any form of monophonic source data may enhance the performance of PIT-based separation models. While this had been tested by Jeon et al. (2023) for combining speech and vocal harmony data, the results from this work suggest that all forms of monophonic sources may be combined to potentially train a foundational model for universal separation of monophonic sources from mixtures that could be applicable to music, speech and other environmental sounds as well.

A key observation from the optimization experiments conducted on DPTNet pertains to the enhancement of performance through the manipulation

of filterbank lengths, input frame durations, and the augmentation of repeat units within the separation stack. Notably, the observed improvements are extrapolatable to other models such as Band-split RNN and RoPE transformer, where dual path processing has been integrated into the domain of music demixing.

An important insight has emerged concerning the reduction of the 1-D convolutional filterbank length (and stride length) in the input layer, resulting in a substantial increase in the sequence length of the latent representation. This effect, in turn, significantly influences the overall memory consumption during model training, particularly notable for transformer models compelled to compute pairwise weights for longer sequences.

The application of graph neural network-based separation stacks may serve to mitigate this challenge by constraining the number of pairwise computations executed for prolonged sequences. These have recently been used effectively for other tasks such audio tagging by Singh et al. (2023), which achieves state-of-the-art with a significantly lower number of learnable parameters. Consequently, if the computational demands of the separator stack can be reduced using graph neural networks without significant compromise in performance, it may provide enough computational headroom to reduce the input filterbank length and increase the input audio segment duration, thereby potentially enhancing the capabilities of these separation methods.

Part V

APPENDIX

A

APPENDIX A

In this appendix, a number of analysis experiments are included that are referred to in this thesis.

A.1 ALTERNATIVE HARMONIC OVERLAP HYPOTHESIS

In this section, an alternative to the proposed harmonic overlap score (described in Section 6.2) is presented and evaluated. In this analysis, the average pitch distance between the sources is calculated for an input audio segment.

Given a set of N sources $x_i(t)$ for $i \in \{1, 2, ..., N\}$, we utilise the pYIN pitch detection algorithm from Mauch and Dixon (2014) to estimate their pitches $F_i^0(t)$. The pitch distance based harmonic overlap score (HO_{PD}) between two given sources i_1 and i_2 is then calculated by:

$$HO_{PD} := \sum_{t=0}^{T} min\left(\frac{24 - |F_{i_1}^0(t) - F_{i_2}^0(t)|}{T}, 0\right) \tag{24}$$

Using this average pitch distance as a harmonic overlap measure, and calculating the correlation between the separation performance of a 4-source choral mixture, and its harmonic overlap score (based on pitch distance) is shown in Figure 27.

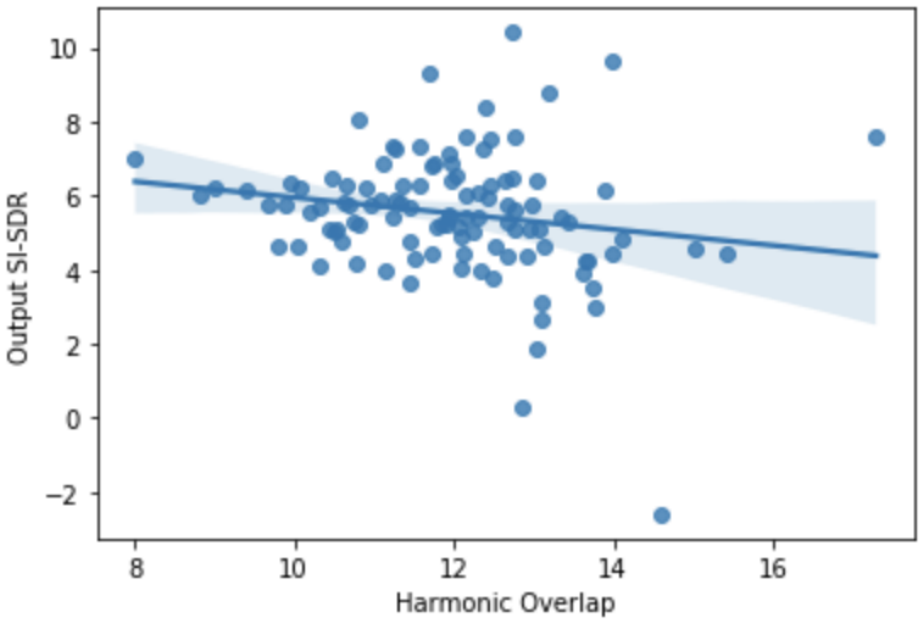

Figure 27: SI-SDR vs. Harmonic Overlap using pitch-distance based Harmonic Overlap measure. Pearson coeff: -0.224

While this measure did not account for the musical relationship of harmonics, the correlation observed w.r.t. to its impact on separation performance was comparable to the one observed with the presented harmonic overlap score in Section 6.2.3. This motivated further investigation into performance correlation based on pitch overlaps (instead of spectral/harmonic overlaps) which is presented in Section 6.5.

A.2 IMPACT OF FINETUNING

Analysis of the difference in performance caused by fine-tuning the models with target domain data is presented in Figure 28. The fine-tuning process as explained in Section 5.5 involved retraining all model weights at a very low learning rate using only 1 song from the URMP dataset. In this experiment, the song used from the URMP dataset for finetuning was a 4-minute track of a string quartet. The impact of this is evident in Figure 28 where it is

