Simulating realistic multiparty speech data

for the development of distant microphone ASR systems

Jack Aaron Deadman

Supervisor: Professor Jon Barker

Department of Computer Science
University of Sheffield

This dissertation is submitted for the degree of
Doctor of Philosophy

Speech and Hearing Research Group July 2023

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Jack Aaron Deadman
July 2023

Acknowledgements

First and foremost I would like to thank my supervisor Prof. Jon Barker for his unwavering support both academically and personally throughout my Ph.D. I extend my gratitude to my mum and sisters for their love and support during my Ph.D and far beyond. I would like to thank the EPSRC for providing the funding to make this project possible. I give thanks to Dr. Yoshi Gotoh and Prof. Haiping Lu for their insightful comments during my progress meetings. I would not have been able to complete this Ph.D without the support of my friends to keep me going. I would like to give special thanks to [redacted], [redacted], [redacted] and [redacted] for their love and support during this time. Finally, I would like to thank all members of the SpandH research group past and present for bringing many happy memories.

Abstract

Automatic speech recognition has become a ubiquitous technology integrated into our daily lives. However, the problem remains challenging when the speaker is far away from the microphone. In such scenarios, the speech is degraded both by reverberation and by the presence of additive noise. This situation is particularly challenging when there are competing speakers present (i.e. multi-party scenarios)

Acoustic scene simulation has been a major tool for training and developing distant microphone speech recognition systems, and is now being used to develop solutions for mult-party scenarios. It has been used both in training – as it allows cheap generation of limitless amounts of data – and for evaluation – because it can provide easy access to a ground truth (i.e. a noise-free target signal). However, whilst much work has been conducted to produce realistic artificial scene simulators, the signals produced from such simulators are only as good as the 'metadata' being used to define the setups, i.e., the data describing, for example, the number of speakers and their distribution relative to the microphones.

This thesis looks at how realistic metadata can be derived by analysing how speakers behave in real domestic environments. In particular, how to produce scenes that provide a realistic distribution for various factors that are known to influence the 'difficulty' of the scene, including the separation angle between speakers, the absolute and relative distances of speakers to microphones, and the pattern of temporal overlap of speech. Using an existing audio-visual multi-party conversational dataset, CHiME-5, each of these aspects has been studied in turn.

First, producing a realistic angular separation between speakers allows for algorithms which enhance signals based on the direction of arrival to be fairly evaluated, reducing the mismatch between real and simulated data. This was estimated using automatic people detection techniques in video recordings from CHiME-5. Results show that commonly used datasets of simulated signals do not follow a realistic distribution, and when a realistic distribution is enforced, a significant drop in performance is observed.

Second, by using multiple cameras it has been possible to estimate the 2-D positions of people inside each scene. This has allowed the estimation of realistic distributions for the absolute distance to the microphone and relative distance to the competing speaker. The

results show grouping behaviour among participants when located in a room and the impact this has on performance depends on the room size considered.

Finally, the amount of overlap and points in the mixture which contain overlap were explored using finite-state models. These models allowed for mixtures to be generated, which approached the overlap patterns observed in the real data. Features derived from these models were also shown to be a predictor of the difficulty of the mixture.

At each stage of the project, simulated datasets derived using the realistic metadata distributions have been compared to existing standard datasets that use naive or uninformed metadata distributions, and implications for speech recognition performance are observed and discussed. This work has demonstrated how unrealistic approaches can produce over-promising results, and can bias research towards techniques that might not work well in practice. Results will also be valuable in informing the design of future simulated datasets.

Table of contents

List of figures

List of tables

Nomenclature

Variables, Symbols and Operations

M	Number of microphones
J	Number of sources
P	Number of components in a mixture model
N	Number of filterbank components
L	Size of sliding analyse window
H	Hop length of sliding analyse window
i	Microphone index
j	Source index or the imaginary unit
t	Discrete timestep
k	Analysis frame index
n	Frequency bin index
p	Mixture component index
$\pi_n^{(p)}$	Mixture component weight for frequency n and component p
τ_i^θ	Time delay for wavefront to reach microphone i after reaching the reference position given a direction of arrival θ
d_i	Distance between microphone i and the reference position
$x_i[t]$	Single-channel signal at microphone i at timestep t
$\mathbf{x}[t]$	Multi-channel signal, where $\mathbf{x}[t] = \begin{bmatrix} x_1[t] & \dots & x_M[t] \end{bmatrix}^\top$

$c_{ji}[t]$	Clean source signal j at microphone i
$\mathbf{c}_j[t]$	Clean multi-channel source signals where, $\mathbf{c}_j[t] = \begin{bmatrix} c_{j1}[t] & \dots & c_{jM}[t] \end{bmatrix}^\top$
$\mathbf{C}[t]$	$J \times M$ matrix of multi-channel clean source signals where, $\mathbf{C}[t] = \begin{bmatrix} \mathbf{c}_1[t] & \dots & \mathbf{c}_J[t] \end{bmatrix}^\top$
$n_i[t]$	Noisey signal at microphone i
$\mathbf{n}[t]$	Multi-channel noisey signal where $\mathbf{n}[t] = \begin{bmatrix} n_1[t] & \dots & n_M[t] \end{bmatrix}^\top$
$\delta[t]$	Dirac delta function
$s_j[t]$	Clean signal at the source j
$r_{ij}[t]$	Room impulse response at microphone i from source j
$u_n[t]$	The n-th filterbank kernel used to transform from time-domain to frequency domain
$v_n[t]$	The n-th filterbank kernel used to transform from frequency domain to time-domain
$x_i[k,n]$	Frequency domain representation of signal at microphone i
$\mathbf{z}[k,n]$	Normalised frequency domain representation of multi-channel signal (directional statistics)
$\mathbf{x}[k,n]$	Frequency domain representation of multi-channel signal $\mathbf{x}[t] = \begin{bmatrix} x_1[k,n] & \dots & x_M[k,n] \end{bmatrix}^\top$
$\mathbf{X}$	A $K \times N$ matrix representation of $x[k,n]$
$\mathbf{d}_n$	Steering vector for frequency component n, where $\mathbf{w}[n] \in \mathbb{C}^M$
$\mathbf{w}_n$	Beamformer weight vector for frequency compoenent n, where $\mathbf{w}[n] \in \mathbb{C}^M$
$\mathbf{\Phi}_n$	$M \times M$ spatial covariance matrix for frequency bin n
$\mathbf{p}_j$	Vector representation of the j-th speaker's cartesian coordinate. Either $\begin{bmatrix} p_j^x & p_j^y \end{bmatrix}^\top$ or $\begin{bmatrix} p_j^x & p_j^y & p_j^z \end{bmatrix}^\top$ depending on the context

$\boldsymbol{\varphi}(\mathbf{s}_1, \mathbf{s}_2)$	Angular separation between speaker $\mathbf{s}_1$ and $\mathbf{s}_2$
$\boldsymbol{\theta}_i(\mathbf{s}_i)$	Azimuth angle of speaker position $\mathbf{s}_i$ from the perspective of the i-th microphone array.
$\mathbf{Y}$	$J \times T$ binary matrix representing speaker activity at each frame
$\boldsymbol{\phi}$	Feature representation of a turntaking model
$\mathscr{H}(\cdot)$	Hilbert transform
$H(\cdot)$	Linear time-invariant system
$\mathscr{M}(\cdot)$	Mask prediction function $\mathscr{M} : \mathbb{C}^{K \times N} \mapsto \mathbb{C}^{J \times K \times N}$
$\exp(x)$	Exponential function, $\exp(x) = \sum_{n=0}^{\infty} \frac{x^n}{n!}$
$\sum_{i=1}^{N} x_i$	Summation from $n = 1$ up to N i.e., $x_1 + x_2 + \cdots + x_N$
$\prod_{i=1}^{N} x_i$	Product from $n = 1$ up to N i.e., $x_1 \times x_2 \times \ldots \times x_N$
$\mathbb{E}[X]$	Expected value of the random variable X, $\mathbb{E}[X] = \sum_{i=0}^{N} x_i P(x_i)$
$a[t] \circledast b[t]$	The discrete convolution between signals $a[t]$ and $b[t]$
$\arg\max_x f(x)$	The value x which produces the maximum value for $f(x)$
$\arg\min_x f(x)$	The value x which produces the minimum value for $f(x)$
$\mathbf{A} \odot \mathbf{B}$	Element-wise multiplication of two matrices $\mathbf{A}$ and $\mathbf{B}$
$\mathbf{A}^\top$	Matrix transpose of matrix $\mathbf{A}$
$\mathbf{A}^H$	Hermitian transpose of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	Matrix inverse of $\mathbf{A}$
$\angle z$	The argument of $z \in \mathbb{C}$
$\|\mathbf{v}\|$	Vector l_2 norm of $\mathbf{v}$
$\|x\|$	Absolute value of x or set cadinality

Acronyms / Abbreviations

ASR	Automatic Speech Recognition

BRDF	Bidirectional Reflectance Distribution Function
cACGMM	complex Angular Central Gaussian Mixture Model
CNN	Convolutional Neural Network
CSD	Cross Spectral Density
DOA	Direction of Arrival
DRR	Direct-to-Reverberant Energy Ratio
DSR	Distant Speech Recognition
FIR	Finite Impulse Response
GEV	Generalized Eigenvalue
GMM	Gaussian Mixture Model
GSS	Guided Source Separation
HMM	Hidden Markov Models
IBM	Ideal Binary Mask
IPD	Inter-channel Phase Difference
ISTFT	Inverse Short-time Fourier Transform
LTI	Linear Time Invariant
MVDR	Minimum Variance Distortionless Response
PESQ	Perceptual Evaluation of Speech Quality
RIR	Room Impulse Response
SDK	Software Developer Kit
SDR	Signal-to-Distortion Ratio
SI-SDR	Scale-Invariant Signal-to-Distortion Ratio
SNR	Signal-to-Noise Ratio
SOTA	State of the Art

SRP-PHAT	Steered-Response Power Phase Transform
STFT	Short-time Fourier Transform
STOI	Short-Time Objective Intelligibility
TDNN-F	Factorised Time-delayed Neural Network
T-F	Time-Frequency
t-SNE	t-distributed Stochastic Neighbor Embedding
TS-VAD	Target Speaker Voice Activity Detection
VAD	Voice Activity Detection
WER	Word Error Rate
WPE	Weighted Prediction Error

Chapter 1

Intoduction

1.1 Motivation

Automatic speech recognition (ASR) is the task of taking an acoustic signal spoken by a person and producing a text transcript of the words uttered. ASR technology has become commonplace in our everyday lives in recent years. It has become customary to speak to our phones to set a reminder or to shout across the room to ask our smart speaker to play a song. Even though the technology has become "good enough" to be helpful, it still leaves users with many frustrations when the recogniser is incorrect. It may be surprising to hear that the performance of these recognisers has surpassed human-level performance on many recognition tasks (Bermuth et al., 2021; Zhang et al., 2020b).

However, in many everyday situations, this technology can perform poorly. Although there can be a lot of specific reasons for poor performance, many can be traced back to a mismatch between the data the systems have been trained on and the data that they encounter when they are later deployed.

A particular ASR problem known as distant speech recognition (DSR) has some unique characteristics making it more susceptible to this mismatch problem that is not often discussed in the literature. In DSR, a target speaker whose speech we would like to recognise is located far from the recording device, which can be several metres across a room. In this setup, there are often *interferers* (i.e., competing speakers and other sound sources) whose signals we may not want to recognise but instead want to filter out to prevent them from corrupting the speech we do want to recognise. Some of these competing sound sources may have well-defined locations, while others may be diffuse and arrive at the microphone from no clear direction. The task of effectively filtering out distracting sounds is something that *humans* can perform naturally when attending to a conversational partner at, for example, a dinner party, i.e., the "cocktail party effect" (Cherry and Bowles, 1960; Haykin and Chen,

2005). However, although humans perform this task seemingly effortlessly, it remains a surprisingly challenging signal processing problem.

In addition to the competing noise source, the DSR task is also made challenging by the effect of *reverberation*. Reverberation is caused by the acoustic signal having many paths to the recording device: there will generally be one 'direct path' but then many paths that result from one or more reflections off the walls and other surfaces in the environment. The reverberation can be split into two parts, *early* and *late*. The early part is composed of the energy arriving by paths that reach the microphone soon after the direct path. These early reflections are highly correlated with the original speech and can aid recognition. In contrast, the late part is uncorrelated and can be detrimental to recognition performance (Kinoshita et al., 2009).

Automatic methods for filtering the target speech can exploit the fact that speakers have a well-defined direction relative to the recording device (Xiao et al., 2016). The device used to capture the acoustic signal in practice tends to be an array of microphones as opposed to a single microphone. Having an array of microphones means that the time delays between the signal reaching the different microphones can be used to infer spatial information of the source location of the signals reaching the device (Knapp and Carter, 1976). So, a typical DSR system will consist of a pre-processing step that involves extracting the target speaker's speech from the noisy signal(s) reaching the recording device. In the multi-channel case, statistically optimal 'beamformers' are commonly used to enhance a target speaker direction whilst suppressing signals from other directions (Breed and Strauss, 2002; Ferguson, 1998). This extraction procedure can be achieved through separate tasks such as target-speaker extraction (Delcroix et al., 2020), speech enhancement (Chaudhari and Dhonde, 2015) and speech separation (Choi et al., 2005; Wang and Chen, 2018).

When building DSR systems with source separation front-ends, *simulation* is typically used to both train (Hershey et al., 2016; Luo and Mesgarani, 2019) and evaluate (Le Roux et al., 2019; Vincent et al., 2006) the performance. Simulation involves taking a clean, isolated speech signal and corrupting it such that it has the acoustic properties of a signal propagating through a room from the source (speaker) to the sink (recording device); this is known as *spatialising*. These spatialised signals can then be combined by simply summing the signals together, creating a mixture. The role of the DSR's front-end is then to map this noisy mixture back to a clean, isolated version of the speech signal that can be fed to the speech recognition stage.

Simulation is useful because it allows for the source separation front-end to be directly *evaluated*, i.e., we have access to the pre-mixed target speech signal that can be compared to the front-end's output. Without simulation, a proxy to the target speech signal would

be required, for example, a near-field microphone could be used. However, such signals still contain some undesired noise. Using simulated signals also allows the front-end to be directly *trained*, i.e., it can learn a noisy to clean speech mapping by using the clean signals as a target in a training procedure. Directly learning this mapping produces by far the best results in commonly used datasets of simulated signals (Luo and Mesgarani, 2019; Subakan et al., 2021). Finally, simulated data is also useful as it allows for arbitrary large datasets to be generated with various spatial properties. In contrast, when recording a real[1] DSR dataset, we are limited to the spatial properties of the original recording.

However, despite its advantages, simulation has a major potential drawback. In practice, training and evaluating systems on simulated data are often found to lead to poor performance when deploying systems on real signals (Haeb-Umbach et al., 2020). Instead, the state-of-the-art in practice relies on an unsupervised backend which by definition is not biased by any training data (Du et al., 2020a; Ito et al., 2016). The problems with simulation are due to the, often, large mismatch between the simulated training data and the eventual real deployment data. Understanding these sources of mismatch and understanding how to produce simulated DSR speech data that is better matched to real scenarios is therefore the key aim of this thesis.

1.2 Thesis overview

1.2.1 Thesis aims

This thesis aims to address some of the mismatches between the evaluation data used in DSR and the eventual deployment data of these recognisers. In particular, the thesis explores data-driven approaches to model the behaviour of real people at small social gatherings, e.g., spatial characteristics of speakers relative to interfering speakers and their turn-taking patterns. These models are used to create datasets that are more challenging and more realistic than those that are currently commonly used, and which, because they are guided by observed data, allow the construction of ASR and pre-processing components that will generalise better to real datasets.

This analysis is performed by exploring a large existing multi-modal dinner party dataset, i.e., CHiME-5 (Barker et al., 2018).

The dataset consists of unscripted, conversational speech that has been captured by microphone arrays with four channels and a high-definition camera integrated into the unit. Whilst the audio is fully-transcribed, the video data is entirely unlabelled. A project aim is to

[1] Real in terms of spatial properties, i.e., artificial signal datasets are not fake.

use the camera feed to infer the location statistics of speakers relative to the microphones. Further, the transcripts are analysed to understand the statistics of speaker turn-taking and the resulting speaker overlap in casual conversational settings. With models of speaker location and turn-taking, realistic DSR data can be simulated. We can then look in detail at how the properties of the simulation impact the evaluation of DSR recognition systems, comparing conclusions drawn from the more realistic data set with simpler existing ones.

1.2.2 Research questions

RQ1 To use spatial filtering to extract the speech of the desired speaker in the presence of an interfering speaker, multiple channels are used to exploit differences in signals reaching the microphones. The differences in the signals are a direct result of different angles of incidence with the microphone array for each of the sources. If two signals (target and interferer) are coming from similar directions the signals will be similar for each of the channels in the array. The difference between the two directions of arrivals is referred to as the angular separation. The amount of angular separation is a parameter that is not often discussed in the design of simulations or measured when reporting results. Therefore this thesis explores, **how well do simulated datasets represent the *angular separation* found in real data? And how does poorly representing real data affect ASR evaluation?**

RQ2 The signal-to-noise (SNR) ratio between the target source and noise (competing speaker plus background) governs a large part of the difficulty of a mixture for both speech separation and ASR. For multiparty speech data, the SNR is a result of the loudness of each of the initial signals and the distances the signals need to travel to reach the microphones in the array. This means the SNR is largely the result of the relative distance of sources i.e., the distance the desired speaker is away from the microphone *relative* to the interfering speaker. Therefore, this thesis explores, **how well do simulated datasets represent the *relative distance* found in real data? And how does poorly representing real data affect ASR evaluation?**

RQ3 To address **RQ1** and **RQ2** methods for locating the positions of speakers are required. Requiring people to wear intrusive devices to track their positions inside rooms is not a practical requirement in everyday life. In practice, under real-world usage, smart-home devices tend to be placed ad-hoc and "out-of-the-way" with little regard to optimising their placement for estimating speaker position unintrusively e.g., by using video cameras. Therefore, this thesis explores, **how well can integrated cameras from**

ad-hoc placements of devices be used to estimate speaker positioning inside of rooms?

RQ4 Speech separation algorithms often perform poorly when the number of sources present is different from the number of sources being extracted. Therefore the number of overlapping speakers is important to model when simulating data. Speech separation algorithms also exploit context when extracting sources, therefore the placement of the overlap affects the difficulty. This thesis therefore explores, **how well do simulated datasets represent the overlap patterns found in real data? And how does poorly representing real data affect speaker extraction evaluation?**

RQ5 The behaviour of speakers in multiparty scenarios will result in different degrees of difficulty for the scenario, e.g., more or less speaker overlap depending on the degree of 'formality'. Modelling parties potentially allows for similar behaving parties to be grouped together and the difficulty of the generated data from participants to be predicted. This is a useful tool when considering the performance of a DSR system for a particular recording. Therefore this thesis explores, **how can representations be created to best model the difficulty of parties for ASR?**

1.3 Contributions

Analysis of speaker separation

In Chapter 4 an analysis of the angular separation distribution used in commonly used simulated datasets is shown. This is contrasted with estimates of the angular separation found in a real dataset, these estimates are later refined in Chapter 5. It was found that there is a large mismatch between the different simulated datasets and further still with the real data. This results in simulated datasets overemphasising the performance of separation angles that do not often occur in real data.

It was found that once we enforce a realistic angular separation into the simulation, the performance of a state-of-the-art baseline system drastically decreases. The degree to which the performance decreases depends on the technique being used. This could potentially lead to a claimed improved system in simulation not resulting in an improvement in real data i.e., its simulated performance improvement was due to improving separations that do not occur in real data.

Analysis of the impact of speaker distance

Chapter 5 contributes an impact analysis of modelling speaker distances in simulations. Using multiple cameras with overlapping views, the 2-D positions of people inside of rooms were estimated as well as angular separation. These 2-D positions enabled estimates of relative distances to the microphone to be modelled. It was found that when constraining estimates to be within a small room, the placement of people can be modelled using uniform random positioning. But when larger rooms are used the participants place themselves relatively closer together i.e., form groups.

When enforcing a realistic relative distance the results showed a complicated relationship which showed the impact of the realistic relative distance distribution depending on the angular separation distribution being used.

Generative modelling framework for realistic overlap

In Chapter 6, a generative framework for modelling the turn-taking of multiparty scenarios using a fixed number of speakers is presented and released as a Python package[2]. The framework produces overlap distributions that can approximate those found in the real data as well as the placement of the overlaps. This framework allows for arbitrarily long turn-taking patterns to be generated which can be used to create isolated mixtures for speech separation as well as other tasks such as diarisation.

Representations for analysing the difficulty of multi-speaker parties

Chapter 6 shows representations that can be computed from the generative model presented in the same chapter. The representations show a way to visualise the behaviour of speakers based on their turn-taking. These representations were shown to be a predictor for the performance of target-speaker extraction. They could further be used to evaluate other tasks such as diarisation, or they can be used in training ASR systems conditioned on the behaviour of speakers.

Speaker position data

To analyse the data in this thesis, speaker position data needed to be gathered from videos. This was achieved through annotating the data as well as using automatic people detection algorithms as described in Chapter 3. This data has been made available online[3] to reproduce

[2]https://github.com/jackdeadman/turn-taking
[3]https://chime.jackdeadman.com

the work. In addition to this, the data could be used by researchers to train ASR systems on speaker position information as the video data is not publicly available for use.

Annotation tools

Finally, the annotation tools used in this thesis were tailor-made to speed up the process of annotation for locating speakers. These tools have been made publicly available to benefit researchers who may want to annotate similar datasets[4].

1.4 Organisation of the thesis

The diagram in Figure 1.1 shows the structure of the thesis. In the introduction, we have discussed the motivation of the thesis and the gap in the literature that this work aims to address. That is we motivate the need for good simulation in order to drive research towards algorithms that perform well in simulation and real deployment data. In Chapter 2, we explore the literature on creating spatialised speech through simulation and how these signals are processed. This reviews the signal-processing techniques required for transforming an isolated clean-speech signal into a multi-channel reverberant signal. This then extends to exploring how simulated signals play a role in the development of automatic speech recognition systems. This involves looking at simulated datasets that are used for pre-processing as well as other roles such as pre-training and data augmentation. Chapter 3 introduces the CHiME-5 dataset, an existing multi-modal, multi-channel dinner party corpus, this is then compared with alternative datasets that could have been used for this work. The chapter includes tools developed to extract the location of speakers within the videos of the corpus both automatically and manually. Chapter 4 explores estimating the angular separation between speakers in a real recorded scenario by using single cameras. The results are then projected into a simulated environment to measure the impact of the mismatch. Chapter 5 extends this work by combining multiple cameras to estimate the 2D position of people inside the rooms This allows for distance estimates and refinement of the angular separation estimate. Chapter 6 looks into analysing speaker behaviour from a temporal perspective as opposed to a spatial one, this investigates how speaker overlap can be modelled as well as representations for the turn-taking behaviour of speakers. Finally, Chapter 7 concludes by discussing how the work addresses the research questions proposed in this introduction chapter.

[4]https://github.com/jackdeadman/video-annotation-tools

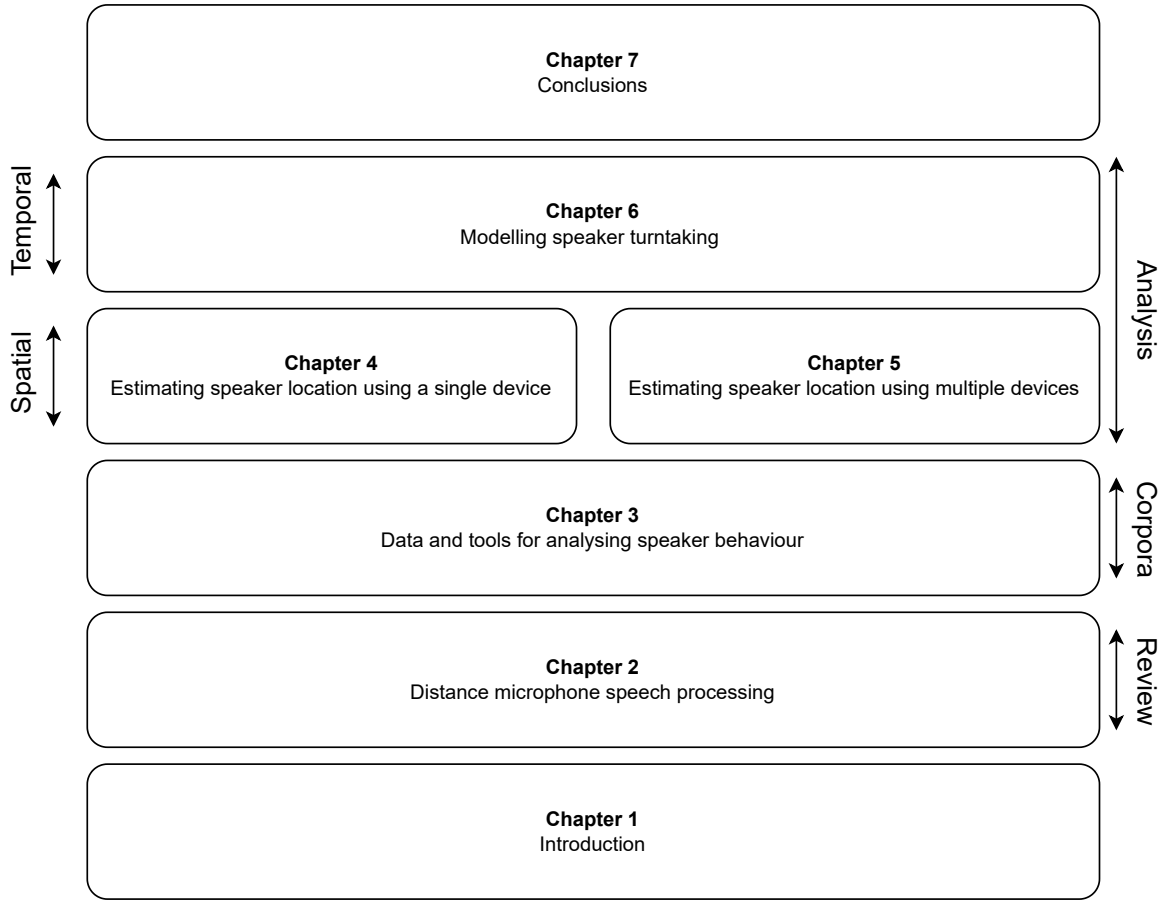

Fig. 1.1 Structure of the thesis chapters. The thesis begins with a review of the literature on distant microphone speech processing. It then goes on to review the data available to do the required analysis and the additional data requirements. The three experimental chapters then follow, with work on spatial analysis using speaker positions in Chapters 4 and 5. This is then followed by temporal analysis using speaker turn-taking in Chapter 6. This is then followed by the conclusions.

1.4.1 List of publications

During the course of the author's Ph.D they have published four papers, three (1, 2, 4) of which have directly contributed to the outputs presented in this thesis.

1. **Deadman. Jack** and Barker. Jon "Simulating realistically-spatialised simultaneous speech using video-driven speaker detection and the CHiME-5 dataset". *In proc. INTERSPEECH. 2020*

2. **Deadman. Jack** and Barker. Jon "Improved simulation of realistically-spatialised simultaneous speech using multi-camera analysis in the CHiME-5 dataset". *In proc. ICASSP. 2022*

3. Tu. Zehai, **Deadman. Jack**, Ma. Ning, Barker. Jon “Auditory-Based data augmentation for end-to-end automatic speech recognition”. *In proc. ICASSP. 2022*

4. **Deadman. Jack** and Barker. Jon “Modelling turn-taking in multispeaker parties for realistic data simulation”. *In proc. INTERSPEECH. 2022*

Chapter 2

Distant microphone speech processing

2.1 Introduction

Automatic speech recognition (ASR) technology has progressed rapidly over recent decades. This has been largely due to algorithmic development, data availability and computation resources providing the advancements needed for complex recognition tasks to be solved. In order to encourage development in the field, simplified tasks are first created by constraining real application scenarios to create simpler problems, which are sometimes referred to as "toy problems" that are more approachable. Once these simplified problems have been adequately solved more complex tasks can be addressed using what was learned from the prior research.

If research is to progress efficiently, great care needs to be taken when defining suitable constrained tasks. The constrained tasks need to be simple enough that progress can be made given state-of-the-art at the time of the research. Making a task too difficult will lead to little progress that can be measured and motivation will be lost. However, it is also necessary that the task is simplified in such a way it still progresses science in the correct direction that will later lead to the development of harder and more realistic tasks. If a problem is simplified in a way such that algorithms can exploit aspects of the task that will never be available in data from the real scenario, then research will be lead down the wrong path, promoting techniques that do not generalise beyond the toy problem. For example, if an ASR system exploits spectral properties that are only present in extremely clean signals then once noise is added to the signal the techniques will fail. Therefore, it is important when simplifying the tasks to understand what part of the tasks are *realistic* and what *unrealistic* parts could potentially be exploited.

Recognising speech from distant microphone recordings is one of the most difficult unsolved problems in ASR. The difficulty comes from the many aspects of complexity that occur due to the behaviour of people in the environment and the way the acoustic signal is

corrupted due to the properties of the environment. This leads to algorithms that work in simplified tasks failing in real environments due to the mismatch. Worse, it is often unknown *why* these algorithms fail in such a scenario. This failure may be due to a mismatch in one or more of the many unmeasured variables in the real data. Therefore, it is important to add complexity to the simplified task in a way that approaches the eventual real data. It is important to note, constrained tasks are not made more realistic simply by making them more 'difficult': a task can be made arbitrarily (e.g., extremely noisy), but if this difficulty is not moving the task closer to the real data then this too is going to lead research down a fruitless path.

This chapter aims to give an overview of the task of distant microphone speech recognition and the associated literature in multi-party environments. When reviewing this literature we shall be doing so in light of the discussion above. The key aim will be to understand why a distant microphone is a challenging problem and to understand something about the most popular approaches that go towards solving it. With this knowledge in hand, we can understand the aspects of the problem that need to be captured when designing the simplified datasets, i.e. the toy problems, that can be used as stepping stones towards producing systems for real applications. This understanding will then contribute to the design of the datasets introduced in the main thesis chapters.

The remainder of this chapter will proceed as follows. First Section 2.2 will provide a more formal description of what we mean by distant microphone and multi-party environment. Section 2.3 will describe the components that make up an ASR system for this setup. Next, speech separation techniques will be explored in Section 2.4 and 2.5 as this is a key component in multiparty scenarios. With Section 2.4 focusing on spectral-temporal filtering and Section 2.5 focusing on spatial filtering. Next, the chapter looks at artificial room simulation in Section 2.6. Then in Section 2.7 we explore where simulated data fits into the development on distant microphone ASR.

2.2 Problem setup

In this thesis, we will be considering 'multi-party environments'. By multi-party this simply means environments in which more than one person is present and more than one person may be speaking at any moment. The environments will also contain a mix of non-speech sources. To formally define the setup, a multiparty environment signal is captured with a microphone array denoted by $\mathbf{x}[t] = \begin{bmatrix} x_1[t] & \cdots & x_M[t] \end{bmatrix}^\top$. The array consists of M microphones and each microphone i captures a signal $x_i[t]$ at time t. The signal captured at the microphone

array will consist of a combination of J speech signals $\mathbf{C}[t] = \begin{bmatrix}\mathbf{c}_1[t] & \cdots & \mathbf{c}_J[t]\end{bmatrix}^\top$ and some background noise $\mathbf{n}[t] = \begin{bmatrix}n_1[t] & \cdots & n_M[t]\end{bmatrix}^\top$,

$$\mathbf{x}[t] = \sum_{j=1}^{J} \mathbf{c}_j[t] + \mathbf{n}[t]. \tag{2.1}$$

The *wet* speech signals $\mathbf{c}_j[t]$ are the result of clean signals being subjected to distortion from the containing environment and $\mathbf{n}[t]$ captures all remaining noise e.g., sensor and diffuse noise. The room corruption process can be modelled as a linear time-invariant (LTI) system. The response of a signal $s[t]$ with some LTI system H can be computed through convolution ($\circledast$) with the impulse response of that system i.e., $s[t] \circledast H(\delta[t]) = H(s[t])$. The convolution operation is defined for some signals $s[t]$ and $h[t]$ as,

$$s[t] \circledast h[t] = \sum_{k=-\infty}^{\infty} s[k]h[t-k]. \tag{2.2}$$

Therefore, the wet speech signals $\mathbf{c}_j[t]$ at the microphone array for source j are defined as,

$$\mathbf{c}_j[t] = \begin{bmatrix}s_j[t] \circledast r_{1j}[t] & \cdots & s_j[t] \circledast r_{Mj}[t]\end{bmatrix}^\top, \tag{2.3}$$

where r_{ij} is system response of the room of microphone i for the source position j. This signal is commonly known as the room impulse response (RIR) and characterises how the *dry* source signal $s_j[k]$ is corrupted by traveling from location of source j to microphone i.

Speech recognition in a multi-party environment aims to extract the speech sources $s_j[t]$ from the mixture and produce a transcript of each of the utterances. Alternatively, it may only be necessary to extract the speech of some target source $s'[t] \in \{s_1[t], ..., s_J[t]\}$ with all remaining sources treated as noise. This chapter aims to review techniques for producing a transcript for the target source(s) in the mixtures as well as discuss the challenges involved in this problem domain and the assumptions being made. Training and evaluating such systems often requires knowledge of $s_j[t]$ but this cannot be directly captured in real multi-party data. Data can be synthesised to address this through generating mixtures from close-talk microphone recordings $s_j[t]$ and known RIRs $r_{ij}[t]$. Methods for generating RIRs through measurement and simulation are discussed at the end of the chapter.

The experimental work of this thesis is situated within this framework. Chapter 4 and Chapter 5 consist of experimental work on producing more realistic RIRs (Equation (2.3)). Whilst Chapter 6 explores the mixing process of the sources (Equation (2.1)) i.e., the turn-taking behaviour of the speakers.

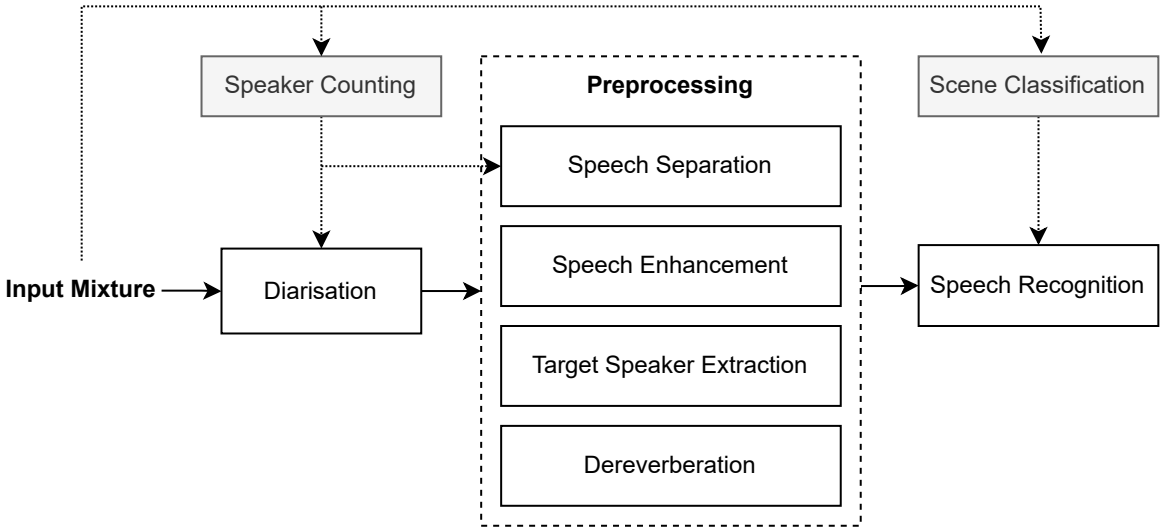

Fig. 2.1 Schematic overview of how distant microphone processing tasks can be combined to create a speech recognition system. Each of the individual components can be evaluated as a means to an end.

2.3 Components within distant speech recognition

In order to develop systems for distant microphone speech recognition the process is often broken down into several components. These components represent tasks that can be considered separately, i.e. trained and evaluated independently before being used to construct a full system. Example components include speech detection and speaker counting, source separation, dereverberation, speech recognition, etc. However, recent work has seen that it can be possible to train systems in an end-to-end fashion if the data is available (Li et al., 2022). The diagram in Fig 2.1 presents an example of how components can be combined to form a distant microphone ASR system.

Given a stream of multiparty audio containing the speech that we want to recognise, we must first determine the segments containing the individual utterances. The process of determining who is speaking and when is known as *diarisation*. Diarisation is often treated as an independent task with its own performance metrics (Bredin, 2017). Although more recent approaches treat diarisation and speech recognition *jointly*, i.e. solving both problems simultaneously rather than sequentially (Mao et al., 2020), we here consider more classical approaches that apply diarisation as a precursor to speech recognition.

After diarisation, smaller segments of speech can be processed in an enhancement step. These segments will contain the desired speaker but may also contain noise such as from the environment or interfering speakers. Enhancement aims to extract the desired signal from the mixture removing the undesired noise. Again, enhancement is often treated as a separate

task. Enhancement research considers many desired tasks, for example, telecommunications, hearing aids as well as a preprocessing before speech recognition. It is well known that an improved perception of audio quality from a human perspective may not lead to improved recognition performance. Therefore the loss functions the enhancement systems use may vary but the underlying architectures are unchanged.

Finally, this enhanced signal is fed into a speech recognition backend. For the system to perform well, it still needs to be robust to the variability in the data in distant microphone recognition. To achieve this systems are often trained on a variety of noisy data, e.g., corrupted by many different environments. This could consist of artificially created data (Ko et al., 2017) or recorded vast amounts of data (Chen et al., 2021). Other effective strategies for creating robust distant microphone ASR consist of removing the effects of the environment as a feature level with robust feature transforms (Gales and Woodland, 1996).

The component that has arguably the biggest governance over WERs is speech separation. Therefore, the extent to which systems can separate speech well governs their performance. The extent to which datasets are realistic is governed by the extent to which they capture the complexity of the real source separation cues.

2.4 Spectral-temporal filtering

The following section will discuss speech separation cues and the techniques to extract sources. These can be broken into *spectral-temporal* cues which exploit cues in the time-frequency domain and *spatial* cues which exploit cues from the source's physical location.

2.4.1 General framework

In this section, the general framework will be described in the context of single-channel enhancement networks. The next section describes how this framework can be extended to incorporate multiple channels this is performed by incorporating spatial information.

Given the mixtures $\mathbf{x}[k]$ the aim of spectral-temporal filtering is to extract the target source $s_j[k]$ or J target sources $\begin{bmatrix} s_1[k] & \cdots & s_J[k] \end{bmatrix}^\top$ using spectral information. In this section, $x_*[t]$ denotes a selected channel in $\mathbf{x}[t]$ i.e., $x_*[t] \in \{x_1[t], \cdots, x_M[t]\}$ which is known as the reference channel and may be arbitrarily chosen or through an estimate of audio quality. Note the distortion due to reverberation of the RIR r_{ij} can either be treated as noise and therefore part of the separation task or through further processing through dereverberation e.g., using weighted-prediction-error (Drude et al., 2018). The specifics usually depend on

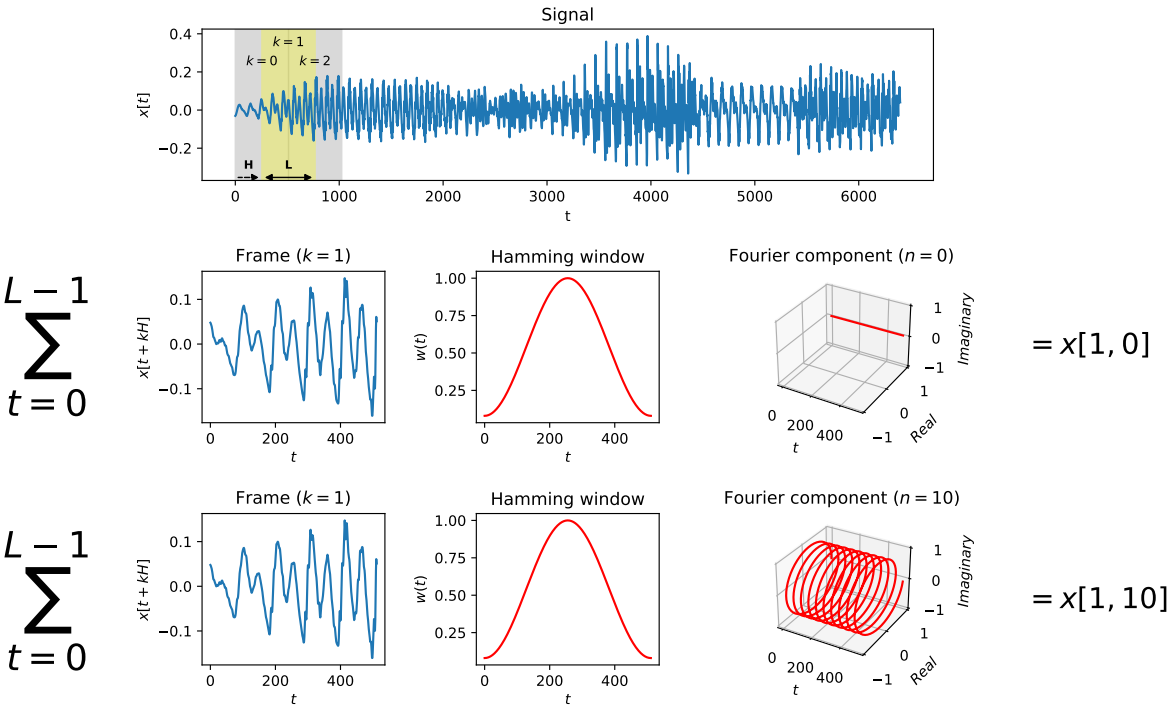

Fig. 2.2 A depiction of how the short-term Fourier transform (STFT) converts a signal $x[t]$ into a time-frequency representation. The time-domain signal is broken down into K overlapping frames by sliding a window with a shift size H. Each frame consists of L samples, and frequency information is then extracted from the frame using N filterbanks ($n = 1$ and $n = 10$ are shown above as examples). Above in red shows the filterbanks are a combination of a windowing function and a Fourier basis component. Instead of a fixed filterbank, these kernels can be learned from data for representations that are more task-specific than the fixed STFT.

the targets used in supervised training but nevertheless does not need to be considered when describing the framework.