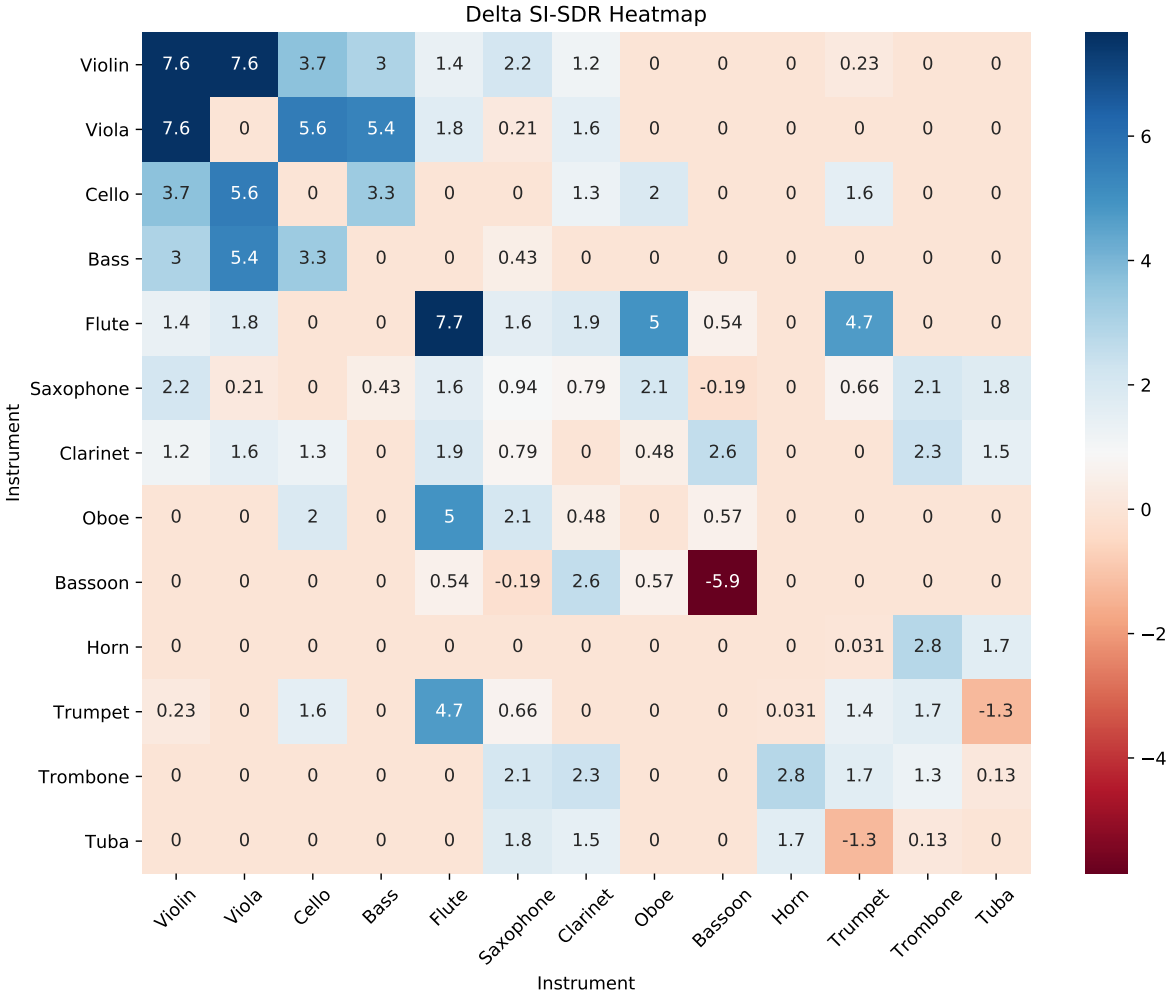

Figure 28: Instrument pairwise ΔSI-SDR for 2 source DPTNet based ensemble separation model trained on EnsembleSet after fine-tuning with a string quartet example from URMP.

observed that string instruments are the examples with the most significant improvement.

While most other instrument-pairs do not get negatively impacted by the fine-tuning process, two outliers are observed both of which are monotimbral. All scenarios with Flutes see a consistent performance improvement even though no flutes were presented in the fine-tuning data. This could be due to the similar pitch ranges present in the fine-tuning data (consisting of violins and viola predominantly) and the pitch range of Flutes. Meanwhile, monotimbral mixtures of Bassoon see a significant performance degradation.

These changes may be a result of either the model adapting to the timbral characteristics of the test dataset, or the model adapting to the temporal dynamics of the real performance in the URMP dataset. The latter may be the more realistic explanation as the training data EnsembleSet is completely synthesised and would typically result in highly synchronised note onsets and dynamics, while the URMP dataset was generated by musicians performing individually with a backing track. Further exploration of the impact of musical context vs. instrument timbres on the separation performance of these models is presented in Section 6.5.

A.3 DISTRIBUTION OF FAILURE CASES

The worst-performing examples from the test data with less than 5 dB output SI-SDR are aggregated. Amongst them, the 10 most occurring instrument pairs are filtered and shown in Figure 29. While only 3 of the 10 categories consist of monotimbral mixtures, 9 of them consist of instrument pairs which have significant overlap in their pitch ranges. Based on the observations presented in Section 6.6, it is found that these instrument pairs are likely suffering from pitch overlap related performance degradation. While timbral similarity between the sources cannot be ruled out as a potential cause for poor performance, the evidence suggests that the model's performance is more dependent on pitch trajectories of the constituent sources rather than the timbral similarities between the sources.

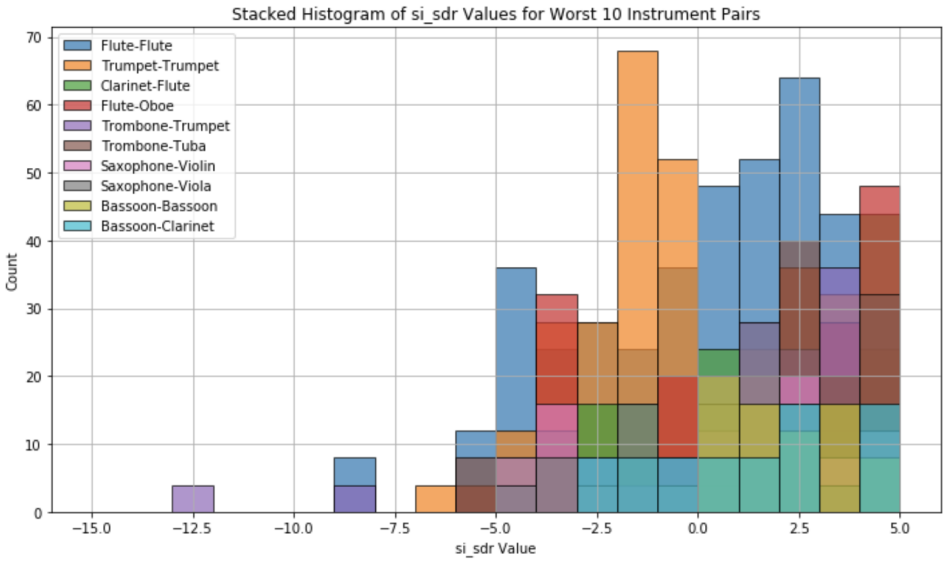

Figure 29: Distribution of top 10 categories of worst performing test cases

B

APPENDIX B

This is an excerpt from an interview with Jake Jackson, Recording Engineer for the Spitfire Audio BBC Symphony Orchestra Sample Library. The interview was conducted via Zoom on 14th October 2021. The interviewee consented to the interview being recorded and the information shared was permitted to be used for this research. Details from this interview were used to better understand the audio plugin, its recording conditions and design considerations which were crucial for the EnsembleSet dataset generation process described in Chapter 3.