The discussion in the following section lies within the framework of modern deep neural network (DNN) approaches, however, the classical approach began long before this e.g., Wiener filtering (Lim and Oppenheim, 1979). However, this is outside of the scope of this thesis. In modern DNN approaches, the task of separating sources from a mixture is often broken down into three stages, encoding, masking and decoding (Pariente et al., 2020). The encoder takes the time-domain signal $x_*[t]$ and transforms it into a new domain using a series of N filterbanks, using a sliding window of size L and a hop length H,

$$x_*[k,n] = \sum_{t=0}^{L-1} x_*[t+kH]u_n[t], \quad \text{for } n \in \{0,\ldots,N-1\}, \tag{2.4}$$

where k is the frame number and n is the frequency bin. Traditionally in digital signal processing, the kernel $u_n[t]$ in this formulation is fixed to be,

$$u_n^{\text{stft}}[t] = w(t)\exp\left(-j\frac{2\pi tn}{N}\right), \tag{2.5}$$

where $w(t)$ is a windowing function such as Hamming. Using this well-known fixed kernel $u_n^{\text{stft}}[t]$ would result in the Short-time Fourier transform (STFT). The STFT domain is very popular and is well-understood in signal processing. A diagram illustrating this process is shown in Figure 2.2.

In the general case for $u_n[t]$, the kernel can be parameterised and the weights of the kernel can be learned. Using a convolutional neural network to learn this kernel was a breakthrough in the source separation with the introduction of Conv-Tasnet (Luo and Mesgarani, 2019). This work showed that the performance of masking could be improved when the signal is transformed into a learned task-specific domain rather than the general-purpose Fourier domain. This work led to further kernel designs that have fewer parameters and make assumptions about their shape, e.g., a SincNet (Ravanelli and Bengio, 2018) and GaussNet (Loweimi et al., 2019). The choice of which depends on the application and the amount of data available.

Given the transformed signal $x_*[k,n]$, a masking network can be trained to estimate the function $\mathscr{M}$ which predicts J masks i.e., one for each of the sources. Using $\mathbf{X}_*$ as a $K \times N$ matrix representation of $x_*[k,n]$, the mask prediction function is defined as,

$$\mathscr{M}(\mathbf{X}_*) = \left[\mathbf{M}_1, \cdots, \mathbf{M}_J\right]^\top, \tag{2.6}$$

where $\mathbf{M}_j$ is a mask with the same shape as $\mathbf{X}_*$, allowing for each of the sources to be separated through element-wise multiplication ($\odot$),

$$\hat{\mathbf{S}}_j = \mathbf{M}_j \odot \mathbf{X}, \tag{2.7}$$

where $\hat{\mathbf{S}}_j$ is the matrix representation of the estimated source signal $\hat{s}_j[k,n]$, which is in the transformed domain. The source can then be fed into further steps e.g., into a recognition system or transformed back into the time domain using a decoder function,

$$\hat{s}_j[t] = \sum_{k=0}^{K-1}\sum_{n=0}^{N-1} \hat{s}_j[k,n]v_n[t-kH], \tag{2.8}$$

where the kernel $v_n[t]$ can be jointly learned alongside $u_n[t]$ or alternatively the pseudo-inverse of v_n can be computed using singular-value decomposition to reduce the number of parameters. In the specific STFT case, the corresponding fixed transform for the STFT

is known as the inverse short-term Fourier transform (ISTFT). Often the three components (encoding, masking and decoding) are jointly trained in an “end-to-end" manner using time-domain loss functions. When fixed transforms are used the loss function can be in the frequency domain e.g., the error between the predicted mask and an ideal mask. Therefore, the task of source separation can be mainly reduced to a mask prediction task.

2.5 Spatial filtering

As well as being able to filter a signal based on its time-frequency properties, it is also possible to filter based on the source location of the signal. Cues for the location of the signal can be captured by recording the signal simultaneously through multiple microphones at different spatial locations (i.e., by a so-called ‘microphone array’). The relative delays between the signal arriving at the separate microphones are indicative of the location of the sound source. First, in this section, beamforming approaches inspired by signal processing techniques in antennas are explained, then spatial probabilistic models are explained followed by general spatial features that can be computed and used to estimate masks.

2.5.1 Beamforming

Beamforming is a common approach for *spatially* filtering signals. A beamformer is able to enhance signals arriving from one or more directions while suppressing signals arriving from others. Beamforming algorithms use multiple (sample-synchronised) microphones (i.e. microphone arrays) and exploit the signal time-delay of arrival discussed earlier.

We will start by considering the case of a linear microphone array, i.e. one in which the microphones are arranged in a line. If we assume a signal is sufficiently far away that the waves can be approximated as planar, then the sounds coming from directly in front of the array can be enhanced relative to sounds from competing directions by simply summing the microphone signals. The sounds being emitted from in front of the array will be enhanced as they will be received by all the microphones at the same time (due to plane wave assumption). The signal will be enhanced due to them being the same signals and all in phase; therefore, the interference will be constructive. Sounds being emitted from different directions will receive a lower gain as they will not be in phase and hence will not have constructive interference.

To enhance a source in a different direction, the received signals can be delayed before summing them together, effectively steering the direction of enhancement. The amount of time needed to delay the signal to achieve the desired angle can be calculated by,

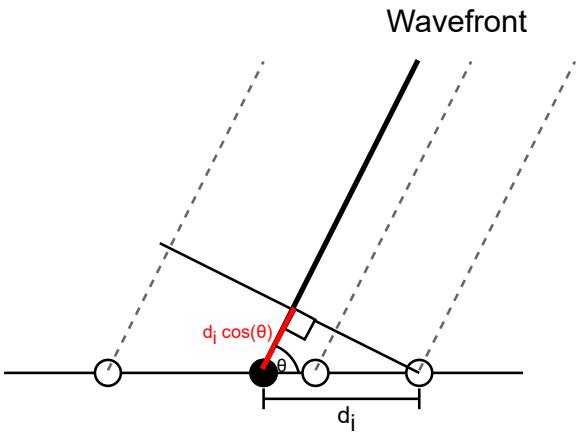

Fig. 2.3 Geometry for calculating the delays used in delay and sum beamformer. Here, we assume that the source is sufficiently far away that we can assume the wavefront is a plane wave. This means that when the wavefront reaches a reference point, a right-angle triangle is formed and the distance the wavefront needs to travel to reach any point in the array can be computed using trigonometry. Given a distance d_i away from a reference point, the distance the wave will travel before reaching the microphone i is given by $d_i \cos(\theta)$ where θ is the direction the beamformer is pointing. From this distance the delay can be computed by using the speed of sound in the medium.

$$\tau_i^\theta = \frac{-d_i \cdot \sin(\theta)}{C}, \tag{2.9}$$

where d_i is the distance between the microphone i and a reference point (e.g., centre or some reference microphone), θ is the desired angle to steer to, and C is the speed of sound in the medium.

The signal can be enhanced by applying the delay and then summing the channels together and then normalising the gain, this results in the delay and sum beamformer (Grythe and Norsonic, 2015),

$$\hat{s}[t] = \frac{1}{M}\sum_{i=1}^{M} x_i[t - \tau_i^\theta], \tag{2.10}$$

where $\hat{s}[t]$ is an estimate of the source in the direction θ. The delay and sum beamformer can be applied in the frequency domain through a filter that changes the phase. Let the multi-channel, time-frequency domain signal be denoted by $\mathbf{x}[k,n] = \begin{bmatrix} x_1[k,n] & \cdots & x_M[k,n] \end{bmatrix}^\top$. This leads to a class of beamformers known as filter and sum, as they can be formulated as follows in the frequency domain,

$$\hat{s}[k,n] = \mathbf{w}_n^H \mathbf{x}[k,n], \tag{2.11}$$

for the delay and sum beamformer the weight vector $\mathbf{w}_n^{\mathrm{DS}}$ is defined as,

$$\mathbf{w}_n^{\mathrm{DS}} = \frac{1}{M}\mathbf{d}_n^{\theta}, \tag{2.12}$$

$$\mathbf{d}_n^{\theta} = \begin{bmatrix} \exp(-j\omega_n \tau_1^{\theta}) & \exp(-j\omega_n \tau_2^{\theta}) & \cdots & \exp(-j\omega_n \tau_M^{\theta}) \end{bmatrix}^{\top}, \tag{2.13}$$

where $\mathbf{d}_n^{\theta}$ is the *steering vector* representing the plane wave propagation in direction θ and $\omega_n = 2\pi n/N$ is the radial frequency in radians per seconds. Note that the weight vector is frequency dependent; this leads to the beamformer obtaining different characteristics per frequency as shown in Figure 2.4.

The class of filter and sum beamformers optimise the weight vector $\mathbf{w}$ to create filters that are optimal according to some criteria and are known as adaptive beamformers. Such filters are optimal with respect to an estimate of the spatial covariance matrix defined by the cross-spectral density (CSD) between the signals received at all the pairs of microphones in the array. For a series of vectors $\mathbf{a}[k,n] \in \mathbb{C}^M$ the frequency-dependent spatial covariance matrix $\mathbf{\Phi}_n^{(aa)} \in \mathbb{C}^{M \times M}$ is defined as,

$$\mathbf{\Phi}_n^{(aa)} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{a}[k,n]\mathbf{a}[k,n]^H, \tag{2.14}$$

most of the time, the true value of the covariance can only be estimated, for example, we may want to know the spatial covariance of the noise, this would require estimating the noise at each of the microphones without the target speech. Through using a masking approach described previously this can be estimated by filtering the signals and then computing the statistics. Alternatively, the noise covariance can be estimated by finding portions of the signal that only contain noise and no target speech. Assuming independence between the noise and the target speech the covariance matrices can be broken down (Souden et al., 2009),

$$\mathbf{\Phi}_n^{(xx)} = \mathbf{\Phi}_n^{(yy)} + \mathbf{\Phi}_n^{(nn)}, \tag{2.15}$$

where $\mathbf{\Phi}_n^{(xx)}$, $\mathbf{\Phi}_n^{(yy)}$ and $\mathbf{\Phi}_n^{(nn)}$ are the received signal covariance, target covariance and noise covariance respectively. The received signal covariance can be computed without filtering, therefore an estimate of the target covariance can be computed with the knowledge of a noise covariance.

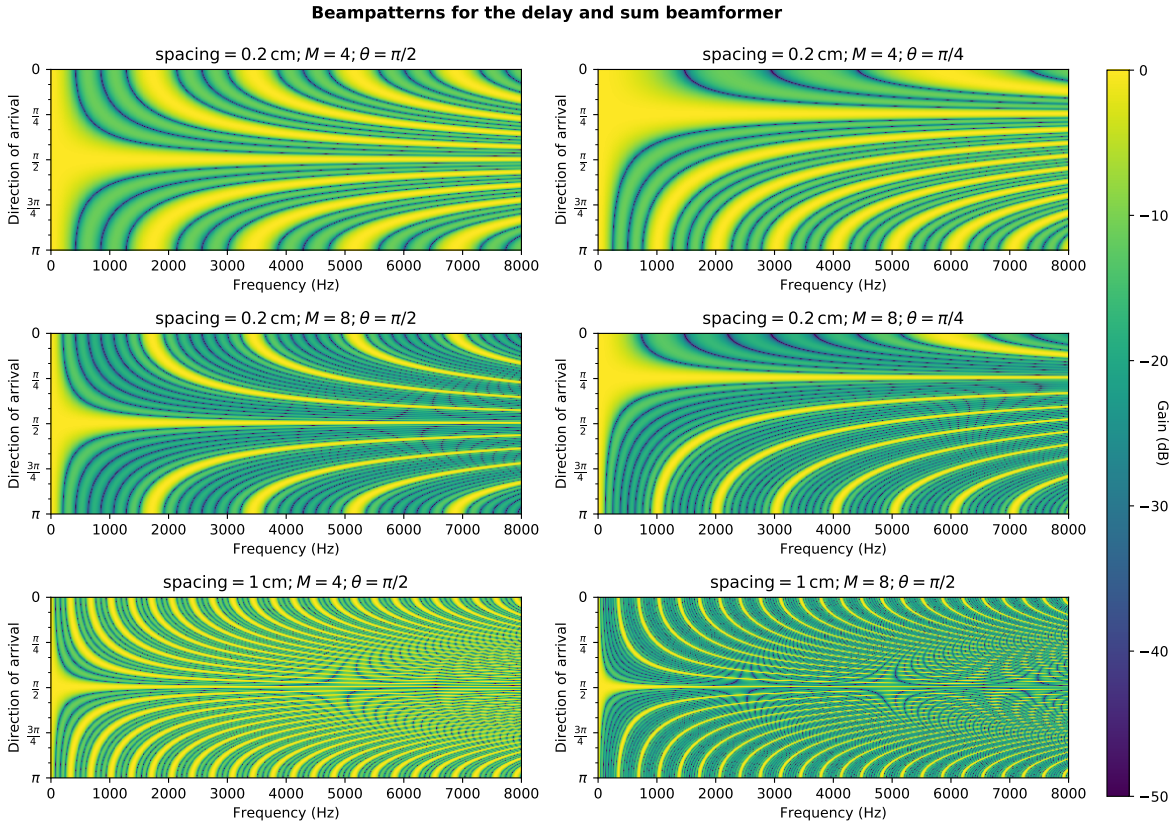

Fig. 2.4 Beampatterns of the delay and sum beamformer pointing in different directions (θ) from linear microphone array with M channels. The spacing between the microphones is given in the plots. The direction of arrival indicates the angle of incident of the sound source and the colour represents the gain. Lower gains result in sound sources from those directions and frequencies being suppressed. The plots shows how changing the spacing between the microphones and the number of microphones can have a large impact on the characteristics of the beamformer such as the spatial aliasing and the beamwidth.

One of the most popular choices of statistically optimal beamformers is the Minimum Variance Distortionless Response (MVDR) (Capon, 1969) beamformer which minimises the energy of the noise with the constraint that the signal in the direction of the beam is not distorted,

$$\mathbf{w}_n^{\text{MVDR}} = \arg\min_{\mathbf{w}} \mathbf{w}_n^H \mathbf{\Phi}_n^{(nn)} \mathbf{w}_n \qquad \text{s.t. } \mathbf{w}_n^H \mathbf{d}_n^\theta = 1, \tag{2.16}$$

where $\mathbf{\Phi}_n^{(nn)}$ is the spatial covariance matrix for the noise and $\mathbf{d}_n^\theta$ is the steering vector in the direction of the beam, note this is applied for each frequency bin n. For the steering vector $\mathbf{d}_n$ the delays can be estimated (DiBiase et al., 2001; Knapp and Carter, 1976; Schmidt, 1986) or the principal component of an estimate of the signal covariance ($\mathbf{\Phi}^{(yy)}$) matrix can be used. The constrained MVDR problem has a solution,

$$\mathbf{w}_n^{\text{MVDR}} = \frac{\mathbf{\Phi}_n^{(nn)-1} \mathbf{d}_n^\theta}{(\mathbf{d}_n^\theta)^H \mathbf{\Phi}_n^{(nn)-1} \mathbf{d}_n^\theta}, \tag{2.17}$$

where $\mathbf{\Phi}^{(nn)-1}$ is the inverse of the noise spatial covariance matrix, in practice, due to numerical stability issues, the inverse is not directly computed and alternative formulations are used. The solution requires estimating a model of the diffuse uncorrelated noise and the direction to enhance.

The Generalized Eigenvalue (GEV) (Warsitz and Haeb-Umbach, 2007) beamformer is also a popular choice and does not require the direction vector, instead, the spatial covariance matrix of the source signal is estimated which has the direction information embedded into the matrix via its principal components. The GEV beamformer finds a weight vector $\mathbf{w}_n^{\text{GEV}}$ which maximises the energy of the signal whilst minimising the energy of the noise. This leads to the following optimisation problem,

$$\mathbf{w}_n^{\text{GEV}} = \arg\max_{\mathbf{w}} \frac{\mathbf{w}_n^H \mathbf{\Phi}_n^{(yy)} \mathbf{w}_n}{\mathbf{w}_n^H \mathbf{\Phi}_n^{(nn)} \mathbf{w}_n}, \tag{2.18}$$

where $\mathbf{\Phi}^{(yy)}$ and $\mathbf{\Phi}^{(nn)}$ are the signal and noise spatial covariance matrices respectively. The solution to Equation 2.18 leads to a generalized eigenvalue problem which has the solution,

$$\mathbf{\Phi}_n^{(\text{yy})} \mathbf{w}_n^{\text{GEV}} = \lambda_n \mathbf{\Phi}_n^{(\text{nn})} \mathbf{w}_n^{\text{GEV}}, \tag{2.19}$$

where the optimal weight vector $\mathbf{w}_n^{\text{GEV}}$ is the *generalized principal component* with the corresponding eigenvalue λ_n.

2.5.2 Spatial models

Recall for multi-channel recordings the observations for a time-frequency are represented by an $M-$dimensional complex vector, i.e., $\mathbf{x}[k,n] \in \mathbb{C}^M$. Given the phase gives information about the location of a signal we can exploit this by clustering similar T-F bins together which have similar phases. One of the difficulties of using phase information is that it wraps, therefore care needs to be taken with the model choice and distance measure. A common approach to this problem is to use *directional statistics*, which are unit vectors around a complex hypersphere. A probabilistic model over this domain can be used to cluster similar vectors. The complex vector x_{kn} is whitened, removing the magnitude information from the vector,

$$\mathbf{z}[k,n] = \frac{\mathbf{x}[k,n]}{\|\mathbf{x}[k,n]\|}, \tag{2.20}$$

using directional statistics, unsupervised clustering techniques can be used to separate the sources in an utterance by grouping each $\mathbf{z}[k,n]$ into clusters or assigning an affiliation probability for each cluster. These assignments (or affiliations) can be used directly to create a mask, e.g., a binary mask from hard assignments. The resulting masks can then be used directly to enhance the signal through convolution, or they can be used to gather the required statistics to use in beamforming techniques.

Clustering directional statistics requires specialised models that operate over the confined space and in the complex domain. To use these models for source separation a mixture of P models is used. Each model represents one of J sources and an additional one to catch all remaining noise (i.e., $P = J+1$),

$$p(\mathbf{z}[k,n];\Theta_n) = \sum_{p=1}^{P} \pi_n^{(p)} \mathscr{A}(\mathbf{z}[k,n];\theta_n^{(p)}), \tag{2.21}$$

where $\Theta_n = \{\pi_n^{(1)}, \theta_n^{(1)}, ..., \pi_n^{(P)}, \theta_n^{(P)}\}$ with $\pi_n^{(p)}$ being the component weight and $\theta_n^{(p)}$ being the distribution parameters for the spatial model $\mathscr{A}$. It is important to note that each of the frequency bins n is modelled independently and therefore correlations between frequency bins are not exploited.

One distribution used for $\mathscr{A}$ is the complex Watson distribution (denoted by $\mathscr{W}$) (Mardia and Dryden, 1999),

$$\mathscr{W}(\mathbf{z};\boldsymbol{\alpha},\kappa) = \frac{(M-1)!}{2\pi^M F_1(1,M;\kappa)} \exp(\kappa \|\boldsymbol{\alpha}^H \mathbf{z}\|^2), \tag{2.22}$$

where M is the number of microphones, $\boldsymbol{\alpha}$ is the mean direction (located on the same unit hypersphere as $\mathbf{z}$), κ is the concentration factor and F_1 is the Kummer function. The concentration defines how narrow the distribution will be around the mean direction, i.e., a high value for k indicates the members of the distribution are located close to the mean vector. The complex Watson distribution is rotationally symmetric around $\boldsymbol{\alpha}$. The symmetric nature does not necessarily reflect the true distribution of $\mathbf{z}[k,n]$. To allow for non-symmetrical shapes around the mean vector the complex angular central Gaussian (cACG)(Ito et al., 2016) was introduced, denoted by $\mathscr{C}$. The shape of the distribution is parameterised by a positive-definite Hermitian matrix $\mathbf{B}$ allowing for non-symmetric distributions where the mean direction vector is the principal component of the matrix, and the concentration is the eigenvalue,

$$\mathscr{C}(\mathbf{z}|\mathbf{B}) = \frac{(M-1)!}{2\pi^M \det(\mathbf{B})} \frac{1}{(\mathbf{z}^H \mathbf{B}^{-1} \mathbf{z})^M}. \tag{2.23}$$

Both these models have parameters that are fit through using iterations of expectation maximisation procedures. After fitting the parameters of the distribution, the affiliations to each of the sources can be used to filter the signal and extract source component p. This results in P source signal estimates $\hat{\mathbf{c}}_p[k,n]$ at the microphone array,

$$\hat{\mathbf{c}}_p[k,n] = \mathbf{x}[k,n] \pi_n^{(p)} \mathscr{A}\left(\frac{\mathbf{x}[k,n]}{\|\mathbf{x}[k,n]\|}; \theta_n^{(p)}\right), \tag{2.24}$$

therefore, for the cACGMM this would be,

$$\hat{\mathbf{c}}_p[k,n] = \mathbf{x}[k,n] \pi_n^{(p)} \mathscr{C}\left(\frac{\mathbf{x}[k,n]}{\|\mathbf{x}[k,n]\|}; \mathbf{B}_n^{(p)}\right). \tag{2.25}$$

The parameters of the probability distributions that are being fit for the directional statistics $\mathbf{z}[k,n]$ are not shared between frequency bins; this leads to the frequency permutation problem. Between different frequencies, it can not be guaranteed that a component index in one bin corresponds to the same component index in another bin, e.g., the noise cluster may be index 0 for one bin and index 1 for another bin. To mitigate this permutation problem, an alignment can be calculated by maximising the correlation of neighbouring frequencies (Sawada et al., 2010). In addition to this, a final step to associate a mixture component p with a desired source component j is required.

2.5.3 Spatial features

In Section 2.4 we saw how spectro-temporal features extracted from a single channel could be used to estimate spectro-temporal masks for source separation, e.g. conv-tasnet. The

performance of these mask estimates can be improved by including 'spatial features' extracted from microphone arrays. Spatial features that indicate the location of sources can be used to inform masking networks beyond spectral information. The spatial features are concatenated as input alongside a selected reference channel's spectral features (Gu et al., 2019; Zhang et al., 2020a). It is also possible to use all the channels as input directly with the internals of the network learning the spatial information (Luo and Mesgarani, 2020).

The inter-channel phase difference (IPD) is a commonly used spatial feature. IPDs are used in the frequency domain and are successful in source localisaton (Evers et al., 2020) as well as being used as additional input features for enhancement networks (Gu et al., 2019). The IPD features are simply the difference between the phases for the same time-frequency bin across the two channels $x_i[k,n], x_j[k,n] \in \{x_1[k,n], x_2[k,n], \ldots, x_M[k,n]\}$,

$$\text{IPD}(x_i[k,n], x_j[k,n]) = \angle x_i[k,n] - \angle x_j[k,n], \tag{2.26}$$

where $\angle$ is the argument of the complex number (i.e., $\angle(a+bj) = \text{atan2}\left(\frac{b}{a}\right)$). In frequency domains, as a result of learned filter banks (e.g., conv-tasnet), IPD features can still be computed if the filterbank consists of analytic filters. In Gu et al. (2019), an analytic filter $u_n[t]$ was formed by learning real kernels $w_n[t]$ resulting in the subsequent derived real and imaginary parts,

$$u_n[t] = u_n^{\text{real}}[t] + j u_n^{\text{im}}[t], \tag{2.27}$$

$$u_n^{\text{real}}[t] = w_n[t] \cos\left(\frac{2\pi t n}{L}\right), \tag{2.28}$$

$$u_n^{\text{im}}[t] = w_n[t] \sin\left(\frac{2\pi t n}{L}\right), \tag{2.29}$$

where L is the length of the analysis window and n is the filterbank index. Alternatively, an unconstrained filter can be learned and then the Hilbert transform can be used to form an analytic filter (Pariente et al., 2020) i.e., $u'[t] = u[t] + j\mathscr{H}(u[t])$ where $\mathscr{H}$ is the Hilbert transform. Given any analytic filter, the IPD can be computed as follows,

$$\text{IPD}(x_i[t], x_j[t]) = \text{atan2}\left(\frac{x_i[t] \circledast u^{\text{im}}[t]}{x_i[t] \circledast u^{\text{real}}[t]}\right) - \text{atan2}\left(\frac{x_j[t] \circledast u^{\text{im}}[t]}{x_j[t] \circledast u^{\text{real}}[t]}\right), \tag{2.30}$$

where $x_i[t], x_j[t]$ are the time-domain versions of $x_i[k,n], x_j[k,n]$. A problem with using IPD features is that they require a larger analysis window compared with the windows used in enhancement, therefore there is a problem with alignment. In Zhang et al. (2020a), spatial

features are learned using a 2D convolution across channels in the same domain as the masking network, avoiding the alignment issue and surpassing the performance of using frequency domain IPD features.

The aforementioned techniques use spatial and spectral cues, and these cues are typically 'data-driven', i.e. learned from representative data using DNNs. This is potentially problematic as cues learned from simulation may not generalise to real data. At the very least, we need to take great care when simulating training data. In the next section, we will look at commonly used techniques for data simulation and consider their strengths and weaknesses.

2.6 Artificial room simulation

When developing systems for distant microphone speech recognition it is often not practical to deal directly with real signals. As explained previously, real data may be too complex, i.e. too far beyond the capabilities of the state of the art. Recording real data in carefully controlled environments is an option, but this is often expensive, and inconvenient and still leaves problems with collecting accurate 'ground truth' measurements of individual sound sources. For these reasons, many of the datasets commonly used for development in distant microphone ASR are produced by simulation. In this section, we review the approaches that are typically employed.

When we are talking with a person, the sounds we produce from our articulators resonate in the environment we are speaking in before reaching the ears of the listener. In the case of a recording, the listener is the microphone or microphone array. The acoustic properties of the signals change drastically depending on the environment, for example, a large empty room will be 'echoey' and a small furnished room will be absent of such echoes. The distance between the source (the person speaking) and the sink (listener or recording device), plays a role in the acoustic properties. Intuitively, we would expect a person talking who is far away from us to sound quieter than a closer person. The delays and level differences between the signals reaching our ears also give us cues for the location of the source. Therefore, we can understand that in the real-world speech sources have spatial properties defined by the location of the speaker, location of the listener and the containing room. In this work we call simulating a signal being corrupted by the environment like this, a *spatialised* signal, in the wider literature this is also known as auralization.

Auralization is a larger field that encompases many aims. For example producing audio that mimics the acoustics of large concert halls, which can be used to aid the design. As well as for entertainment purposes in creating immersive video games and music production. It is also used in the design and evaluation of signal-processing techniques such as source

separation and spatial audio. Whilst the goals are similar the tradeoff to reach the goals is not, a video game requires the processing to be real-time with a believable perception of the environment. A tool for evaluating processing techniques will benefit a spatialised signal more representative of real life.

If a near-field microphone is used to record speech then these spatial properties are almost entirely removed by design i.e., special recording setups may be used to mitigate any undesirable spatial effects. These near-field recordings can then be 'spatialised' through artificial room simulation in order for them to have the desired spatial properties for a defined experimental setup or through measuring the properties of a room. In this section, we will explore the assumptions and techniques used when modelling spatialised speech.

2.6.1 Acoustic rendering

To render an acoustic scene the sound wave propagates around the room before reaching the microphone. Inspired by the rendering equation used in computer graphics the acoustic rendering equation was proposed in (Siltanen et al., 2007). The specific details of which are beyond the scope of this thesis but the interested reader should refer to the original paper. However, it will be described enough to compare different methods for approximating the equation. The equation is defined as follows,

$$l(\hat{\mathbf{p}},\boldsymbol{\omega}) = \overbrace{l_0(\hat{\mathbf{p}},\boldsymbol{\omega})}^{\text{Emitted}} + \underbrace{\int_{\mathscr{G}} R(\mathbf{p},\hat{\mathbf{p}},\boldsymbol{\omega}) l\left(\mathbf{p}, \frac{\hat{\mathbf{p}}-\mathbf{p}}{\|\hat{\mathbf{p}}-\mathbf{p}\|}\right) \mathrm{d}\mathbf{p}}_{\text{Reflections from all incoming directions}}. \tag{2.31}$$

At a high level, the equation reduces the rendering process to formulating the acoustic energy at the 3-D point $\hat{\mathbf{p}}$ traveling in the direction $\boldsymbol{\omega}$. The directivity pattern of the microphone can then be considered when simulating the acoustics captured by summing the energy in the appropriate directions. The equation is recursively defined as the energy being emitted at that point and direction (l_0) plus the energy reflected from all possible incoming directions i.e., $\mathscr{G} \subset \mathbb{R}^3$ is all the possible surface points the reflection could have come from. R encompasses the reflectivity of the surface (e.g., how much scattering occurs when bouncing off the surface), the geometry of the surface (e.g., the rotation of the surface) and whether the surface is visible,

$$R(\mathbf{p},\hat{\mathbf{p}},\boldsymbol{\omega}) = \overbrace{V(\mathbf{p},\hat{\mathbf{p}})}^{\text{Surface visibility}} \underbrace{\rho\left(\frac{\mathbf{p}-\hat{\mathbf{p}}}{\|\mathbf{p}-\hat{\mathbf{p}}\|}, \boldsymbol{\omega}; \hat{\mathbf{p}}\right)}_{\text{Reflection characteristics}} \overbrace{g(\mathbf{p}-\hat{\mathbf{p}})}^{\text{Geometry}}. \tag{2.32}$$

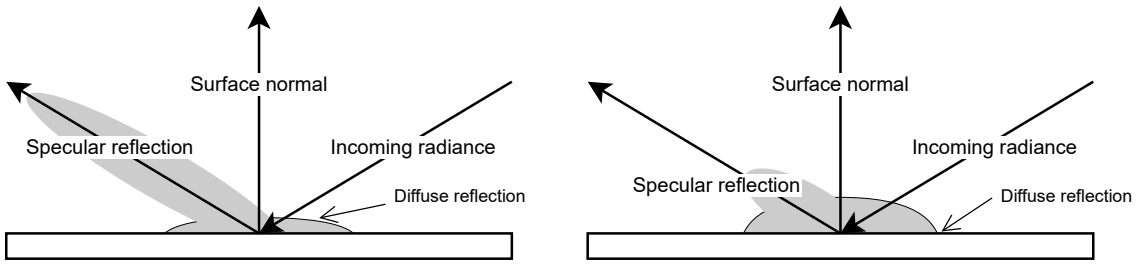

Fig. 2.5 An original illustration of example bidirectional reflectance distribution functions (BRDF) used in artificial room simulation. The figure on the left shows a highly reflective surface, whilst the right shows a more diffuse surface. BRDFs are used to model how sound sources scatter across a surface.

The visibility predicate function $V(\cdot)$ simply indicates whether or not the line between points $\mathbf{p}$ and $\hat{\mathbf{p}}$ is visible, allowing for non-convex environments to be modelled. $g(\cdot)$ describes how the energy flows for the geometry of the surface (e.g., using surface normals) and $\rho(\cdot)$ is the bidirectional reflectance distribution function (BRDF), describing how the sound scatters when reflecting. An example schematic of two BRDFs is shown in Figure 2.5. A highly reflective surface (left) will have less diffuse reflections and more energy in the direction of the reflection compared to a less reflective surface (right) which has more diffuse energy. The complete BRDF can be measured from real materials but often they are parameterised from absorption and scattering coefficients. It is important to note that these coefficients are frequency dependent and this is typically considered in bands of frequencies and the simulation is run multiple times at different frequencies.

Image Method

The simplest approach to room simulation is the image method (Allen and Berkley, 1979). This method is by far the most commonly used in simulated distant microphone scenarios, this is largely due to its efficiency. Often many thousands of environments need to be simulated to be used in speech separation, therefore it is not practical to use computationally expensive tools that can take several hours per scene to render.

The image method works by simulating the sound wave propagating around the room reflecting off walls along the mirror reflections i.e., no diffuse reflections in the BRDF. Non-cuboidal rooms are possible using the image method, however, most simulated datasets do not take advantage of this due to the performance gains possible with the cuboidal assumption.

Ray-tracing

More advanced techniques borrow ideas from computer graphics. Ray tracing approaches model the acoustic environment through propagating rays (not waves) through scenes and bouncing off walls. After each reflection, additional rays are generated in several directions accounting for the scatting of the sound. A ray continues bouncing around the environment until a minimum energy level is reached or a maximum number of reflections has occurred. The computational complexity of this method is a large reason why it is not commonly used and many of the implementations are behind expensive commercial software (Schröder and Vorländer, 2011). In recent years, pyroomacoustics (Scheibler et al., 2018) an open source, tool has provided a free implementation allowing for more advanced scenes to be modelled but this still lags behind commercial software significantly. For example, RAVERN (Schröder and Vorländer, 2011) can render scenes with 3-D models within them e.g, furniture.

In the work in this thesis, the image method is used for simulation[1]. The main reason for this decision is due to the fact that the majority of benchmark datasets use this technique which allows for a fairer comparison by changing parameters within the simulation and then keeping simulation software constant. Using more complex simulation techniques such as ray-tracing would provide overall more realistic simulations, however, this would be extremely computationally expensive to use as in this work as a large number of different datasets will be constructed and compared. However, the analysis of the speakers is general and applicable to all simulations. The simulations require speaker behaviours to be defined inside of their scenarios. The parameters of interest are the subject of the next section.

2.6.2 Parameters of a simulation

Given a toolkit for providing a simulation technique, we still need to provide (realistic) metadata (i.e., scene description parameters) of the simulation. There are numerous aspects of a simulation that needs to be decided beforehand (e.g. receiver characteristics, source characteristics - location, directivity - room geometry, surface absorption characteristics, air temperature and humidity etc.).

First, for a room to be simulated the geometry of the room needs to be decided. The allowable complexity of the room will depend on the modelling technique being employed. Early work required cuboidal rooms to be used, but modern software allows for arbitrary shapes to be used. Advanced commercial software allows for full 3-D models to be used in the simulation allowing not just rooms to be modelled but also the contents of the room (Schröder and Vorländer, 2011). There is a clear trade-off between the complexity of the

[1]Toolkit: https://github.com/ehabets/RIR-Generator

rooms and the number of rooms designed. With a simple cuboidal room, the width and height can be a random value chosen allowing for many configurations to be generated. For a detailed scene, randomly generating configurations is more complicated.

The next parameter that can be modelled is the amount of reverberation. One of the values used to measure reverberation is T60 which is the amount of time it takes the sound pressure level to reduce by 60 dB. When using the image method, this value can be directly chosen as a parameter of the simulation. This is achieved by using the inverse of Sabine's equation to calculate the absorption coefficient. For more complicated simulations this is the result of the materials used in the acoustic scene. Instead, absorption coefficients for walls in the scene are chosen and then a T60 can be measured.

Next, the position of the people in the room and the microphone placement are metadata that needs to be decided. Typically this is chosen uniformly within the room (Hershey et al., 2016; Maciejewski et al., 2020). The distances between speakers will affect the difficulty of the mixtures as spatial cues become less useful as discriminating features when in close proximity.

As well as the position of the speaker, the direction they are facing also plays a role in the way the sound propagates around the room. The sound emitted from a person talking is not uniformly dispersed spherically from a point sound but has some directivity. Modelling the directivity of sound sources was largely only possible in commercial software, but recently it has been made available in free software (Scheibler et al., 2018).

For speech separation datasets, scenes are often simulated with talkers performing single utterances and starting co-temporaneously (e.g., Hershey et al. (2016)). More recently, there has been an interest in producing datasets suitable for developing and evaluating diarization with multiple speakers and attempts made at reproducing realistic conversation turn-taking patterns (e.g., Chen et al. (2020)). Specific examples are discussed in more detail in the following section.

2.7 The role of simulated data in speech recognition

Simulated data plays a large role in the development of distant microphone ASR systems, both in providing training data as well as evaluation data. In this section, an overview of commonly used datasets is shown, then an overview of how simulated data is used to *train* as well as *evaluate* ASR and components of ASR systems.

2.7.1 Simulated datasets

A simulated dataset in this work is any database of signals which were generated through a computer program either in its entirety or in part. This is in contrast to *real* data which is directly captured in an environment. Typically the speech signal used in artificial data is captured from a real recording, often this is recorded in a quiet environment, close to the microphone, this is denoted as a *clean* signal. Artificial data can then be created by processing the clean signal. For example, background noise can be added to the signal by simply combining the clean signal with some background noise which could be recorded independently or generated e.g., white noise. The background noise could be a competing signal that needs to either be separated or suppressed. It is important to note that simply adding the signals together ignores many of the phenomena that occur in the real world. However, it does provide a cheap way to generate a large amount of data.

To create artificial signals which have some of the acoustic properties of the signal moving around the room, the artificial simulation techniques discussed in the previous section are used. To create such a signal the clean signal is convolved with the RIR generated by the simulation software. RIR can also be recorded instead of generated, the recorded RIR would then be used to create the artificial data which sounds like it was recorded in the room the RIR was recorded in. However, capturing a large dataset of RIRs is expensive and the variety of rooms will always be limited.

Finally, a combination of mixing and room simulation approaches is used in practice when creating artificial datasets. For single-channel enhancement room simulation is used to create reverberant signals. In the multi-channel case, it is essential to have some kind of environmental simulation either through recorded or simulated RIR. By their nature, a simulation of the environment is required for the spatialised datasets as the amount of delay between the microphones for the signal when reaching the array is exploited in the spatial filtering. To simulate two people talking inside a room the two spatialsied signals are created and then they are combined together through addition. The inclusion of background noise into the mixture is a non-trivial problem. For example, adding single-channel recordings of background noise to each of the artificial channels will result in a source in front of the array. Recording multi-channel background noise and positioning the microphones the same distance away as the recording device can be a method to use the background noise but this limits the amount of data available and there will still be a mismatch in the data e.g., directivity patterns of the microphones and the geometry of the room the data was recorded in will not be the same as the simulation.

The use of simulated data has been widespread across the literature. The work in this thesis is particularly interested in overlapping speech in distant microphone ASR. Speech

separation algorithms aim to separate the overlapped speech into separated clean signals. This field was largely initiated with Deep Clustering (Hershey et al., 2016) which introduced an artificial dataset, typically known as *WSJ0-2mix* in the literature. The dataset consists of mixing together pairs of utterances from the WSJ corpus together through adding the signals, creating highly overlapped mixtures. The dataset became the defacto benchmark for speech separation, but quickly become saturated with the latest techniques able to separate the sources with near perfection (Subakan et al., 2021). Further versions of the dataset with additional speakers are also used to add to the complexity.

For the dataset to be used for multi-channel algorithms, the spatialised version was introduced in (Wang et al., 2018), which used the image method to create artificial RIR by placing speakers randomly inside a room. The same pairs of utterances were used as WSJ0-2Mix to allow for some comparison.

Both versions of WSJ0-2Mix contain no background noise. The WHAM! (Wichern et al., 2019) corpus was created to address this weakness by recording ambient noise in urban areas. WHAM! uses the same mixtures as WSJ0-2mix but with the additional background noise. Later the WHAMR! corpus (Maciejewski et al., 2020) was released which uses the same speaker positioning and mixing as the spatialised version of WSJ0-2Mix, creating a reverberant dataset. However, this time the background noise from WHAM! adding to the complexity. However, due to the background noise being a single channel, the reverberant dataset can only be a single channel.

Libri2Mix and Libri3Mix (Cosentino et al., 2020) were introduced to compliment WSJ0-2mix to offer another commonly used benchmark, with spatialised versions created in related works. When introducing these datasets they showed an improvement could be made by training across corpora.

There has been a clear trend in these corpora to make them more challenging as the state-of-the-art algorithms perfect the separation. The aforementioned datasets have attempted to make challenges more difficult through the addition of more speakers and real background noise. The set of Clarity challenges (Graetzer et al., 2021) has attempted to make the datasets more realistic by using more advanced simulation techniques. The second enhancement challenge increased this complexity with the introduction of more advanced background noise such as music as well as moving microphone receivers. The challenges were created for enhancement for hearing aid users (hence the moving receivers) but it can been used to benchmark speech enhancement in general.

The role of these datasets is to provide a clean target that can be used in supervised neural network training. But potentially more impactful they are used to provide a reference signal used to evaluate the performance of source separation algorithms. The details of these are

discussed in the next sections followed by a discussion of the need for good simulation and the potential pitfalls.

2.7.2 Training

Simulation plays a crucial role in training *supervised* speech enhancement and speech separation techniques as it provides a target for the models to estimate. The targets may not directly be the clean source but instead the mask as discussed previously. In (Heymann et al., 2015), ideal binary masks and ideal ratio masks were shown to be effective for the tasks. These oracle masks can be computed as we have access to the mixture and access to the clean target, so the required mask to filter the mixture into the clean signal can be computed. The loss function is then the root mean square error between the target and the estimate.

Instead of masks being explicitly used as the target, many modern techniques use the target signal directly when computing the loss function, using more task-specific objectives e.g., reducing distortions (Luo and Mesgarani, 2019) or maximising the perceptual quality (Martin-Donas et al., 2018).

In (Hershey et al., 2016), instead of a mask being estimated an embedding network is learned. The embedding networks provide a representation for each of the time-frequency bins in the signal such that bins coming from the same source are close to each other in the embedding space. These embeddings are then clustered together to create a set of masks (e.g., using k-mean clustering). Work has been conducted on combining these spectral embedding features with the spatial probabilistic features discussed before (Drude and Haeb-Umbach, 2017).

Finally, simulation can be used to aid the training of the acoustic model in ASR through data augmentation allowing for multi-condition training using simulated data. Instead of training a model to remove the reverberation from a signal, an acoustic model can be trained directly on the reverberant audio and learn the relevant features to focus on (Lucas et al., 1981).

2.7.3 Evaluation

The desired goal of the separation in this context is to improve ASR performance (i.e., reduce WER). However, often it is inconvenient to compute this value as it requires an entire ASR to be trained. Therefore, a proxy to this evaluation metric is used to guide the development of these frontends.