B.1 INTERVIEW WITH JAKE JACKSON

Saurjya: Could you provide a brief introduction to how the BBCSO library is recorded and rendered and primarily how is it superior to like using a generic synth?

Jake: Essentially, the idea was to record it in a way that we would record a film score.

So therefore, we had all the microphones that you would use if you were recording this for a film score and for the people who would use it. But then also we decided to make it trying to future prove it as much as possible by making sure we had microphones that could be used for Atmos and but then also, you know, go take signals that were kind of luxuries really in terms of, you know, you might, it might use them for certain projects.

Basically wanted to have something that was basically an entire toolkit for one set of recording everything. So we had everything from the traditional like, to Decca tree recording, if you're doing it that way, outriggers, ambience, because we had the really far balcony mics that were right at the very back of the room, and we had the atmos mics, we had the side mics, and then we had the close mics so that you can have, you know, so you can change the perspective.

Basically the idea is that you can have whatever perspective of any instrument you wanted you could have basically. And so we had, you know, like, it's just anything I've ever kind of used in a recording session, we put it out.

And then it just made sense to make it available if you bought the professional one that you could have any combination of those. And then also a person called John Powell asked what would it be like if we had recorded all the microphones at once so that you could hear you know what the spill mics were so that if you were, because often when we do recordings, if you're recording just the first violins you record it, you know, like or in a recording when you hear the first violins playing by themselves they're still, they're single, still coming out of the microphones on the other side of the room.

So the idea was to have those as well. And because it became a kind of massive project we decided we'd just release it that way anyway. That's the idea because it was there and because somebody would find it useful and there was no point in trying to make it too light.

We did a, you know, it was a, it's a, it's a possibility where everything is possible. So you have every single thing we could ever imagine.

Saurjya: Thank you. That was quite useful. So one thing that I noticed, I'm not sure if I did not, like if it's, I'm hearing it wrong, but I did not notice any of the time delays or the phase differences across the difference mic when I was rendering it through multiple mics. So is that because like whatever

was the real time difference between the mics when it was recorded has been corrected during rendering?

Jake: No, there's no time correction at all. So with most of the kind of 25 years I've been doing this, most of the recordings we do we don't time align any microphones. They're all just as they are in the room. On occasion we've done this like I've worked with Deutsche Grammarphone and they go around and they do some time align. And we've done experiments where you time align and you listen and it sounds very different. You have a much more present sound if you time align all the microphones. But because you'd have to I think it would be actually virtually impossible to time align with all the spill mics but all the different sections.

So no absolutely not. No at any point. Nothing's time aligned. So it's all just as it is from. That makes it slightly awkward in terms of making something, when you do a sample, putting it in time, because you have to tell what is the first sample of time.

Is it the close micro? Is it the tree? That kind of thing, because obviously the speed of sound is relatively slow. But no, nothing's timeline. So it's all just literally as it goes. So the only difference would be where the start of the sample is compared to where the start of the violin sample is, compared to where the start of something recorded in the back of the room. So like a timpani, for example, there's quite a big difference in where the time it takes from hitting the tree

Because obviously if you're thinking about how an orchestra play, they're playing it so the conductor hears it at the same time. Which doesn't mean rather than, there's an inherent delay of when, if everyone played one staccato note at the same time, where they start is all different to reach the A point, a single point in the room.

If you all wanted them to hit the single point that was 10 meters behind the conductor at the same time, then they'd all have to start in different

places. Wouldn't they say everyone? So certain instruments anticipate the beat, which is fine when you're recording, but when you're sampling, where do you move that to?

Do you decide that's gonna be the start of the closed mic or is that the tree mic or the ambient mic? Do you see what I mean? So that's the, I don't know how they chose that. And I guess you can look at that by looking at a sample and seeing where they've decided that is.

And that's the only thing that makes it slightly more awkward, whether it's if you're playing with an orchestra. And obviously when you have 100 people playing a single note at the same time, they wouldn't all be exactly the same time anyway.

Saurjya: I was actually completely unaware of the fact that, from a performer's perspective, they would actually play the note considering what is the target, like do they want the note to sound at the same time for the conductor behind them?

Jake: Yeah, I mean, the reason I say that is because when we're recording, for like when we're doing things separately, we, the musicians talk about people playing ahead of the beat or on the beat, on the click, you know, like the brass players tend to play fractionately early because, because by the time it reaches microphones, they know that they have to make it sound like in a control room, it's in time, they have to play fractionately early so that so that it's in time for us because we don't really listen to close mics that much, we listen to the room mics.

BIBLIOGRAPHY

Adavanne, Sharath, Pasi Pertila, and Tuomas Virtanen (2017). "Sound event detection using spatial features and convolutional recurrent neural network." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 771–775. ISBN: 9781509041176. DOI: 10.1109/ICASSP.2017.7952260. arXiv: 1706.02291.

Araki, Shoko, Francesco Nesta, Emmanuel Vincent, Zbyněk Koldovskỳ, Guido Nolte, Andreas Ziehe, and Alexis Benichoux (2012). "The 2011 signal separation evaluation campaign (SiSEC2011):-audio source separation." In: *Latent Variable Analysis and Signal Separation: 10th International Conference, LVA/ICA 2012, Tel Aviv, Israel, March 12-15, 2012. Proceedings 10*. Springer, pp. 414–422.