To evaluate the performance of speech separation a dedicated metric is often used. When describing the metrics, vector representations of the signals will be used i.e., $\mathbf{s} \in \mathbb{R}^T$, where

T is the number of samples. The most commonly used one until recently was the signal-to-distortion ratio (SDR) (Févotte et al., 2005) which considers an estimated signal $\hat{\mathbf{s}}$ as the composition of the following parts[2],

$$\hat{\mathbf{s}} = \mathbf{s}_{\text{target}} + \mathbf{e}_{\text{intefer}} + \mathbf{e}_{\text{artifact}}. \tag{2.33}$$

The signal $\mathbf{s}_{\text{target}}$ is an altered version of the desired target signal based on how much of the desired signal is in the estimate given by,

$$\mathbf{s}_{\text{target}} = f(\mathbf{s}) \qquad \text{where} f \in \mathbb{F}. \tag{2.34}$$

The set $\mathbb{F}$ is a user-designed parameter indicating which distortions are allowed to incur to the signal without penalty. The quantity $\mathbf{e}_{\text{interfer}}$ is the signal from other sources (and noise) still remaining in the estimated source and $\mathbf{e}_{\text{artifact}}$ is the remaining unaccounted signal which is not the desired signal or from the other original sources i.e., artifacts created from the signal processing.

Together these quantities can be used to compute the SDR as defined as the following ratio,

$$\text{SDR} = 10\log_{10}\frac{\|\mathbf{s}_{\text{target}}\|^2}{\|\mathbf{e}_{\text{intefer}} + \mathbf{e}_{\text{artifact}}\|^2}. \tag{2.35}$$

One of the underlying assumptions of the metric is that it assumes the residual error i.e., $\hat{\mathbf{s}} - \mathbf{s}_{\text{target}} = \mathbf{e}_{\text{intefer}} + \mathbf{e}_{\text{artifact}}$ is orthogonal to the target signal. However, this is not always the case and simply rescaling the estimated signal will result in a change in SDR performance even though the amount of distortion is effectively unchanged. To make the metric agnostic to the scale of the estimate, the SI-SDR metric (Le Roux et al., 2019) is defined as,

$$\text{SI-SDR} = 10\log_{10}\frac{||\alpha\mathbf{s}_{\text{target}}||^2}{||\alpha\mathbf{s}_{\text{target}} - \hat{\mathbf{s}}||^2} \quad \text{for}\, \alpha = \arg\min_{\alpha}\|\alpha\mathbf{s} - \hat{\mathbf{s}}\|^2, \tag{2.36}$$

where $\alpha = \frac{\hat{\mathbf{s}}^\top \mathbf{s}}{\|\mathbf{s}\|^2}$ i.e., scales the reference signal to be as close to the estimated signal as possible. Later versions of SDR in the *bss_eval* toolkit attempt to address the scaling issue as well time offset between the reference target and the estimated through learning an optimal filter. This allows for a source target signal to be used as a reference instead of a spatialised signal that may not be available. SI-SDR is the preferred metric used in speech separation, however, when spatialised signals are used then SDR has been shown to be the better metric (Drude et al., 2019a).

[2]Sometimes a noise term is considered but this can be considered as part of the interferer

Given these metrics, we can see how simulated data plays a vital role in evaluating the performance of speech separation. In order to compute SDR and SI-SDR as well as many other metrics (Taal et al., 2011) we have to have access to the clean signal before any noise is adding the signal. In the real world is it difficult to have access to that clean signal. If we are recording a distant microphone scenario we may be able to record the clean signal by using a microphone close to the speaker as a proxy for the clean signal. This clean signal may not have many of the environmental effects present e.g., reverberation. However, if we are recording a conversation it is unlikely we will be able to record the signal without the interfering speaker also being captured.

One way to have access to the clean signal and the noisy signal is to have access to the signal that is being generated. This is possible if the real dataset is captured by playing audio through speakers and then recording the playback from a distant microphone. The clean reference signal that is being played from the speakers can be used as the reference. Producing such as dataset will of course be costly and limited in size.

Artificial simulation is the typical way speech separation is evaluated as we have access to the reference signal before mixing. As the simulation is fully controlled it also allows for analysis of the performance with respect to many aspects of the setup e.g., the nature of the speaker, the positioning of the speakers and the room configuration.

2.7.4 Discussion

Now that the role of simulation has been described it is important to discuss the potential impact relying on simulation has on speech separation and distant microphone ASR in general.

Through looking at the brief history of the development of simulated datasets we could see the need to create more and more complexity as the algorithms developed got better. The way complexity has been added does not necessarily reflect the way simulation mismatches with the real data. For example, should developing an algorithm that can separate a large number of speakers be a priority when in reality this will likely rarely occur?

When training models on simulated datasets which are then tested datasets of real signals, it is quick to see that that mismatch results in poor performance. Where this mismatch comes from is not always easy to see, given how many uncontrolled variables are present in real data

What is harder to see, and potentially more troublesome is when the simulation is used for evaluation. A system may perform well on the simulated evaluation data because it is exploiting aspects of the simulation. This problem may be solved by updating the training data to closely match the evaluation data. However, some aspects may be more fundamental

to the nature of the technique. For example, a technique that only works when sources are completely stationary will work in a simulated environment where the sources are stationary, however, when the sources inevitably move in the real data, that technique will fail. This mismatch may result in pursuing algorithms that work in simulated data but will never work in real data.

2.8 Conclusions

In this chapter, an overview of techniques used in distant microphone speech processing has been presented. The techniques discussed have largely focussed on the overlapped speech aspect and how we can separate the sources. Separating sources in distant microphone ASR exploits spectral filtering techniques as well as spatial filtering techniques.

This chapter has shown how simulation plays a crucial role in the development and evaluation of speech separation. Simulation is used to provide the targets in the supervised training of models but also used to provide the reference signal when evaluating the performance of source separation.

The chapter has discussed the aspects of the simulation that can be controlled when designing the scenario that is being simulated, such as the speaker positioning and the amount of overlap. Typically, these values of chosen uniformly random with little motivation. If we are using simulated data to evaluate the performance of these techniques the data must be a fair reflection of what we will expect to see in the real data.

In the following chapters experimental work on providing the metadata for the simulation, driven by analysing the behaviour of people in real scenarios. To be specific the positioning of speakers relative to one another and the amount of overlap in speech will be addressed.

Chapter 3

Data and tools for analysing speaker behaviour

3.1 Introduction

The way in which people interact in social environments is a well-researched field in the social sciences. In Psychology, the field of Proxemics (Hall et al., 1968) studies how the physical space between people varies depending on the type of social discourse. For example, in more intimate settings, conversational partners position themselves closer to each other than compared to when conversing with colleagues. Whilst there is a wealth of research in this field (Norris, 2004), the work typically relies on specialist knowledge using a specialised notation (Hall, 1963) to annotate the behaviour of participants in these studies. If we want to produce simulations with characteristics of realistic speakers, we must first understand the behaviours we want to mimic.

In the previous chapter, we looked at the variables of a simulation that affect the realism of the artificial acoustic environment. The realism of the simulation can be factored into two parts, first how well the simulator reproduces physical phenomena and second the naturalness of the metadata provided to parameterise the simulation, e.g., speaker positioning and turn-taking. Whilst the simulators are informed by physical models, the metadata is typically informed by heuristics which are not often presented in a well-motivated manner. This work, therefore, focuses on how this metadata can be informed in a data-driven manner using data extracted from the behaviour of real people. To accomplish this, we must first have access to a dataset containing natural behaviour; we must then analyse this data to extract the important behavioural information of the speakers.

The structure of the chapter is as follows, first in Section 3.2 a review of the potential speech corpora that can be used for speaker behaviour analysis is presented. In order to achieve this, a set of requirements for the dataset is established. Corpora are then reviewed with respect to these requirements. The result of the review showcases that not one of the potential datasets meets all the requirements. However, in lieu of recording an entirely new dataset, the CHiME-5 (Barker et al., 2018) dataset, a collection of unscripted audio-visual dinner party recordings, is judged to be the most appropriate. Details of the CHiME-5 dataset are presented in Section 3.3. However, its video component lacks the annotations required for analysing speaker behaviour. Therefore, the remainder of the chapter focuses on the CHiME-5 dataset and tools developed to extract speaker positional information. Methods to extract positional information from the videos are then established in Section 3.4. This is achieved through automatic people detection methods as well as through novel tools developed to produce annotations. Evaluation of these tools is then presented in Section 3.5. This extracted data provides the basis for the work in Chapter 4 and Chapter 5 using single and multiple devices respectively.

3.2 Corpora

The size of the models underlying modern automatic speech recognition (ASR) systems have grown incredibly in recent years, such that current state-of-the-art systems require millions of parameters to be trained. Adequately fitting these parameters can require 100s or even 1000s of hours of speech as training data. This has led to the development of many large-scale corpora such as LibriSpeech (Panayotov et al., 2015), CommonVoice (Ardila et al., 2019) and Gigaspeech (Chen et al., 2021). These corpora allow for large continuous speech recognition systems to be trained and to be robust to many variations in speech quality.

However, these datasets consist of recordings of segmented speech of mostly individual speakers. This neglects many crucial difficulties that occur when we place a microphone at a distance to transcribe conversations. In the real world, the people talking will walk around their environment, this changes the characteristics of the acoustics of the signal such as direct-to-reverberant energy ratio (DRR) and signal-to-noise ratio (SNR) when we take into account an interfering speaker. Spatial cues are typically exploited when using multi-channel recording devices to enhance signals based on their source location. It is therefore more challenging for these enhancement systems when the signals come from the same direction. In addition to this, people do not take clean turns when having a conversation, speech is often filled with backchannels and interruptions.

Therefore to transcribe long-form recordings a system needs to be developed to break the long recordings into speech segments to be fed into a recognition system. This is a non-trivial problem which involves determining who is speaking and when a task known as diarisation. In addition to this, there may be multiple people speaking at once in order to separate these speech sources into separate streams the known number of speakers is required for the prominently used separation techniques (Ito et al., 2016; Luo et al., 2020; Luo and Mesgarani, 2019).

The task of transcribing long-form recordings offers many real-world challenges, but also many real-world opportunities for solutions to the problem. In order to enhance a speech signal it is often required to build a background model of the noise. Having a large context window (Kanda et al., 2019) can be beneficial for modelling this background. Furthermore, steering a multi-channel enhancement system towards a target speaker requires knowing the location of the speaker either through explicit steering vectors or implicitly with spatial models of the target speech, both of which requires observing enough of the target speech to estimate these values. However, with a long-form party, prior information can be used to establish the speaker's location based on where they were previously observed.

This thesis aims to provide methods for simulating such distant microphone setups driven by observing *real-life* recordings. This allows for state-of-the-art systems to be evaluated in setups more closely matched with what they will face in deployment which will provide more reliable results and avoid over-emphasising techniques exploiting unrealistic setups. A realistic simulation also provides the opportunity to provide a controllable setup to explore techniques to exploit the characteristics that do appear in real data.

3.2.1 Corpora requirements

Establishing a set of requirements allows for the possible datasets to be compared objectively with respect to their suitability for this research. This work aims to produce simulations that more realistically mimic how people behave in the real world when distant microphones are used to observe social interactions in environments. Therefore the behaviour of people needs to be recorded in a similar setting in a scenario with few constraints on how the participants should behave. There is a trade-off between the number of constraints placed on participants in such recordings setups (McGrath and Hollingshead, 1994), placement of too many constraints results in data not representative of the real scenario, and placing too few constraints leads to too many variable factors making analysis and interpretations of results difficult. The datasets will be evaluated with respect to the *situation setup*, that is how closely it will match a typical distant microphone scenario. The *task* the dataset is designed for and what applications it can be amenable to e.g., ASR, diarization, separation etc. Finally with

respect to the dataset's *tracking capability* of the participants within the recordings in order to capture speaker behaviour.

Situation Setup The speech should be elicited from people having real conversations in a natural recording setup. Actors playing roles produce different *turn-taking patterns* to what we would expect from a real conversation. In real data this is especially the case if the person talking is conversing with someone familiar. For example, friends are more likely to interrupt each other and finish each other's sentences. The formality of the setting also plays a role in this behaviour, a formal meeting may contain lots of well-articulated utterances that can be very long e.g., a presentation. Contrast this with a heated conversation over dinner or during a board game where turns will not be clearly taken and words may not be fully completed.

Task The dataset should consist of recordings from long-form conversations. Therefore, the dataset should be amenable to the task of ASR and diarization. The dataset does not have to be fully transcribed, knowing that speech is taking place is sufficient. However, a fully-transcribed dataset is beneficial as the end goal of this work is to produce simulations to benefit speech recognition research and being able to benchmark the source material for the analysis is a nice feature. Lower-level transcriptions such as word-level alignment and phone-level are not required. The dataset being amenable for speech separation research is also a nice feature but is not required. For real recordings worn microphones can be used as a proxy for the clean signal and the distant microphone as the noisy input needing enhancement.

Tracking capability The final key component of the dataset is a method for understanding speaker positioning. The detail of speaker tracking can be on several levels. Given microphone arrays in distant speech recognition typically have microphones placed along a horizontal plane, the position of speakers along the azimuth angle of the array is the lowest level of useful positional information of the speaker. Tracking the 2-D position (top-down view) of the speaker is a level above this i.e., the angle plus the distance away from the microphone. This provides information about the distance speakers are away from the microphone as well as their distance to a competing speaker relative to the microphone. Tracking the direction people are facing can provide further information on the acoustics e.g., facing towards the microphone or away. Simulating directivity patterns is not possible in many simulation packages freely available (and commonly used), but is available in some commercial software (Schröder and Vorländer, 2011). Finally, full 3-D position information

of speakers including their skeletal posture would provide the most detailed description of their position, however, modelling this in simulation is beyond the scope of this work.

3.2.2 Review of potential speech corpora

In this section, a review of potential speech corpora will be conducted. Datasets that capture recordings of real environments are considered. The related literature in distant microphone speech recognition can be split into two categories. First, a 'noise robustness' perspective where the problems related to processing speech signals in the presence of additive noise and reverberant distortions. The second perspective is 'multi-party interaction' where the difficulty comes from recognising speech from people having conversations.

Whilst multi-party corpora often have some noise robustness aspect e.g., a distant microphone capturing the recordings of the speakers, however, difficulty also comes from the fact multiple people are talking which could contain overlapping speech in a difficult-to-predict manner. The multi-party interaction corpora have largely focused on the 'meeting room' problem involving the task of diarization prior to speech recognition. On the other hand, the noise-robust community has largely focused on recording challenging realistic backgrounds and room impulse responses to artificially create mixtures. Simulation is key in noise robust speech recognition to evaluate pre-processing tasks such as dereverbaration, speech enhancement and speech separation. A timeline of influential datasets across these communities is depicted in Figure 3.1. The figure showcases corpora which contain real data as part of the simulation or in its entirety. Not all the datasets listed are distant microphones, but their development has led to progress in related work in distant microphones. These datasets are often released in the context of a challenge providing the opportunity to benchmark the performance of speech-processing techniques, this is also depicted in the diagram.

Noise Robust Speech Processing. The robust speech processing community is focused on developing signal processing techniques to remove noise created by background noise and reverberation, as well as developing speech recognition systems robust to variations caused by these distortions. A robust speech recognition system will contain several of the following, a frontend enhancement system, robust feature transforms, robust features and multi-conditioning training. Early work in robust speech recognition treated the enhancement tasks independently with their own evaluation metrics (Kinoshita et al., 2013; Vincent et al., 2006). Each of the individual evaluation metrics does not necessarily correspond with eventual WERs (Iwamoto et al., 2022).

Datasets for noise robust research largely focus on creating artificial mixtures which involve recording background noise in real environments and then additively mixing this

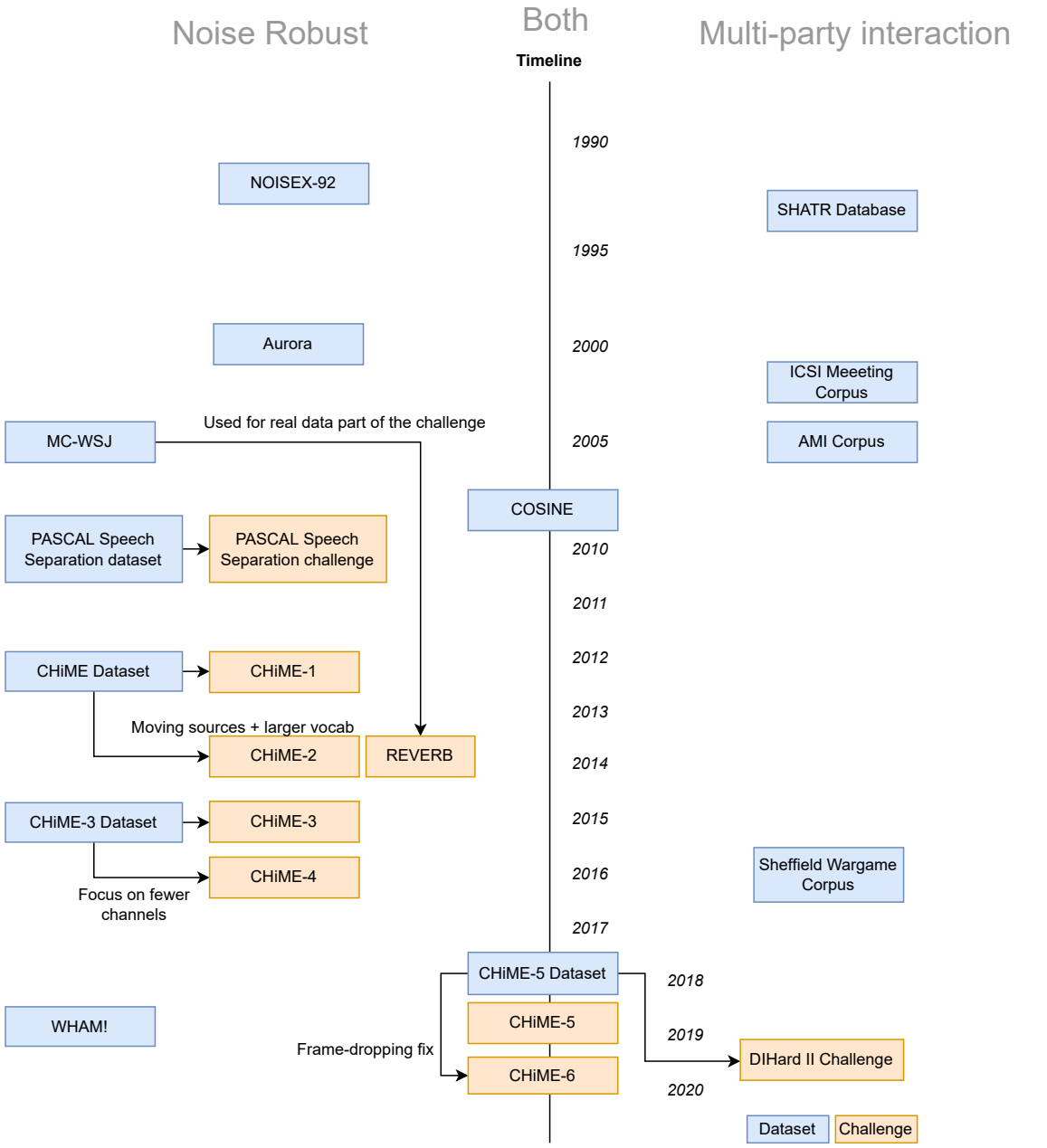

Fig. 3.1 Depictions of influential datasets in noise robust speech processing and multi-party interaction. The diagram illustrates how datasets are often been driven by challenges to benchmark their performance. Often multiple challenges are associated with a single dataset.

noise into clean close-talking microphones. The first known dataset to do this was NOISEX-92 (Varga and Steeneken, 1993), which created samples with backgrounds recorded in military-related environments e.g., 'helicopter', 'machine gun', which the task of recognising continuous string of digits. Following a similar scheme, the AURORA (Nakamura et al., 2003) dataset contains spoken digits with separately recorded backgrounds added to the mixtures. This dataset is larger with a greater variety in the number of speakers. Recording speech and then adding the noise into the mixture negates the fact that people change how they speak in the presence of noise (Zollinger and Brumm, 2011). More recently, The MC-WSJ (Lincoln et al., 2005) records speech in a real environment of overlapping speech. In the dataset, people are seated in a defined location around a meeting room table and told to read utterances from the Wall Street Journal corpus (Paul and Baker, 1992). This dataset has been used for the REVERB challenge (Kinoshita et al., 2013) to benchmark dereverberation techniques on real data.

The series of CHiME challenges have looked to motivate research in robust speech recognition by combining signal processing and statistical modelling communities.

The first CHiME challenge (Barker et al., 2013) used the CHiME dataset (Christensen et al., 2010), extending foundational work using single-channel (Cooke et al., 2010). The data contains background recordings captured inside a real home environment which contains many different sound sources such as noise from children playing and the television. In addition to this binaural room impulse responses (RIRs) are measured inside the room capturing the audio. Artificial mixtures were then created by mixing utterances from the GRID corpus (Cooke et al., 2006) convolved with the RIR and adding the background recordings.

CHiME-2 (Vincent et al., 2013) addresses the limitations of the first challenge by using the same dataset but creating a larger vocabulary task by using the Wall Street Journal corpus (WSJ) as source material. The RIRs were also extended to be time-varying by interpolating a grid of measured RIRs to simulate the movement of speakers across a distance of around 5 cm. The results from the challenge concluded that this movement added into the simulation had little impact on the recognition performance.

The third CHiME challenge (Barker et al., 2015) was released alongside a new dataset. A tablet with 6 microphones attached to the device along with a close-talking microphone was used to record prompted read speech in real environments. The impulse responses were estimated as an optimal filter in terms of least squares using the close-talking microphone as a proxy for the clean signal. The motion of the speaker was tracked using SRP-PHAT which was used to create a time-varying filter. The combination of impulse response and time-varying filter was used to simulate clean utterances in the environment. For background

noise, recordings without speech were added to the simulated datasets. Therefore, the dataset consists of real data with matched simulated data. Analysis of the challenge concluded that simulation overestimated the usefulness of beamforming approaches. The spatial statistics required for these methods could not be accurately estimated using the approaches that worked in simulation. Further analysis was more critical towards the minimum variance distortionless response (MVDR) beamformer, however, the largest impact on performance was due to the number of microphones. This resulted in a fourth challenge limiting the number of microphones to only two (Vincent et al., 2017). As of writing the final dataset for the CHiME series is from CHiME-5 (Barker et al., 2018) which was used in the CHiME-5 challenge and the CHiME-6 challenge (Watanabe et al., 2020). This involves distance microphone recordings of unscripted dinner parties consisting of a variety of domestic noise from the homes of real people, more details will follow in this chapter.

Multi-party interaction On the other hand, the literature on multi-party interaction has largely focused on the 'meeting room' problem. Meeting room datasets are created either through recording real meetings or eliciting meetings through participants playing roles. Whilst noise is present in these settings (e.g., air conditioning and computer fans), the noise is relatively less dominant and varied compared to environments investigated in noise robust. The field of the literature arguably starts with the ShATR database[1] which is the first to incorporate multi-speaker interactions through eliciting conversations between five people seated around a table solving crossword puzzles.

The largest and most impactful of these datasets is the AMI Corpus (McCowan et al., 2005), which has been influential in the field of distant microphone speech recognition spanning eight years of development. The AMI dataset was the result of a project which aimed to give insight into human interaction during *meetings*. Distant microphone audio recordings, as well as videos, were captured using separate device recordings at several research sites using "instrumented meeting rooms". In addition to this, other accompanying files such as notes and slides are also released. The dataset is fairly large, with 100 hours of meetings recorded containing a varying number of participants. In the AMI corpus, the meetings are not scripted. However, participants are given roles to play within a fictional company and given a scenario. A subset of the dataset does come from real meetings that are not elicited. The corpus consists of recordings of meetings across several locations consisting of mostly non-native English speech. The high number of non-native speakers is an interesting ASR problem to solve but contributes to a further complexity that needs to be addressed when developing speech recognition systems and analysing results.

[1] https://spandh.dcs.shef.ac.uk/projects/shatrweb/papers/ioa94.html

The Sheffield Wargames Corpus (SWC) (Fox et al., 2013) was developed to address some of the weaknesses of the AMI dataset by using games of Warhammer (a table-top strategy game) as a surrogate to a meeting. They argue that playing Warhammer elicits similar behaviours that are observed in meetings. AMI uses volunteers playing roles who do not know each other, which arguably does not result in the same behaviour we would expect in real meetings containing people who do know each other. Like AMI, SWC also provides multiple modalities such as audio and video captured on different devices. SWC also provides positional tracking of the participants using Ubisense tracking devices. However, SWC has some major drawbacks, such as only using one small room to record the data. The small room also contains a large table to assist in their game, further limiting the movement of the participants. The constrained task of Warhammer also limits the variety in speaker behaviour across the possible space of human social behaviour. In addition to this, the dataset consists of entirely male speakers in the initial recording, which limits the dataset variety in spectral properties and behavioural properties. Later recording sessions (Liu et al., 2016) of the dataset address this with a day of a recording involving female participants. However, this was addressed through a setup where male players taught females how to play the game, creating a different dynamic from the rest of the dataset.

The CHiME-5 dataset was developed and released alongside the CHiME-5 (Barker et al., 2018) challenge and was also used in the subsequent CHiME-6 challenge (Watanabe et al., 2020). The dataset consists of recordings of "Dinner parties" inside real homes. These dinner parties are unscripted and the topics can be freely chosen. The only requirement was that a party contained three stages. A cooking stage where participants prepared their meals, a dining stage and some form of after-dinner socialising. A party is recorded using multiple Kinect v2 devices which have a 4-channel microphone array and an integrated 1080p camera on the unit. Six devices are placed around the apartments with at least two devices in each of the rooms. Given that these are all real homes, the location of the activities may be in the same room e.g., some households may dine and socialise in the same room. In total there are 20 parties with a total of 50 hours of speech.

Through surveying the literature, three potential datasets have been identified that could meet or partially meet the requirements established in the previous section, namely AMI (McCowan et al., 2005), Sheffield Wargames Corpus (Fox et al., 2013) and CHiME-5. Whilst other potential meeting corpora could be used to evaluate speech recognisers such as the ICSI Meeting corpus (Janin et al., 2003) or more conversational datasets such as COSINE (Stupakov et al., 2009). These datasets have been immediately ruled out as they do not provide additional modalities to extract positional information.

Table 3.1 Comparison of different datasets of conversational speech.

Dataset	Pros	Cons
AMI	• Large dataset (100 hours) • Audio and video data recorded • Well established in the research community • Unscripted speech • Multiple locations	• Formal meeting scenario • Participants acting roles • No position tracking provided
SWC	• Position tracking information provided • Unscripted speech • Natural behaviour (not acting)	• Limited to game playing scenario • Limited to a single room • Small dataset (24 hours) • Female participants retrofitted into the dataset
CHiME-5	• Large dataset (50 hours) • Unscripted speech • Multiple Locations • Natural behaviour (not acting) • Social setting • Audio and video data recorded	• No position tracking provided

From the potential datasets available, the most appropriate for this work is CHiME-5. This is because it directly uses data recorded inside a variety of domestic environments in unscripted scenarios. Although positional information is not provided like with SWC, it does provide multi-camera recordings, which can be used to locate participants. A breakdown of the pros and cons of the datasets is given in Table 3.1.

Given that the CHiME-5 does not provide any tracking system data but does provide video, techniques to extract speaker location needs to be explored and developed. The remainder of this chapter provides a deep analysis of the CHiME-5 dataset and the tools used to explore the video data.

3.3 CHiME-5 dataset

Given it has been established that CHiME-5 is the most appropriate dataset to conduct this research, the remainder of this chapter conducts an analysis of the dataset and its impact on the research community. The CHiME-5 data was released in 2018 alongside the CHiME-5 challenge.

3.3.1 Overview

The CHiME-5 dataset consists of *twenty* recordings of dinner parties inside real homes, each party having *four* participants. There are 48 participants in total, 23 of which are female and the remaining 25 are male. An attempt was made to have an even balance of male and female participants in each of the parties, however, for only 10 out of the 20 parties this is the case[2]. A dinner party consists of three phases, cooking, dining and after-dinner socialising. Each of these phases typically took place in different rooms of the home. The dinner parties are captured using multiple devices. In each of the rooms of the parties, two Microsoft V2 Kinects are placed at the edge of the rooms. A Kinect consists of a 4-channel microphone array with an integrated camera into the unit, as illustrated in Figure 3.2. The devices provide audio recordings from each of the microphones sampled synchronised between microphones on a device. However, recordings are not necessarily synchronised between different devices. The devices also provide 1080p video recordings from their cameras.

As well as the distant Kinects, near-field microphones also capture the session through in-ear devices (OKM Soundman II) worn by each of the participants. The in-ear microphones can be used in training components of an ASR system such as the acoustic model or speech enhancement system. However, the systems are evaluated on their performance on the Kinect

[2]For a breakdown see: https://spandh.dcs.shef.ac.uk/chime_challenge/CHiME5/data.html

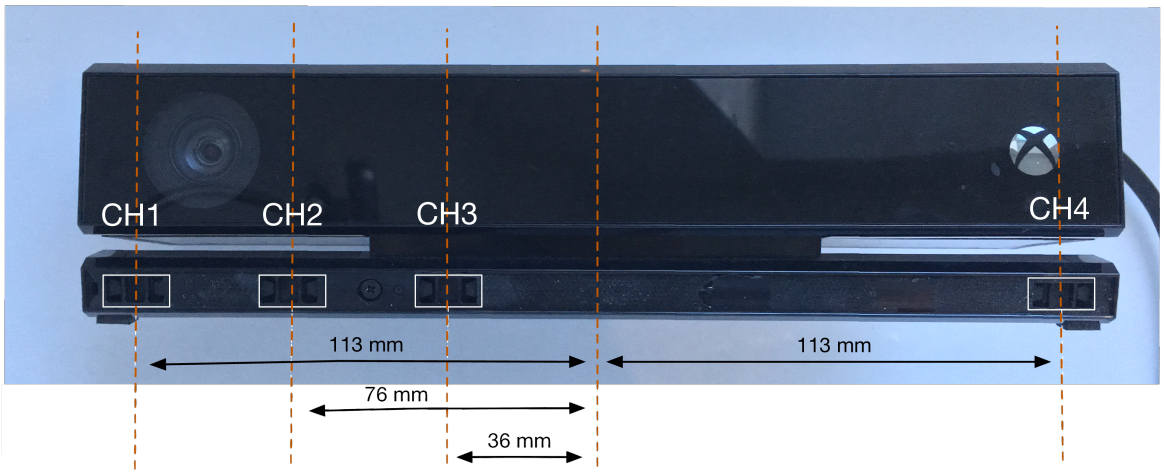

Fig. 3.2 Diagram showing the layout of the Microsoft Kinect v2 device. The device contains a 4-channel linear microphone array with an integrated 1080p camera. Used with permission from The University of Sheffield http://spandh.dcs.shef.ac.uk/chime_challenge/CHiME5/overview.html

devices. The primary motivation for the in-ear microphones was for aiding the transcription of the dataset and therefore it was not a priority for this data to be a clean reference, only an improvement over the distant microphones. Whilst the in-ear microphones may have a higher SNR they are not clean signals. The in-ear microphones often pick up more cross-talk due to their location and omnidirectional nature compared with traditional close-talking microphones located near the speaker's mouth.

The structure and layout of each of the apartments vary greatly and give a good representation of what we would expect when deployed in the real world. In Figure. 3.3 we can see four examples of the floorplans. In each of the apartments, six recording devices (Microsoft Kinect v2) are placed around the apartments, with at least two devices in each of the rooms. The figure demonstrates that the definition of a 'room' can be blurry, for example, in S12, the entire apartment is open plan with no walls separating each of the rooms. In contrast, if we look at S18, we can see a clearly separate kitchen area in the apartment. In more open-plan apartments, devices placed to listen in one room will also be able to capture audio from the other rooms. Figure. 3.4 shows how the separation of the speakers can depend largely on the layout of the room.

3.3.2 Impact of CHiME-5

The dataset has been used as the basis for open automatic speech recognition evaluations. The first of these, the CHiME-5 challenge (Barker et al., 2018) was launched with a baseline

Fig. 3.3 Example of the floorplans of four apartments in the CHiME-5 dataset. The figure demonstrates the variety of room layouts. Some are open-plan like S12 and others have more distinct rooms like S18.

Fig. 3.4 The left-hand side shows participants being narrowly separated whilst the right-hand side shows widely separated participants. Faces have been manually blurred to protect the privacy of the participants.

system with a performance of a WER of 73.3% on the eval dataset; this was later reduced to 41.6% (Kanda et al., 2019) after the challenge. Following this, the CHiME-6 challenge was released with the same dataset with official procedures to address frame-dropping and array de-synchronisation issues that arose while recording the dataset. The challenge also expanded the task to also include diarisation, i.e., the task of figuring out who is speaking and when. The result of this challenge was a state-of-the-art performance of 31.0% (Du et al., 2020a). Throughout the thesis, when referring to the CHiME-5 dataset, this will refer to the dataset created *including* the procedures provided in the CHiME-6 challenge to address recording issues.

Since the release of the CHiME-5 dataset, the corpus has led to many advancements in the field of ASR and speech enhancement. The challenges themselves have drawn in many participants, with the WERs significantly reduced over time. One of the most impactful contributions comes from research within Guided Source Separation (GSS) (Boeddeker et al., 2018) which is a powerful unsupervised technique for separating speakers in a mixture. More details on this approach are presented in Chapter 3. In addition to this, advancements in voice activity detection (VAD) in this domain with target-speaker VAD (TS-VAD) were presented in (Du et al., 2020a). CHiME-5 has also been used to develop simulated corpora, this uses CHiME-5 background noise along with signals spatialised using RIRs with array geometries matching the Kinects (Sivasankaran et al., 2021). The outcomes of this corpus showcased previous state-of-the-art techniques fail to work, such as the delay-and-sum based beamformer, BeamformIt (Anguera et al., 2007) which finds the optimal path of delays using GCC-PHAT (Knapp and Carter, 1976) and dynamic programming. Beamformit provided large improvements in the AMI corpus. When used in CHiME-5 the beamformer provided minimal improvement to the recognition results, simply choosing the best channel provided better recognition results (Barker et al., 2018). The reason for this difference is not immediately clear and given the nature of the dataset, it does not provide easy analysis due to the number of variables that can not be controlled when observing real behaviour. Providing a method to approximate the complexity of CHiME-5 across multiple dimensions e.g., noise, room geometry, speaking styles, speaker positioning, speaker turn-taking etc. lends itself to providing insight into why CHiME-5 is so difficult as we can peel back the layers of complexity.

3.4 Tools created for analysis

3.4.1 General strategy

In the previous section, we have established that the CHiME-5 dataset is the most suitable corpus to analyse the behaviour of people in a social setting. As stated, the corpus contains video recordings of the participants which will be used to locate the speakers. Therefore, it is left to decide how this speaker information can be extracted from the videos. In this following section, the tools used to extract speaker information from the raw videos are described and evaluated. The end goal of these tools is to locate the mouth position of each of the speakers i.e., the source of the speech signal. The tools were developed from scratch in order to meet the objectives of the thesis and the tools have now been realised to aid in reproducing this work or annotating similar datasets. The code alongside installation instructions can be found in the GitHub repository[3].

Given there is a large amount of data that needs to be analysed in the corpus, it is not feasible to accurately hand-annotate the entire dataset in a reasonably efficient manner. Therefore, the first automatic tools to locate people in videos are explored, which are used across a variety of different domains. Novel annotation tools are also developed which have two roles, first to establish the error in the automatic tools, and second to provide their own raw data. Both the data generated from the automatic tools and hand-annotated tools are used in Chapter 4 and Chapter 5 when exploring the impact of using this analysis in simulation.

Manual annotation tools To trade-off between fast annotations and highly accurate annotation tools, two different sets of tools are developed: highly accurate annotations allow for automated tools to be evaluated whilst the faster tool allows more data to be annotated compromising on some accuracy.

Automatic tools Another trade-off needs to be made with the automatic tools. In the section on automatic tools we will explore different approaches namely *face* detection and *pose* estimation.

First, a description of the annotation and automated approaches are described, followed by the methodology to extract the mouth position of the speakers. These mouth positions are then evaluated with respect to the most accurate isolated frame annotation tool.

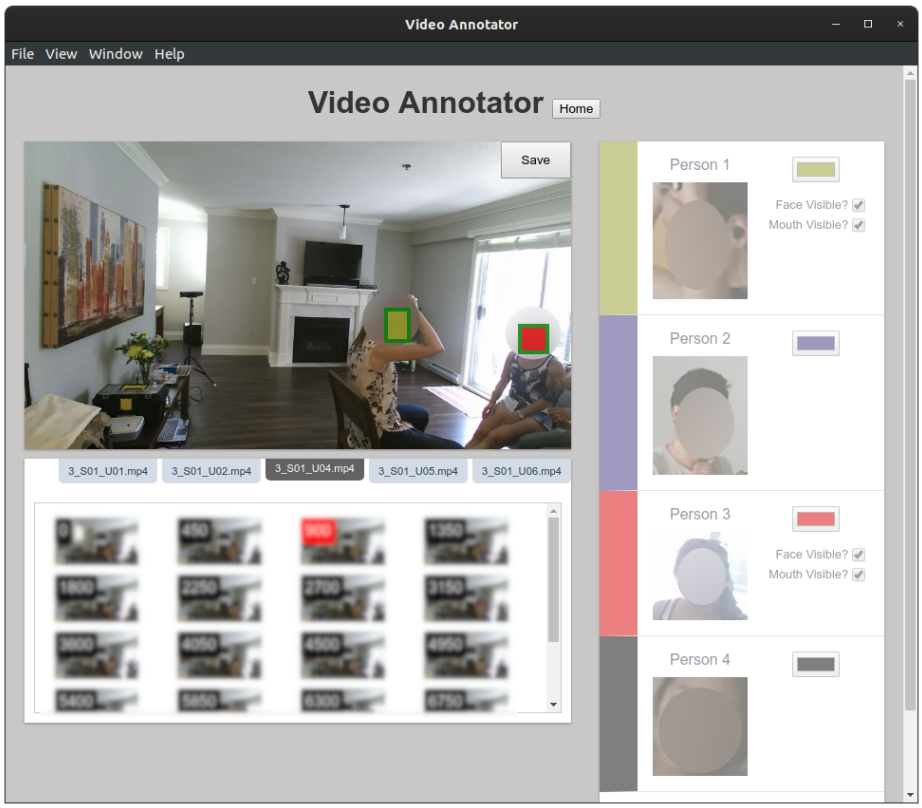

Fig. 3.5 Example screenshot is taken from the isolated frame annotation tool. The faces of the participants have been blurred to protect their privacy. When annotating the data, faces were visible.

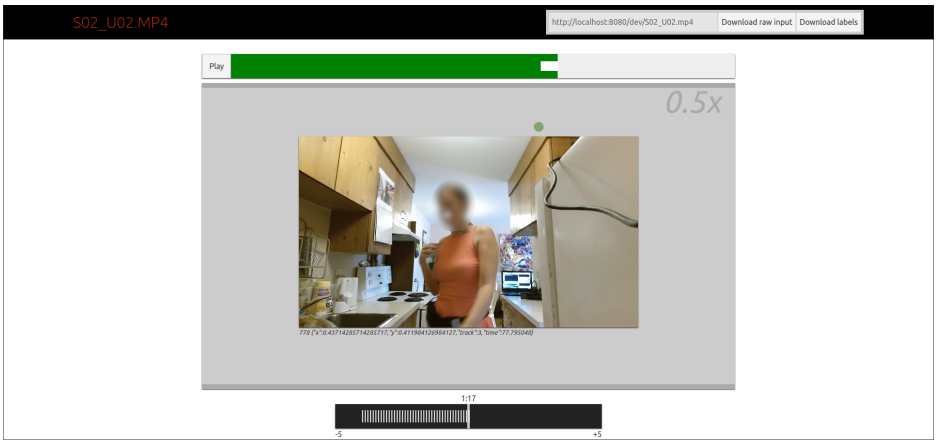

Fig. 3.6 Screenshot is taken from the real-time annotation tool. The faces of the participants have been blurred to protect their privacy. When annotating the data, faces were visible.

3.4.2 Isolated-frame annotation tool

The *isolated-frame* annotation tool aims to provide the most accurate estimations of the mouth position of the speaker. This tool is developed to work on a set of images rather than videos, therefore frames from the videos are first extracted before using the tool. In Figure. 3.5 an example of annotating a frame of a video is shown. When annotating a person a bounding box around the head of the person is first created by the person annotating. Then from within this bounding box a point can be placed indicating where the mouth is positioned.

The annotation tools work by annotating *projects*, a project is created for each of the sessions annotated. Within a project, multiple frames from a video can be loaded, in addition to multiple camera perspectives. In the figure, on the right-hand side, the list of speakers is displayed. The annotator selects the speaker they want to annotate before they create their bounding box. The speaker they select is consistent across the frames and cameras in the project, this allows for there to be an association between frames and across devices. These associations can later be linked to a speaker in the audio data by inspecting the transcript and video.

Isolated frames were annotated by sampling every five minutes in all the videos in the development set and then annotating a bounding box around the speaker's face as well as annotating the position of the mouth.

3.4.3 Real-time annotation tool

The *real-time* annotation tool works directly with the video data as opposed to extracted frames. The goal of this tool is to provide a larger amount of data whilst compromising on the accuracy of locating the mouth position. In the *isolated-frame* tool, box a bounding box was annotated as well as the mouth position. In the real-time tool, only the mouth position is annotated.

The real-time annotation tool works by the annotator playing the video and clicking and dragging on the person as they move across the screen. In addition to this a tracking point can be placed and by using the Lucas–Kanade method of optical flow (Lucas et al., 1981), further annotations can be automatically created automatically, guided by the annotator. The tracking point allows for annotations to be created which react quickly to sudden movements from the speakers. In cases where the optical flow cannot easily track someone e.g., occlusions, the human annotator can correct the errors made and replace the tracking point when appropriate. Due to the real-time nature of the tool, it is not always possible for the tool to keep up with the video being annotated, in this case, frames are dropped and the gaps in between the frames are linearly interpolated. An example of using the real-time annotation tool is shown in Figure 3.6, where a single speaker is being annotated in the centre of the frame. There is a trade-off between using the speed of the real-time tool and the accuracy of the isolated-frame tool. An example of comparing annotated frames using the two tools is shown in Figure 3.7. Here we can see the real-time tool slightly missing the mouth position of the speaker when we compare it to the accurate isolated-frame tool. Later, this difference in accuracy with be evaluated.

3.4.4 Face detection automatic tool

Automatic tools allow for a far greater amount of data to be annotated. The first automatic tool to be explored is *face-detection*. The reason for using a face-detection approach is that they are efficient and provide a clear way of estimating the mouth position.

The task of face-detection is well-established in the literature. Initial approaches (Kumar et al., 2019) used a classifier with a sliding window to determine regions which contained faces, these regions are then combined to find the location of the faces. Modern approaches use deep learning techniques that can perform the entire inference in a single pass (Redmon et al., 2016).

The open source toolkit Dlib (King, 2009) is used, which is well-established and provides many tools for Computer Vision tasks. In particular, the toolkit has a Convolutional Neural

[3]https://github.com/jackdeadman/video-annotation-tools

Fig. 3.7 The annotations on the left show bounding boxes and mouth positions annotated from the isolated frame tool. The annotations on the right show the corresponding frames in the real-time annotation tool. In the real-time tool, only mouth positions are annotated. The faces of the participants have been blurred to protect their privacy. When annotating the data, faces were visible.

Network (CNN) based face-detection system. Specifically, the tool being used in this work is the *cnn_face_detection_model_v1*[4], this model is trained on a dataset of faces created from a subset of ImageNet, AFLW, Pascal VOC, the VGG dataset, WIDER, and face scrub[5]. The model is trained by optimising the maximum-margin optimisation function. During inference the images are upsampled to be twice as large, this was found to improve the number of detections significantly. Even though no information is being added when upsampling it was found faces that were far away and therefore smaller are missed without this upsampling step. The pose detection system is run on all the videos in CHiME-5.

3.4.5 Pose estimation automatic tool

The final tool used in this work is for pose estimation, in particular, the open source toolkit *OpenPose* (Cao et al., 2019) is used to find the poses of the people in the videos. Again, this is a well-established toolkit. Estimating the pose of a person aims to locate keypoints of a human skeleton, for example, arms, legs and neck. Estimating the entire skeleton of a person may seem excessive as we are only really interested in the mouth position. However, this approach has the benefit of being able to detect people when they are facing away from the camera, in the face detection case, no face would be found, however, a pose could be still estimated. The pose detection system is run on all the videos in CHiME-5.

3.5 Evaluation

Now that both the automatic detection and annotation tools have been described, the next section looks at establishing the accuracy and therefore the confidence we expect to have in the tools with respect to the CHiME-5 dataset. It is important to establish this confidence as the speakers inside of the CHiME-5 dataset is very different to the datasets the automatic methods are normally evaluated on i.e., often the cameras in CHiME-5 have occlusions. Therefore in the following section a methodology for this evaluation is established and then the metrics that will be used are defined, this leads to the next section where the results are presented with respect to this methodology.

[4]https://github.com/davisking/dlib/blob/8d4df7c0b3fa7c4c1e4175951161b01ccf4541b5/tools/python/src/cnn_face_detector.cpp

[5]https://github.com/davisking/dlib-models

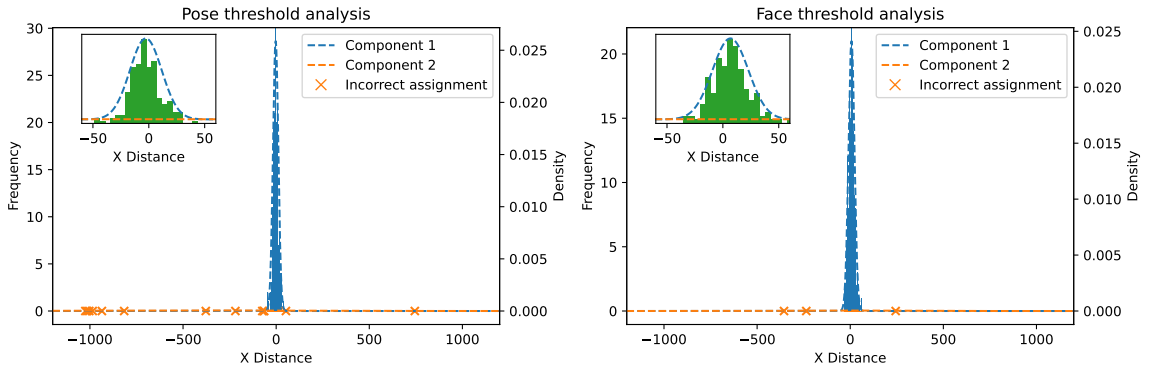

Fig. 3.8 Automatic detections were paired with the annotated data through minimising the total euclidean distance between annotated position and detected position. Then a two-component Gaussian mixture model is fit on the paired data to find the threshold values for pose and face detection methods to determine if a detected point is close enough to be considered correctly detecting the paired person. Using these models a threshold of 53 was chosen for the pose system and 64 was chosen for face. The crosses inside of the plot indicate misclassified detections, which shows the face detection produces far fewer mistakes.

3.5.1 Methodology

Given these automatic methods provide different estimates of speakers i.e, a face bounding box vs a skeleton, we eventually want to estimate a mouth position. This mouth position needs to be assigned to a labelled speaker in order to determine the accuracy of the detection system.

For the pose-detection system, the mouth position is simply chosen to be the *nose* keypoint as this is the closest to the mouth that OpenPose provides. To estimate the mouth position of the speaker from the *face* detection tool we can use the centre of the box for the x-axis position of the mouth. For the y-axis 74% down the face is chosen, this minimises the distance from the labelled data.

The assignment of the *isolated-frame* annotations and the detections are chosen by looking at all the possible permutations and choosing the pairing of detections and annotations which miminises the total euclidean distance.

Given the assignments determined from the previous method, it is left to decide if the two points are close enough to be considered detecting the annotated person or whether it is the result of a false positive and a false negative. In order to determine if an assignment is a detection correctly assigned to an annotation a threshold value needs to be determined. This threshold is estimated through fitting a two-component Gaussian Mixture Model (GMM) on the paired data as depicted in Figure 3.8 where *pose* assignments are considered correct if they are within an x pixel distance of 53 and for *face* this value is 64.

3.5.2 Evaluation metrics

Detection Using the assignments and the thresholds defined in the methodology section, the following standard metrics can be defined to evaluate the *detection* capability of the automatic methods.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}, \tag{3.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}, \tag{3.2}$$

$$\text{F-Score} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} = \frac{2\times\text{TP}}{2\times\text{TP}+\text{FP}+\text{FN}}, \tag{3.3}$$

where TP, FN, FP are the total True Positives, False Negatives and False Positives respectively.

Accuracy To determine the accuracy the distance a speaker is away from the detection is used. In particular, the accuracy of the x-Distance, y-Distance and Euclidean distances are of interest,

$$x\text{-Offset} = a_x - b_x, \tag{3.4}$$

$$x\text{-Distance} = |a_x - b_x|, \tag{3.5}$$

$$y\text{-Distance} = |a_y - b_y|, \tag{3.6}$$

$$\text{Euclidean} = \|\mathbf{a} - \mathbf{b}\|, \tag{3.7}$$

where $\mathbf{a} = \begin{bmatrix} a_x & a_y \end{bmatrix}^\top$ is the *annotated* position and $\mathbf{b} = \begin{bmatrix} b_x & b_y \end{bmatrix}^\top$ is the *detected* position of the automated method. The absolute value and the euclidean norm are given by $|\cdot|$ and $\|\cdot\|$ respectively.

3.6 Results

3.6.1 Automatic detection results

The results of the automatic methods are shown in Table 3.2. The face-detection system finds fewer people than pose as it fails when people are facing away from the camera. However, high precision means we can be fairly confident that it is finding true faces. Pose complements these errors as it misses fewer people but yields more false positives. Note, a low recall will

Table 3.2 Results are shown from eight devices in two different sessions after 558 faces have been hand-annotated and paired with detections by automatic methods. Video resolution: 1920×1080. Accuracies are mean $\pm$ standard error.

	Detection			**Accuracy (px)**		
	Precision	Recall	F-Score	*x*-Distance	*y*-Distance	Euclidean
Face	98.7%	36.6%	52.5%	$23_{\pm 2}$	$18_{\pm 1}$	$32_{\pm 2}$
Pose	94.1%	60.5%	72.9%	$24_{\pm 3}$	$27_{\pm 2}$	$40_{\pm 4}$

Table 3.3 The results of re-annotating a segment in a session in the CHiME-5 dataset. The distances are in pixels and are the result of annotating 1080p videos (1920×1080). Showing the mean and standard deviation (std) difference between the two annotation runs.

Measure	Mean	Std
x-Offset	-3	21
x-Distance	15	15
Euclidean	19	17

not hinder the separation analysis that follows as long as the position of persons missed is at random with respect to screen position. Next, we look at the accuracy of these detections by measuring the horizontal and vertical distance in pixels to the mouth, *x*-Distance and *y*-Distance (Table 3.2, rhs). (Note, in one device an oddly placed mirror mislead the detection systems leading to many large unrepresentative errors. This effect was not seen in any of the other 113 cameras and so was treated as an outlier and the device was removed from the evaluation).

3.6.2 Re-annotation accuracy of the real-time tool

There is a trade-off between the speed of the annotations and the accuracy that they provide with the real-time tool. When not using the tracking feature, the annotated position of the speaker can be slightly incorrect as the human annotator has to react to the sudden change if the person changes their direction and therefore some error can be introduced.

The error in the real-time tool can be measured by re-annotating the same data twice and comparing the differences between the two. This process was performed for a 5-minute segment in all the devices in one of the sessions in the dataset. The difference in the re-annotations is shown in Table. 3.3. Given that the videos have 1920 pixels horizontally and 1080 pixels vertically, the error is very small. The table shows that, on average, the number

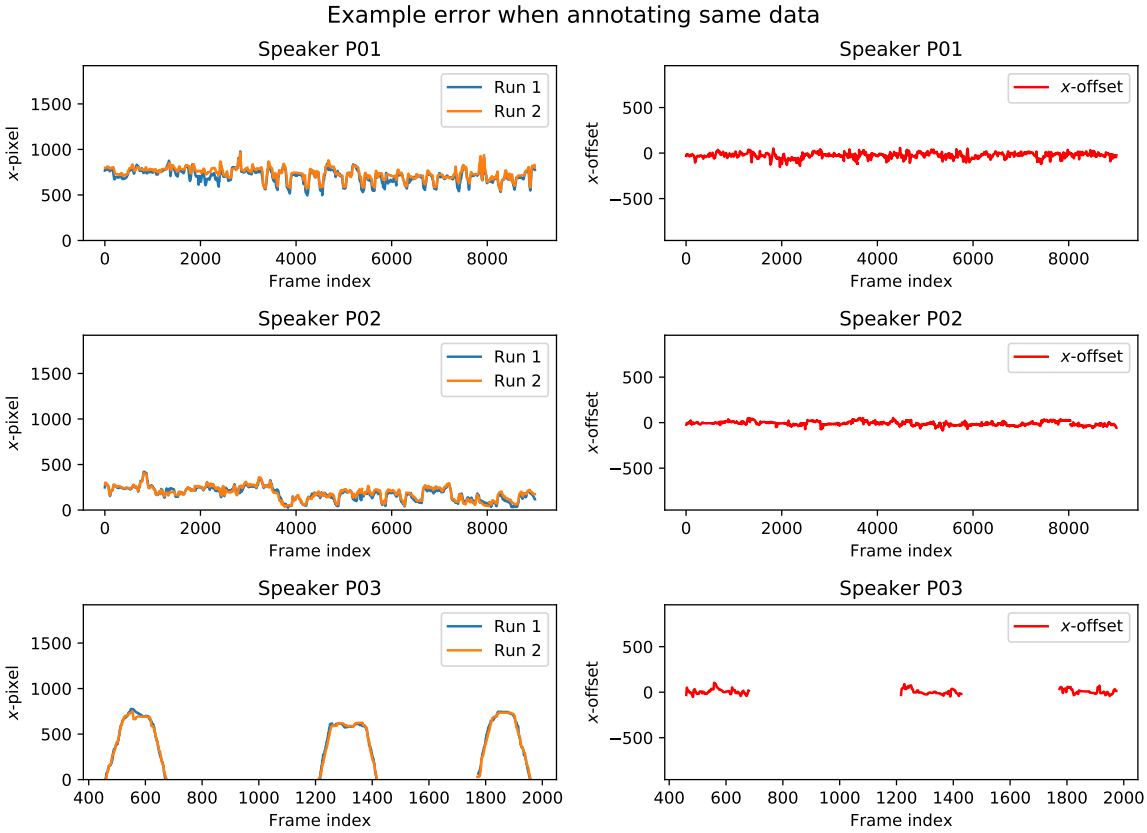

Fig. 3.9 An example of comparing annotations from two different runs of the same segment of data using the real-time annotator. The left side shows the x pixel index of the two runs for three speakers and then the right-hand side shows the x-offset (i.e., the difference between the runs).

of pixels difference between the re-annotations is only *15 pixels*. This estimate of error will later be used in Chapter 5 when combining camera estimates to locate speakers.

The error in re-annotating is caused by not being able to consistently keep track of the mouth position of the speaker as the video is being played in real-time, and people can move their heads very quickly. When annotations are very poor, the tool allows the annotator to go back in time in the video and redo the annotation. The degree of inaccuracy depends on the distance the speaker is away from the camera, i.e., if they are far away from the device, movements can more easily be tracked. The plots in Figure 3.9 illustrate two different runs of annotating the same segment. The left side shows the x-position for both the annotations for the three speakers that were visible during this segment. On the right side the difference between the two runs is show and we can see the result is close to 0.

3.7 Conclusions

This chapter presented the motivation for using the CHiME-5 corpus. The dataset provides a unique opportunity to analyse people's natural behaviour in a social setting using unobtrusive recording devices. In addition to this, CHiME-5 is a well-established dataset that has already impacted the research community. Continuing to use this data by providing position information of the speaker contributes further to this work.

This chapter also presented annotation tools to track speakers in real-time, as well as isolated frames. This is because tools are needed to extract this position information from the videos to create a ground truth. Now that automatic tools to automatically extract speaker positions are evaluated using the annotations. The positions will later be used in further analysis in the following chapters.

Chapter 4

Speaker spatial analysis: estimating speaker location using a single device

4.1 Introduction

It is becoming commonplace to find smart devices with voice assistants in homes. These devices use an array of microphones to exploit spatial cues to enhance speech in the desired direction whilst suppressing competing sounds such as noise and competing speakers in other directions. It is well known that these systems perform better when there is a greater angular separation between speakers. In a meeting room scenario, a microphone array is typically placed in the centre of the table to maximise the separation angle between speakers. This is in contrast to smart speakers, where the primary focus of the user is not maximising angular separation, but usability. Their preference is more likely to be to place the device “out of the way" i.e., this is typically at the edges of rooms avoiding obstructions. In addition to this, studies in the behaviour of people have shown that in a social setting, people tend to stand close to each-other (Hall et al., 1968). These two factors combined result in a smaller angle of separation between the talkers than one may expect. We will explore this conflict by analysing the behaviour of people in social settings and the impact it has on current speech enhancement techniques and automatic speech recognition (ASR). Knowing the true behaviour of speakers will help in understanding how best to design future microphone array algorithms and hardware.

As described in the previous chapter, to benchmark speech enhancement techniques, a controlled environment is required where a version of the audio before distortion is available. This is typically achieved using databases of simulated signals, which are created by generating room impulse responses (RIR) through simulation, e.g., the image method

(Allen and Berkley, 1979), and then convolving the RIR with the clean audio. Simulating the complexities of the real world is an incredibly difficult task but an important gap that needs to be bridged to provide meaningful results before algorithms are tested on real data. It is therefore the aim to produce results that are representative of those that would be achieved on real data. This is because direct testing of real data is not possible, this is because, unlike simulation, the real data does not provide the ground truth separated signals that are needed when computing many performance metrics. Many advances have been made in improving the realism of simulations (Brinkmann et al., 2019), for example, by simulating non-cuboidal room shapes (Scheibler et al., 2018) and high fidelity ray-traced acoustic rendering. However, the distribution of the speaker locations is an aspect that has been largely overlooked when producing the simulated data.

Often in multi-channel speech enhancement, when reporting results, overall performance is presented with no information about how the performance relates to the speaker separation distribution in the dataset being used. This is a surprise given the attention that is paid to other aspects of evaluation, e.g., the reverberation times, the SNR ranges, and the choice of metric to be reported (Le Roux et al., 2019). Although these are all very important factors when evaluating the performance of speech enhancement systems, this chapter argues that the separation of speakers is an additional factor that is equally important to simulate correctly, particularly because different techniques may perform with different effectiveness at different points on the speaker separation distribution. An unrealistic separation distribution can therefore lead to misleading conclusions regarding the relative effectiveness of source separation and recognition techniques under consideration.

In particular, this chapter argues that current simulated datasets such as (Drude et al., 2019b; Maciejewski et al., 2020; Wang et al., 2018) have separation distributions that have unrealistically large mean separations and therefore may lead to overpromising results, i.e. demonstrating large gains due to spatially-based source separation that are not realised in a real-situation where the separation angles are much lower. This is because they do not represent the spatial separation of speakers in typical social settings and therefore, may produce overpromising results. For this study, the video data captured during the recording of the CHiME-5 dataset is used for the analysis of speaker separation. This chapter uses analysis from cameras capturing videos from the perspective of 114 microphone arrays recording 50 hours of social interaction in 20 homes. For a full description of the dataset, see Chapter 3 and (Barker et al., 2018).

The aim of the chapter is to compare the difficulty of a simulated overlapping speech recognition task that uses a realistic distribution of speaker separation, with that of the difficulty of existing commonly-used datasets. This will be achieved by first learning

the speaker separation from the real data, i.e., by analysing the CHiME-5 data using its video component to locate speakers from the perspective of every one of the microphone devices. Second, these distributions will be used to generate overlapped speech datasets that are equivalent to existing evaluation datasets in every respect apart from the separation distribution. Third, the performance of state-of-the-art multichannel distant microphone speech recognition systems will be compared using both the unrealistic existing datasets and the more realistic new datasets. This chapter, therefore, aims to address questions such as *How much impact does separation distribution have?* and *Do the realistic separations lead to significant differences in the estimate of ASR performance?* Worse, do different separation distributions lead to different answers when ranking ASR techniques, i.e., might previous datasets have led researchers to make erroneous conclusions about what might work best in a real environment.

The chapter is organised, as follows. Section 4.2 outlines the methodology for estimating and testing the separation angle. Section 4.3 details the approach for automatically locating people in the living spaces. Next, in Section 4.4 the distributions of the real dataset are compared with existing benchmark datasets. The experimental work in Section 4.5 and Section 4.6 show the impact of using the realistic distribution. Finally the concluding remarks in Section 4.7.

4.2 Methodolody

The aim of this chapter is to investigate the impact that imposing a realistic separation angle distribution has on distant microphone ASR and speech separation. To achieve this, first, we need to establish what is a realistic distribution, then, we need to determine how to generate data according to this distribution and finally, we need to evaluate the impact that this has on performance. This section will give a brief overview of the methodology adopted in this chapter, first starting with the definition of angular separation.

The separation angle in this work is defined as follows. Given a microphone array l with a centre at position $\mathbf{m}_l = \begin{bmatrix} m_l^x & m_l^y \end{bmatrix}^\top$ and speakers at position $\mathbf{p}_1 = \begin{bmatrix} p_1^x & p_1^y \end{bmatrix}^\top$ and $\mathbf{p}_2 = \begin{bmatrix} p_2^x & p_2^y \end{bmatrix}^\top$ the angular separation is defined as,

$$\varphi_l(\mathbf{p}_1, \mathbf{p}_2) = \arccos\left(\frac{(\mathbf{p}_1 - \mathbf{m}_l)^T (\mathbf{p}_2 - \mathbf{m}_l)}{\|\mathbf{p}_1 - \mathbf{m}_l\| \|\mathbf{p}_2 - \mathbf{m}_l\|}\right). \tag{4.1}$$

A depiction of the angular separation between people is shown in Figure 4.1 where two people are in the room with two randomly placed microphone arrays. The possible separation

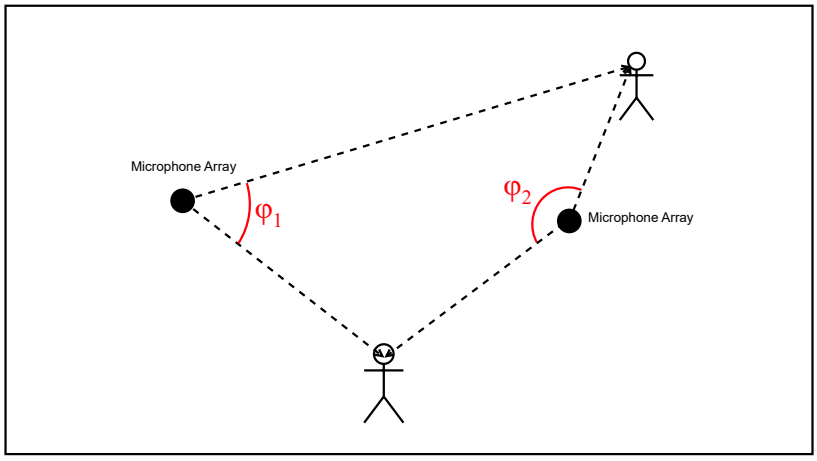

Fig. 4.1 Depiction of the definition of angular separation. The separation angle is the absolute difference between the two angles of the speakers from the perspective of the microphone array. It has a range of 0 to 180 degrees.

angles are between 0 and 180 degrees. We can clearly see that the range of angles that are possible depends on the placement of the devices as well as on the Euclidean distance between the speakers. The separation angle function φ_l can be reformulated to be in terms of the difference between the azimuth angles of the speakers relative to the device as depicted in Figure 4.2. This is formulated as,

$$\varphi_l = |\text{wrap}\left(\theta_l\left(\mathbf{p}_1\right) - \theta_l\left(\mathbf{p}_2\right)\right)|, \tag{4.2}$$

where,

$$\theta_l\left(\mathbf{p}_j\right) = \text{atan2}\left(\frac{p_j^y - m_l^y}{p_j^x - m_l^x}\right), \tag{4.3}$$

$$\text{wrap}(\theta) = \text{atan2}\left(\frac{\sin(\theta)}{\cos(\theta)}\right), \tag{4.4}$$

The wrap function normalises the angle between π and $-\pi$. Therefore after taking the absolute value the angle lies between 0 and π.

The video component of the CHiME-5 dataset will be used to estimate the position of the people from the perspective of each of the cameras in the screen space. Given that the cameras are integrated into the microphone array device, the speaker azimuth angle to the device ($\theta_l(\mathbf{p}_j)$) can be estimated from the screen space without knowledge of the talker's true location. Screen space positions of speakers are detected using the automatic methods, *pose* detection and *face* detection, details of which are outlined in the previous chapter. For

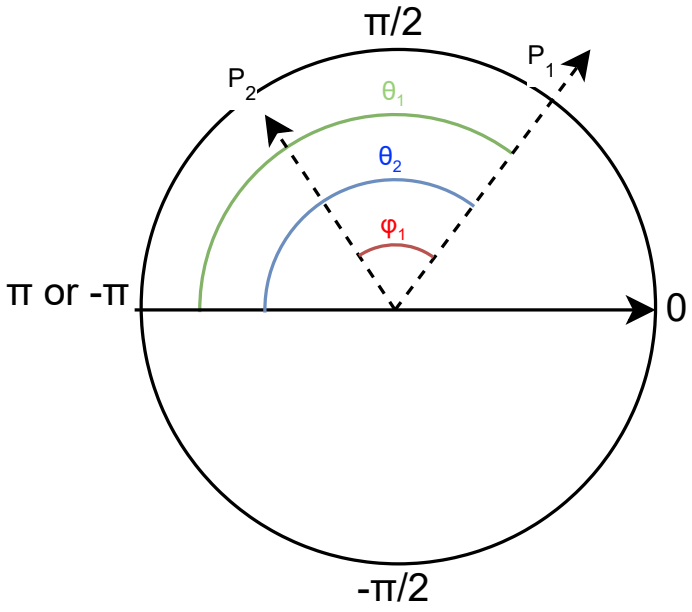

Fig. 4.2 The angle between the two speakers (φ_j) at microphone array l can also be computed through looking at the azimuth angle of the two speakers and computing the difference. When computing this difference care needs to be taken due to the wrapping nature of circles. To account for this, the difference needs to be normalised to be around the unit circle before taking the absolute value.

every frame in the video, i.e., every session and every device (total 114 videos) the detection systems are run. From this, a distribution of angular separation can be estimated.

This distribution can then be used directly for sampling an angular separation or as a reference to match when using other sampling methods. By creating a dataset that has separation angles that are expected in the real data and then evaluating systems, the overall performance will implicitly weigh the performance of the system across angles that are more important. This means focusing on improving the angles we see in the real data will improve overall performance more than improving wide angles. Finally, to compare the potential impact of using unrealistic separation angles, the experiments in this work use identical datasets that have the same mixing process but with separation angle distribution being the only difference.

4.3 Estimation of the real speaker spatial separation distribution

The CHiME-5 dataset consists of 20 dinner party sessions, with each party broken into three stages: cooking, dining and after-dinner socialising. Each of these stages typically takes place in different rooms of the house, i.e., *Kitchen*, *Dining*, *Living* rooms respectively. A room is captured by two Microsoft Kinect V2 devices, consisting of a 4-channel linear microphone array and a 1080p camera. The location of the devices was chosen such that they were not obstructing the participants, i.e., at the edge of the room looking into the party. This means the placement of the devices does not necessarily maximise the separation of speakers but more closely mimics the placement of a device in a real home use case.

4.3.1 Linear approximation of the relationship between screen and angle

To find the angle of the speakers, the position of the speakers in the image of device l needs to be mapped to the azimuth angle. The azimuth is the target because, like most linear arrays, the Kinect is linear in the horizontal plane, as this is where most spatial diversity occurs. This means for our analysis, the x coordinate (measured in pixels) of a speaker's mouth in the image is the most important feature to capture. The angle of azimuth can be approximated from the l-th device's screen x-coordinate using,

$$\theta_l(\mathbf{p}_j) \approx \frac{x_{lj}^{\text{screen}} \times 84.1}{1920}, \tag{4.5}$$

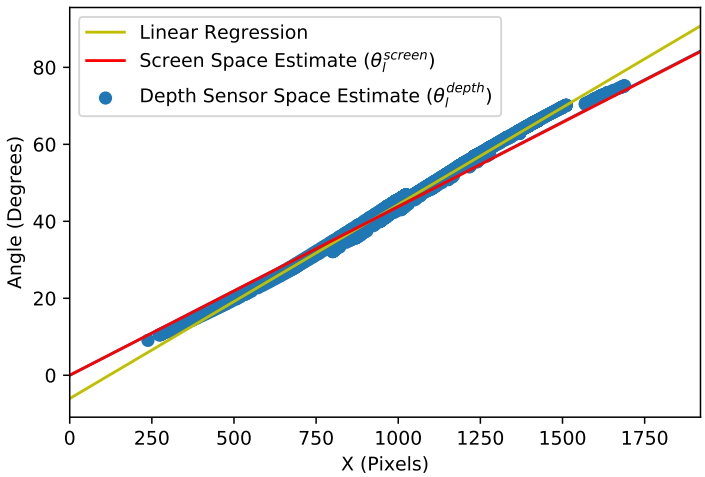

Fig. 4.3 Validation of the linear relationship between device screen space and azimuth angle. This assumption was validated by using the depth sensor in the Microsoft Kinect and projecting the position (θ_l^{depth}) to an angle and then comparing this with the screen space estimate of angle (θ_l^{screen}).

where x_{lj}^{screen} is the x pixel index in the screen space for device l and source j, 84.1 is the field of view of the camera measured in degrees, and 1920 is the resolution of the video. However, due to the nature of lenses, this linear approximation is not completely valid, i.e., there is typically distortion at the edges of the frames which can become extreme for wide-angle lenses. To validate the linear assumption, an informal experiment using the depth sensor within the Kinect V2 device was conducted. Using the Kinect Software Developer Kit[1] (SDK) a skeleton of a person with 3-D points of their position relative to the device is given. The SDK also provides a screen position of each of the joins in the pose. In order to estimate the amount of distortion and therefore the validity of the linear assumption, the mapping from the 3-D depth sensor position to the screen space position is shown in Figure 4.3. The figure was created by walking around a room and periodically capturing 3-D positions and screen space positions of all visible joints. All these points are then used in the data in the plot. In the plot we can see the data roughly fits the linear relationship between screen space and azimuth angle. A function could be fit directly on this data but for simplicity, the linear relationship will be used in the following experiments in this chapter and the further chapters.

[1] https://learn.microsoft.com/en-us/azure/kinect-dk/

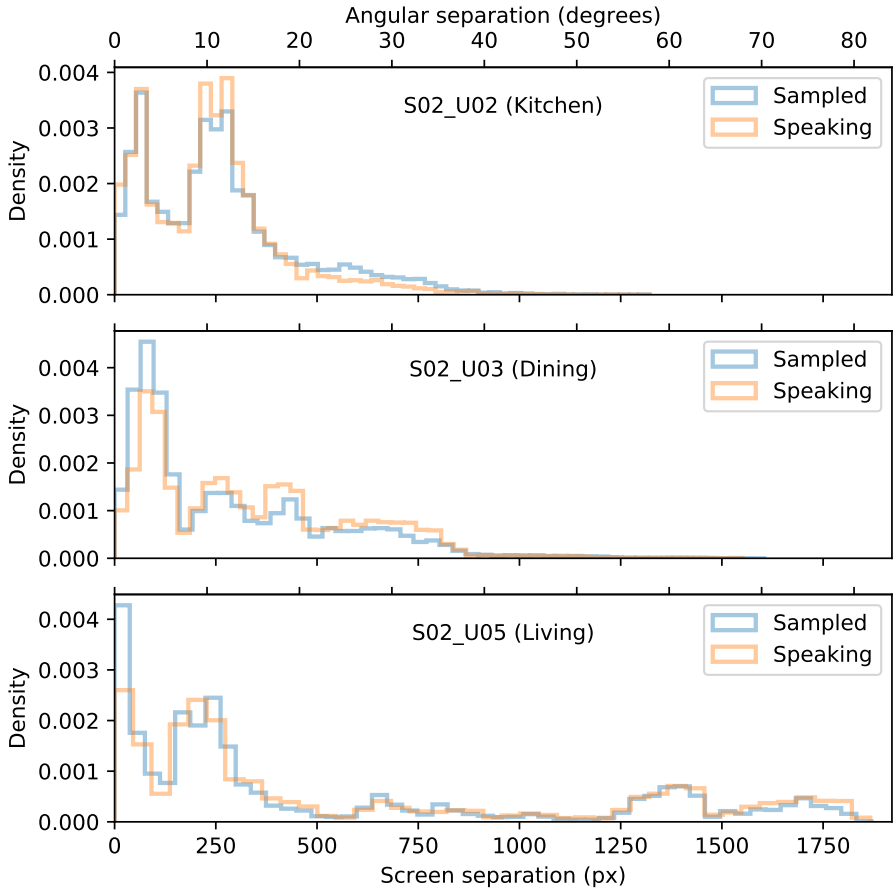

Fig. 4.4 Validation of the assumption of randomly choosing speakers instead of active speakers. The plot compares the distribution of separation angle between active speakers with the distribution when choosing two people at random. The similarity of the two distributions suggests that the separation angle is independent of speaker activity state. This means that the separation of *active* speakers can be modelled using measurements of the separation between all pairs of speakers in the scenes.

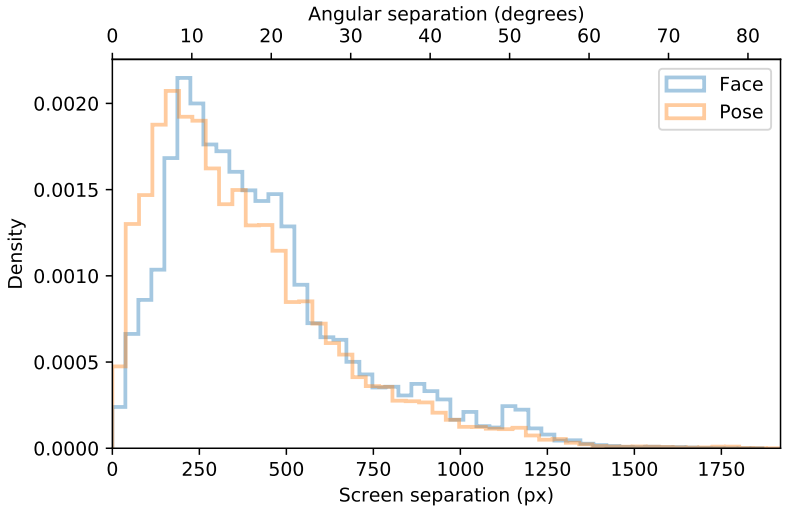

Fig. 4.5 Comparison of the separation distributions created from using all the frame data in the entirety of CHiME-5.

4.3.2 Estimated speaker separations

To detect speakers, two different 'out-of-the-box' tools were used: the Dlib CNN face detector (King, 2009) *(face)* and the OpenPose keypoint detection library (Cao et al., 2019) *(pose)*. These tools can both be regarded as state-of-the-art but have different strengths and weaknesses. The face detection system is only able to locate a person if they are facing the camera or if their profile view is visible. The pose detection system is able to locate people turned away from the camera but suffers from more false detections. These detection systems were run on each of the frames in the 114 videos in isolation.

Ideally, we wish to use the angular positions from the automatic methods to estimate angular separation between *active speakers*. Although the CHiME-5 transcript can be used to recover the speaker activity state of *identified* speakers, the identity of the speakers can not easily be determined reliably using the automatic methods. Therefore, the analysis makes the assumption that separation is independent of speaker activity state, i.e., that we can measure angles between pairs of people in the scene regardless of whether they are talking and then use the distribution of these separations as an estimate of cases where both people are talking.

This assumption has been tested using a fully annotated subset of the data. Using the real-time annotation tool described previously, three cameras are annotated for the entirety of session S02, i.e., with the speaker identities so that the people in the scene can be linked to the transcript. For each video frame in which two or more active speakers are detected by the *face* system, two random people are chosen and the angular separation is computed. Note, this approach is valid even considering the low recall of the detector assuming the missed

Table 4.1 Position and separation of speakers throughout the dinner parties. The centre of the screen is 0 pixels/degrees. Results are average ± standard deviation.

	Position		**Separation**	
	Screen (px)	*Angle (°)*	*Screen (px)*	*Angle (°)*
Pose	-23 ± 323	-1 ± 14	380 ± 268	17 ± 12
Face	-35 ± 302	-2 ± 13	427 ± 274	19 ± 12

detections are missed at random with respect to location. This is then repeated but now sampling pairs of people regardless of speaking activity state. The resulting distributions are compared in Figure 4.4. The similarity of the distributions suggests that person separation is largely independent of the speaking state. This may seem unusual, i.e., people speaking at the same time might be expected to be closer together. However, overlapping speakers may be from competing conversations, and inactive speakers are still 'socially engaged' and therefore standing at conversational distances from each other. The figure also highlights the variety in the distributions between the different devices. The distributions have clear distinct peaks indicating speakers are often in the same locations.

We can now measure person separation across all 114 devices without regard for speaker activity state and take this as a proxy for overlapped speaker separation. Analysis is repeated with both *face* and *pose* detectors (Figure 4.5). Even though the two systems have complementary errors, the resulting distributions are similar. Both distributions show that few detections have a separation around 0 pixels. This observation is likely due to the fact the detection systems are not able to detect a person if that person is being occluded by another person, rather than being directly caused by any specific human behaviour (this is one of the limitations that can be overcome using the multiple camera approaches that will be introduced in the next chapter.)

In Table 4.1 the overall statistics for the dataset are shown. The mean and standard deviation for the position results are the averages of the mean and standard deviation of each of the sessions. We average over sessions as the initial placement of the device will affect these statistics. Both detection systems have a small skew to the left, indicating a bias in the placement of the devices. Both detection systems show how small the separation angle is between the speakers, with both showing similar separation angles even though the two different approaches have different characteristics.

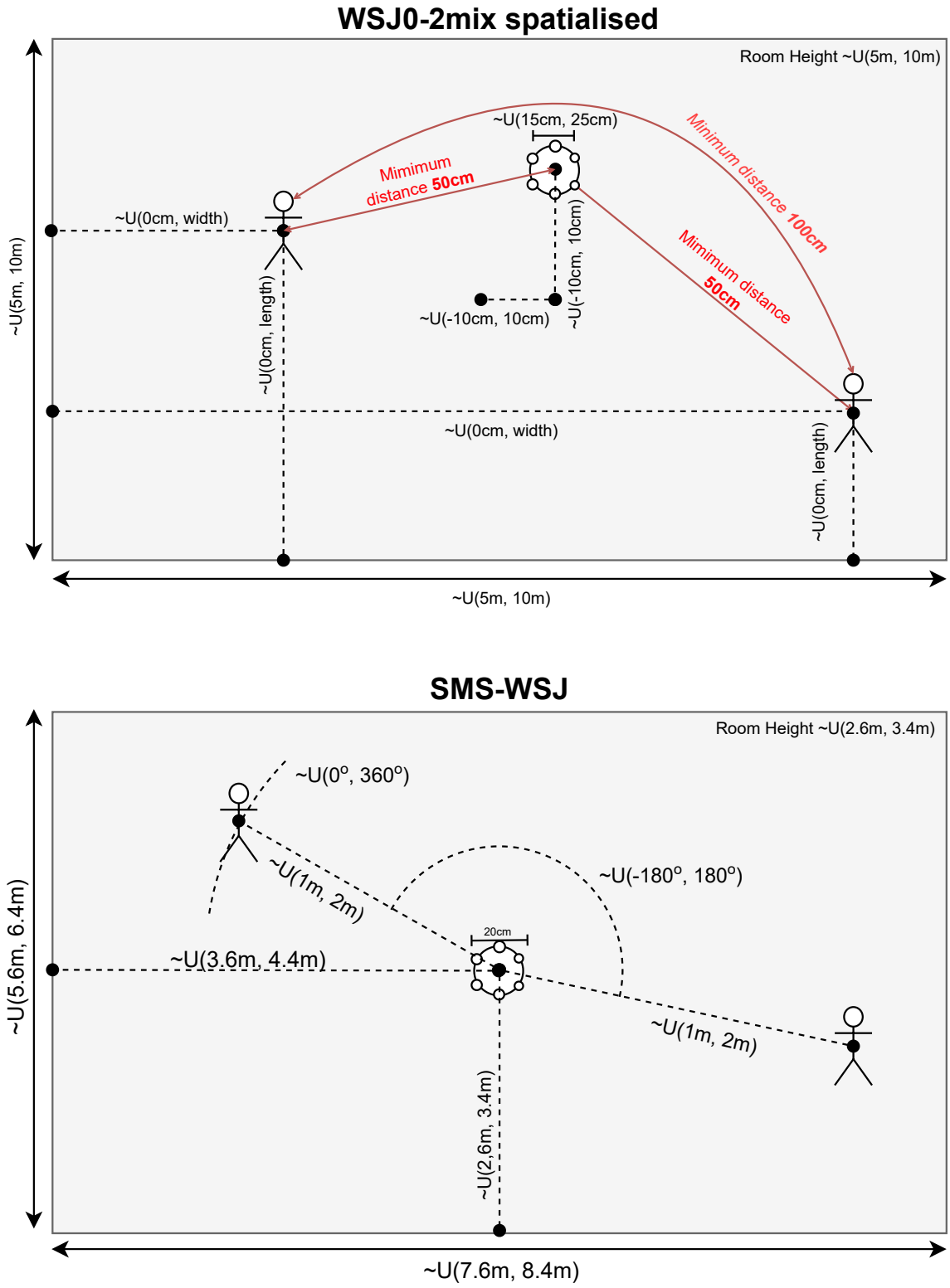

Fig. 4.6 Comparison of the different methods for generating the metadata for the room configurations in WSJ0-2Mix and SMS-WSJ. Original diagrams based on descriptions of the simulation configurations reported in (Wang et al., 2018) and (Drude et al., 2019b), respectively.

4.4 Existing spatialised speech datasets

Now that the angular separation in the real data has been established, we will look at how this compares to two publicly available spatialised datasets that are commonly used to benchmark multi-channel speech separation and distant microphone ASR. This section will first describe two datasets, namely *WSJ0-2Mix Spatialised* and *SMS-WSJ* both of which use the Wall Street Journal (WSJ) corpus in full or in part as source material for creating the mixtures. The WSJ corpus consists of clean recordings of primarily read speech, which comes from speakers reading articles from the newspaper *Wall Street Journal*. The dataset has two releases commonly referred to WSJ0[2] and WSJ1[3]. On face value, *WSJ0-2Mix Spatialised* and *SMS-WSJ* seem similar but we shall see the design decisions for the two datasets are quite different, especially with respect to their source positioning.

4.4.1 WSJ0-2Mix Spatialised

WSJ0-2Mix is a well-established corpus within speech separation that contains mixtures of two speakers taken from WSJ0, mixed at signal-to-noise ratios between 0 dB and 10 dB. Mixtures have been constructed by first choosing two random speakers and then one random utterance from each of those speakers. The training data for WSJ0-2Mix takes utterances from the training data of WSJ and the validation dataset also samples from the training data of WSJ (i.e., this is a closed set). The evaluation dataset, however, samples from the combined development and evaluation datasets of WSJ (i.e., creating an open set). In total the dataset consists of 30 hours of training, 10 hours of validation and 5 hours of evaluation data. WSJ0-2Mix uses instantaneous addition of the pair of speech samples, i.e., there is no spatialisation nor modelling of room acoustics.

Later, in further work, a spatialised version of the WSJ0-2Mix dataset was created using the same utterance pairings and SNRs (rescaled factoring in the sound decay when spatialised). The dataset uses the image method to create the RIRs which are then used to spatialise the mixtures. For the room simulation, shoebox rooms are employed with T60 reverberation times sampled between 0.2 s and 0.6 s. The geometries of the rooms are created with random widths, lengths and heights. The size of the circular microphone array aperture and its offset from the centre of the room are randomly sampled. The ranges used when sampling these values are depicted in Figure 4.6 (LHS). The positions of the talkers in the room are chosen by sampling a random position inside the room. Once two positions are chosen some validations are run such as checking the speakers are not within a

[2]LDC Catalog No. LDC93S6A
[3]LDC Catalog No. LDC94S13A

minimum distance of each other and that both are not too close to the microphone array. If the conditions are not met then the random values are sampled again. It is worth mentioning that these constraints were not documented in the corresponding publication. It is only through inspecting the code for generating the parameters that we can see this procedure. We will later see the impact this has on the distribution of angular separations.