Araki, Shoko, Alexey Ozerov, Vikrham Gowreesunker, Hiroshi Sawada, Fabian Theis, Guido Nolte, Dominik Lutter, and Ngoc Q K Duong (2010). "The 2010 Signal Separation Evaluation Campaign (SiSEC2010): Audio Source Separation." In: *Latent Variable Analysis and Signal Separation*. Ed. by Vincent Vigneron, Vicente Zarzoso, Eric Moreau, Rémi Gribonval, and Emmanuel Vincent. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 114–122. ISBN: 978-3-642-15995-4.

BS, ITUR (1770). "Algorithms to measure audio programme loudness and true-peak audio level." In: *International Telecommunication Union, Tech. Rep* 4, p. 2015.

Baevski, Alexei, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli (2020). "wav2vec 2.0: A framework for self-supervised learning of speech representations." In: *Advances in neural information processing systems* 33, pp. 12449–12460.

Barry, Dan, Robert Lawlor, and Eugene Coyle (2004). "Real-time sound source separation: Azimuth discrimination and resynthesis." In.

Bittner, Rachel M, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan Pablo Bello (2014). "Medleydb: A multitrack dataset for annotation-intensive mir research." In: *Proc. of the 14th International Society for Music Information Retrieval Conference (ISMIR)*. Vol. 14, pp. 155–160.

Cakir, Emre, Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, and Tuomas Virtanen (2017). "Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection." In: *IEEE/ACM Transactions on Audio Speech and Language Processing* 25.6, pp. 1291–1303. ISSN: 23299290. DOI: 10.1109/TASLP.2017.2690575. arXiv: 1702.06286.

Chandna, Pritish, Helena Cuesta, Darius Petermann, and Emilia Gómez (2022). "A Deep-Learning Based Framework for Source Separation, Analysis, and Synthesis of Choral Ensembles." In: *Frontiers in Signal Processing* 2. ISSN: 2673-8198. DOI: 10.3389/frsip.2022.808594. URL: https://www.frontiersin.org/article/10.3389/frsip.2022.808594.

Chen, Jingjing, Qirong Mao, and Dong Liu (2020). "Dual-Path Transformer Network: Direct Context-Aware Modeling for End-to-End Monaural Speech Separation." In: *Proc. Interspeech 2020*, pp. 2642–2646. DOI: 10.21437/Interspeech.2020-2205.

Chen, Zhuo, Takuya Yoshioka, Liang Lu, Tianyan Zhou, Zhong Meng, Yi Luo, Jian Wu, Xiong Xiao, and Jinyu Li (2020). "Continuous speech separation: Dataset and analysis." In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 7284–7288.

Chiu, Chung Cheng et al. (2018). "State-of-the-Art Speech Recognition with Sequence-to-Sequence Models." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Vol. 2018-April. Institute of Electrical and Electronics Engineers Inc., pp. 4774–4778. ISBN: 9781538646588. DOI: 10.1109/ICASSP.2018.8462105. arXiv: 1712.01769.

Choi, Hyeong-Seok, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyogu Lee (2019). "Phase-aware speech enhancement with deep complex u-net." In: *arXiv preprint arXiv:1903.03107*.

Choi, Seungjin, Andrzej Cichocki, Hyung-Min Park, and Soo-Young Lee (2005). "Blind source separation and independent component analysis: A review." In: *Neural Information Processing-Letters and Reviews* 6.1, pp. 1–57.

Choi, Woosung, Minseok Kim, Jaehwa Chung, and Soonyoung Jung (2021). "LaSAFT: Latent Source Attentive Frequency Transformation for Conditioned Source Separation." In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 171–175.

Cockos (2006). *Reaper: Digital Audio Workstation*. URL: https://www.reaper.fm/.

Cong, Fu'Ze, Shuchang Liu, Li Guo, and Geraint A. Wiggins (2018). "A Parallel Fusion Approach to Piano Music Transcription Based on Convolutional Neural Network." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Vol. 2018-April. Institute of Electrical and Electronics Engineers Inc., pp. 391–395. ISBN: 9781538646588. DOI: 10.1109/ICASSP.2018.8461794.

Cosentino, Joris, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent (2020). "Librimix: An open-source dataset for generalizable speech separation." In: *arXiv preprint arXiv:2005.11262*.

Cuesta, Helena, Emilia Gómez Gutiérrez, Agustín Martorell Domínguez, and Felipe Loáiciga (2018). "Analysis of intonation in unison choir singing." In: *Proc. of the 15th International Conference on Music Perception and Cognition (ICMPC)*.

Dabike, Gerardo Roa, Scott Bannister, Jennifer Firth, Simone Graetzer, Rebecca Vos, Michael A Akeroyd, Jon Barker, Trevor J Cox, Bruno Fazenda, Alinka Greasley, et al. (2023). "The First Cadenza Signal Processing Chal-

lenge: Improving Music for Those With a Hearing Loss." In: *arXiv preprint arXiv:2310.05799*.

Dai, Jiajie and Simon Dixon (2017). "Analysis of Interactive Intonation in Unaccompanied SATB Ensembles." In: *Proc. of the 18th International Society for Music Information Retrieval Conference (ISMIR)*.

Défossez, Alexandre (2021). "Hybrid spectrogram and waveform source separation." In: *arXiv preprint arXiv:2111.03600*.

Défossez, Alexandre, Nicolas Usunier, Léon Bottou, and Francis Bach (2019). "Music source separation in the waveform domain." In: *arXiv preprint arXiv:1911.13254*.

Durrieu, Jean-Louis, Bertrand David, and Gaël Richard (2011). "A musically motivated mid-level representation for pitch estimation and musical audio source separation." In: *IEEE Journal of Selected Topics in Signal Processing* 5.6, pp. 1180–1191.

Elizalde, Benjamin, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang (2023). "Clap learning audio concepts from natural language supervision." In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 1–5.

Emiya, Valentin, Emmanuel Vincent, Niklas Harlander, and Volker Hohmann (2010). "The PEASS Toolkit-Perceptual Evaluation methods for Audio Source Separation." In: *9th Int. Conf. on Latent Variable Analysis and Signal Separation*.