4.4.2 SMS-WSJ

The Spatialised Multi-Speaker Wall Street Journal (SMS-WSJ) corpus was created to address some of the shortcomings of the WSJ0-2Mix and Spatialised WSJ0-2Mix corpora for use in ASR. WSJ0-2Mix was created for speech separation and not ASR and thus there is no agreed-upon set of data to be used for acoustic model training, i.e, should the same training data be used for acoustic modelling as used for training supervised speech separation? The training data for WSJ0-2Mix contains 20,000 mixtures, however, due to the sampling nature, only 8769 are unique, making it less ideal for ASR training. Therefore, when using the dataset, some researchers train the ASR system on different data, e.g., the full WSJ corpus (Paul and Baker, 1992). In addition to this, the development data for WSJ0-2Mix contains speakers that are used in training. SMS-WSJ addresses these problems through creating mixtures derived from WSJ0 *and WSJ1* and by taking care to maximise the number of unique utterances used.

In terms of spatialisation, SMS-WSJ and WSJ0-2Mix have some similarities such as both opting to use shoebox rooms and using the image method to generate the RIRs. Again, the details of the generation method are shown in Figure 4.6 (RHS). The most noticeable difference between the two datasets is their approach to selecting the positions of the sources: whilst WSJ0-2Mix chooses random positioning in the room, SMS-WSJ chooses the position relative to the microphone array. This works by choosing an angle for the first speaker, then an angular separation is sampled, and finally, two microphone distances are sampled.

SMS-WSJ exploits the simulated nature of the corpus to provide extra targets that can be used for evaluation and training. The authors provide a split for the RIRs to separate them between early and late reflections, with a cutoff chosen at 50 ms. This allows for the decomposition of the spatial image to be,

$$c_j^{(\text{image})}[t] = c_j^{(\text{early})}[t] + c_j^{(\text{late})}[t] \tag{4.6}$$

$$= r_{ij}^{(\text{early})}[t] \circledast s_j[t] + r_{ij}^{(\text{late})}[t] \circledast s_j[t], \tag{4.7}$$

where $c_j^{(\text{image})}[t]$ is the reverberant speech signal, $c_j^{(\text{early})}[t]$ is the reverberant speech from early reflections, $c_j^{(\text{late})}[t]$ is the uncorrelated late reflections for source j, $r_{ij}^{(\text{early})}[t]$ is the early part of the RIR, $r_{ij}^{(\text{late})}[t]$ is the late part of the RIR for source j at microphone i, and $s_j[t]$ is the clean signal for source j. A schematic overview of the dataset in shown in Figure 4.7. The dataset is defined by the scenario metadata which describes the utterances being mixed together and their spatial positioning. SMS-WSJ has generated Gaussian noise for the background, the seed for this is derived from the metadata. The blocks in orange describe databases of speech signals that can be used for training and evaluation. For example, the early reflections of the spatial image may be a better target for training speech enhancement if an anechoic output is desired.

4.4.3 Comparison of the angular separation between the datasets

Next, we will look at the difference in the distributions of the sampling methods imposed in WSJ0-2Mix and SMS-WSJ. Instead of the instantiated data in the datasets, we will look at the underlying separation distributions of the generative methods used. For SMS-WSJ, the separation angle is drawn directly whereas in WSJ0-2Mix the distribution is more complicated as the angle is the outcome of other parameters. In order to estimate the theoretic angular separation distributions of the two datasets, 1,000,000 speaker position pairs are generated and the resulting separation angles are computed, allowing for the true distribution to be closely approximated. The result of this sampling is shown in Figure 4.8. Immediately we can see that the two datasets produce very different separation distributions. As expected SMS-WSJ has a uniform distribution across all angles. However, WSJ0-2Mix produces very few mixtures with speakers with a low separation angle. Although this distribution may be initially surprising, the underlying cause is clear and results from the constraints placed on minimum inter-speaker distance and minimum speaker-microphone distance: In order for the angle to be narrow the competing speaker needs to be in front or behind the first speaker and not within 1 metre. This results in far fewer valid positions compared to if the two speakers have a very wide separation.

Both of these distributions are in stark contrast to the true separation angle distribution estimated in the previous section. WSJ0-2Mix, in particular, produces very few mixtures with separation angles we would expect in real data. This means that WSJ0-2Mix is favouring systems that are better at separating mixtures with wide angular separation and almost completely disregarding their performance on narrow angles. Without knowing how well a system performs on narrow-angle data, it will be hard to predict how well it will perform on real data. This same argument can be made for SMS-WSJ which weighs all angles equally.

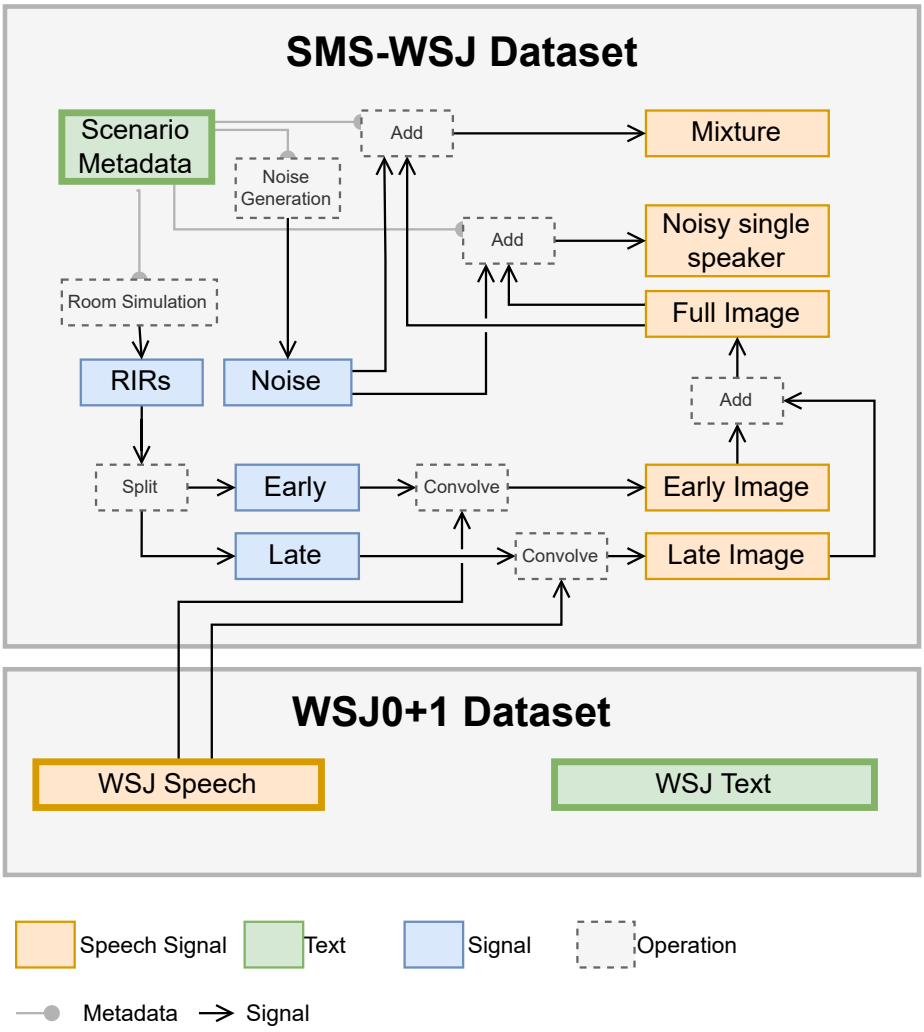

Fig. 4.7 Overview of the data inside of SMS-WSJ. The dataset provides several versions of targets that can be used for training and evaluation. In particular, they decompose the spatial images into early and late reflection parts.

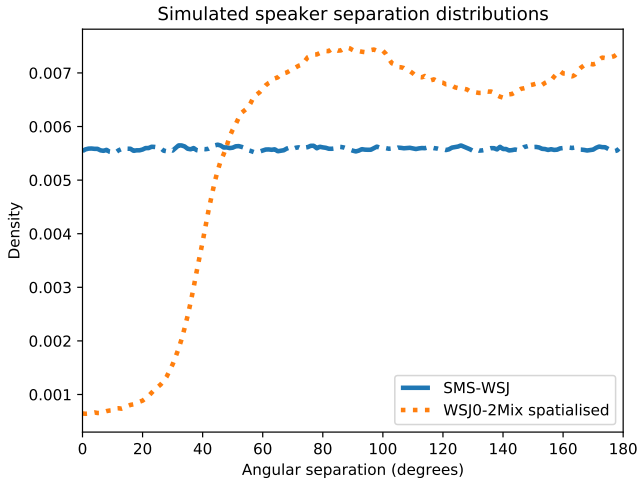

Fig. 4.8 The distributions used for generating the separation angle between speakers in SMS-WSJ and WSJ0-2mix. The plot illustrates a clear mismatch between the two approaches. With SMS-WSJ having a uniform distribution of separation angles and WSJ0-2Mix spatialised having a large dip in narrow angles.

4.5 Effect of realistic angular separation

So far in this chapter, we have established the speaker separation distribution in a real dataset (Section 4.3) and shown how this is greatly mismatched to the distributions of two commonly used simulated datasets (Section 4.4). In this section, we will investigate the impact of this mismatch by comparing the performance of identical speech separation and recognition systems under these various distributions.

4.5.1 Motivation

The aim of the following experiment is to explore the potential impact of using datasets that favour separation angles not found in the real world. A potentially very effective system on simulated data can fail on a real dataset (which is often the case) for numerous reasons such as this mismatch in separation angle. This could happen for two reasons. First, supervised methods are *trained* directly on data with very few narrowly separated speaker samples, therefore they will be unable to learn to discriminate between sources in narrowly separated mixtures. Second, and potentially more troublesome, is that approaches may be more fundamentally flawed when it comes to discriminating narrowly separated speakers, but the impact of this has not been realised because they have not been *tested* at such angles.

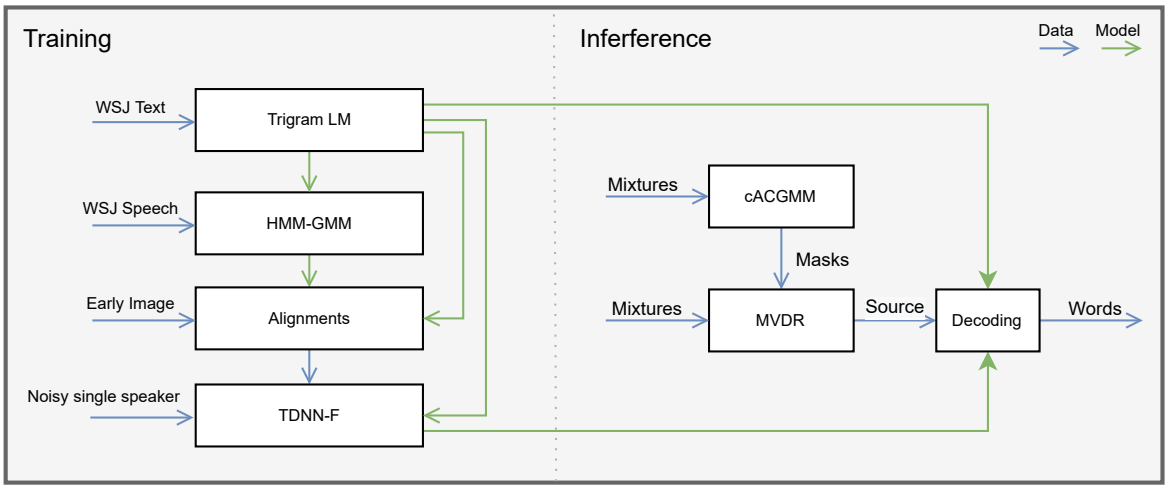

Fig. 4.9 Schematic diagram of the SMS-WSJ baseline system.

4.5.2 Method

Experiments use the baseline system[4] described in (Drude et al., 2019b), namely, a complex angular central Gaussian mixture model (cACGMM) (Ito et al., 2016) mask estimator is used with a Minimum Variance Distortionless Response (MVDR) beamformer and a factorised time-delayed neural network (TDNN-F) based acoustic model. The acoustic model is trained by first training a HMM-GMM hybrid system on *clean* WSJ utterances following the WSJ recipe in the Kaldi repository[5]. Using the standard Kaldi training procedure of bootstrapping lower-order models, a triphone HMM-GMM is then used to create the alignments for the neural model training. The early images of the reverberant single speakers are used to obtain alignments, these phone alignments are then used not in the early images but full images with additional noise, i.e., *Noisy single speaker*. This allows the neural model to be trained robustly but with more accurately estimated alignments. The TDNN-F is trained using the lattice-free maximum mutual information loss function. Note the acoustic model is trained solely on isolated speech and not mixtures, which allows for the mixing process to be the subject of experiments for evaluation without needing to retrain the acoustic model.

The first set of experiments measures how the baseline performance changes when the SMS-WSJ dataset enforces a realistic spatial distribution. In the original SMS-WSJ setup, the target speaker was placed in the room by randomly sampling a distance and an angle from the microphone array, and a competing speaker is placed at a uniformly sampled angular distance. To generate the two speaker positions for the 'realistic' distribution, an angle is uniformly sampled around the array. Using a Gaussian distribution with a standard deviation

[4]Code available online: https://github.com/fgnt/sms_wsj
[5]https://github.com/kaldi-asr/kaldi/tree/master/egs/wsj/s5

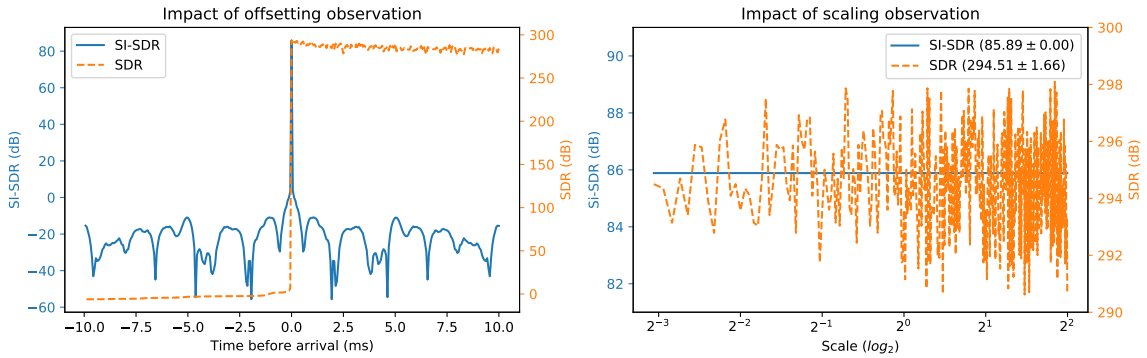

Fig. 4.10 Comparison of the sensitivity of speech separation metrics. SDR is widely known to be sensitive to the scale of the signals. SI-SDR is sensitive to the offset of the signal, making it a less ideal metric for multi-channel microphones where each of the microphones has different offsets but each is equally valid.

of 14 degrees and a mean set by that chosen direction, the two speaker directions are sampled. The value 14 degrees is used based on the standard deviation of people positions given by the Pose estimator shown in Table 4.1. The speaker distances are then chosen by sampling uniformly from 1-2 metres, i.e., the same as SMS-WSJ. The remaining random parameters are identical to SMS-WSJ. This does not necessarily create a realistic setup because in CHiME-5 the arrays were placed at the edge of the room and here they are placed in the centre of the room. However, it does let us see how the performance of the system changes when speakers have the distribution of separations that are observed in real data.

Source separation metrics and word error rate (WER) will be reported in the results. In particular, the signal-to-distortion ratio (SDR) will be reported in preference to the scale-invariant version of the metric (SI-SDR). While the SI-SDR metric is ubiquitous in single-channel speech separation where it was designed to be used, it is not well defined in the multi-channel case, even though it is still widely used. The toolkit used to compute the SDR (*bss_eval* (Févotte et al., 2005)) finds a finite impulse response (FIR) filter to allow the metric to align the observation and reference, i.e., to model the room impulse response (RIR). This FIR filter allows for clean utterances to be used as a reference instead of spatial images. This is important because, in multi-channel setups, each of the audio streams is not aligned, i.e., as the sources take different amounts of time to reach the microphones. The SI-SDR metric does not have this RIR filter and therefore expects the reference and the observation to be aligned. An illustration of this problem is shown in Figure 4.10 (LHS). The plot shows the SI-SDR and SDR computed with the inputs being the same signal (an utterance from WSJ). The x-axis indicates an artificial delay added to the signal to simulate the time-of-arrival. A positive value indicates the amount of time before the signal reaches

the microphone. The plot shows that SI-SDR is very sensitive to this offset and any change results in a very negative result. The SDR is mostly invariant to this change in offset. On the right-hand side of the same figure we can see how this compares with the sensitivity of SDR with respect to scale. Of course, the SI-SDR metric is (by design) completely invariant to scale as shown in the figure. Whilst SDR is not invariant to scale, it is far less sensitive to the change in scale compared with the sensitivity of SI-SDR with offset. If anechoic spatial images are used for the reference when computing SI-SDR, a massive change in performance will be observed by simply changing the reference channel. In some enhancement systems, a reference channel may be an input and therefore for evaluation we know which reference spatial image to compare to. However, some modern systems even use all audio streams as input and therefore a reference spatial image is not known. This lack of definition for SI-SDR for multi-channel speech separation makes the SDR metric the preferred metric. Alongside SDR, PESQ (Rix et al., 2001) to measure speech quality and STOI (Taal et al., 2011) to measure intelligibility will also be reported, this is to show how changes in these metrics correspond with the WER performance. When computing the WER the sources are separated first and then inference using the speech recognition system is run on each of the sources.

Only the *evaluation* dataset is changed in the experiments. Therefore the original SMS-WSJ dataset is used to train the acoustic model. This is valid because the acoustic model is trained on single-speaker utterances rather than on mixtures, and the updated mixtures will only have changed angular separation between speakers (the rotation angle distribution relative to the microphone array will be unchanged). It is also valid because the enhancement system under test is the cACGMM (Ito et al., 2016) model which is unsupervised and therefore only requires the evaluation data. Experiments with training supervised separation methods on mixtures of narrowly separated talkers is a potential area for future work.

The cACGMM model is being used as this is the key component of the current state-of-the-art in speech separation in real datasets such as CHiME-5 (Barker et al., 2018). The model clusters the time-frequency representations of the channels to provide a mask. These masks can either be used directly to enhance a channel or used to compute the spatial covariance matrices to be used in beamforming, e.g., MVDR. The experiments will compare the two approaches to see if one is more sensitive to the angular separation. In addition to this, oracle masks will be used instead of the estimate from the cACGMM. This allows us to distinguish between two possible causes of poor results at low angular separations: the system may perform poorly because of the poor spatial clustering due to the narrow angles or because the MVDR beamformer cannot filter such narrow separation.

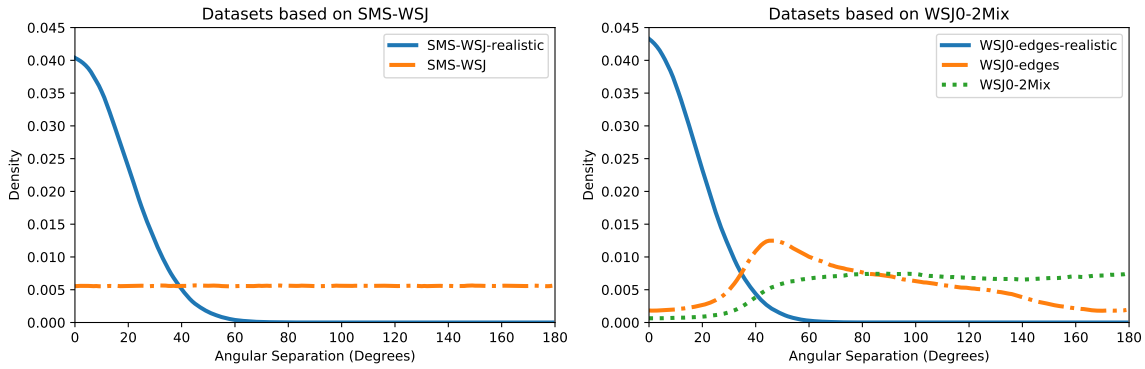

Fig. 4.11 Comparison of the angular separation in simulated datasets. We compare the datasets SMS-WSJ (Drude et al., 2019b) and WSJ0-2mix spatialised (Wang et al., 2018) with adapted versions of their setup.

Table 4.2 The effect of changing the positions of the speakers in the SMS-WSJ database. Oracle results are shown in grey. When Enhancement is 'None', the first channel in the microphone array is chosen.

Mask	Enhancement	Data	SDR	PESQ	STOI	WER
cACGMM	MVDR	Realistic	9.0	1.85	0.74	31.49
cACGMM	MVDR	SMS-WSJ	12.3	2.07	0.82	18.15
cACGMM	Mask	Realistic	7.1	1.73	0.71	49.09
cACGMM	Mask	SMS-WSJ	9.5	1.83	0.78	40.01
None	None	Realistic	-0.4	1.49	0.66	78.93
None	None	SMS-WSJ	-0.4	1.50	0.66	78.73
IBM	MVDR	Proposed	10.4	1.88	0.77	21.23
IBM	MVDR	SMS-WSJ	12.9	2.06	0.83	14.23

4.5.3 Results

The results from changing the placement of sources in SMS-WSJ are shown in Table 4.2. We can see that by only changing the location of speakers the WER of the cACGMM system increases by over 13% absolute (73.5% relative) when using the MVDR beamformer. Note, this is a system that contains similar components to the best performing systems on the CHiME-5 dataset (Du et al., 2020b). If a system can be made that is more robust to smaller separation angles, then there is huge potential to create an overall better-performing ASR system. The oracle ideal binary mask (IBM) comparison shows that even with perfect knowledge, the beamformer approach performs significantly worse with the new dataset.

Multi-channel approaches that do not use a beamformer may offer a solution to this (Wang et al., 2018; Zhang et al., 2020a); however, they rely on closely matched training data.

The performance difference between the two datasets when comparing using the cACGMM mask directly with the result with MVDR shows that beamforming is potentially more impacted by the narrow distribution compared with the spatial clustering.

The results of using no enhancement and no masking indicate that changing the speaker positioning has not greatly changed the lower bound performance, i.e., this is expected as the mixture utterances have not changed.

Next, when we look at the speech separation metrics, in this case, we can see they all largely correspond with WER when comparing the scores across different techniques and across the two datasets.

4.5.4 Discussion

The results have shown that imposing a realistic separation angle has a large impact on the performance of both source separation and ASR. One of the reasons for the low separation angles in CHiME-5 is the devices are placed at the edges of the room in order to avoid obstructing people and well as mimicking the placement you would expect such devices to be placed naturally in homes. However, in SMS-WSJ the devices are placed in the center of the room instead of the edges. It is important to consider how much of the angular separation angle is due to the placement of the devices and not just the behaviour of the people in the room. Therefore, the next experiments will consider placing the devices at the edges of the room.

4.6 Microphone location versus speaker distribution

The next set of experiments considers the placement of the microphones separately from the distribution of the speaker positions. For these experiments, WSJ0-2Mix is used to show the impact of changing the device placement. In the real data, people are interacting with each other rather than talking towards any one of the devices, therefore their position should be independent of the device placement. This makes WSJ0-2mix's method for generating speaker positioning more ideal when comparing the effect of device placement and the effect it has on the resulting angular separation.

4.6.1 Motivation

In the previous experiments, in which the microphones were placed in the centre of the room, the simulation was artificial because the narrow speaker separation angles that were imposed would not naturally occur if the microphones were in this position. Therefore the next experiments will consider how much of the difficulty can be added into the simulation without altering the speaker location distribution from the original SMS-WSJ but by simply resimulating with the devices moved to the edges of the rooms. Does microphone position alone account for the difficulty observed in the CHiME-5 set-up?

4.6.2 Method

These experiments compare the WSJ0-2mix spatialised setup, with a variation of the setup where the microphones are placed at the edge of the room *(WSJ0-edges)*, in both cases speakers are positioned uniformly in the room with constraints on minimum distances. A device is placed at the edge by first randomly choosing one of the four sides of the room and then a random position along the wall. Devices are placed such that there is a 50 cm padding room from the closest wall. The two datasets are then compared with a setup with microphones at the edges but with the realistic distribution enforced. This uses the same angle generation method as the previous experiment but with a distance sampled between 1 and 3 metres *(WSJ0-edges-realistic)*. The comparison of the distribution created from this setup is shown in Figure 4.11 (right). Placing the microphones at the edge of the room resulted in a distribution closer to the real data, but the tail is still far larger than that observed in the real data. Note the distribution of this realistic setup is slightly different from the distribution created in the previous setup. This is due to the resampling of points when they are outside of the room.

4.6.3 Results

Table 4.3 presents performance results of the cACGMM MVDR source separation system when adapting the WSJ0-2mix spatialised setup to be a more realistically distributed dataset. Here the *min* version of the dataset is used which trims the longer utterance to be the length of the shorter one. This version is not appropriate for speech recognition as some words may be cut off. Surprisingly, placing the microphones at the edge of the room does not make the dataset any more challenging than the original setup, i.e., the performances reported in the 1st and 2nd table rows are fairly comparable. This is likely due to the minimum distance constraint still limiting the minimum angular separation possible. Now when we look at the

Table 4.3 Source separation results

Dataset	SDR	PESQ	STOI
WSJ0-2Mix	15.1	2.50	0.83
WSJ0-edges	15.2	2.61	0.82
WSJ0-edges-realistic	14.5	2.28	0.71

results of enforcing the angular separation distribution with the devices at the edges we can see the impact of the distribution. The impact is not as big as that observed in the previous experiment. This may be attributed to the simplicity of this setup, i.e., no background noise and 100% overlapping mixtures. The *realistic* variant of the 'edges' dataset is sampled from the perspective of the device and as such a microphone-source distance also has to be sampled. The distribution of speaker distances resulting from this may also be contributing to the performance difference.

4.6.4 Discussion

Placing the devices at the edge of the room results in an angular separation closer to that of what was observed in the real data but still not what we could expect to see if the real data. By enforcing the realistic distribution the performance decreased but this may also be due to the distance speakers are away from the microphone and their relative distance. Only the separation angle was explored in this chapter. Estimation of the distances will further provide realism in the simulation.

4.7 Conclusion

Often the methodology for generating speaker positions in generated datasets is to make it completely random, but as discussed throughout this work, this is not realistic. Constraints such as enforcing a minimum distance between sources seem sensible at first but can yield unrealistic distributions. Without reporting either the separation distribution of the dataset or the performance of the source separation system with respect to the separation angle, it is difficult to compare results across different works. For example, it has been shown in this chapter the WERs can change by over 73.5% relative due to changes in the distribution of source locations alone. It is suggested that when generating simulated evaluation data err towards sources being closer together rather than using a uniform distribution in order to more closely match real data. This work has focused on just one parameter of simulation design, however, other equally important parameters are often overlooked such as directivity patterns

(i.e., the direction speakers are facing), the distance they are away from the microphone and the degree of speaker overlap (Chen et al., 2020).

In this chapter, automatic methods were employed to estimate the angular distribution of speakers in a multi-speaker distant microphone scenario using face-detection and pose-detection techniques. Using this analysis, the chapter showed that in the CHiME-5 scenario where the camera has a field of view of 84.1 degrees, the speakers that are visible have an average angular separation of 17 degrees. This distribution was compared with common simulated datasets that are used to benchmark the state-of-the-art in speech separation and found there is a large disparity. We then showed that this disparity could have consequences for the research community, such as leading research down the wrong path by pursuing systems that optimise unrealistic angular separations.

The next chapter will explore how 2-D estimates of speaker locations inside rooms can be estimated by combining estimates of angle from multiple cameras. This will allow for a better estimate of angular the separation between speakers as occlusions will not hinder the detections. Further, it allows the microphone-source distances to be calculated. Accurately modelling these distances allows for realistic SNRs levels to be estimated (i.e., stemming from the relative distance of competing speakers) and for distance-dependent direct-to-reverberant energy ratios to be modelled. Modelling these aspects correctly can further improve the realism of the simulated datasets.

Chapter 5

Speaker spatial analysis: estimating speaker location using multiple devices

5.1 Introduction

Acoustic room simulation (Allen and Berkley, 1979; Scheibler et al., 2018; Schröder and Vorländer, 2011) is an essential tool for developing distant microphone automatic speech recognition (ASR) systems. Simulation allows for clean reference signals to be used in evaluating speech enhancement (Févotte et al., 2005; Le Roux et al., 2019), for arbitrary large training data (Ko et al., 2017) to be constructed and for targeted evaluation and analysis of the performance of ASR systems (Vincent et al., 2017). Simulation is commonly used for generating *training data*. For example, for augmenting real training data, or for providing ground truth information when training supervised speech enhancement systems. For the simulated data to be useful, it needs to match the distribution of the real target data (Cosentino et al., 2020). However, simulation is also often used for generating *evaluation data*. In such cases, the need for realism is even more crucial: a poor simulation can result in wasted effort, i.e., by promoting approaches that work in simulation but not in real situations.

Although modern methods for acoustic room simulation can accurately model the physics of sound propagation, e.g., (Schröder and Vorländer, 2011), this is only one part of the problem. Room simulations are driven by their metadata, e.g., the room size, location of sources, T60 time and so on. The distribution of this metadata needs to be carefully considered. If it is poorly motivated the resulting dataset can overemphasise the importance of one component of a speech processing system over another. For example, the importance of beamforming approaches can be overplayed if simulations have unrealistically large angular separations between speakers (Pan et al., 2014).

In our work, simulating multiparty conversations for distant microphone speech recognition research is the focus. Previously, a large real audio-visual dataset (CHiME-5 (Barker et al., 2018)) was used to look at one aspect of this problem, angular speaker separation (Deadman and Barker, 2020). Our methodology was to use camera data from single devices to estimate and hence simulate realistic angles between overlapping speakers. In this work, the analysis is extended by using multiple cameras. This allows the 2-D room location of the target and interference speakers to be estimated. This data can then be used to correctly simulate the full spatial distribution of speakers, and hence produce data with realistic speaker properties such as signal-to-noise ratio (SNR), angular separation and direct-to-reverberant energy ratio (DRR).

The chapter is organised as follows. Section 5.2 reviews previous simulated spatialised-speech datasets and their role in automatic speech recognition research. In Section, 5.3 the general methodology is outlined. Section 5.4 details the method for calculating speaker locations using multiple devices from annotated single-device data. These positions are then used in Section 5.5 to estimate the relative distance of speakers and interferers in the CHiME-5 datasets. This analysis is used to inform a simulation with an improved estimate of angular separation and speaker distance which is evaluated in Section 5.6. The chapter concludes with a short discussion and a summary of the findings.

5.2 Background

The speech enhancement and source separation fields rely heavily on simulated datasets constructed by convolving room impulse response (RIRs) with clean utterances, for example from WSJ (Paul and Baker, 1992) and LibriSpeech (Panayotov et al., 2015). The spatialised version of WSJ0-2MIX was introduced in (Wang et al., 2018), which became a common benchmark for multi-channel source separation algorithms. In recent years, deep learning techniques have performed so well in these scenarios that more challenging datasets have been required. WHAM! (Wichern et al., 2019) increased the challenge by adding real background noise and then WHAMR! (Maciejewski et al., 2020) extended WHAM! by using *reverberant* noisy mixtures. Both multi-channel WSJ0-2Mix and WHAM! use the WSJ corpus (Paul and Baker, 1992) as their source for clean speech signals, and both randomise speaker positions uniformly in the room. LibriMix (Cosentino et al., 2020) was introduced to compliment WHAM! and WSJ0-2Mix

Attempts have been made towards creating simulations that mimic more realistic temporal overlap in simulation (Chen et al., 2020; Fujita et al., 2019). However, little progress has been made towards generating data-driven speaker positioning in these setups (Deadman and

Barker, 2020), even though there is a wealth of behavioural research showing that people in multi-party conversations observe social rules that govern how they are spaced, i.e., the field of proxemics (Hall, 1963). Due to these spatial mismatches, amongst others, deep learning techniques may perform well in simulated environments but then perform poorly in real domestic scenarios (Maciejewski et al., 2019).

5.3 Methodology

This chapter follows the same general methodology presented in the previous chapter, first, the behaviour of speakers will be analysed and modelled. Using this modelling a set of evaluation datasets are created to measure the impact imposing realistic positioning has on speech separation and speech recognition.

First, in order to estimate the speaker position information, multiple devices with overlapping views of the CHiME-5 living spaces are used. Alongside the CHiME-5 video recordings, the CHiME-5 data provides rough floorplan sketches that were produced by the recording engineers. These sketches provide the *approximate* position and orientation of the recording devices. With this information, the 2-D location of people in the scenes can be estimated using a process akin to triangulation. However, device positions have only been marked approximately, and so this data has to be refined using a calibration process. This process works by iteratively adjusting device location and orientation parameters so as to reduce the apparent mismatch between estimates of speaker positions in each of the devices. The devices are then combined to give 2-D positions of people in the rooms throughout the parties.

Once the 2-D person positions have been recovered, they can then be used to produce estimates of the absolute distances speakers are away from the microphones and the relative distances compared with a competing speaker. The 2-D positions also allow for a refinement of the angular separation estimates provided in the previous chapter. Together the angular separation distribution and the relative distance to competing speaker distribution are the focus of the experimental work. With the belief that angular separation is the more important distribution to be modelled for separation performance and relative distance is more impactful for ASR performance.

5.4 Estimating 2-D positions using multiple devices

Realistic speaker location distributions are learned from the CHiME-5 dataset, a unique dataset that contains long unscripted recordings of informal 4-person ‘parties’ recorded

across many homes. Analysing CHiME-5 allows us to gain insight into the natural behaviour of people in conversational settings. The data comprises recordings from Microsoft Kinect v2 devices placed unobtrusively at the edges of rooms. The devices contain a microphone array with an integrated camera. The video recordings, which have overlapping fields of view, allow speaker location to be estimated.

In order to accurately estimate the position of speakers in the room, several challenges need to be addressed. First, accurate speaker locations need to be estimated in the image space of each of the devices.

Second, in order to map from the device image spaces to the physical room space, the location and orientation of each of the devices need to be known. This step uses a calibration procedure that estimates the actual location of the devices given initial rough sketch estimates provided with CHiME-5 (Section 5.3). Finally, a procedure for mapping into room space is required that is robust to errors in the video annotation and camera parameter estimation (Section 5.4).

5.4.1 Speaker location annotation

An annotation tool[1] is used that employs a mixture of optical flow tracking and manual guidance to allow an annotator to efficiently and accurately track the location of each person's mouth (or estimated location in case of occlusion). Annotations are made at 100 ms intervals with occasional dropped frames in-filled via linear interpolation. Annotated tracks are reviewed and corrected as necessary.

The CHiME recordings are each around 150 minutes in duration. Each is composed of three separate phases of roughly equal length, focusing on activity in different areas of the living space (kitchen, dining, living room). A sample of the data is used, composed of 5-minute segments from the middle of each phase. There are 20 separate CHiME party recordings and so there are a total of 60 5-minute segments. Each of these is recorded with 5 or 6 devices, making a grand total of 342 video segments. The video may feature between 0 and 4 participants depending on the party phase and the device location. 186 of the segments were seen to contain at least one participant requiring annotation. Note that many of the environments are 'open plan' flats, so devices located in a living room, or kitchen area, can detect participants in the dining area, for example.

[1]The tool is available to use, https://github.com/jackdeadman/tracking-annotator

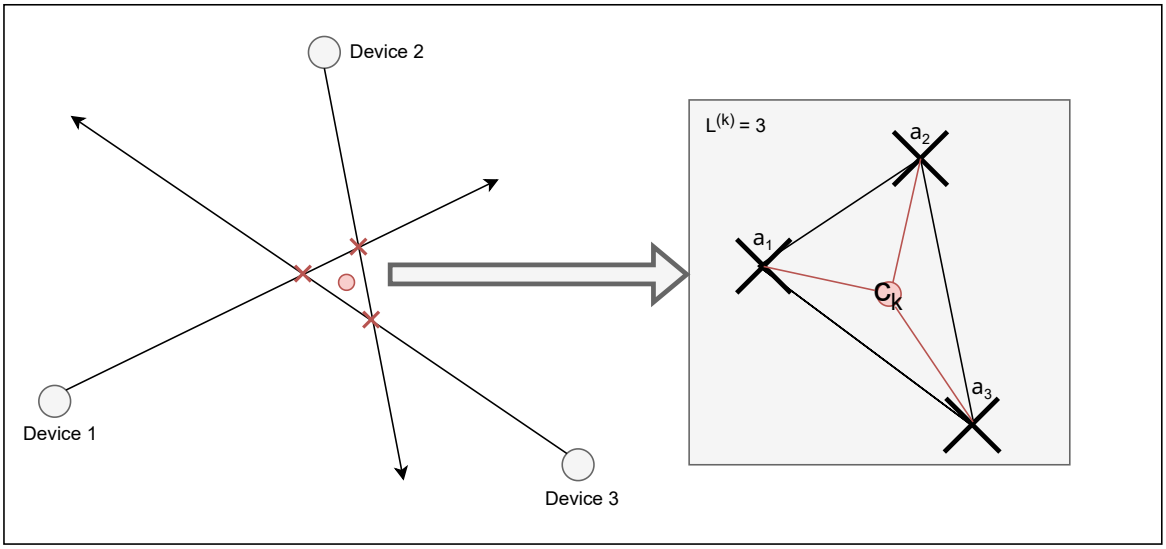

Fig. 5.1 Illustration of the aim of the loss function. Given some number of devices, each one of them will have a hypothesis of the angle of the speaker based on the camera estimate. In a perfectly calibrated system all the cameras will have hypothesis angles such that the lines coming out of the devices intersect at the true location of the speaker. Therefore the objective is to minimise the error in these misalignments. This is archieved through iteratively reducing the error between the centre of the intersections and each of the intersections.

5.4.2 Camera calibration

In the CHiME-5 dataset, the floorplan of each of the rooms is provided through sketches. These sketches include the walls and their measured length and rough locations of the devices and their rotations. This provides a good initial starting point for the true location of the devices, but if they are used naïvely, the final estimate of the participant positions will be poor.

To address this issue, the devices are calibrated using an optimisation procedure. If three cameras detect the same person, then the vectors produced from their observation angle should intersect at the same point. This is formulated by minimising the following objective function,

$$J(\Theta) = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{L^{(k)}}\sum_{l=1}^{L^{(k)}}\|\mathbf{a}_l^{(k)} - \mathbf{c}_k\|, \tag{5.1}$$

where K is the number of samples, $L^{(k)}$ is the number of intersections for sample k, $\mathbf{a}_l^{(k)}$ is the l point of intersection for sample k, and $\mathbf{c}_k$ is the centre point of the intersections, i.e., the objective is to minimise the distance between the intersections and the mean intersection point. Therefore, an optimal solution places the device such that all the devices "agree" with each other, a depiction of this loss function is shown in Figure 5.1. This objective function is

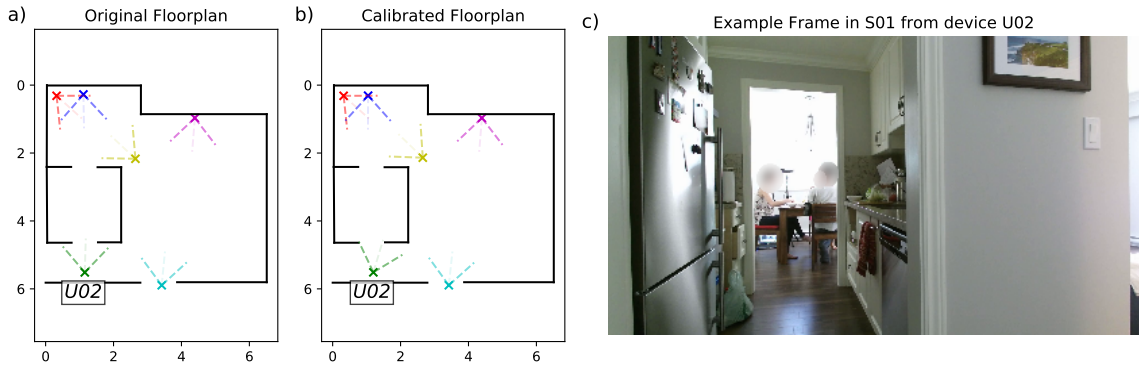

Fig. 5.2 Results from running the calibration process. The image shows the process has successfully calculated that the device U02 should be rotated in the floorplan. This calibration process has resulted in estimates of positions that are more plausible.

then minimised using stochastic gradient descent. The parameters (Θ) are the x, y coordinates and rotation of each of the devices (18 parameters in total for a session). A sample in this formulation is a vector containing the detected angle in each of the devices. If a camera does not detect the person, the computed gradient is set to zero for the corresponding parameters for that device.

The calibration procedure assumes that the devices remain stationary. The devices are supposed to remain stationary throughout a session. Although the devices are at fixed locations, analysis of the data indicates that small movements occasionally occur, presumably when they have been accidentally disturbed by participants.

Note also that the calibration process is using a 2-D geometry. For each device, three parameters are estimated (x, y, yaw) and three ignored $(z, \text{pitch}, \text{roll})$. Throughout the work, we are assuming that we are dealing with linear microphone arrays that are in the horizontal plane. Cues for source separation arise due to differences in the azimuthal angle of sources, not in their elevation. Hence, 2-D person locations are required to be estimated and the z-coordinate is ignored.

An example of the result of running this calibration procedure is shown in Figure 5.2. The figure shows how the calibration procedure has adjusted the orientation of device U02 to be rotated slightly clockwise relative to the original sketch. By then examining an example frame from the video, we can see this is a sensible adjustment as the camera direction after calibration better matches what we can see in the video, i.e., the initial floorplan indicates that the device is aligned to be facing parallel to the walls of the kitchen area, whilst in fact, it is rotated slightly to the right. Similar assessments were made for the other sessions to verify the calibration was working. (Note, however, the technique requires at least three cameras to see a person so it is not always possible for it to be applied).

5.4.3 Estimating speaker location

After camera calibration, for each frame, a person's true angle to the device can be estimated given the annotated observations in the image. We model this using a Gaussian distribution with the mean set to the angle given by the annotation d,

$$\theta \sim \mathcal{N}(\mu = d, \sigma^2). \tag{5.2}$$

The variance, σ^2, models inaccuracies in the annotation and in the estimation of the device angle. This parameter has been set empirically and is tuned to 10 degrees in the experiments that follow. This value was chosen by using the estimate of the errors within annotations shown in Section 3.6[2].

Given these annotations in isolated cameras, we can estimate the most probable location of speakers in a two-dimensional space. Given a 2D position in the room, an angle can be computed in each of the devices by projecting the position $\mathbf{p} = \begin{bmatrix} p^x & p^y \end{bmatrix}^\top$ into an angle for each of the devices where η_l is the rotation of the device l,

$$\text{wrap}(\theta) = \text{atan2}\left(\frac{\sin(\theta)}{\cos(\theta)}\right), \tag{5.3}$$

$$\text{project}_i(p^x, p^y) = \text{wrap}(\text{atan2}(p^y, p^x) - \eta_l), \tag{5.4}$$

These observation angles can then be combined to give a probability of a position given the annotations $o_1, ..., o_L$, where L is the number of devices,

$$P(p^x, p^y | o_1, ..., o_L) = \prod_{l=1}^{L} \mathcal{N}\left(\text{project}_l(p^x, p^y); \mu = o_l, \sigma^2\right), \tag{5.5}$$

If a speaker is not detected in a camera then the probability mass is distributed uniformly across all angles.

The probability function in Equation 5.5 is depicted in Figure 5.3. In the figure, an artificial setup is created with the true location of the person placed at the position (3, 3). An error has been artificially added to the simulated detections to show how the probabilities of the position change as more devices are added to the formula. Even though the final camera provides a poor estimate of the position, it does not skew the distribution away from the true location. Given this probability distribution, a location can be estimated by either choosing

[2]By using the linear relationship: $\sigma = 15\,\text{px} \approx \sqrt{10}^\circ$.

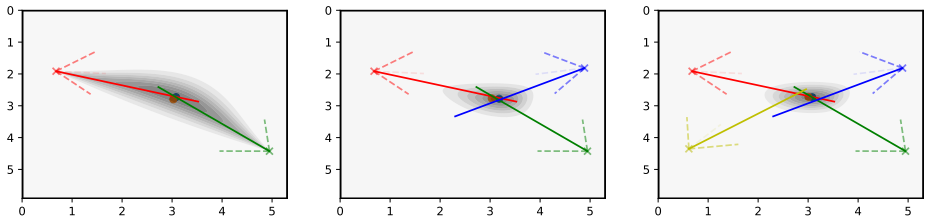

Fig. 5.3 Illustration of Equation 5.5 showing how adding more cameras changes the estimate of positions. The darker areas indicate a higher probability of the person being in that location given the detections in each of the cameras.

the peak,

$$\mathbf{pos}_{\max} = \underset{x,y \in W \times H}{\arg\max}\, P(x,y) \tag{5.6}$$

where the catesian product $W \times H$ represents the discrete grid of all possible positions in the room. Alternatively the expected value can be computed,

$$\mathbf{pos}_{\exp} = \left[\sum_{x \in W} x P_X(x) \quad \sum_{y \in H} y P_Y(y) \right]^\top. \tag{5.7}$$

The max point provides the most plausible estimates when the devices are well-calibrated and close to each other. In the more difficult cases, i.e., devices facing each other, the expected point resulted in more plausible estimates and the max point was found to be very sensitive to small changes in the image-space location estimates. For this work, the $\mathbf{pos}_{\exp}$ is used as the estimate of the speaker position.

5.5 Using 2-D positions to estimate mixture statistics

Next, we will explore how the methodology outlined is used to estimate positions of people in CHiME-5 and the information that can be extracted from the 2-D position estimates. Information such as an improved angular separation estimate and distance to the microphone.

5.5.1 CHiME-5 position estimates

Using the procedure described in the methodology, the 2-D positions of people in the dataset are computed. Without being able to evaluate the position estimates of the talkers directly with groundtruth positions, the estimates will be visualised and qualitatively evaluated.

The visualisation will compare the use of $\mathbf{pos}_{\max}$ against $\mathbf{pos}_{\exp}$ and the effect of using the calibrated device estimates. First, in Figure 5.4 the position estimates of the people for the

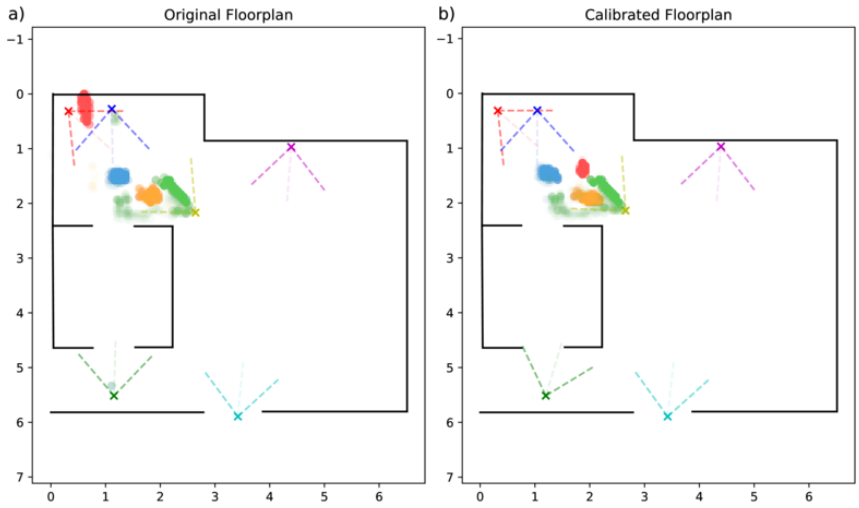

Fig. 5.4 **(S01, Segment 1, pos_{max}):** Without well-aligned devices the position estimates using the max results is some very implausible estimates, such as the red speaker being far away from the group when they are all eating their dinner. Once calibrated, all the position estimates seem sensible, even showing the green person moving around the orange.

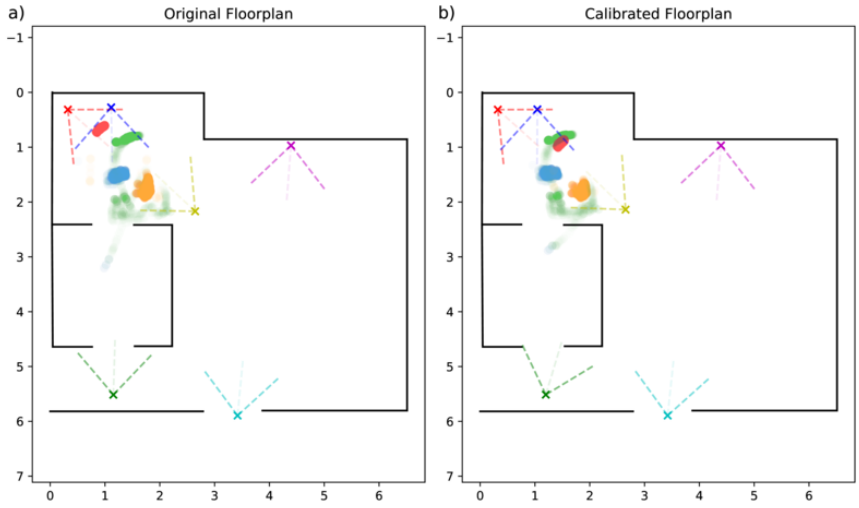

Fig. 5.5 **(S01, Segment 1, pos_{exp}):** Using the expected point results in estimates less sensitive to the camera misalignment problem. However, the improvement when aligned is less significant.

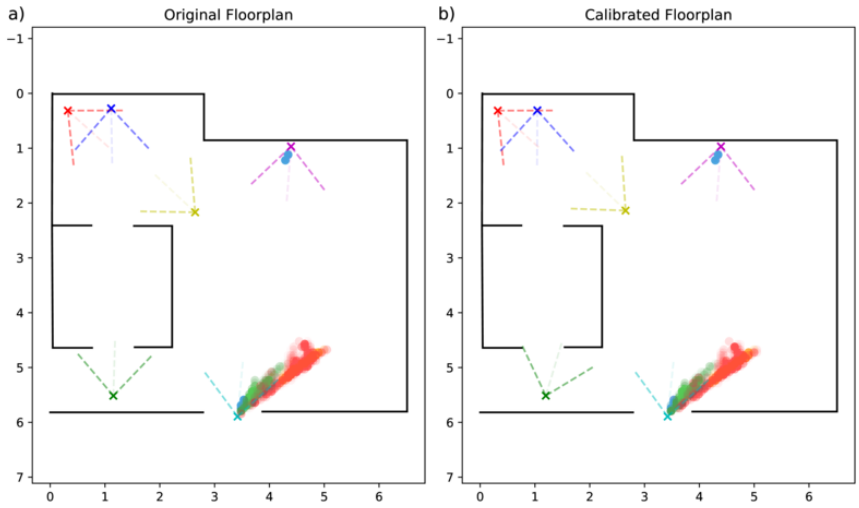

Fig. 5.6 **(S01, Segment 3, $\mathbf{pos}_{\text{max}}$):** When the maximum point is used the estimates are very poor when the cameras are facing toward each other.

first segment of session S01 are shown[3]. In these estimates, we can see that calibration has a large impact on the estimates. Given that during this segment, all the participants are eating food and seated around a dinner table, we would expect their positions to be stationary and located around a dinner table. In the uncalibrated plot (left) we can see that the red person is situated far away from the group. After calibration (right) we can see all the participants are now in more plausible locations. The reason for this large change in the estimate is due to the nature of the relationship between the distance away from the camera and the amount of the space that is visible e.g., if we are looking at a wall and then step back, we will see more of the wall. This means the error in the 2-D position estimate will be larger for devices that are far away from the true position of the speaker.

Next, if we look at the same session and segment but now use $\mathbf{pos}_{\text{exp}}$ as the estimate of speaker location. A visualisation of these positions is shown in Figure 5.5, here we can see the estimates are less sensitive to the misaligned devices. That is the red person is closer to the rest of the group compared to when using the $\mathbf{pos}_{\text{max}}$ as the estimator. However, the estimates of positions seem less plausible than the calibrated version of $\mathbf{pos}_{\text{max}}$. In $\mathbf{pos}_{\text{exp}}$ it is not clear where the green person would have been seated. That being said, the other speakers' positions all seem plausible and the movement of the green person has been captured. Both these figures have demonstrated the need for the calibration process.

Next, we will look at a later part in the same session. Using $\mathbf{pos}_{\text{max}}$ the positions from session S01 and Segment 3 are shown in Figure 5.6. In this part of the session, the participants

[3]Segment 1 is typically the cooking phase but there was no specified time limit for cooking. In this session the participants finished cooking quickly and therefore was not represented in this particular segmentation.

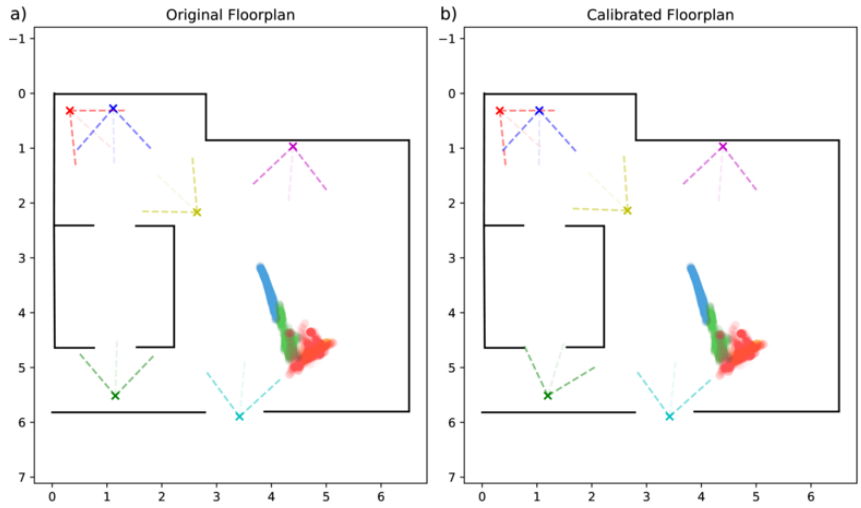

Fig. 5.7 **(S01, Segment 3, $\mathbf{pos}_{\text{exp}}$):** Using the expected point provides better estimates even when the cameras are facing toward each other.

are placed in the bottom right of the room playing a party game that involves some movement. This part of the room is more challenging for estimating the positions from devices. This is because the cameras are facing almost opposite each other, meaning the small movements in one of the estimates can lead to large changes to the position estimates. In the figure, we can see that the blue person has several estimates far away from the group and has appeared to have teleported across the room, which is very unlikely to have been the case. Again if we contrast this with the estimate from using $\mathbf{pos}_{\text{exp}}$ (Figure 5.7) we can see these erroneous estimates are no longer being made and the overall distribution of the positions is more spread out. Through manually inspecting the videos, these estimates appeared more plausible.

Finally, to conclude looking at the positions, another session in shown in Figures 5.8 and 5.9. In these figures, session S08 is shown for the third segment. Again similar observations can be made with $\mathbf{pos}_{\text{max}}$ producing less reliable results compared with $\mathbf{pos}_{\text{exp}}$. The plot also again shows the importance of a calibration process. Even though the estimates seem reliable they are not perfect, as demonstrated in Figure 5.9, here speakers are located outside of the room. Initially, this may seem like a very poor estimate as they cannot be outside of the room. However, when looking at the videos the participants were sitting on a sofa very close to the wall. This error could be due to the misalignment in the devices which calibration did not perfectly fix, or due to the floorplans provided not having been perfectly measured. With that being said, from inspecting the videos, the positions seem roughly correct. The green and blue people are seated on one sofa, whilst the orange and red are on another.

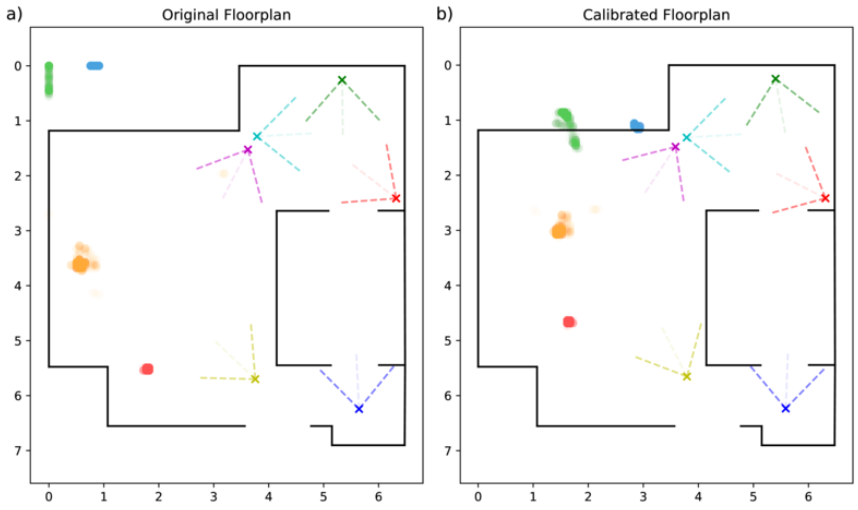

Fig. 5.8 **(S08, Segment 3, pos_{max}):** The estimates are not perfect, sometimes people can be predicted as being outside of the house. However, these estimates are still roughly correct given the participants were sitting on a sofa next to the wall.

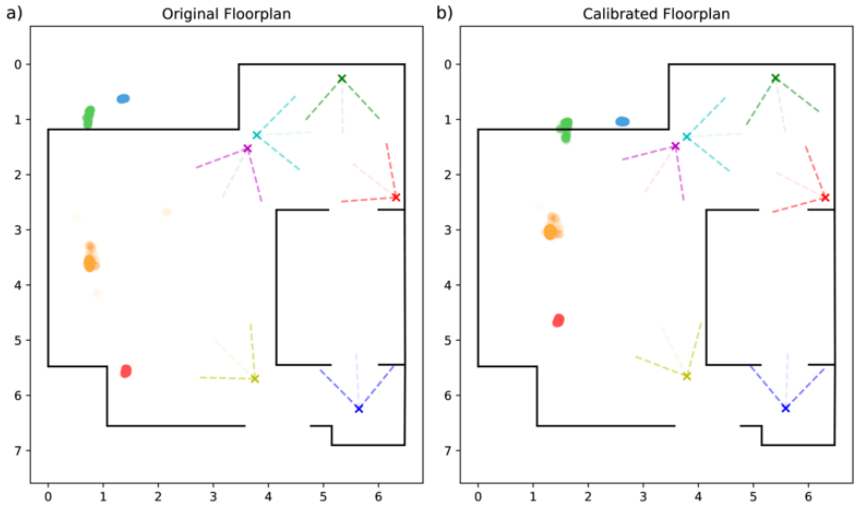

Fig. 5.9 **(S08, Segment 3, pos_{exp}):** Again the expected point has provided a better estimate of speaker positions when the cameras are misaligned. However, after, calibration the position estimates are outside of the room. This could potentially be caused by the walls being poorly sketched.

5.5.2 Estimating angular separation using 2-D positions

Using the procedure described in the methodology, the 2-D positions of people in the dataset are computed. This allows us to refine the estimate of the angular separation we previously reported. The automatic pose detection method used in the previous chapter was limited by the field of view of the device, i.e., separations of more than 84.1 degrees cannot be observed because at least one of the speakers would not be visible, and the probability of one speaker being out of view increases as the angular separation increases towards this limit. The automatic method also underestimates extremely small angles due to the speakers occluding each other. Now that the 2-D positions of speakers have been estimated, we can project these positions onto the devices to measure their separation angle relative to the device (even in cases where they cannot both be seen by the camera). This is important because even if the participant is not visible, their speech would still be recorded.

The plot in Figure 5.10 shows the updated angular separation from projecting the positions into a reference device. The reference device is chosen by selecting the device which on average throughout a segment people are closest to, this device is then constant for the entire segment. A reference device is chosen as not all devices will be relevant, e.g., a device that is in a room that the speech signal does not reach.

The single-device approach would not have detected people in that room but when using 2-D positions an angle can be computed no matter the device's relevance. The single-device estimates use all the devices that can see speakers and not a reference device as speaker distance is not available to those techniques and it is the techniques for estimating separation that is being compared.

The plot in Figure 5.10 shows that the average separation angle between speakers is still very low but not as low as was previously estimated when using single-device analysis. The plot also shows the separation of the speakers if the labels are used directly, which shows a similar distribution to that which was previously reported, validating the belief that the automatic methods provided a good estimate of speaker location. The labelled data also shows that the lack of separations at 0 degrees was due to the occlusions as this gap has now been filled in. The angles from using projections from 2-D positions show that the first estimated distribution was accurate but with a long tail that was missed, presumably due to the limited field of views of the individual devices.

5.5.3 Estimating speaker distance

Now that estimates of the 2-D positions have been computed, this now enables the estimate of speaker distances, both relative to one another and with respect to microphone arrays.

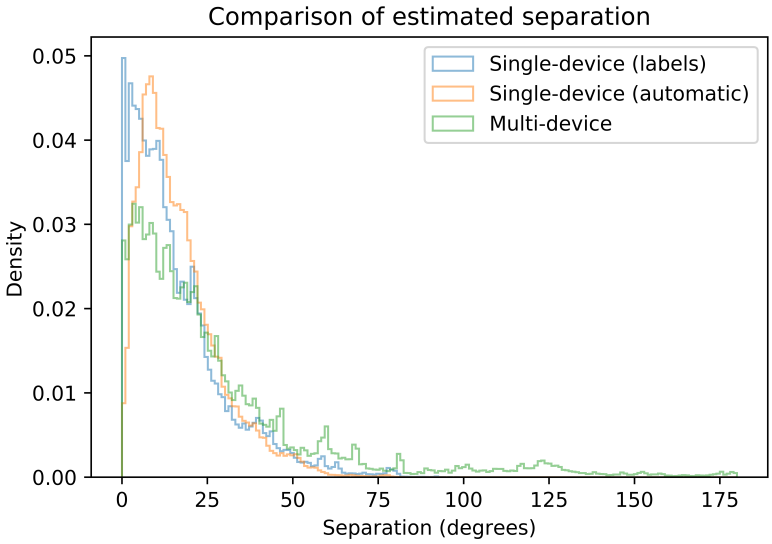

Fig. 5.10 Distribution of the angular separation estimates for different estimation approaches. Shown are two single-device approaches (one automatic and one using labelled data) and a multi-device approach that uses a combination of cameras to produce 2-D position estimates, which are then projected into the reference device.

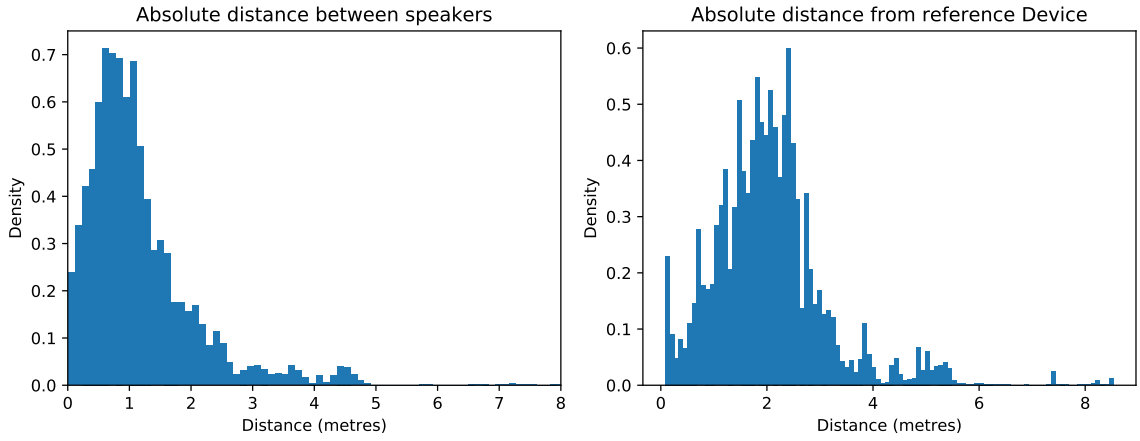

Fig. 5.11 Comparison of the absolute distance between speakers (left) and the absolute distances to the reference device (right).

Again using the reference device, for a frame in the video that has two or more people with estimated positions, two random people are selected and one is assigned to be the target speaker. In Figure 5.11 (left), the absolute distances between the speakers are shown. Here we can see that the participants are often within a metre of one another. This lines up with the literature of proxemics with states humans tend to be between 46 cm and 122 cm during interactions between friends and families (Hall et al., 1968), i.e., the scenario implemented in CHiME-5. It is worth noting a limitation in the position estimates: the distributions show some estimates where speakers are very close to 0 cm away from each other, clearly, this is not possible. However, the general shape of the distribution is encouraging, with a strong dip in distances classed at *intimate*, i.e., 1 cm to 46 cm. Next in Figure 5.11 (right), we can see the absolute distances speakers are away from the reference devices. Given that the devices are intended to be placed out of the way we would not expect participants to be close to the devices. The plot shows that on average people tend to be around 2 metres away from the devices but these distances can go as far as 5 metres indicating maybe one speaker is in a different room from the reference device (and therefore other participants).

Next, we will look at the relative distance ratio of speakers and interferes,

$$D = \frac{d_{\text{target}}}{d_{\text{interferer}}} \tag{5.8}$$

where d_{target} and $d_{\text{interferer}}$ are the distances away from the device for the target speaker and interferer respectively. The ratio of these distances allows us to interpret the potential impact with respect to SNR. Just looking at the absolute distance (Euclidean) between speakers is insufficient for approximating the relative level of their speech signals at the microphone. For example, speakers being 1 metre apart when the microphone is 1 metre away from the target speaker will produce very different relative levels than if the target is 5 metres from the microphone.

The division in the relative distance distribution will produce a long tail therefore when visualising the resulting data it is easier to see the distribution in the log domain. Next, we will look at how these real positions compare with the positions we get from randomly placing people uniformly inside rooms. The plot in Figure 5.12 compares the absolute value of the log ratio of the target speaker and interferer, i.e., $|log_{10}(D)|$. The absolute value is taken as the interferer and target is randomly chosen and therefore the ordering is arbitrary[4]. In Figure 5.12 we can see that, on average, people stand closer together in this social setting as compared to positioning randomly. This means that given a large room, people place themselves relatively close to each other rather than spacing themselves across the entirety

[4] Alternatively, the target could have been chosen to always be the closest person.

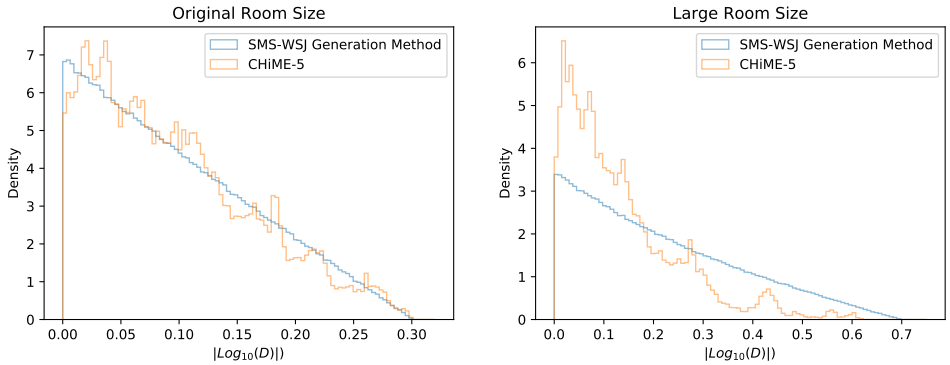

Fig. 5.12 Comparison of the absolute log of the ratio between speaker and competing speaker. Under the constraint, speakers are between 1 and 2 metres (left) from the device, and speakers position themselves somewhat randomly. In a larger room (right) setting they position themselves closer to each other i.e., form a group.

of the room. This is an expected result but is confirmed by the data. The plot on the left of Figure 5.12 shows the relative distance of speakers when we constrain the estimates to be in the range of positions that can be found in SMS-WSJ. In the plot on the right, the distribution in SMS-WSJ is extended to between 1 and 5 metres and the range in CHiME-5 has been matched. From this plot we can see that the distribution is random when looking at a small room. But when looking at a larger room, people tend to gather in groups.

Next we will look at the range of absolute distances and relative distances observed in the dataset. Figure 5.13 looks at the joint distribution of d_{target} against $\frac{d_{\text{target}}}{d_{\text{interferer}}}$. Here we can see that most of the data is concentrated around small absolute distances. Again, this plot would benefit from being displayed in the log domain. In Figure 5.14, the same data is shown in a log-log plot. This now highlights the spareness of the data. Even though the dataset consists of many hours of parties, people will only move around the room for a limited amount of time, and the distances possible will be limited to very few room geometries. The plot also shows the parallelogram nature of the distribution. This is due to the fact the relative distance's maximum is limited by the room size. The distribution shape can be explained by looking at the extremes. When the target speaker is very close to the microphone array all of the possible distances for the interfering speaker will be *at least* as far as them i.e., the left side of the plot. On the other extreme, where the target speaker is as far away from the microphone as possible i.e., the right-hand side of the plot. The interfering speaker will be *at most* as far as the target and most likely closer to the microphone.

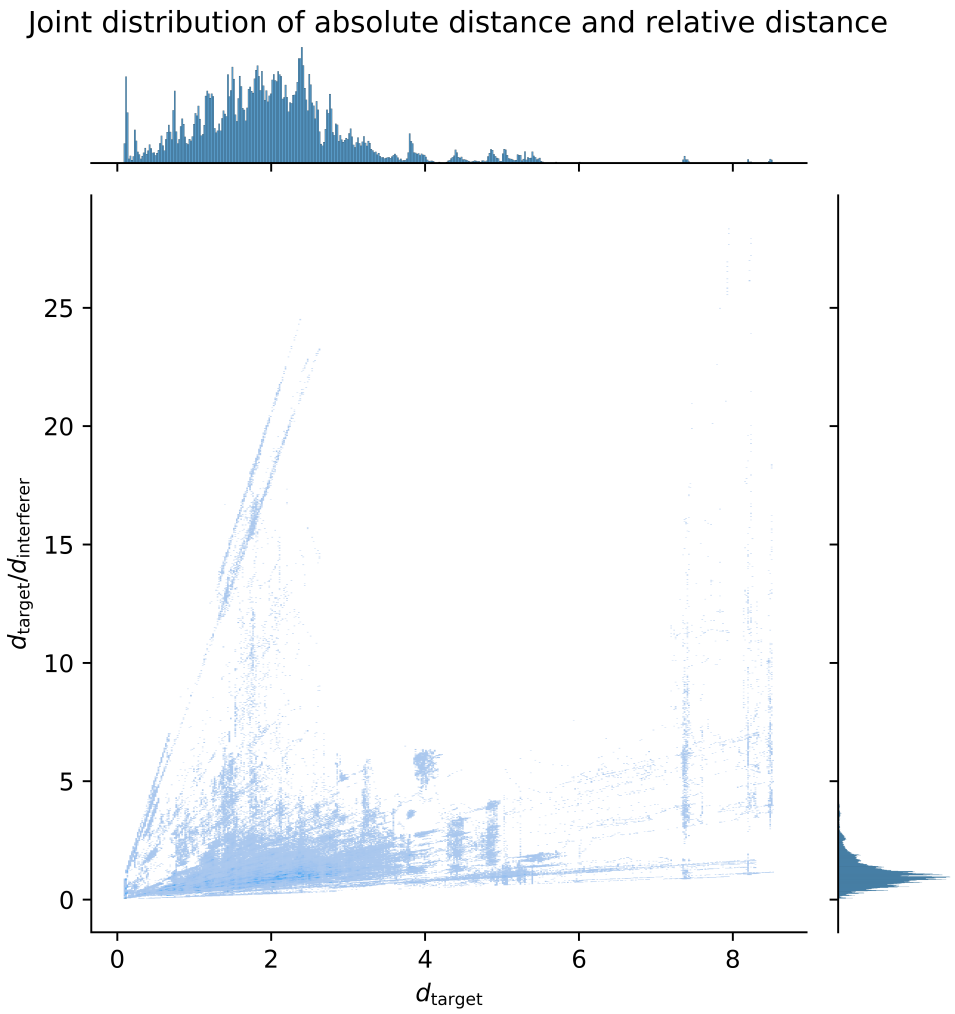

Fig. 5.13 Joint distribution of the target speaker's absolute distance from the microphone against the relative distance to a competing speaker.

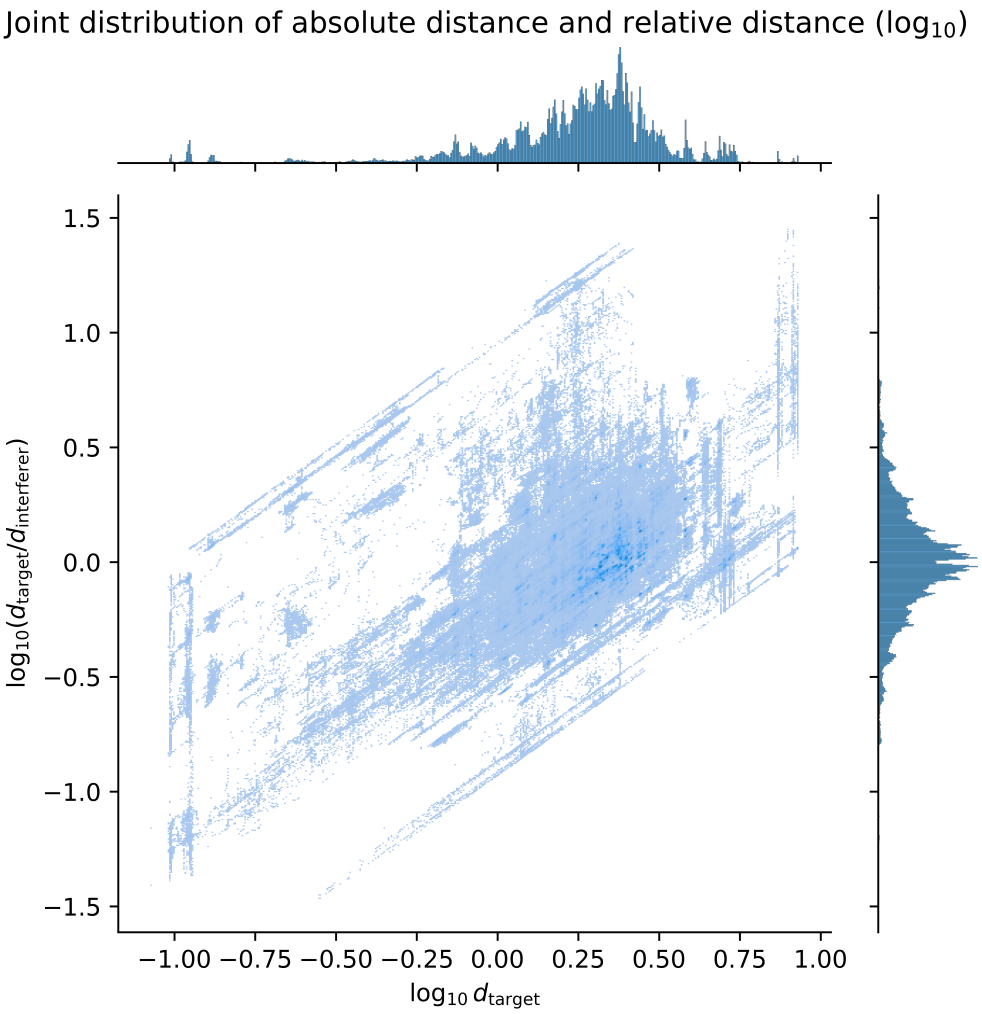

Fig. 5.14 Joint distribution of the target speaker's absolute distance from the microphone against the relative distance to a competing speaker.

5.6 Realistic speaker location in simulation

Following a similar procedure to that of the previous chapter, to evaluate the impact that speaker positioning has on source separation and ASR, a series of datasets are created that use data-driven speaker location estimates to provide realistic setups. Using the same baseline system presented in Chapter 4 and trained on the same data. ASR experiments are run to show the impact that these setups have on the performance of speech separation and recognition to illustrate the potential impact of the mismatch between typical simulation and real data. In particular, the experiments aim to explore if the updated angular separation still has a large impact on the difference in performance between real and simulated data that we observed in the previous chapter. The experiments also aim to explore if enforcing the relative distance distribution also impacts speech separation and ASR. Again speech separation, intelligibility and ASR metrics are reported in the results to see how an overall impact on performance for the different datasets.

5.6.1 Experimental setup

Experiments use the baseline system described in Drude et al. (2019b), namely, a complex angular central Gaussian mixture model (cACGMM) mask estimator is used with a minimum variance distortionless response (MVDR) beamformer and a factorised time-delayed neural network (TDNN-F) based acoustic model. The experiments measure how the baseline performance changes when the SMS-WSJ dataset enforces realistic speaker distributions. This compares the impact of the relative distance distribution and the updated angular separation distribution. To account for the dependency on the change in relative distance distribution when the absolute distance is larger, an additional set of datasets are created with a room size mean set to (12, 8) which samples speaker distances between 1 and 5 metres, which is named *large*. The *original* dataset has a mean room size of (8, 6) and samples distances between 1 and 2 metres.

For realistic separation, the angular separation of the speakers is sampled from a Gaussian kernel density estimate of the angular separation distribution from multi-device as shown in Figure 5.10. For realistic relative distance, the first speaker's absolute distance is sampled in the same way as SMS-WSJ. The competing speaker's distance is then drawn by sampling from a conditional distribution. The conditional distribution is computed by first modelling the joint distribution of absolute distance and relative distance, again using a Gaussian kernel density estimate.

Table 5.1 Results from the complex angular central Gaussian mixture mode (cACGMM) baseline system comparing several datasets with *fit* (F) and an *uninformed* (U) distributions for angular separation (Φ) and relative distance (D).

	Name	Φ	D	PESQ	STOI	SDR	WER
Original	SMS-WSJ (Drude et al., 2019b)	U	U	2.07	0.82	12.35	18.25
	Single Device (Chapter 4)	F*	U	1.85	0.74	9.0	31.49
	Original+Φ	F	U	1.91	0.76	9.80	28.25
	Original+D	U	F	2.06	0.82	12.17	18.49
	Original+Φ+D	F	F	1.90	0.76	9.79	28.09
Large	Large-SMS	U	U	2.01	0.78	11.38	22.49
	Large+Φ	F	U	1.84	0.73	8.40	34.21
	Large+D	U	F	2.04	0.80	11.59	21.73
	Large+Φ+D	F	F	1.83	0.73	8.09	36.07

**Angular separation distribution fit on different data.*

5.6.2 Comparing the use of the distributions in large and small rooms

The results in Table 5.1 show the outcome of the experiments. An experiment is denoted by the format {room size}+Φ+D, where Φ+D are included if the angular separation and relative distance are using the distributions fit on the real data. When an *uniformed* i.e., original distribution is used, these symbols are not present in the name (e.g., Large+Φ means the angular separation was fit on the real data but the relative distance distribution is uninformed, and the room size is the large variant).

The results show that the updated angular separation still has a large impact on the performance of ASR and speech separation, but slightly less extreme than first reported in the previous chapter, this is due to the distribution producing very few narrow mixtures. The impact of fitting the relative distances shows a more complex relationship. In the original room size of SMS-WSJ the performance of the system decreases by a very small amount when the relative distance distribution is enforced. The performance however increases when this is combined with the angular separation, again by a small amount. When extended to a large room the results show that a realistic distance results in an easier dataset for separation. To explore why this may be the case, the results of the large room experiments are broken down next.

Table 5.2 Enhancement and ASR performances when using MVDR with estimated masks (cACGMM), oracle masks (IBM), or directly using pre-mixed signals (Image) for large rooms under various speaker spatial distributions: baseline (Large+SMS), baseline plus realistic distances (Large+D), plus realistic angular separation (Large+Φ), or both (Large+D+Φ).

Dataset	Mask	Enh	PESQ	STOI	SDR	WER
Large-SMS	cACGMM	MVDR	2.01	0.78	11.38	22.49
	IBM	MVDR	2.01	0.80	12.10	16.68
	Image		2.00	0.80	13.20	9.66
Large+D	cACGMM	MVDR	2.04	0.80	11.59	21.73
	IBM	MVDR	2.05	0.81	12.36	16.41
	Image		2.03	0.81	13.53	9.63
Large+Φ	cACGMM	MVDR	1.84	0.73	8.40	34.21
	IBM	MVDR	1.88	0.76	10.23	21.81
	Image		2.00	0.80	13.24	9.80
Large+Φ+D	cACGMM	MVDR	1.83	0.73	8.09	36.07
	IBM	MVDR	1.88	0.76	10.13	21.96
	Image		2.02	0.81	13.42	9.45

5.6.3 Analysis of the impact of positioning in large rooms

Next, the results in Table 5.2 show the breakdown of the performance within a large room. First looking at the "images" which represent the spatialised versions of the utterances before any mixing. All datasets give similar WER performances on these images, as expected as the utterances have not changed and they are contained in the same set of room sizes. The WERs for the images show the best performance we could expect for the utterances based on their location i.e, how much the reverberation is affecting performance. We can see that Large $+ D$ produces very similar results. But comparing Large $+ \Phi$ and Large $+ \Phi + D$ we get some interesting results. The image for Large $+ \Phi + D$ shows the raw spatial images are the easiest. But after mixing and then separating the results switch i.e., Large $+ \Phi$ now performs better than Large $+ \Phi + D$. This could be because without the Φ distribution the separation is performing so well (competing source completely removed) the impact of the relative distance is not felt. Therefore it is important for both of these distributions to be modelled together and simply the relative distance alone is not enough. The results also show how intelligibility metrics can be a poor predictor of ASR performance. In Large $+ D$ the intelligibility metrics are very similar between cACGMM and the ideal binary mask (IBM) but the WER has a large difference, with the IBM being far better.

5.7 Discussion

In this chapter the need for a data-driven approach to relative distance has been shown to be room size dependent, therefore this should be another consideration for possible consideration when designing simulations. Room sizes are often quite arbitrarily defined and their shapes are simple rectangles (i.e, shoe boxes). Further analysis considering the room geometry would give further insight into this relationship.

In the results, the importance of both modelling the relative distance alongside the angular separation has been shown. If sources are being well separated and then the relative distance and therefore SNR of the initial mixture is irrelevant if the noise is being completely removed. Therefore further work in considering training acoustic models on partially separated mixtures under the Large$+\Phi+D$ conditions could potentially improve the performance. This complex relationship shows an important motivation for why we should have realistic simulations. If we believe our systems can separate mixtures this well because of wide angles, then we will never see that when people are closer to each other our acoustic model is failing. Rewarding improving separation across all angles evenly (as currently being done) will not benefit eventual ASR tasks, if, we need to improve robustness to narrow angles and narrow relative distances.

5.8 Conclusions

In this study, an analysis of the relative distance between speakers and competing speakers at unscripted dinner parties has been contributed. A methodology and the challenges involved in deriving this estimate due to uncalibrated cameras from rough floorplan sketches. The analysis also contributed an updated estimate of the angular separation of speakers in the CHiME-5 dataset, a refinement of the previous chapter. The experimental work shows the relationship between angular separation and the challenges it produces when the angle narrows. The work has also demonstrated the complicated relationship of relative distance and its effect on performance and the importance of modelling it alongside the angular separation.

The speaker location labels for a subset of CHiME-5, are released alongside this work[5], allowing for the analysis to be reproduced. Derived 2-D positions are also released alongside this, to allow for further analysis. In addition to this, the RIRs and metadata for the datasets produced have been released, which can be used as additional benchmarks, i.e., allowing

[5]https://chime.jackdeadman.com

the community to analyse the performance of their ASR systems with respect to angular separation and relative microphone distance in a comparable manner.

Chapter 6

Speaker temporal analysis: modelling speaker turn-taking

6.1 Introduction

Automatic speech recognition (ASR) is a challenging task with numerous factors contributing to its difficulty, with recognising multi-party conversations in real, noisy environments being one of the most difficult scenarios. Recent ASR challenges have shown that much work is left to be done (Barker et al., 2018; Watanabe et al., 2020). It is therefore vital to break down the factors that are contributing to the difficulty. In this thesis so far, we have explored the impact of the angular separation between speakers and the microphone distance by analysing videos in the CHiME-5 dataset (Barker et al., 2018). This chapter extends this analysis by looking at speakers' temporal behaviour, i.e., their turn-taking behaviour.