— (2011). "Subjective and objective quality assessment of audio source separation." In: *IEEE Transactions on Audio, Speech and Language Processing* 19.7, pp. 2046–2057. ISSN: 15587916. DOI: 10.1109/TASL.2011.2109381.

Fabbro, Giorgio et al. (2023). *The Sound Demixing Challenge 2023 x*2013 *Music Demixing Track*. arXiv: 2308.06979 [eess.AS].

Fan, Zhe-Cheng, Yen-Lin Lai, and Jyh-Shing R Jang (2018). "Svsgan: Singing voice separation via generative adversarial network." In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 726–730.

Fischinger, Timo, Klaus Frieler, and Jukka Louhivuori (2015). "Influence of virtual room acoustics on choir singing." In: *Psychomusicology: Music, Mind, and Brain* 25.3, p. 208.

Fitzgerald, Derry, Antoine Liutkus, and Roland Badeau (2016). "Projet—spatial audio separation using projections." In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 36–40.

Fritsch, Joachim (2012). *High Quality Musical Audio Source Separation*.

Fu, Szu Wei, Chien Feng Liao, and Yu Tsao (2020). "Learning with Learned Loss Function: Speech Enhancement with Quality-Net to Improve Perceptual Evaluation of Speech Quality." In: *IEEE Signal Processing Letters* 27, pp. 26–30. ISSN: 15582361. DOI: 10.1109/LSP.2019.2953810. arXiv: 1905.01898.

Garcia, Hugo Flores, Aldo Aguilar, Ethan Manilow, and Bryan Pardo (2021). "Leveraging Hierarchical Structures for Few-Shot Musical Instrument Recognition." In: *arXiv preprint arXiv:2107.07029*.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio (2014). "Generative adversarial nets." In: *Advances in neural information processing systems*, pp. 2672–2680.

Goto, Masataka, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka (2002). "RWC Music Database: Popular, Classical and Jazz Music Databases." In: *Proc. of the 2nd International Society for Music Information Retrieval Conference (ISMIR)*. Vol. 2, pp. 287–288.

Gover, Matan and Philippe Depalle (2019). "Score-informed source separation of choral music." PhD thesis. Ph. D. dissertation, McGill University.

Grais, Emad M, Gerard Roma, Andrew JR Simpson, and Mark D Plumbley (2016). "Single-channel audio source separation using deep neural network ensembles." In: *Audio Engineering Society Convention 140*. Audio Engineering Society.

Grais, Emad M., Hagen Wierstorf, Dominic Ward, Russell Mason, and Mark D. Plumbley (2019). "Referenceless performance evaluation of audio source

separation using deep neural networks." In: *European Signal Processing Conference*. Vol. 2019-September. European Signal Processing Conference, EUSIPCO. ISBN: 9789082797039. DOI: 10.23919/EUSIPCO.2019.8902932. arXiv: 1811.00454.

Graves, Alex, Abdel Rahman Mohamed, and Geoffrey Hinton (2013). "Speech recognition with deep recurrent neural networks." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 6645–6649. ISBN: 9781479903566. DOI: 10.1109/ICASSP.2013.6638947. arXiv: 1303.5778.

Griffin, D. and Jae Lim (1984). "Signal estimation from modified short-time Fourier transform." In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 32.2, pp. 236–243. DOI: 10.1109/TASSP.1984.1164317.

Heitkaemper, Jens, Darius Jakobeit, Christoph Boeddeker, Lukas Drude, and Reinhold Haeb-Umbach (2020). "Demystifying TasNet: A dissecting approach." In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 6359–6363.

Hennequin, Romain, Anis Khlif, Felix Voituret, and Manuel Moussalam (2019). "Spleeter: A fast and state-of-the art music source separation tool with pre-trained models." In: *Proc. International Society for Music Information Retrieval Conference*.

Hershey, John R, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe (2016). "Deep clustering: Discriminative embeddings for segmentation and separation." In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 31–35.

Hershey, Shawn et al. (2017). "CNN architectures for large-scale audio classification." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 131–135. ISBN: 9781509041176. DOI: 10.1109/ICASSP.2017.7952132. arXiv: 1609.09430.

Hu, Yanxin, Yun Liu, Shubo Lv, Mengtao Xing, Shimin Zhang, Yihui Fu, Jian Wu, Bihong Zhang, and Lei Xie (2020). "DCCRN: Deep complex

convolution recurrent network for phase-aware speech enhancement." In: *arXiv preprint arXiv:2008.00264*.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger (2017). "Densely connected convolutional networks." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Ihalainen, Kirsi (2008). "Methods of choir recording for an audio engineer." PhD thesis. Tampere Polytechnic.

Isola, Phillip, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros (2017). "Image-to-image translation with conditional adversarial networks." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134.

Jansson, Andreas, Eric Humphrey, Nicola Montecchio, Rachel Bittner, Aparna Kumar, and Tillman Weyde (2017). "Singing voice separation with deep U-Net convolutional networks." In.

Jeon, Chang-Bin, Hyeongi Moon, Keunwoo Choi, Ben Sangbae Chon, and Kyogu Lee (2023). "Medleyvox: An Evaluation Dataset for Multiple Singing Voices Separation." In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 1–5.

Jordal, Iver (2021). *torch-audiomentations: Audio data augmentation in PyTorch*. URL: https://github.com/asteroid-team/torch-audiomentations.

Kolbæk, Morten, Dong Yu, Zheng-Hua Tan, and Jesper Jensen (2017). "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.10, pp. 1901–1913.

Kong, Qiuqiang, Yin Cao, Haohe Liu, Keunwoo Choi, and Yuxuan Wang (2021). "Decoupling Magnitude and Phase Estimation with Deep ResUNet for Music Source Separation." In: *ISMIR*. Citeseer.

Le Roux, Jonathan, Gordon Wichern, Shinji Watanabe, Andy Sarroff, and John R Hershey (2019). "Phasebook and friends: Leveraging discrete representations for source separation." In: *IEEE Journal of Selected Topics in Signal Processing* 13.2, pp. 370–382.

Lee, Yuan-Shan, Chien-Yao Wang, Shu-Fan Wang, Jia-Ching Wang, and Chung-Hsien Wu (2017). "Fully complex deep neural network for phase-incorporating monaural source separation." In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 281–285. DOI: 10.1109/ICASSP.2017.7952162.

Li, Bochen, Xinzhao Liu, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma (2018). "Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications." In: *IEEE Transactions on Multimedia* 21.2, pp. 522–535.

LilyPond (2016). *python-ly: Python library containing various Python modules to parse, manipulate or create documents in LilyPond format*. URL: https://github.com/frescobaldi/python-ly.

Lin, Liwei, Qiuqiang Kong, Junyan Jiang, and Gus Xia (2021). "A Unified Model for Zero-shot Music Source Separation, Transcription and Synthesis." In: *Proceedings of 21st International Conference on Music Information Retrieval, ISMIR*.

Liu, Xubo, Qiuqiang Kong, Yan Zhao, Haohe Liu, Yi Yuan, Yuzhuo Liu, Rui Xia, Yuxuan Wang, Mark D Plumbley, and Wenwu Wang (2023). "Separate Anything You Describe." In: *arXiv preprint arXiv:2308.05037*.

Liutkus, Antoine and Roland Badeau (2015). "Generalized Wiener filtering with fractional power spectrograms." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Vol. 2015-August. Institute of Electrical and Electronics Engineers Inc., pp. 266–270. ISBN: 9781467369978. DOI: 10.1109/ICASSP.2015.7177973.

Liutkus, Antoine, Jonathan Pinel, Roland Badeau, Laurent Girin, and Gaël Richard (2012). "Informed source separation through spectrogram coding and data embedding." In: *Signal Processing* 92.8, pp. 1937–1949.

Liutkus, Antoine, Fabian Robert Stöter, Zafar Rafii, Daichi Kitamura, Bertrand Rivet, Nobutaka Ito, Nobutaka Ono, and Julie Fontecave (2017). "The 2016 signal separation evaluation campaign." In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and*

Lecture Notes in Bioinformatics). Vol. 10169 LNCS. Springer Verlag, pp. 323–332. ISBN: 9783319535463. DOI: 10.1007/978-3-319-53547-0_31.

Lu, Wei-Tsung, Ju-Chiang Wang, Qiuqiang Kong, and Yun-Ning Hung (2023). *Music Source Separation with Band-Split RoPE Transformer*. arXiv: 2309.02612 [cs.SD].

Luo, Yi, Zhuo Chen, and Nima Mesgarani (2018). "Speaker-independent speech separation with deep attractor network." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 26.4, pp. 787–796.

Luo, Yi, Zhuo Chen, Nima Mesgarani, and Takuya Yoshioka (2020). "End-to-end Microphone Permutation and Number Invariant Multi-channel Speech Separation." In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6394–6398. DOI: 10.1109/ICASSP40776.2020.9054177.

Luo, Yi, Zhuo Chen, and Takuya Yoshioka (2020). "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, pp. 46–50.

Luo, Yi and Nima Mesgarani (2018). "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, pp. 696–700.

— (2019). "Conv-TasNet: Surpassing Ideal Time-Frequency Magnitude Masking for Speech Separation." In: *IEEE/ACM Transactions on Audio Speech and Language Processing* 27.8, pp. 1256–1266. ISSN: 23299304. DOI: 10.1109/TASLP.2019.2915167. arXiv: 1809.07454.

— (2020). "Separating varying numbers of sources with auxiliary autoencoding loss." In: *arXiv preprint arXiv:2003.12326*.

Luo, Yi and Jianwei Yu (2023). "Music Source Separation With Band-Split RNN." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Manilow, Ethan, Gordon Wichern, Prem Seetharaman, and Jonathan Le Roux (2019). "Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity." In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. Vol. 2019-October. Institute of Electrical and Electronics Engineers Inc., pp. 45–49. ISBN: 9781728111230. DOI: 10.1109/WASPAA.2019.8937170. arXiv: 1909.08494.

Mathur, Akhil, Anton Isopoussu, Fahim Kawsar, Nadia Berthouze, and Nicholas D Lane (2019). "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*, pp. 169–180.

Mauch, Matthias and Simon Dixon (2014). "pYIN: A fundamental frequency estimator using probabilistic threshold distributions." In: *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 659–663.

Mitsufuji, Yuki, Giorgio Fabbro, Stefan Uhlich, and Fabian-Robert Stöter (2021). "Music demixing challenge at ISMIR 2021." In: *arXiv e-prints*, arXiv–2108.

Mitsufuji, Yuki, Giorgio Fabbro, Stefan Uhlich, Fabian-Robert Stöter, Alexandre Défossez, Minseok Kim, Woosung Choi, Chin-Yun Yu, and Kin-Wai Cheuk (2022). "Music demixing challenge 2021." In: *Frontiers in Signal Processing* 1, p. 18.

Nakamura, Tomohiko, Shinnosuke Takamichi, Naoko Tanji, Satoru Fukayama, and Hiroshi Saruwatari (2023). "jaCappella Corpus: A Japanese a Cappella Vocal Ensemble Corpus." In: *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. DOI: 10.1109/ICASSP49357.2023.10095569.

Ono, Nobutaka, Zbyněk Koldovskỳ, Shigeki Miyabe, and Nobutaka Ito (2013). "The 2013 signal separation evaluation campaign." In: *2013 IEEE International workshop on machine learning for signal processing (MLSP)*. IEEE, pp. 1–6.

Ono, Nobutaka, Zafar Rafii, Daichi Kitamura, Nobutaka Ito, and Antoine Liutkus (2015). "The 2015 Signal Separation Evaluation Campaign." In: *Latent Variable Analysis and Signal Separation*. Ed. by Emmanuel Vincent, Arie Yeredor, Zbyněk Koldovský, and Petr Tichavský. Cham: Springer International Publishing, pp. 387–395. ISBN: 978-3-319-22482-4.