Generating realistically overlapped speech data is crucial for the development of better conversational ASR systems. There has been much recent work in speech separation (a crucial ASR component) but it has mainly focused on highly-overlapped mixtures (Cosentino et al., 2020; Drude et al., 2019b; Hershey et al., 2016). This data poorly models the real challenges. For example, applying separation techniques can be detrimental to ASR if the models attempt to extract more sources than those that are present (Sato et al., 2021), hence good estimates of the number of active speakers are needed. However, this is trivial in fully-overlapped cases. More recently, attention has turned to *sparse* versions of commonly used datasets (Menne et al., 2019) that use a parameter to govern the amount of overlap when creating mixtures. This is a step towards creating more *realistic* simulations but still fails to model the complexity of real conversations.

The simulated dataset most closely capturing real conversation dynamics is LibriCSS (Chen et al., 2020), which creates long-form parties containing many utterances from several speakers. It can be processed into segmented mixtures, directly used for tasks such as diarisation, or used in experiments with enhancement systems requiring long context windows (Kanda et al., 2019). However, LibriCSS contains just 10 hours of data, which are recordings of audio played back in a room. Therefore, the dataset is only appropriate for evaluation rather than training.

The work presented in this chapter aims to greatly extend this by allowing for arbitrarily large amounts of data to be produced from generative models. These models will be derived from analysis of the 50 hours of conversations recorded from 20 parties in the CHiME-5 dataset (Barker et al., 2018).

The uses of the generative overlapped speech simulation techniques presented in this chapter extend beyond ASR to the task of diarisation. The recent trends in diarisation has been towards jointly optimising voice activity detection and segmentation (Bredin and Laurent, 2021), and towards end-to-end systems (Fujita et al., 2019). If such systems are to be trained using simulated datasets, the modelling of realistic turn-taking behaviour is essential.

In this work, we will establish a framework for analysing the turn-taking behaviour of people in real-life recordings and establish a method for extracting how much of the difficulty of a recording can be attributed to the turn-taking behaviour. In Section 6.2, we discuss the complexity of human turn-taking. In Section 6.3, we introduce a framework for modelling the turn-taking behaviour of multi-person "parties" using a simple finite-state representation. Section 6.4 shows that representations can be created that can be used to characterise recordings in real datasets with interpretable meanings. These representations are then evaluated in Section 6.5 in the context of target-speaker extraction, where the representations can predict the difficulty of a mixture purely based on turn-taking behaviour. Finally, Sections 6.6 and 6.7 concludes our findings.

6.2 Background

Modelling the turn-taking behaviour of humans is a well-established research field (Schegloff, 2000; Skantze, 2021). The behaviour of people changes depending on numerous factors such as their environment, who they are talking to and whether the conversation is physical or virtual. Modelling turn-taking is also a multi-modal activity, where cues are not always verbal. Gazes are often used to select the next speaker, or head nods are used to indicate confirmation and encourage the speaker to continue talking.

Predicting who is speaking next is of interest across a wide range of applications. For example, more natural human-computer interactions can be realised if the virtual agent can naturally interject and wait their turn when appropriate. Virtual agents themselves have a wide range of applications such as in clinical settings (Mirheidari et al., 2019) and within smart-home echo systems.

Much work has been done towards *predicting* how the turns will develop, allowing for conversational agents to naturally take their turn and add backchannels without the perception of interruption (Ekstedt and Skantze, 2020).

In this chapter, instead of predicting turn-taking behaviour, we aim to observe real turn-taking and generate more data following the observed distribution. Whilst prediction will require observations of the speaker (e.g., a video recording), a generator will not require a recording. The generator can model characteristics of this behaviour by looking at the resulting turns that were taken even if that turn occurs due to non-verbal behaviour (i.e., to describe the patterns we only need to observe them, and do not need to understand their cause.)

In previous work, when generating simulated parties for the use in diarisation, simplistic approaches have been used, such as in (Fujita et al., 2019), where speakers are treated independently. This completely neglects how speakers interact with each other. Agent-based models for generating speaker turns have been explored (Padilha, 2006), where participants are parameterised by engineered features such as *talkativeness* (desire to talk), *confidence* (persistence to talk when others are talking), *verbosity* (desire to continue talking). Motivated by this, we aim to build a structure that can learn these parameters from the data.

6.3 Framework for modelling turns

In this section, we introduce a framework[1] for modelling turn-taking solely based on utterance timings, i.e., looking at the turn-taking behaviour of speakers while ignoring any linguistic and acoustic cues that will come with these signals. This simple approach allows for easily computed models that can be fitted to a wide array of datasets, i.e., not all datasets provide fully transcribed text but most provide end-pointing and speaker identity.

With a transcript providing start and endpoints of utterances in a party, a discrete representation of K observations can be created through sampling at a predefined frame rate f_s with no overlapping frames. In this chapter the value $f_s = 100$ is used for all experiments. Given a party with a set of speakers $\mathbb{P} = \{a, b, c, d, ...\}$. The state of the speaker activity of the

[1] Python Package: https://github.com/jackdeadman/turn-taking

party is defined as matrix $\mathbf{Y} = \left[\mathbf{y}_1, ..., \mathbf{y}_K\right]^\top$ where $\mathbf{y}_k \in \{0,1\}^J$ and $J = |\mathbb{P}|$ i.e., the number of speakers. The value $\mathbf{y}_k^j = 1$ if speaker j is speaking at frame k and $\mathbf{y}_k^j = 0$ otherwise. A speaker is speaking in a frame if at any point in the frame, the transcript denotes they are speaking.

6.3.1 Finite-state model formulation

We want to build a vector representation $\boldsymbol{\phi}$ to summarise the speaker activity matrix, $\mathbf{Y}$. We propose to achieve this through training a generative model $M_{\boldsymbol{\theta}}$ and computing features from the learnt parameters $\boldsymbol{\theta}$, i.e., $f(\boldsymbol{\theta}) = \boldsymbol{\phi}$. Finite-state models have been ubiquitous in speech recognition history (Trentin and Gori, 2001) with their relatively simple design and well-defined mathematical properties. They have also been shown to be useful for modelling and predicting turn-taking (Raux and Eskenazi, 2009). Using the parameters of hidden-Markov models has been well established in many tasks requiring speaker representations, e.g., speaker recognition (Garcia-Romero and Espy-Wilson, 2011), speaker verification (Campbell et al., 2006) and adaptation for robust speech recognition (Kuhn et al., 2000; Senior and Lopez-Moreno, 2014).

We propose to model the behaviour of speakers as a series of observations $\mathbf{y}_k$ being generated by a model depending only on $\mathbf{y}_{k-1}$. This can be achieved through Markov models where states indicate the active speakers at that point in time. First, *fully-connected* Markov model, where every possible combination of speakers has a state, $\mathbb{S}^{\text{full}} = \mathscr{P}(\mathbb{P})$ and an $|\mathbb{S}| \times |\mathbb{S}|$ transition matrix $\mathbf{T}^{\text{full}}$ where $\mathbf{T}_{mn}$ is the probability of transitioning from state m to state n. The full model requires many parameters ($\mathscr{O}(2^J)$). Therefore we will also explore a further model which requires fewer parameters ($\mathscr{O}(J)$) by treating speakers as independent generators with their own Markov models. An *independent* model with a set of independent models $\mathbb{S}^{\text{ind}} = \{s_1^{\text{ind}}, ..., s_J^{\text{ind}}\}$ where $s_j^{\text{ind}} = \{\{\mathbb{P}_j\}, \varnothing\}$, with the corresponding transition matrices for each of the sub-models $\mathbf{T}^{\text{ind}} = \{\mathbf{T}_1^{\text{ind}}, ..., \mathbf{T}_J^{\text{ind}}\}$. In all these models the null state $\varnothing$ represents the silent state.

The total time spent inside of state follows an exponential distribution, which means very small durations are highly probable, but this will not be the case in real data. To account for this the modelling power of all Markov models can be further extended through fitting a time distribution $P_s(D = d; \Theta_s)$ on each of the states s for a time d, making the model semi-Markovian (Janssen and Limnios, 1999). The time spent in the state is drawn from this distribution instead of being a function of a state self-transition probability. Graphic representations of these models are presented in Figure. 6.1 and Figure. 6.2 for the *independent* and *fully-connected* models respectively.

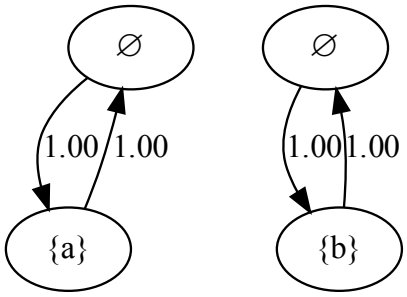

Fig. 6.1 Diagram of the finite-state representation for the *independent* model. The state names represent the active speakers when in that state. Each of the speakers has a their own sub-model inside of the larger turn-taking model. The time spent inside each of the states is drawn from a time distribution before transitioning to the next state.

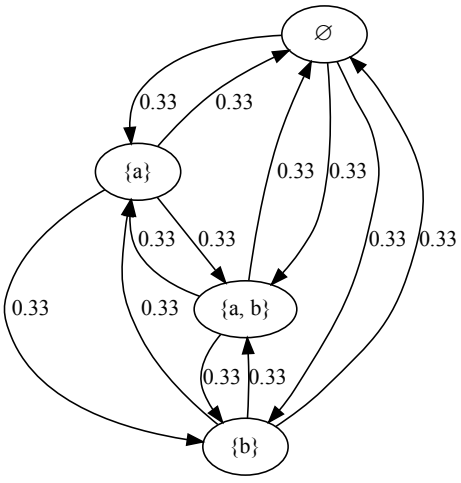

Fig. 6.2 Diagram of the *fully-connected* model. The states represent all the possible combinations of speakers.

6.3.2 Training models

In order to train a model, the training data needs to be gathered by transcribing real turn-taking behaviour. This can be derived from fully-transcribed text such as for ASR tasks or endpoints with speaker identity such as from diarisation tasks. The endpoints for each of the speakers should be time-aligned, i.e, if individual close-talking microphones are used, they should be synchronised upon the beginning of the recording.

Given the endpoints for a session, sampling at the sample rate f_s produces the activity matrix $\mathbf{Y}$. This can be interpreted as a sequence of states e.g., $\begin{bmatrix} s_1 & s_1 & s_1 & s_2 & s_2 & s_2 & \dots \end{bmatrix}^\top$, this data then needs to be converted into data that can be used to fit the model. The parameters of the model are a combination of the transition weights between the states and the parameters of the time distributions. First, the transition weights can be estimated by counting the number of times transitions are made between state pairs,

$$\mathbf{T}_{mn} = \frac{\eta_{mn}}{\sum_m \eta_{mn}}, \tag{6.1}$$

where η_{mn} is the count of the transitions from s_m to s_n. This is the same process for all the turn-taking models.

Next, the state duration distributions $P_s(D;\Theta_s)$ are fit by counting the number of consecutive samples that have the same state, i.e., this is computing the duration in the state. This process will therefore lead to a training dataset where each sample is an integer i.e., the durations it observed in the states. These time datasets are then used to fit the duration models inside each state according to the appropriate parameter estimation technique for the model, e.g., using maximum likelihood estimation (MLE).

It is important to note that the ordering of speakers is arbitrary and reordering the speakers will result in a change in the parameter estimations. In order to mitigate this issue, the speaker identities are assigned based on activity, i.e., $\mathbb{P}_1$ is always the most active, and $\mathbb{P}_K$ is the least.

To make this training procedure concrete and to illustrate some differences between the data between sessions in the CHiME-5 dataset, examples of the state duration distributions are shown in Figure 6.3 and Figure 6.4. In both these examples the *fully-connected* model is trained with activity Y generated using $f_s = 100$ using data from the entire session. Within the states a Wald distribution[2] is used for $P_s(D;\Theta_s)$ and the scale and mean parameters are fit using MLE.

There are several observations we can make from looking at these plots. First, as expected the durations observed in single-speaker states are far longer than when we compare with states with multiple active speakers. This is because people tend to try to not interrupt each

[2]https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.wald.html

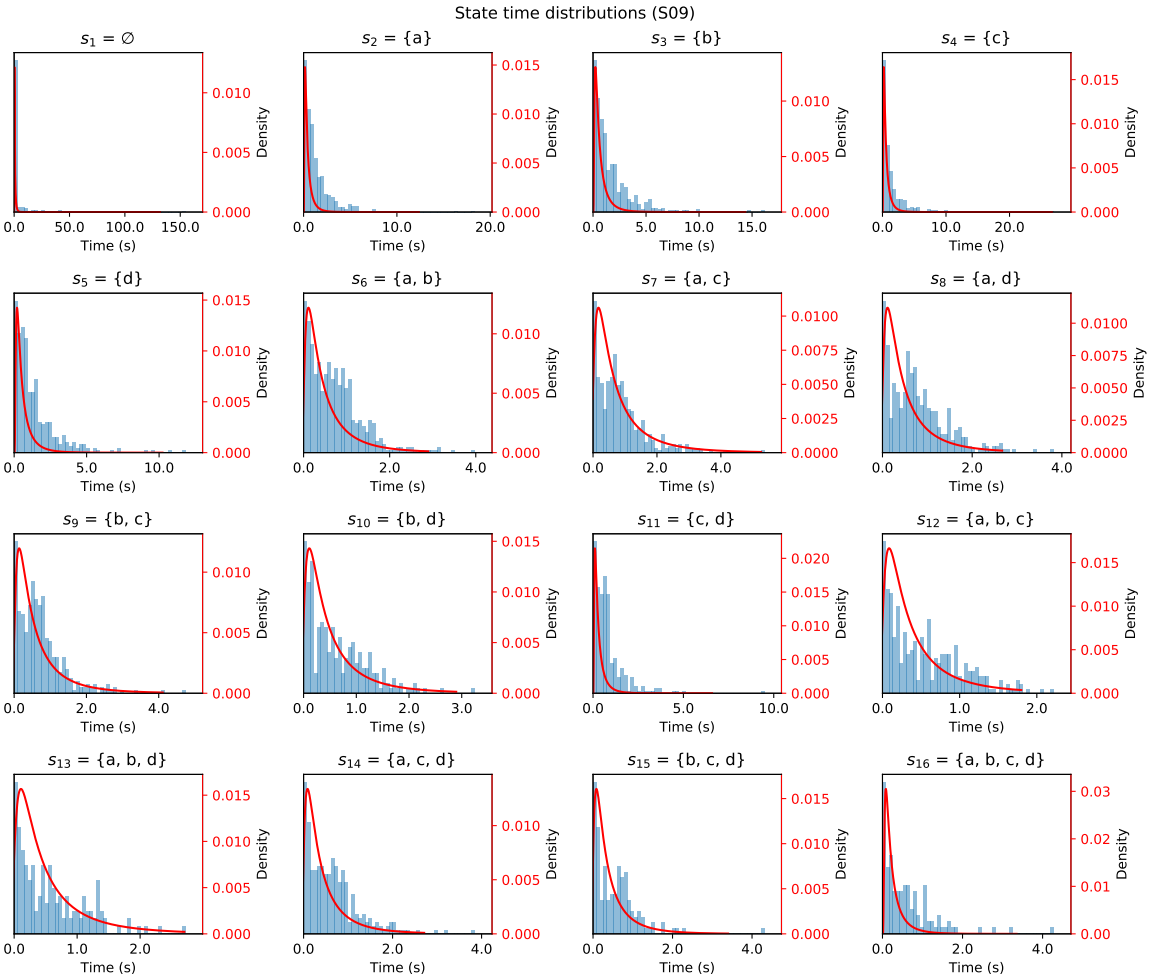

Fig. 6.3 *Fully-connected* model's state distributions fit on **S09** using a Wald distribution to model the duration.

other and when an interruption does occur the other person will quickly stop, it is unlikely long sustained simultaneous speech will occur. Next we can compare across session S09 (Figure 6.3) and S21 (Figure 6.4). Here we can see in session S21, fewer observations were seen for states involving many speakers and when they do occur they are shorter when compared with S09. Indicating the group of individuals in S21 prefer to take turns talking instead of interrupting one another and S09 consists of a group of dominant speakers.

6.3.3 Sampling models

For the model to be used for generating turn-taking data, state sequences need to be sampled from the generative models, this creates the activity matrix Y. This can then be used in applications such as diarisation to generate turns or as we will explore later in the chapter,

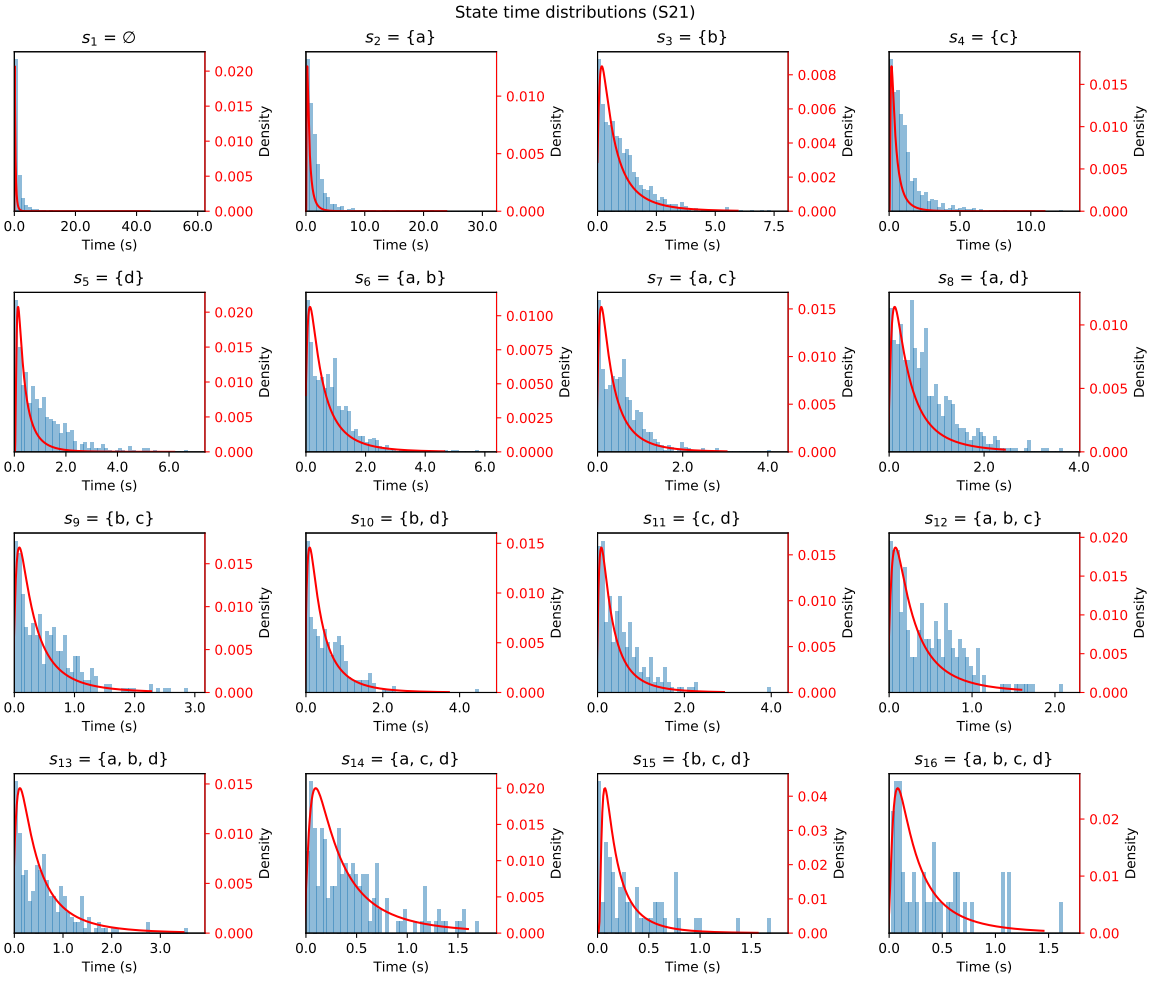

Fig. 6.4 *Fully-connected* model's state distributions fit on **S21** using a Wald distribution to model the duration.

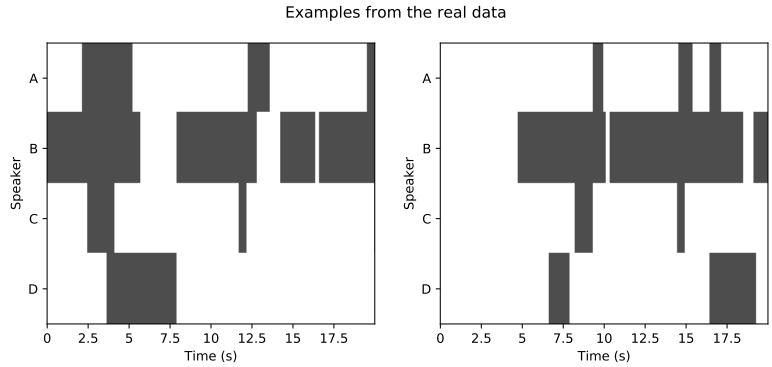

Fig. 6.5 Two 10 second segments taken from CHiME-5. The black regions indicate parts in the audio where people are talking.

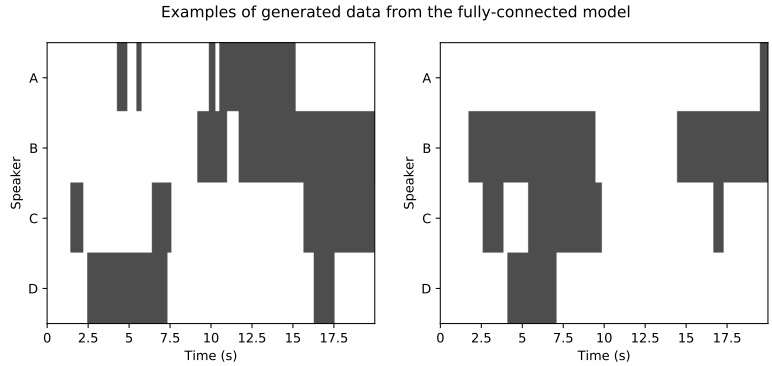

Fig. 6.6 Data generated by the *fully-connected* model. The model is fit using the entire session that Figure 6.5 shows a segment of.

speech separation where mixtures can be created with realistic overlap and placement of overlaps.

To generate an activity matrix $\mathbf{Y}$, the model is initialised at a starting state (e.g., silent state), sampling the duration from the state distribution $P_s(D;\Theta_s)$ and then sampling the next state based on the transition weight. The model then transitions to this state, and a new duration is sampled. The process is repeated until a desired number of samples have been generated. For the *independent* model, this process can be done in parallel across the sub-models.

To compare the generative capabilities of the *fully-connected* and the *independent model*, data from the entirety of session S09 in CHiME-5 is used to fit the models. By using a sample rate $f_s = 100$ and the Wald distribution for state duration distribution P_s the models are fit on the entire sessions and then a set of samples are generated starting at the silent state. In Figure 6.5 a segment from the real data is shown, this illustrates the kind of turn-taking that occurs in the real scenario, we can see it is quite structured with sensible locations for

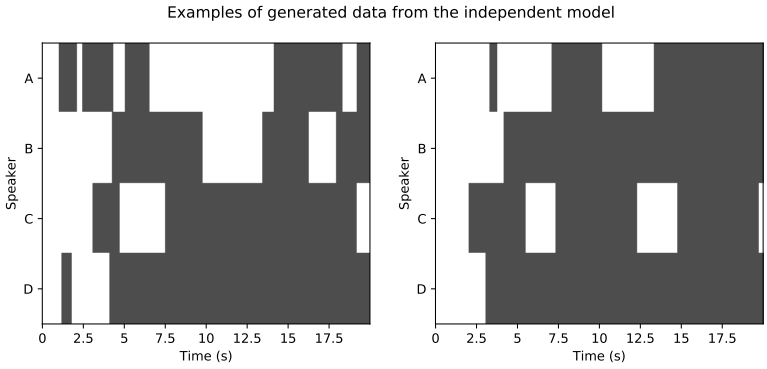

Fig. 6.7 Data generated by the *independent* model. The model is fit using the entire session that Figure 6.5 shows a segment of.

interrupts and that it is not too overlapped. Next in Figure 6.6 we can see two examples of 10-second segments generated by the *fully-connected* model. Here, we can see the model has produced some turn-taken patterns which could be believable, the model appears to place overlaps in positions that we would expect in real conversations. For example, on the left, we can see that overlaps have been placed at the start and end of an utterance, with a quick backchannel in between. However, the model appears to maybe overestimate the amount of overlap. Next, in Figure 6.7 the *independent model* is used to generate the data, here we can see clearly that the model overestimates the amount of data and does not produce turn-taken that seems believable.

Given a trained turn-taking model, we can look at the transition matrix which contributes to its characteristics. The heatmap in Figure 6.8 shows an example of the resulting transition matrix after training a fully-connected model on session S09. Here we can see the spareness of the matrix, it is very unlikely that some of the transitions will occur, i.e., many of the state transitions involving more than one speaker activity changing are very rare.

Next, the turn-taking models will be evaluated more formally by exploring how well they can match the overlap statistics found in the real data.

6.3.4 Comparison of overlap distribution produced from models

Now that the two model types have been described and we have subjectively seen that the *fully-connected* model produces turn-taking that appears to be more realistic than the independent model. Next, we will look at the overlap distribution in terms of the number of people speaking at one time. Accurately modelling this distribution will aid in producing data that can be used for evaluating speech separation as knowing the number of active speakers that are being mixed is important for many techniques used.

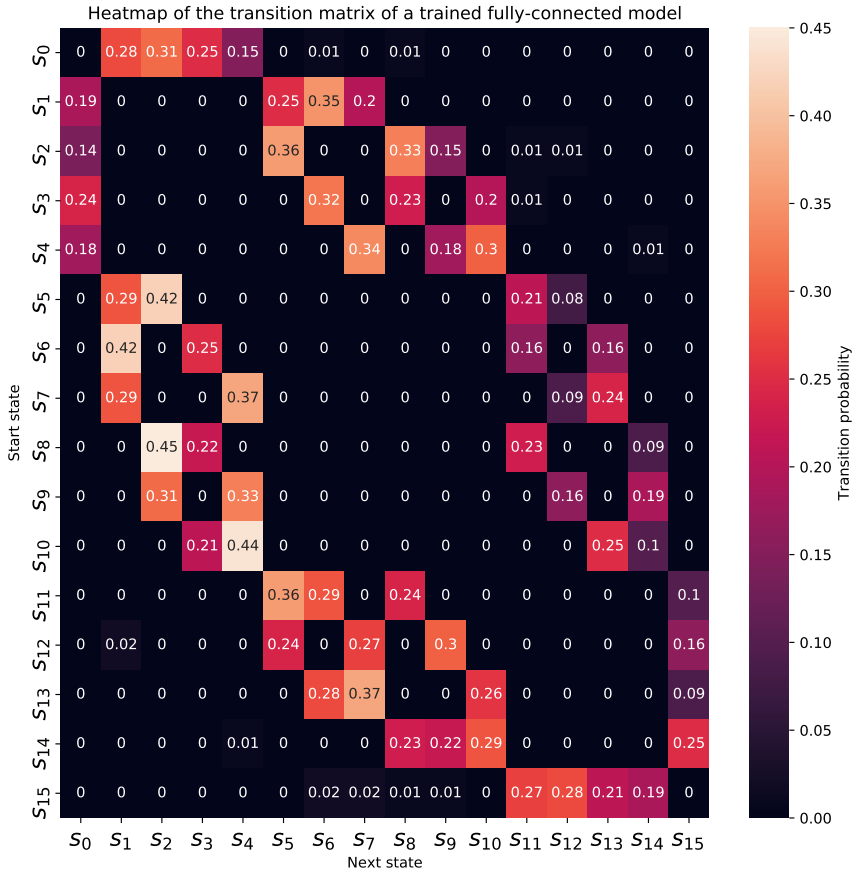

Fig. 6.8 Heatmap plot showing the transition matrix of the fully-connected model after being trained on session S09. The plot shows the spareness of the matrix due to transitions between states involving two speaker changes being rare.

Table 6.1 Comparison of the fully-connected model with the independent model. Computed using transcript, no voice activity detection. Monte Carlo estimation using 500 samples to estimate mean overlap for the models. $\mathbb{E}[X]$ is the expected number of people talking at one time.

	Number of people speaking					
Model	0	1	2	3	4	$\mathbb{E}[X]$
CHiME-5	22%	52%	20%	5%	2%	1.15
Fully-connected	19%	42%	30%	8%	1%	1.30
Independent	6%	25%	37%	25%	6%	1.98

To measure the model's capability of matching the overlap statistics of the sessions it was trained on, the overlap distribution will be compared with the training data used for the model. For each session in the CHiME-5 dataset, a $\mathbf{Y}$ is computed using $f_s = 100$ and a Wald distribution (Seshadri, 1999) is used for the time distribution (P_s) in the states. A model is fit on the entire session, and then data is generated matching the original session length by first starting the sampling process at the silent state. The overlap statistics are then computed. The process is repeated 500 times in order to estimate the mean statistics of the overlap, that is we want to know the steady-state overlap distribution of the generative model, estimated in a Monte-Carlo fashion.

The results in Table 6.1 show the distribution of the CHiME-5 overlap according to the transcript[3]. The numbers reported in the table are the average of this across all the parties. From the table, we can see that the *independent* model produces a larger expected number of speakers than the original dataset. It is also mismatched with respect to the number of people speaking at one time. The *fully-connected* model still produces more overlapped data, but it is far closer to the training data both in terms of the expected number of speakers and the general distribution shape.

Although the model is not *directly trained* on the overlap distribution, it successfully approximates the degree of overlap observed in CHiME-5. The results show us that if we treat speakers as independent generators, the overlap statistics produced by the model vastly overestimate the amount of overlap in real turn-taking. This is because the model has no consideration for how speakers interact with each other and neglects the fact people try to avoid talking over each other when possible when having a conversation.

Table 6.2 Table of results showing how well different distributions $(Q(x))$ can approximate the true distribution $(P(x))$ of speaker overlap. All distributions in scikit-learn that were able to produce an estimate of the state distributions were used in the experiment. The results show that many of the distributions produce similar results and are all appropriate choices for the CHiME-5 dataset. Due to the large standard error values, it is not possible to say if any of the distributions are most appropriate for the CHiME-5 dataset.

Distribution	$D_{KL}(P(x)\|\|Q(x))$	**Standard Error**
Wald	0.055	0.010
Exponnorm	0.056	0.010
Gilbrat	0.057	0.010
Gompertz	0.058	0.011
Genexpon	0.059	0.012
Recipinvgauss	0.059	0.011
Moyal	0.060	0.013
Norminvgauss	0.064	0.015
Lomax	0.086	0.039
Kappa4	0.175	0.126
Foldcauchy	0.186	0.047
Halfcauchy	0.216	0.066
Nct	0.255	0.193

6.3.5 Comparison of the time-distributions

So far the Wald distribution has been used throughout this chapter. This is a design choice that can be made and many other distributions could potentially be used. To compare distributions a model will be trained on session data, and a new session of the same length will be then generated using the model. The resulting overlap distribution of the generated session will then be compared to the original distribution. To compare the distributions, the information gain by using the original distribution $P(x)$ over the new distribution $Q(x)$. Formally, this is defined using the KL divergence,

$$D_{KL}(P(x)||Q(x)) = \sum_{x} P(x) log \frac{P(x)}{Q(x)}, \tag{6.2}$$

where $D_{KL}(P(x)||Q(x))$ tells us how much additional information is needed to encode $P(x)$ if we have the $Q(x)$ distribution. If the distributions are identical this quantity would be equal to 0, therefore a better fitting distribution will have a lower value.

This divergence value will be computed for each of the sessions in the CHiME-5 dataset, and then the average $D_{KL}(P(x)||Q(x))$ value is reported alongside the standard error for

[3]No voice activity detection used in this calculation.

several distributions. All the distributions available in the Python toolkit scikit-learn[4] were preliminarily tested by training them on the session data and the resulting density functions for the distributions were visually inspected. Ones that catastrophically failed (i.e., data likelihoods of infinity for any of the states) were removed from the main experiment.

The results of the overlap experiments using the remaining distributions are presented in Table 6.2. The results show that many of the distributions are able to used for the CHiME-5 dataset. Due to the large standard error values, it is not possible to say if any are the most appropriate. Going forward the Wald distribution will continued to be used for the state distribution.

6.4 Party representations

Using the Markov-models we can compute features from the parameters learnt from the data. For the independent generator, features are computed for each model and then concatenated together, creating a larger representation. As noted the growth of the parameters for the *fully-connected* is $\mathscr{O}(2^J)$ where J is the number of speakers. When looking for meaningful representations this is going to potentially have a profound effect. Lots of the parameters of the models will not be well fit because transitions will never be seen during training. Therefore, an additional model is introduced to mitigate this issue, which will be named *competing*. The *competing* speaker model has sub-models that have dependencies on other speakers, $\mathbb{S}_j^{\text{comp}} = \mathscr{P}(\{\mathbb{P}_j, \xi\})$ where ξ symbolises some other person speaking, the model also has independent transition matrices $\mathbf{T}^{\text{comp}} = \{\mathbf{T}_1^{\text{comp}}, ..., \mathbf{T}_J^{\text{comp}}\}$. A diagram of this model is shown in Figure 6.9, again using two speakers as an example.

A comparison of the growth in parameters between the models introduced in this chapter is shown in Figure 6.10. Here we can see both the *independent* and *competing* models produce parameter growths linearly with the number of speakers.

6.4.1 Extracting features from models

Given these models, next we will explore how representations can be computed from the models to give a vector representation to characterise speaker behaviour. A representation will allow for better analysis of speech technology results conditioned on the type of speaker interaction. First, given a transition matrix for a Markov-model, a steady-state distribution can be computed to give the probability of being in each of the states (Gagniuc, 2017),

[4] https://docs.scipy.org/doc/scipy/reference/stats.html

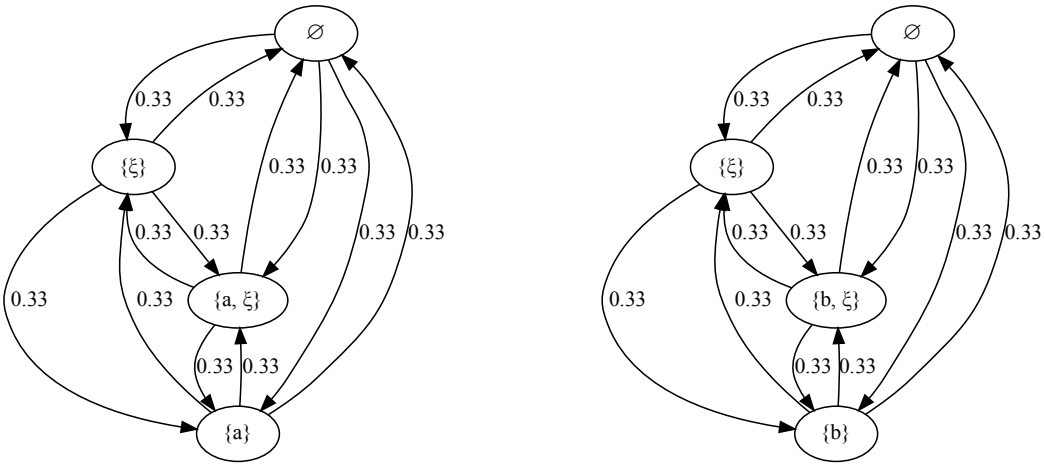

Fig. 6.9 *Competing* speaker model. A speaker now has their own sub-model which has a state conditioned of whether or not someone else is talking (ξ).

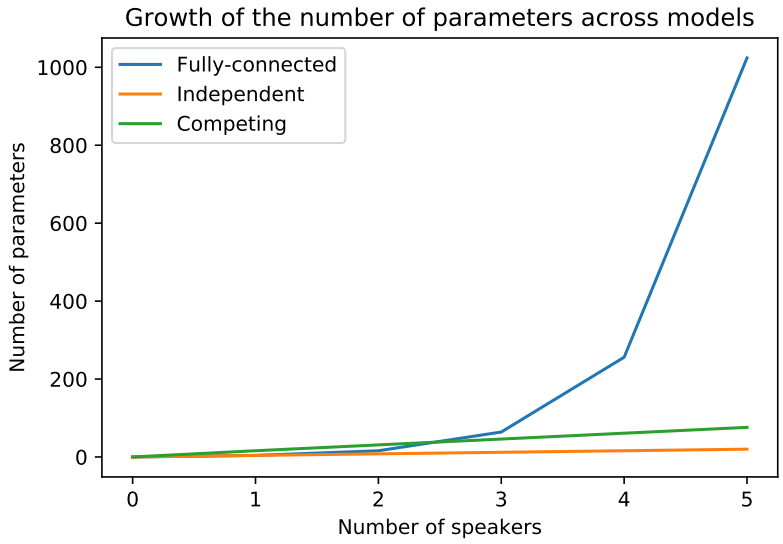

Fig. 6.10 Each of the models presented in this chapter has varying complexities with respect to their number of parameters. The *fully-connected* model requires $\mathcal{O}(2^J)$ parameters, where J is the number of speakers. Making it not practical for modelling large groups.

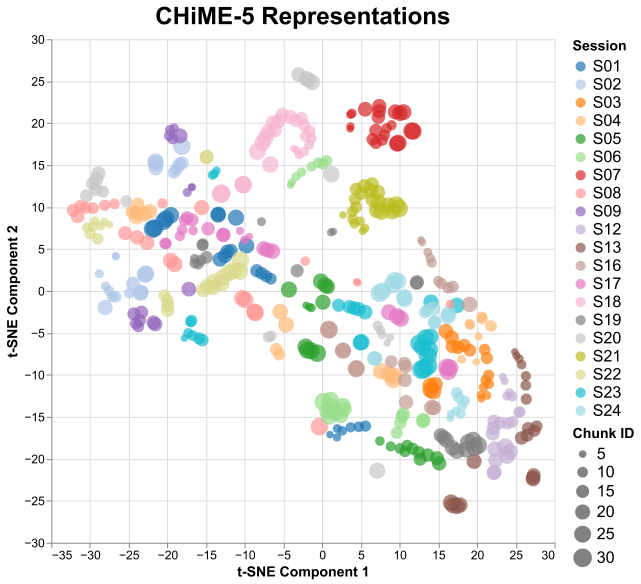

Fig. 6.11 t-SNE plot of the sessions in CHiME-5. Chunk ID shows how the points move around the space over time.

$$\boldsymbol{\phi}^{\text{state}} = \left[(\mathbf{Q}^T \mathbf{Q})^{-1} \mathbf{1} \right]^\top, \tag{6.3}$$

where $\mathbf{Q} = \begin{bmatrix} \mathbf{T} - \mathbf{I} & \mathbf{1} \end{bmatrix}$ and $\mathbf{1}$ is a column vector of ones with length $|\mathbb{S}|$. From the steady-state, we can compute a distribution over the transitions to give the probability of a transition occurring,

$$\boldsymbol{\Phi}^{\text{trans}}_{mn} = \boldsymbol{\phi}^{\text{state}}_{m} \mathbf{T}_{mn}, \tag{6.4}$$

which is flattened into a vector $\boldsymbol{\phi}^{\text{trans}} \in \mathbb{R}^{|\mathbb{S}||\mathbb{S}|}$. Finally, using the state time distributions, an expected duration in the state can be computed,

$$\boldsymbol{\phi}^{\text{dur}} = \boldsymbol{\phi}^{\text{state}} \odot \begin{bmatrix} \mathbb{E}[P_{\mathbb{S}_1}] & \cdots & \mathbb{E}[P_{\mathbb{S}_{|\mathbb{S}|}}] \end{bmatrix}^\top, \tag{6.5}$$

where $\odot$ is the element-wise product.

6.4.2 Visualisation

To illustrate the embeddings, we will train models on a real-world dataset, CHiME-5. For visualisation, we train the *fully-connected* model on 40-minute chunks with an overlap of 5 minutes using a sample rate $f_s = 100$ and Wald for state distribution P_s. The speaker IDs are assigned based on activity, i.e., $\mathbb{P}_1$ is always the most active, and $\mathbb{P}_K$ is the least. This is

done to mitigate the effect of speaker ordering being arbitrary but affecting the feature order. The plot in Figure 6.11 shows a t-SNE representation of $\mathbf{\Phi}^{\text{all}} = \left[\boldsymbol{\phi}^{\text{state}};\ \boldsymbol{\phi}^{\text{trans}};\ \boldsymbol{\phi}^{\text{dur}}\right]^{\top}$. The t-SNE representation is a low dimensionality projection of the high dimensional data with the objective of keeping together points which are close in the original space are also close in the projected space. Unlike principal component analysis which is limited to linear transformations, t-SNE projections are non-linear and are more appropriate for visualisation rather than preprocessing for modelling.

The plot illustrates how sessions in CHiME-5 form clusters, i.e., the statistics of the parties remain fairly consistent. The size of the points in the plot is derived from their position in time in the party i.e., showcasing the movements of the points around the space over time. The plot illustrates some interesting behaviour amongst parties. S07 and S21 exhibit homogeneous behaviour throughout the parties, whilst other parties show a more varied distribution. Overall this visualisation shows that there is turn-taking structure that can be captured within these models that characterises behaviour. The t-SNE algorithm has no knowledge of the labels and yet clusters within same parties are formed. It would be interesting to see if the same group of speakers participated in a party on another day, would we see those parties close together?

Next, we will look at how turn-taking changes across different datasets. In Figure 6.12 CHiME-5 is compared with AMI (McCowan et al., 2005) and an altered version of LibriParty[5]. LibriParty is a simulated dataset for generating long-form parties, the method treats each of the speakers independently. LibriParty originally is configured to generate parties of two people this has been altered using four people i.e., the same as CHiME-5. LibriParty scenarios are extended to be 40 minutes long in order to use the same chunk size of 40 minutes, 50 of these scenarios are generated. For AMI, the scenarios are limited to only those containing four speakers. The figure shows that there is a large overlap in turn-taking behaviour across AMI and CHiME-5. However, CHiME-5 shows a larger diversity in the behaviours and AMI appears to be a subset of this behaviour. Now when we compare the results with LibriParty, we can see this approach is very homogeneous and does not cover much of the space of possible behaviours.

6.5 Evaluation

Now that a representation has been described, we will evaluate the representation with respect to the task of speaker extraction. That is, we aim to investigate how SI-SDR (Le Roux et al., 2019) changes with respect to the location of parties in the embedding space.