Ozerov, Alexey, Antoine Liutkus, Roland Badeau, Gaël Richard, and Senior Member (2013). "Coding-based informed source separation: Nonnegative tensor factorization approach." In: *IEEE Transactions on Audio, Speech and Language Processing* 21.8, pp. 1699–1712. DOI: 10.1109/TASL.2013.2260153Ãŕ. URL: https://hal.inria.fr/hal-00869603.

Ozerov, Alexey, Emmanuel Vincent, and Frédéric Bimbot (2012). "A general flexible framework for the handling of prior information in audio source separation." In: *IEEE Transactions on Audio, Speech and Language Processing* 20.4, pp. 1118–1133. ISSN: 15587916. DOI: 10.1109/TASL.2011.2172425.

Pariente, Manuel, Samuele Cornell, Joris Cosentino, Sunit Sivasankaran, Efthymios Tzinis, Jens Heitkaemper, Michel Olvera, Fabian-Robert Stöter, Mathieu Hu, Juan M Martín-Doñas, et al. (2020). "Asteroid: the PyTorch-based audio source separation toolkit for researchers." In: *arXiv preprint arXiv:2005.04132*.

Pascual, Santiago, Antonio Bonafonte, and Joan Serrà (2017). "SEGAN: Speech Enhancement Generative Adversarial Network." In: *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH* 2017-August, pp. 3642–3646. arXiv: 1703.09452. URL: http://arxiv.org/abs/1703.09452.

Pascual, Santiago, Antonio Bonafonte, and Joan Serra (2017). "SEGAN: Speech enhancement generative adversarial network." In: *arXiv preprint arXiv:1703.09452*.

Perez, Ethan, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville (2018). "Film: Visual reasoning with a general conditioning layer." In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

Petermann, Darius, Pritish Chandna, Helena Cuesta, Jordi Bonada, and Emilia Gomez (2020). "Deep learning based source separation applied to choir ensembles." In: *arXiv preprint arXiv:2008.07645*.

Petermann, Darius, Gordon Wichern, Aswin Subramanian, and Jonathan Le Roux (2023). "Hyperbolic Audio Source Separation." In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 1–5.

Praetzel, Eric (2000). *Mutopia Project: Free Sheet Music for Everyone*. URL: https://www.mutopiaproject.org/index.html.

Raffel, Colin (2016). "Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching." PhD thesis. Columbia University.

Rafii, Zafar, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner (2017). "MUSDB18-a corpus for music separation." In.

— (2019). *MUSDB18-HQ - an uncompressed version of MUSDB18*. DOI: 10.5281/zenodo.3338373. URL: https://doi.org/10.5281/zenodo.3338373.

Reiss, Joshua D. (2011). "Intelligent systems for mixing multichannel audio." In: *2011 17th International Conference on Digital Signal Processing (DSP)*, pp. 1–6. DOI: 10.1109/ICDSP.2011.6004988.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-net: Convolutional networks for biomedical image segmentation." In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 9351. Springer Verlag, pp. 234–241. ISBN: 9783319245737. DOI: 10.1007/978-3-319-24574-4_28. arXiv: 1505.04597.

Rosenzweig, Sebastian, Helena Cuesta, Christof Weiß, Frank Scherbaum, Emilia Gómez, and Meinard Müller (2020). "Dagstuhl ChoirSet: A multitrack dataset for MIR research on choral singing." In: *Transactions of the International Society for Music Information Retrieval* 3.1.

Roux, J. L., S. Wisdom, H. Erdogan, and J. R. Hershey (2019). "SDR – Half-baked or Well Done?" In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 626–630. DOI: 10.1109/ICASSP.2019.8683855.

Sapp, Craig Stuart (2005). "Online Database of Scores in the Humdrum File Format." In: *ISMIR*, pp. 664–665.

Sarkar, Saurjya, Emmanouil Benetos, and Mark Sandler (2021). "Vocal Harmony Separation Using Time-Domain Neural Networks." In: *Proc. Interspeech 2021*, pp. 3515–3519. DOI: 10.21437/Interspeech.2021-1531.

— (2022). "EnsembleSet: a new high quality synthesised dataset for chamber ensemble separation." In: *Proc. of the 23rd International Society for Music Information Retrieval Conference (ISMIR)*, pp. 625–632.

Schlüter, Jan and Sebastian Böck (2013). "Musical onset detection with convolutional neural networks." In: *6th international workshop on machine learning and music (MML), Prague, Czech Republic*.

Schramm, Rodrigo, Emmanouil Benetos, et al. (2017). "Automatic transcription of a cappella recordings from multiple singers." In: Audio Engineering Society.

Shewan, Robert (1979). "Voice classification: An examination of methodology." In: *The NATS Bulletin* 35.3, pp. 17–25.

Shi, Ziqiang, Huibin Lin, Liu Liu, Rujie Liu, Shoji Hayakawa, Shouji Harada, and Jiqing Han (2019). "Furcanet: An end-to-end deep gated convolutional, long short-term memory, deep neural networks for single channel speech separation." In: *arXiv preprint arXiv:1902.00651*.

Sigtia, Siddharth, Emmanouil Benetos, and Simon Dixon (2016). "An end-to-end neural network for polyphonic piano music transcription." In: *IEEE/ACM Transactions on Audio Speech and Language Processing* 24.5, pp. 927–939. ISSN: 23299290. DOI: 10.1109/TASLP.2016.2533858. arXiv: 1508.01774.

Singh, Shubhr, Christian J. Steinmetz, Emmanouil Benetos, Huy Phan, and Dan Stowell (2023). *ATGNN: Audio Tagging Graph Neural Network*. arXiv: 2311.01526 [cs.SD].

Smaragdis, Paris and Gautham J Mysore (2009). "Separation by "humming": User-guided sound extraction from monophonic mixtures." In: *2009 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, pp. 69–72.

Spiertz, Martin and Volker Gnann (2009). "Source-filter based clustering for monaural blind source separation." In: *Proceedings of the 12th International Conference on Digital Audio Effects*. Vol. 4, p. 6.

SpitfireAudio (2019). *User Manual BBC Symphony Orchestra Professional*. URL: https://d1t3zg51rvnesz.cloudfront.net/p/files/product-manuals/4126/1648649726/BBCSOPro_Manual_v2.0.pdf.