[5]https://huggingface.co/speechbrain/vad-crdnn-libriparty

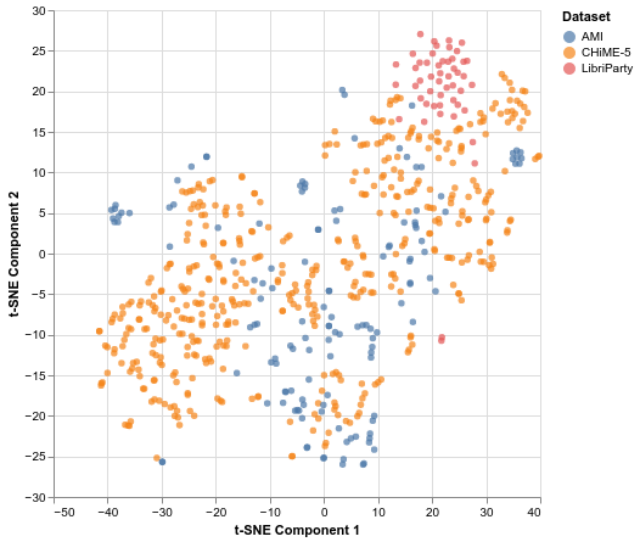

Fig. 6.12 Comparison of the turn-taking behaviour across different datasets. The AMI corpus appears to represent a subset of the behaviour observed in CHiME-5. The simulated corpus LibriParty shows very similar behaviour and does not represent the diversity of the real data.

Speaker extraction involves extracting the speech signal from the desired speaker in a mixture of zero or more other speakers and noise. The target speaker is indicated to a model through an enrollment. For this work, an utterance from a speaker is used that is not present in the mixture.

6.5.1 Target-speaker extraction model

For evaluation, time-domain Speakerbeam model (Delcroix et al., 2020) will be used. The model is based on Conv-Tasnet (Luo and Mesgarani, 2019) with an additional component to learn a speaker embedding encoder jointly. The model is trained on Libri2Mix with WHAM noise (Wichern et al., 2019). The advantage of target-speaker extraction as an evaluation task is that the model works with mixtures with more than two speakers, which will be the case with the mixtures generated from the models.

6.5.2 Data generation

In total, 506 chunks are created using CHiME-5, and AMI (Carletta et al., 2005) transcript data. Chunk lengths are again 40-minutes with a 5-minute overlap. The AMI data is reduced to only sessions containing four speakers and at least 40 minutes in duration. The *fully-connected* model is then fit on the chunks to generate the data, again a Wald distribution is used for states (P_s). To generate the mixture data, first, the activity matrix $\mathbf{Y}$ is generated

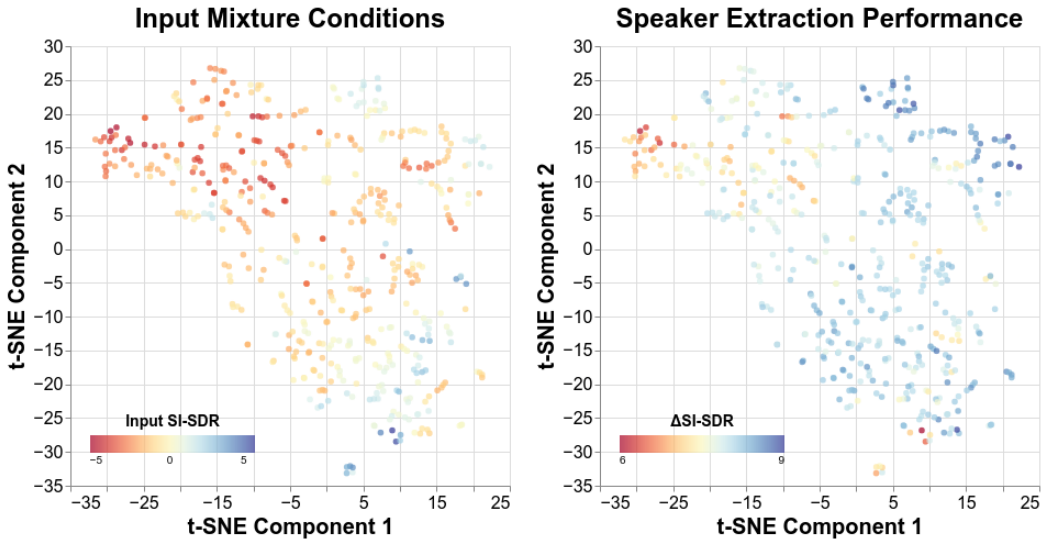

Fig. 6.13 t-SNE representation of learnt space against target-speaker extraction performance.

from the *fully-connected* model, starting the model at the silent state ($\varnothing$). For this evaluation, $\mathbf{Y}$ contains 2.5 hours of activity for each chunk being evaluated.

The large $\mathbf{Y}$ matrix is broken into chunks 1 minute in length. Utterances from LibriSpeech (dev-clean) are then placed onto these segments such that they minimise the difference between LibriSpeech utterance duration and turn-generated duration. This process is repeated for all the chunks, and care is taken such that mixtures do not contain speakers overlapping themselves. If the error between the segments and LibriSpeech utterances is too large, it is discarded[6].

The 1-minute chunks are then broken into segmented mixtures. Each of the utterances in the chunk becomes a mixture where the duration is the length of the target signal.

This allows for the steady-state of this learnt turn-taking behaviour to be evaluated, i.e., the *fully-connected* model has been shown to accurately produce turn-taking behaviour with approximate statistics to the learnt data. Therefore generating from this model allows for statistically stationary turn-taking data to be generated.

6.5.3 Results

Each of the models is evaluated using the 2.5 hours of turn-taking generated from the models. A visualisation of the results is shown in Figure 6.13. The plot illustrates the average performance of the target-speaker extraction for data generated by each of the models. The t-SNE representation shows a clear trend where the parties in the top left of the space start

[6]If the average difference between utterance and segment is greater than 0.5 seconds.

Table 6.3 Predicting SI-SDR using the model embeddings. Where ρ is Pearson Correlation and RMSE is the root mean square error in dB. $\mathbb{E}$[SI-SDR] is using the mean of the training data as the prediction for the test samples.

Feature	*Fully-connected*		*Competing*		*Independent*	
	ρ	**RMSE**	ρ	**RMSE**	ρ	**RMSE**
$\mathbb{E}$[SI-SDR]	-	2.17	-	2.17	-	2.17
$\boldsymbol{\phi}_{\text{state}}$+$\boldsymbol{\phi}_{\text{dur}}$+$\boldsymbol{\phi}_{\text{trans}}$	0.62	1.72	**0.66**	**1.63**	0.38	2.06
$\boldsymbol{\phi}_{\text{state}}$+$\boldsymbol{\phi}_{\text{dur}}$	0.63	1.70	**0.67**	**1.62**	0.38	2.06
$\boldsymbol{\phi}_{\text{state}}$+$\boldsymbol{\phi}_{\text{trans}}$	0.60	1.76	**0.63**	**1.69**	-	2.16
$\boldsymbol{\phi}_{\text{trans}}$+$\boldsymbol{\phi}_{\text{dur}}$	0.63	1.70	**0.66**	**1.64**	0.38	2.06
$\boldsymbol{\phi}_{\text{state}}$	0.60	1.76	**0.63**	**1.69**	-	2.16
$\boldsymbol{\phi}_{\text{trans}}$	0.60	1.76	**0.62**	**1.71**	-	2.16
$\boldsymbol{\phi}_{\text{dur}}$	**0.53**	**1.85**	0.40	2.00	0.38	2.06

with a low SI-SDR and gradually go into an area of higher SI-SDR. We can then compare this with the performance of the speaker extraction scores, where models in the middle of the space provide the largest gains. Models in the top left are too noisy for the extraction to perform well, and models in the bottom right are already too clean, and any enhancement attempt has resulted in signal degradation. Such visualisation gives an overall picture of the performance of an extraction system across a range of parties.

Next, to evaluate the efficacy of the features, we use the application of predicting the SI-SDR scores. A Support Vector Regressor (Awad and Khanna, 2015) is trained with a radial-basis function kernel using combinations of features presented in Section 4. The 506 samples are evaluated using a bootstrap cross-fold validation where 80% sample of sessions is used as training data, and the other 20% is used for the test. Where in each fold all the chunks in a session are evaluated after being trained on all other chunks, we then report Pearson correlation and root mean squared statistics in Table 6.3; this is the average across 100 repeats of this sampling procedure. The results show all the features provide information in predicting SI-SDR with a fairly strong correlation when using all the features together. The result indicates that the embeddings can provide a method of extracting how difficult a mixture can be when the generator is known. In addition to this, the table also shows the performance of the independent generator models in predicting the SI-SDR. The features of the *fully-connected* model are able to predict the performance of SI-SDR well. The *competing* model is able to surpass the performance of the fully-connecting model. This

may be attributed to the fact that the full model is over-parameterised for the embedding task but well-suited for the generation task. The *independent* model lacks representation power. Each of the Markov-models inside of the independent models have the same transition matrix which have a weight $\mathbf{T}_{mn} = 1$ for $i \neq j$ and $\mathbf{T}_{mn} = 0$ for $i = j$, therefore $\boldsymbol{\phi}^{\text{state}}$ and $\boldsymbol{\phi}^{\text{trans}}$ contain no information.

6.6 Discussion

This chapter has presented a framework for modelling turn-taking behaviour. The training procedure implemented to fit the parameters of the model requires a large chunk size in order to see enough observations to fit the parameters well. However, this chapter has shown that there are redundancies in some of the parameters. This was shown by the *competing* model out-performing the fully-connected model in the representation evaluation. However, it is not possible to generate data from the *competing* speaker model due to it requiring the different sub-modules to align with each other but the generation process is independent so this cannot be guaranteed. Therefore, it leads to the question of whether it is possible to exploit the redundancies showcased to aid the training of the fully-connected model. For example, the weights in the fully-connected could be shared across certain transitions. Alternatively, simpler models such as the *independent* model could be used to set the initial values for the parameters of the more complicated model. Similar to how lower order n-gram models are interpolated for unseen higher order n-grams (Kneser and Ney, 1995).

6.7 Conclusions

In this work, we have presented a simple Markov-model approach for generating arbitrary large datasets with statistical behaviour approximating the behaviour of people in real parties in terms of overlap. From these models, we showed features that can be computed to give a vector representation of a party. Using the CHiME-5 dataset, we illustrated that sessions within the dataset provided homogeneous behaviour and clustering.

These embeddings were then evaluated within the task of target-speaker extraction, where they were shown to have an interpretable meaning with respect to the performance of SI-SDR improvement. This was evaluated with respect to a prediction task based solely on the embeddings. We have shown that treating people as independent generators does not provide a realistic way of creating turn-taking behaviour, and it does not provide an adequate way of building a representation to predict difficulty. The *fully-connected* model was shown to be

the best approach for generating turn-taking. The *competing* model was shown to surpass the performance of the *fully-connected* model as a representation.

Chapter 7

Conclusions

This thesis has been an exploration of the mismatches between simulation and real data in the context of automatic speech recognition (ASR). This work set out to critique the metadata used to drive these simulations, which is typically presented without motivation. This work aimed to investigate ways to generate this metadata by analysing the behaviour of people when they are socialising and the impact of imposing realism into the simulation.

Simulated data is used to both train and evaluate speech separation, a key component of distant microphone speech recognition. Benchmarking the performance of techniques against inadequate simulations will result in misleading results which are not likely to lead to a well-performing system on real data. In multi-talker environments, separating the speech signals when the speech overlaps is crucial for a well-performing ASR system. Speech separation for distant microphone ASR often relies on multiple microphones forming an array allowing for signals to be filtered spatially. The work in this thesis has explored the spatial aspect of the simulation i.e., the positioning of the sources, as well as the temporal aspect i.e., the placement and amount of overlap between the sources.

At the outset of this thesis, a number of key research questions were identified which aimed to break down the question of what is required of simulated data. These questions have been addressed through the experimental work presented in the preceding three chapters. They are reviewed below.

RQ1: How well do simulated datasets represent the *angular separation* found in real data? And how does poorly representing real data affect ASR evaluation?

In Chapter 4 the angular separation between speakers in commonly used datasets, namely SMS-WSJ and the spatialised version WSJ0-2Mix were compared. It was found that these datasets have vastly different distributions of separations and the details of what caused this difference was not even presented in the publications. It

was found that SMS-WSJ produces simulated datasets with uniform distributions of angular separation. Whilst WSJ0-2Mix produced a distribution of angular separation that did not focus on narrow angles at all.

Video data from CHiME-5 was used to automatically detect speakers inside the dataset to estimate their separation angle. Using the recordings from single devices in isolation found that the separation angles between speakers are not uniform but biased more towards narrow angles. This is due to the behaviour of people, i.e., people tend to stand close to each other, but also in part due to a limitation in the field of view of the devices. Therefore in Chapter 5, a refinement of the estimate of angular separation was made by estimating 2-D positions of people in the rooms and then projecting them into the devices. It was found the initial estimate of separation was a good approximation but some large separation angles were missed.

The realistic angular separation was then used to create simulated evaluation datasets that showed a large degradation in recognition and separation performance. The performance decrease showed that not all separation techniques are affected equally and therefore this could potentially lead to a mismatch where one technique is believed to be superior because it performs were in exploiting narrow angles but in fact is it not the better approach because the improvement over wide angles is superfluous and focus should be on narrow angles.

RQ2: **How well do simulated datasets represent the *relative distance* found in real data? And how does poorly representing real data affect ASR evaluation?**

In Chapter 5, 2-D estimates of speaker locations were gathered by combining and triangulating individual estimates in single devices. These 2-D positions were then used to estimate the distances talkers were away from the microphones. It was found that under the constraint of small rooms i.e., the ones typically used in simulation the relative distance was accurately modelled as in close proximity, people are positioned randomly. However, if the same naive positioning techniques are used when we extend the room sizes to be larger, the relative distance between speakers is no longer accurate. The data showed that when rooms are large, people position themselves closer to each other, i.e., they form groups. Uniformly positioning people in the room does not capture this phenomenon.

Enforcing this realistic relative distance distribution was then the focus of the experimental work. The results showed a complicated relationship between angular separation and relative distance. It was shown that it was important to model both these distributions accurately. If the relative distance distribution was modelled but

a uniformed angular separation was used, then the impact the relative distance distribution has on ASR was masked by the fact the sources were easy to separate. If a realistic angular separation distribution was used alongside the realistic relative distance distribution then a more challenging dataset was created. This is due to the separation performing poorly and therefore more of the interferer is present in the mixture. When more of the interferer is present the impact of the relative closeness of the sources is felt.

This analysis showed a potential pitfall that could be faced when an unrealistic simulation is used to derive the direction of research. The problem of the ASR system not being able to perform well when sources are relatively close to each other was masked by the fact the sources were too easy to be separated as the angles were too wide.

RQ3: **How well can integrated cameras from ad-hoc placements of devices be used to estimate speaker positioning inside of rooms?**
In the CHiME-5 dataset devices were placed at the edge of rooms to emulate the natural positioning of devices when people use smart-home devices in their everyday living environment i.e., "out-of-the-way" and not a centrepiece. The placement of these devices was not completely unknown, a rough sketch of the floorplans was provided. This resulted in initial placements of the devices with some unknown errors. In Chapter 5, a calibration method was devised which minimises the disagreement of speaker positioning when three or more devices can detect a person. The calibration process accurately repositioned the devices which under a visual inspection of the videos appeared to represent reality.

Through modelling the error in the camera detections in the devices a probabilistic method for estimating the probability of any position in a room given the detection in each of the devices was shown to be an effective method for modelling the additional error in the cameras. Choosing the maximum probability of the position in the room was shown to be more error-prone and less effective at position estimate compared with computing the expected value over the room space.

It was shown this method for estimating positions works well when devices can be calibrated i.e., there are many overlapping views and the devices do not face each other. Two devices facing each other result in poor estimates of speaker location, especially if using the maximum probability.

RQ4: **How well do simulated datasets represent the overlap patterns found in real data? And how does poorly representing real data affect speaker extraction evaluation?**
Simulated datasets vastly overestimate the amount of overlap present in mixtures. With

some by design containing 100% speaker overlap or close to this. These mixtures often contain a fixed number of speakers and do not consider the dynamics of multi-party interactions. Methods for creating longer unsegmented multi-party scenarios treat each of the speakers as independent generators with a set amount of time pause in between utterances. In Chapter 6, this was demonstrated to show unrealistic interactions that do not match real turn-taking behaviour, i.e., it produces scenarios with a high degree of overlap and a high number of people speaking at one time.

A finite-state approach to represent the turn-taking was developed in Chapter 5 using semi-Markov models connecting speaking states representing all combinations of speakers. This method models a fixed number of speakers and was shown to be an effective way of generating data which produced turn-taking behaviour that approximated that observed in a real environment purely based on timing information.

Modelling speakers independently failed to represent the diversity of overlap patterns that were observed in real data. Through visualising the space of models (see **RQ5**) the location of independent models was only a small cluster within the complete space of turn-taking models. The homogeneous nature of the models impacts ASR as some parts of the model space were shown to be more challenging the others (again see **RQ5**). Not evaluating ASR across an array of turn-taking behaviour may result in techniques being preferred which optimise that one part of the model space and not the full diversity of turn-taking behaviours we observe in real data.

RQ5: **How can party representations be created to best model the difficulty of parties for ASR?**

Using the turn-taking models developed for generating data, features were engineered that could be computed from the parameters of the models. Computing the steady-state probability, expected durations and weighted transition probability were effective features for the representations.

Segments of turn-taking were used to train a generative model to produce a large dataset of state-state turn-taking behaviour. Target-speaker extraction was then used for evaluation to explore how difficult the data generated by each of the models was for this task. Through visualising the representations alongside the average speech separation performance, the models showed a clear association between the model space and the difficulty of the data it generated. This evaluation technique allows for different speech extraction techniques to be compared across a range of turn-taking behaviours, allowing for a more insightful comparison between techniques.

Using a regressive model and the representations as input, Chapter 6 demonstrated that the features could be used to predict the performance of target speaker extraction. This allows for measuring the amount of difficulty in simulated data due to the turn-taking behaviour alone.

7.1 Limitations

The work in this thesis has largely benefited from the availability of the CHiME-5 dataset which contains a diverse range of unscripted multi-party interactions, in realistic settings. In addition to this, the author was fortunate to have access to the video data recorded alongside the dataset. However, due to the dataset's uniqueness, it has been difficult to evaluate this work across corpora. For example, DiPCo (Van Segbroeck et al., 2019) which is the most similar dataset to CHiME-5 does not provide any video or speaker position data. Any corpora that do provide the additional modalities such as speaker positioning do not capture the same kind of multi-party interactions that are being modelled in this work. Producing a similar dataset with the additional speaker information was not feasible under the financial and time constraints (without severely limiting the scope of the analytic work).

7.2 Scope for future work

This work has provided several insights into the limitations of the metadata being used to produce simulations. However, only two aspects have been explored, the speaker positioning and the overlap between utterances. A simulation consists of many more parameters that are not being modelled, and as the simulation techniques being used to create reverberant speech signal becomes more advanced, so do the requirements for the metadata. For example, the room geometry is not a large consideration in the design of simulation, rectangular "shoebox" rooms are often used. The materials used to make up rooms affect the amount of reverberation in the rendered spatial images. This work explored mixtures where the sources are stationary, given the position information, it would be possible to model the movement of people. The trajectories of the sources would need to be modelled realistically to avoid systems exploiting that aspect of the simulation. Furthermore, the direction people are facing changes the acoustic properties. In this work sources are treated as point sources, however, sources in fact have directivity patterns in real life. These aspects were not the focus of this thesis as largely they are not modelled in current simulated datasets. This work has shown even the simplest choices of metadata such as speaker positioning are not motivated and this has profound consequences for the conclusions made when evaluating distant microphone

speech recognition. It should be seen as motivation for the need to keep advancements in the analysis as the advancements in simulation techniques increase.

The work presented in this thesis focused on spatial and temporal aspects of the multi-party scenarios separately. The experiments focused on evaluating the impact of these parameters through controlling these variables and creating many different datasets to measure their impact. Therefore, not one dataset is presented but rather a family of datasets. The logical next step would therefore be to develop an overall more realistic new dataset. Combining temporal and spatial statistics is a non-trivial problem, when the spatial characteristics were explored in this thesis, segmented mixtures were created and the spatial distributions were used directly. Broadening this setup to long-form parties will result in the need to model the movement of speakers over time. This could be archieved through modelling the movement of people from the 2-D position estimates and then verifying the separation distributions of the segmented mixtures match the real data. Extra care needs to be taken when modelling the movement of people, if the movement is too predictable this may be exploited by the separation system.

This thesis has presented methods to augment any dataset to be more realistic. Given a simulated dataset, a realistic variant can be created by changing the position estimates according the distributions described in Chapters 4 and 5 and the temporal statistics presented in Chapter 6. For datasets created through playing sounds inside of a room and capturing the results (instead of computer simulation), the performance metrics could be altered by using a weighted average. For example, for angular separation, improvements over narrow angles should be weighted higher than improvements over wide angles. The amount to weigh the angles could be governed by the separation distribution presented in this work.

References

Allen, J. B. and Berkley, D. A. (1979). Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America*, 65(4):943–950.

Anguera, X., Wooters, C., and Hernando, J. (2007). Acoustic beamforming for speaker diarization of meetings. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(7):2011–2021.

Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. M., and Weber, G. (2019). Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Awad, M. and Khanna, R. (2015). Support vector regression. In *Efficient learning machines*, pages 67–80. Springer.

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (2015). The third 'CHiME'speech separation and recognition challenge: Dataset, task and baselines. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 504–511. IEEE.

Barker, J., Vincent, E., Ma, N., Christensen, H., and Green, P. (2013). The pascal chime speech separation and recognition challenge. *Computer Speech & Language*, 27(3):621–633.

Barker, J., Watanabe, S., Vincent, E., and Trmal, J. (2018). The fifth 'CHiME' speech separation and recognition challenge: Dataset, task and baselines. In *Proc. Interspeech 2018*, pages 1561–1565.

Bermuth, D., Poeppel, A., and Reif, W. (2021). Scribosermo: Fast speech-to-text models for german and other languages. *arXiv preprint arXiv:2110.07982*.

Boeddeker, C., Heitkaemper, J., Schmalenstroeer, J., Drude, L., Heymann, J., and Haeb-Umbach, R. (2018). Front-end processing for the chime-5 dinner party scenario. In *CHiME5 Workshop, Hyderabad, India*, volume 1.

Bredin, H. (2017). pyannote. metrics: A toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems. In *INTERSPEECH*, pages 3587–3591.

Bredin, H. and Laurent, A. (2021). End-to-end speaker segmentation for overlap-aware resegmentation. *arXiv preprint arXiv:2104.04045*.

Breed, B. R. and Strauss, J. (2002). A short proof of the equivalence of lcmv and gsc beamforming. *IEEE Signal Processing Letters*, 9(6):168–169.

Brinkmann, F., Aspöck, L., Ackermann, D., Lepa, S., Vorländer, M., and Weinzierl, S. (2019). A round robin on room acoustical simulation and auralization. *The Journal of the Acoustical Society of America*, 145(4):2746–2760.

Campbell, W., Sturim, D., and Reynolds, D. (2006). Support vector machines using GMM supervectors for speaker verification. *IEEE Signal Process Lett.*, 13(5):308–311.

Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., and Sheikh, Y. A. (2019). Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Capon, J. (1969). High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE*, 57(8):1408–1418.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., et al. (2005). The AMI meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Chaudhari, A. and Dhonde, S. (2015). A review on speech enhancement techniques. In *2015 International Conference on Pervasive Computing (ICPC)*, pages 1–3. IEEE.

Chen, G., Chai, S., Wang, G., Du, J., Zhang, W.-Q., Weng, C., Su, D., Povey, D., Trmal, J., Zhang, J., et al. (2021). Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. *arXiv preprint arXiv:2106.06909*.

Chen, Z., Yoshioka, T., Lu, L., Zhou, T., Meng, Z., Luo, Y., Wu, J., Xiao, X., and Li, J. (2020). Continuous speech separation: Dataset and analysis. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7284–7288. IEEE.

Cherry, C. and Bowles, J. (1960). Contribution to a study of the "cocktail party problem". *The Journal of the Acoustical Society of America*, 32(7):884–884.

Choi, S., Cichocki, A., Park, H.-M., and Lee, S.-Y. (2005). Blind source separation and independent component analysis: A review. *Neural Information Processing-Letters and Reviews*, 6(1):1–57.

Christensen, H., Barker, J., Ma, N., and Green, P. D. (2010). The chime corpus: a resource and a challenge for computational hearing in multisource environments. In *Eleventh Annual Conference of the International Speech Communication Association*. Citeseer.

Cooke, M., Barker, J., Cunningham, S., and Shao, X. (2006). An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424.

Cooke, M., Hershey, J. R., and Rennie, S. J. (2010). Monaural speech separation and recognition challenge. *Computer Speech & Language*, 24(1):1–15.

Cosentino, J., Pariente, M., Cornell, S., Deleforge, A., and Vincent, E. (2020). Librimix: An open-source dataset for generalizable speech separation. *arXiv preprint arXiv:2005.11262*.

Deadman, J. and Barker, J. (2020). Simulating Realistically-Spatialised Simultaneous Speech Using Video-Driven Speaker Detection and the CHiME-5 Dataset. In *Proc. Interspeech 2020*, pages 349–353.

Delcroix, M., Ochiai, T., Zmolikova, K., Kinoshita, K., Tawara, N., Nakatani, T., and Araki, S. (2020). Improving speaker discrimination of target speech extraction with time-domain speakerbeam. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 691–695. IEEE, IEEE.

DiBiase, J. H., Silverman, H. F., and Brandstein, M. S. (2001). Robust localization in reverberant rooms. *Microphone arrays: signal processing techniques and applications*, pages 157–180.

Drude, L. and Haeb-Umbach, R. (2017). Tight integration of spatial and spectral features for bss with deep clustering embeddings. In *Interspeech*, pages 2650–2654.

Drude, L., Heitkaemper, J., Boeddeker, C., and Haeb-Umbach, R. (2019a). Sms-wsj: Database, performance measures, and baseline recipe for multi-channel source separation and recognition. *arXiv preprint arXiv:1910.13934*.

Drude, L., Heitkaemper, J., Boeddeker, C., and Haeb-Umbach, R. (2019b). SMS-WSJ: Database, performance measures, and baseline recipe for multi-channel source separation and recognition. *arXiv preprint arXiv:1910.13934*.

Drude, L., Heymann, J., Boeddeker, C., and Haeb-Umbach, R. (2018). Nara-wpe: A python package for weighted prediction error dereverberation in numpy and tensorflow for online and offline processing. In *Speech Communication; 13th ITG-Symposium*, pages 1–5. VDE.

Du, J., Tu, Y.-H., Sun, L., Chai, L., Tang, X., He, M.-K., Ma, F., Pan, J., Gao, J.-Q., Liu, D., et al. (2020a). The ustc-nelslip systems for chime-6 challenge. In *Proc. The 6th International Workshop on Speech Processing in Everyday Environments (CHiME 2020)*, pages 19–23.

Du, J., Tu, Y.-H., Sun, L., Chai, L., Tang, X., He, M.-K., Ma, F., Pan, J., Gao, J.-Q., Liu, D., Lee, C.-H., and Chen, J.-D. (2020b). The USTC-NELSLIP Systems for CHiME-6 Challenge. In *CHiME-6 Workshop, Barcelona, Spain*.

Ekstedt, E. and Skantze, G. (2020). TurnGPT: A transformer-based language model for predicting turn-taking in spoken dialog. *arXiv preprint arXiv:2010.10874*.

Evers, C., Löllmann, H. W., Mellmann, H., Schmidt, A., Barfuss, H., Naylor, P. A., and Kellermann, W. (2020). The locata challenge: Acoustic source localization and tracking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1620–1643.

Ferguson, B. G. (1998). Minimum variance distortionless response beamforming of acoustic array data. *The Journal of the Acoustical Society of America*, 104(2):947–954.

Févotte, C., Gribonval, R., and Vincent, E. (2005). Bss_eval toolbox user guide–revision 2.0.

Fox, C., Liu, Y., Zwyssig, E., and Hain, T. (2013). The sheffield wargames corpus. In *Proceedings of Interspeech 2013*. ISCA.

Fujita, Y., Kanda, N., Horiguchi, S., Xue, Y., Nagamatsu, K., and Watanabe, S. (2019). End-to-end neural speaker diarization with self-attention. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 296–303. IEEE.

Gagniuc, P. A. (2017). *Markov chains: from theory to implementation and experimentation*. John Wiley & Sons.

Gales, M. and Woodland, P. (1996). *Variance compensation within the MLLR framework*. Citeseer.

Garcia-Romero, D. and Espy-Wilson, C. Y. (2011). Analysis of i-vector length normalization in speaker recognition systems. In *Interspeech 2011*. ISCA.

Graetzer, S., Barker, J., Cox, T. J., Akeroyd, M., Culling, J. F., Naylor, G., Porter, E., Viveros Munoz, R., et al. (2021). Clarity-2021 challenges: Machine learning challenges for advancing hearing aid processing. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, volume 2, pages 686–690. International Speech Communication Association (ISCA).

Grythe, J. and Norsonic, A. (2015). Beamforming algorithms-beamformers. *Technical Note, Norsonic AS, Norway*.

Gu, R., Wu, J., Zhang, S.-X., Chen, L., Xu, Y., Yu, M., Su, D., Zou, Y., and Yu, D. (2019). End-to-end multi-channel speech separation. *arXiv preprint arXiv:1905.06286*.

Haeb-Umbach, R., Heymann, J., Drude, L., Watanabe, S., Delcroix, M., and Nakatani, T. (2020). Far-field automatic speech recognition. *Proceedings of the IEEE*, 109(2):124–148.

Hall, E. T. (1963). A system for the notation of proxemic behavior 1. *American anthropologist*, 65(5):1003–1026.

Hall, E. T., Birdwhistell, R. L., Bock, B., Bohannan, P., Diebold, A. R., Durbin, M., Edmonson, M. S., Fischer, J. L., Hymes, D., Kimball, S. T., La Barre, W., , McClellan, J. E., Marshall, D. S., Milner, G. B., Sarles, H. B., Trager, G. L., and Vayda, A. P. (1968). Proxemics [and comments and replies]. *Current Anthropology*, 9(2/3):83–108.

Haykin, S. and Chen, Z. (2005). The cocktail party problem. *Neural computation*, 17(9):1875–1902.

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

Heymann, J., Drude, L., Chinaev, A., and Haeb-Umbach, R. (2015). Blstm supported gev beamformer front-end for the 3rd chime challenge. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 444–451. IEEE.

Ito, N., Araki, S., and Nakatani, T. (2016). Complex angular central gaussian mixture model for directional statistics in mask-based microphone array signal processing. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 1153–1157. IEEE.

Iwamoto, K., Ochiai, T., Delcroix, M., Ikeshita, R., Sato, H., Araki, S., and Katagiri, S. (2022). How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr. *arXiv preprint arXiv:2201.06685*.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., et al. (2003). The icsi meeting corpus. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 1, pages I–I. IEEE.

Janssen, J. and Limnios, N. (1999). *Semi-Markov Models and Applications*. Springer US.

Kanda, N., Boeddeker, C., Heitkaemper, J., Fujita, Y., Horiguchi, S., Nagamatsu, K., and Haeb-Umbach, R. (2019). Guided source separation meets a strong asr backend: Hitachi/paderborn university joint investigation for dinner party asr. *arXiv preprint arXiv:1905.12230*.

King, D. E. (2009). Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758.

Kinoshita, K., Delcroix, M., Nakatani, T., and Miyoshi, M. (2009). Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction. *IEEE transactions on audio, speech, and language processing*, 17(4):534–545.

Kinoshita, K., Delcroix, M., Yoshioka, T., Nakatani, T., Habets, E., Haeb-Umbach, R., Leutnant, V., Sehr, A., Kellermann, W., Maas, R., et al. (2013). The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech. In *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–4. IEEE.

Knapp, C. and Carter, G. (1976). The generalized correlation method for estimation of time delay. *IEEE transactions on acoustics, speech, and signal processing*, 24(4):320–327.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *1995 international conference on acoustics, speech, and signal processing*, volume 1, pages 181–184. IEEE.

Ko, T., Peddinti, V., Povey, D., Seltzer, M. L., and Khudanpur, S. (2017). A study on data augmentation of reverberant speech for robust speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5220–5224. IEEE.

Kuhn, R., Junqua, J.-C., Nguyen, P., and Niedzielski, N. (2000). Rapid speaker adaptation in eigenvoice space. *IEEE Transactions on Speech and Audio Processing*, 8(6):695–707.

Kumar, A., Kaur, A., and Kumar, M. (2019). Face detection techniques: a review. *Artificial Intelligence Review*, 52(2):927–948.

Le Roux, J., Wisdom, S., Erdogan, H., and Hershey, J. R. (2019). Sdr–half-baked or well done? In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630. IEEE.

Li, J. et al. (2022). Recent advances in end-to-end automatic speech recognition. *APSIPA Transactions on Signal and Information Processing*, 11(1).

Lim, J. S. and Oppenheim, A. V. (1979). Enhancement and bandwidth compression of noisy speech. *Proceedings of the IEEE*, 67(12):1586–1604.

Lincoln, M., McCowan, I., Vepa, J., and Maganti, H. K. (2005). The multi-channel wall street journal audio visual corpus (mc-wsj-av): Specification and initial experiments. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.*, pages 357–362. IEEE.

Liu, Y., Fox, C., Hasan, M., and Hain, T. (2016). The sheffield wargame corpus-day two and day three. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 3833–3837. ISCA.

Loweimi, E., Bell, P., and Renals, S. (2019). On learning interpretable cnns with parametric modulated kernel-based filters. In *INTERSPEECH*, pages 3480–3484.

Lucas, B. D., Kanade, T., et al. (1981). *An iterative image registration technique with an application to stereo vision*, volume 81. Vancouver.

Luo, Y., Chen, Z., and Yoshioka, T. (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)*, pages 46–50. IEEE.

Luo, Y. and Mesgarani, N. (2019). Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 27(8):1256–1266.

Luo, Y. and Mesgarani, N. (2020). Implicit filter-and-sum network for multi-channel speech separation. *arXiv preprint arXiv:2011.08401*.

Maciejewski, M., Sell, G., Fujita, Y., Garcia-Perera, L. P., Watanabe, S., and Khudanpur, S. (2019). Analysis of robustness of deep single-channel speech separation using corpora constructed from multiple domains. In *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 165–169.

Maciejewski, M., Wichern, G., McQuinn, E., and Le Roux, J. (2020). Whamr!: Noisy and reverberant single-channel speech separation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 696–700. IEEE.

Maciejewski, M., Wichern, G., McQuinn, E., and Roux, J. L. (2020). WHAMR!: Noisy and reverberant single-channel speech separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 696–700.

Mao, H. H., Li, S., McAuley, J., and Cottrell, G. (2020). Speech recognition and multi-speaker diarization of long conversations. *arXiv preprint arXiv:2005.08072*.

Mardia, K. and Dryden, I. (1999). The complex watson distribution and shape analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(4):913–926.

Martin-Donas, J. M., Gomez, A. M., Gonzalez, J. A., and Peinado, A. M. (2018). A deep learning loss function based on the perceptual evaluation of the speech quality. *IEEE Signal processing letters*, 25(11):1680–1684.

McCowan, I., Carletta, J., Kraaij, W., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., et al. (2005). The ami meeting corpus. In *Proceedings of the 5th international conference on methods and techniques in behavioral research*, volume 88, page 100. Citeseer.

McGrath, J. E. and Hollingshead, A. B. (1994). *Groups interacting with technology: Ideas, evidence, issues, and an agenda.* Sage Publications, Inc.

Menne, T., Sklyar, I., Schlüter, R., and Ney, H. (2019). Analysis of deep clustering as preprocessing for automatic speech recognition of sparsely overlapping speech. *arXiv preprint arXiv:1905.03500*.

Mirheidari, B., Blackburn, D., O'Malley, R., Walker, T., Venneri, A., Reuber, M., and Christensen, H. (2019). Computational cognitive assessment: Investigating the use of an intelligent virtual agent for the detection of early signs of dementia. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2732–2736. IEEE.

Nakamura, S., Yamamoto, K., Takeda, K., Kuroiwa, S., Kitaoka, N., Yamada, T., Mizumachi, M., Nishiura, T., Fujimoto, M., Saso, A., et al. (2003). Data collection and evaluation of aurora-2 japanese corpus [speech recognition applications]. In *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No. 03EX721)*, pages 619–623. IEEE.

Norris, S. (2004). *Analyzing multimodal interaction: A methodological framework*. Routledge.

Padilha, E. G. (2006). *Modelling turn-taking in a simulation of small group discussion*. PhD thesis, University of Edinburgh.

Pan, C., Chen, J., and Benesty, J. (2014). Performance study of the MVDR beamformer as a function of the source incidence angle. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(1):67–79.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.

Pariente, M., Cornell, S., Deleforge, A., and Vincent, E. (2020). Filterbank design for end-to-end speech separation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6364–6368. IEEE.

Paul, D. B. and Baker, J. (1992). The design for the wall street journal-based csr corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Raux, A. and Eskenazi, M. (2009). A finite-state turn-taking model for spoken dialog systems. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on - NAACL '09*, pages 629–637. Association for Computational Linguistics.

Ravanelli, M. and Bengio, Y. (2018). Speaker recognition from raw waveform with sincnet. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 1021–1028. IEEE.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.

Rix, A. W., Beerends, J. G., Hollier, M. P., and Hekstra, A. P. (2001). Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 01CH37221)*, volume 2, pages 749–752. IEEE.

Sato, H., Ochiai, T., Delcroix, M., Kinoshita, K., Moriya, T., and Kamo, N. (2021). Should we always separate?: Switching between enhanced and observed signals for overlapping speech recognition. *arXiv preprint arXiv:2106.00949*.

Sawada, H., Araki, S., and Makino, S. (2010). Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(3):516–527.

Schegloff, E. A. (2000). Overlapping talk and the organization of turn-taking for conversation. *Language in society*, 29(1):1–63.

Scheibler, R., Bezzam, E., and Dokmanić, I. (2018). Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 351–355.

Schmidt, R. (1986). Multiple emitter location and signal parameter estimation. *IEEE transactions on antennas and propagation*, 34(3):276–280.

Schröder, D. and Vorländer, M. (2011). RAVEN: a real-time framework for the auralization of interactive virtual environments. In *Forum acusticum*, pages 1541–1546. Aalborg Denmark.

Senior, A. and Lopez-Moreno, I. (2014). Improving DNN speaker independence with i-vector inputs. In *ICASSP 2014 - 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 225–229. IEEE, IEEE.

Seshadri, V. (1999). *The Inverse Gaussian Distribution*, volume 95. Springer New York.

Siltanen, S., Lokki, T., Kiminki, S., and Savioja, L. (2007). The room acoustic rendering equation. *The Journal of the Acoustical Society of America*, 122(3):1624–1635.

Sivasankaran, S., Vincent, E., and Fohr, D. (2021). Analyzing the impact of speaker localization errors on speech separation for automatic speech recognition. In *2020 28th European Signal Processing Conference (EUSIPCO)*.

Skantze, G. (2021). Turn-taking in conversational systems and human-robot interaction: A review. *Computer Speech & Language*, 67:101178.

Souden, M., Benesty, J., and Affes, S. (2009). On optimal frequency-domain multichannel linear filtering for noise reduction. *IEEE Transactions on audio, speech, and language processing*, 18(2):260–276.

Stupakov, A., Hanusa, E., Bilmes, J., and Fox, D. (2009). Cosine-a corpus of multi-party conversational speech in noisy environments. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4153–4156. IEEE.

Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., and Zhong, J. (2021). Attention is all you need in speech separation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 21–25. IEEE.

Taal, C. H., Hendriks, R. C., Heusdens, R., and Jensen, J. (2011). An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136.

Trentin, E. and Gori, M. (2001). A survey of hybrid ANN/HMM models for automatic speech recognition. *Neurocomputing*, 37(1-4):91–126.

Van Segbroeck, M., Zaid, A., Kutsenko, K., Huerta, C., Nguyen, T., Luo, X., Hoffmeister, B., Trmal, J., Omologo, M., and Maas, R. (2019). Dipco–dinner party corpus. *arXiv preprint arXiv:1909.13447*.

Varga, A. and Steeneken, H. J. (1993). Assessment for automatic speech recognition: Ii. noisex-92: A database and an experiment to study the effect of additive noise on speech recognition systems. *Speech communication*, 12(3):247–251.

Vincent, E., Barker, J., Watanabe, S., Le Roux, J., Nesta, F., and Matassoni, M. (2013). The second 'chime'speech separation and recognition challenge: Datasets, tasks and baselines. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 126–130. IEEE.

Vincent, E., Gribonval, R., and Fevotte, C. (2006). Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1462–1469.

Vincent, E., Watanabe, S., Nugraha, A. A., Barker, J., and Marxer, R. (2017). An analysis of environment, microphone and data simulation mismatches in robust speech recognition. *Computer Speech & Language*, 46:535–557.

Wang, D. and Chen, J. (2018). Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(10):1702–1726.

Wang, Z.-Q., Le Roux, J., and Hershey, J. R. (2018). Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Warsitz, E. and Haeb-Umbach, R. (2007). Blind acoustic beamforming based on generalized eigenvalue decomposition. *IEEE Transactions on audio, speech, and language processing*, 15(5):1529–1539.

Watanabe, S., Mandel, M., Barker, J., Vincent, E., Arora, A., Chang, X., Khudanpur, S., Manohar, V., Povey, D., Raj, D., et al. (2020). Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings. *arXiv preprint arXiv:2004.09249*.

Wichern, G., Antognini, J., Flynn, M., Zhu, L. R., McQuinn, E., Crow, D., Manilow, E., and Roux, J. L. (2019). Wham!: Extending speech separation to noisy environments. *arXiv preprint arXiv:1907.01160*.

Xiao, X., Watanabe, S., Chng, E. S., and Li, H. (2016). Beamforming networks using spatial covariance features for far-field speech recognition. In *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–6. IEEE.

Zhang, J., Zorilă, C., Doddipatla, R., and Barker, J. (2020a). On end-to-end multi-channel time domain speech separation in reverberant environments. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6389–6393.

Zhang, Y., Qin, J., Park, D. S., Han, W., Chiu, C.-C., Pang, R., Le, Q. V., and Wu, Y. (2020b). Pushing the limits of semi-supervised learning for automatic speech recognition. *arXiv preprint arXiv:2010.10504*.

Zollinger, S. A. and Brumm, H. (2011). The lombard effect. *Current Biology*, 21(16):R614–R615.