Steinberg (2016). *Dorico: Music Notation Software | Steinberg*. URL: https://www.steinberg.net/dorico/.

Stoller, Daniel, Sebastian Ewert, and Simon Dixon (2018a). "Adversarial semi-supervised audio source separation applied to singing voice extraction." In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 2391–2395.

— (2018b). "Wave-u-net: A multi-scale neural network for end-to-end audio source separation." In: *arXiv preprint arXiv:1806.03185*.

Stöter, Fabian-Robert, Antoine Liutkus, and Nobutaka Ito (2018). "The 2018 signal separation evaluation campaign." In: *International Conference on Latent Variable Analysis and Signal Separation*. Springer, pp. 293–305.

Stöter, Fabian-Robert, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji (2019). "Open-unmix-a reference implementation for music source separation." In.

Subakan, Cem, Mirco Ravanelli, Samuele Cornell, Mirko Bronzi, and Jianyuan Zhong (2021). "Attention is all you need in speech separation." In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 21–25.

Subakan, Cem, Mirco Ravanelli, Samuele Cornell, François Grondin, and Mirko Bronzi (2023). "Exploring Self-Attention Mechanisms for Speech Separation." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Takahashi, Naoya, Purvi Agrawal, Nabarun Goswami, and Yuki Mitsufuji (2018). "PhaseNet: Discretized Phase Modeling with Deep Neural Networks for Audio Source Separation." In: *Interspeech*, pp. 2713–2717.

Takahashi, Naoya and Yuki Mitsufuji (2017). "Multi-scale multi-band densenets for audio source separation." In: *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, pp. 21–25.

— (2020). "D3Net: Densely connected multidilated DenseNet for music source separation." In: *arXiv preprint arXiv:2010.01733*.

Takahashi, Naoya, Sudarsanam Parthasaarathy, Nabarun Goswami, and Yuki Mitsufuji (2019). "Recursive speech separation for unknown number of speakers." In: *arXiv preprint arXiv:1904.03065*.

Tan, Ke and DeLiang Wang (2019). "Complex Spectral Mapping with a Convolutional Recurrent Network for Monaural Speech Enhancement." In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6865–6869. DOI: 10.1109/ICASSP.2019.8682834.

Uhlich, Stefan, Franck Giron, and Yuki Mitsufuji (2015). "Deep neural network based instrument extraction from music." In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 2135–2139.

Uhlich, Stefan, Marcello Porcu, Franck Giron, Michael Enenkl, Thomas Kemp, Naoya Takahashi, and Yuki Mitsufuji (2017). "Improving music source separation based on deep neural networks through data augmentation and network blending." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Institute of Electrical and Electronics Engineers Inc., pp. 261–265. ISBN: 9781509041176. DOI: 10.1109/ICASSP.2017.7952158.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017). "Attention is all you need." In: *Advances in neural information processing systems* 30.

Vincent, Emmanuel, Shoko Araki, and Pau Bofill (2009). "The 2008 signal separation evaluation campaign: A community-based approach to large-scale evaluation." In: *International Conference on Independent Component Analysis and Signal Separation*. Springer, pp. 734–741.

Vincent, Emmanuel, Rémi Gribonval, and Cédric Févotte (2006). "Performance measurement in blind audio source separation." In: *IEEE transactions on audio, speech, and language processing* 14.4, pp. 1462–1469.

Vincent, Emmanuel, Rémi Gribonval, and Mark D Plumbley (2007). "Oracle estimators for the benchmarking of source separation algorithms." In: *Signal Processing* 87.8, pp. 1933–1950.

Vincent, Emmanuel, Tuomas Virtanen, and Sharon Gannot (2018). *Audio source separation and speech enhancement*. John Wiley & Sons.

Weng, Chao, Dong Yu, Michael L Seltzer, and Jasha Droppo (2015). "Deep neural networks for single-channel multi-talker speech recognition." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.10, pp. 1670–1679.

Weninger, Felix, Jonathan Le Roux, John R Hershey, and Shinji Watanabe (2014). "Discriminative NMF and its application to single-channel source separation." In: *Fifteenth Annual Conference of the International Speech Communication Association*.

Williamson, Donald S., Yuxuan Wang, and DeLiang Wang (2016). "Complex Ratio Masking for Monaural Speech Separation." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.3, pp. 483–492. DOI: 10.1109/TASLP.2015.2512042.

Wisdom, Scott, Efthymios Tzinis, Hakan Erdogan, Ron J Weiss, Kevin Wilson, and John R Hershey (2020). "Unsupervised sound separation using mixtures of mixtures." In: *arXiv preprint arXiv:2006.12701*.

Wu, Yu-Te, Berlin Chen, and Li Su (2020). "Multi-instrument automatic music transcription with self-attention-based instance segmentation." In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28, pp. 2796–2809.

Xu, Yong, Qiuqiang Kong, Wenwu Wang, and Mark D. Plumbley (2018). "Large-Scale Weakly Supervised Audio Classification Using Gated Convolutional Neural Network." In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. Vol. 2018-April. Institute of Electrical and Electronics Engineers Inc., pp. 121–125. ISBN: 9781538646588. DOI: 10.1109/ICASSP.2018.8461975. arXiv: 1710.00343.

Yilmaz, Ozgur, Scott Rickard, and Özgür Yılmaz (2004). "Blind Separation of Speech Mixtures via Time-Frequency Masking Off-The-Grid Low-Rank Matrix Recovery and Seismic Data Reconstruction View project Blind Separation of Speech Mixtures via Time-Frequency Masking." In: *IEEE Transactions on Signal Processing* 52.7. DOI: 10.1109/TSP.2004.828896. URL: http://alum.mit.edu/www/rickard/bss.html.

Yoshii, Kazuyoshi, Ryota Tomioka, Daichi Mochihashi, and Masataka Goto (2013). "Beyond NMF: Time-Domain Audio Source Separation without Phase Reconstruction." In: *ISMIR*, pp. 369–374.

Yu, Dong, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen (2017). "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, pp. 241–245.

Zeghidour, Neil and David Grangier (2020). "Wavesplit: End-to-end speech separation by speaker clustering." In: *arXiv preprint arXiv:2002.08933*.