‘DID THE SPEAKER CHANGE?’:

TEMPORAL TRACKING FOR OVERLAPPING SPEAKER SEGMENTATION IN MULTI-SPEAKER SCENARIOS

by

Aidan O. T. Hogg

A thesis submitted in fulfilment of the requirements for the
Doctor of Philosophy (PhD) Degree of Imperial College London and the
Diploma of Imperial College London (DIC) by research

Speech and Audio Processing Laboratory
Communications and Signal Processing Group
Department of Electrical and Electronic Engineering
Imperial College London
July 2022

COPYRIGHT DECLARATION

STATEMENT OF ORIGINALITY

I hereby certify that this thesis and the research to which it refers are the product of my own work under the guidance and supervision of Prof. Patrick A. Naylor and Dr Christine Evers. Any work that has been previously published and included in this thesis has been fully acknowledged in accordance with the standard referencing practices of this discipline. The material of this thesis has not been accepted for any degree and has not been concurrently submitted for the award of any other degree.

Aidan O. T. Hogg

July 2022

ABSTRACT

Diarization systems are an essential part of many speech processing applications, such as speaker indexing, improving automatic speech recognition (ASR) performance and making single speaker-based algorithms available for use in multi-speaker domains. This thesis will focus on the first task of the diarization process, that being the task of speaker segmentation which can be thought of as trying to answer the question 'Did the speaker change?' in an audio recording.

This thesis starts by showing that time-varying pitch properties can be used advantageously within the segmentation step of a multi-talker diarization system. It is then highlighted that an individual's pitch is smoothly varying and, therefore, can be predicted by means of a Kalman filter. Subsequently, it is shown that if the pitch is not predictable, then this is most likely due to a change in the speaker. Finally, a novel system is proposed that uses this approach of pitch prediction for speaker change detection.

This thesis then goes on to demonstrate how voiced harmonics can be useful in detecting when more than one speaker is talking, such as during overlapping speaker activity. A novel system is proposed to track multiple harmonics simultaneously, allowing for the determination of onsets and end-points of a speaker's utterance in the presence of an additional active speaker.

This thesis then extends this work to explore the use of a new multimodal approach for overlapping speaker segmentation that tracks both the fundamental frequency (F_0) and direction of arrival (DoA) of each speaker simultaneously. The proposed multiple hypothesis tracking system, which simultaneously tracks both features, shows an improvement in segmentation performance when compared to tracking these features separately.

Lastly, this thesis focuses on the DoA estimation part of the newly proposed multimodal approach. It does this by exploring a polynomial extension to the multiple signal classification (MUSIC) algorithm, spatio-spectral polynomial (SSP)-MUSIC, and evaluating its performance when using speech sound sources.

ACKNOWLEDGEMENTS

First and foremost I'd like to thank my supervisors Prof. Patrick A. Naylor and Dr. Christine Evers for always finding time to give me their extremely useful advice and guidance throughout my entire PhD. I can't thank you enough for everything.

I must also thank the entire Speech and Audio Processing lab at Imperial College for all the great times we spent together. Especially Vincent Neo who was beyond kind in so many ways. I hope this is only the start of our friendship and that it may continue long into the future. To Yunhao Liu for being such a great desk neighbour; you were always able to lighten up my day. Finally, also to Simon McKnight for all the enjoyable moments we had working together.

I would also like to thank Donald Sayers who was the first person to ever teach me Electronics. All I can say is that he was an awesome teacher and most definitely changed my life for the better. I would also delight in the chance to thank Captain Stuart Ellins and Michael Franklin who really helped me find my feet at the start of my university journey. Your guidance was invaluable. It is also essential to thank Heather Symonds for all the long hours she spent helping me with my PhD. Her charm and wit were always able to make me smile.

I must also thank my family including my loving parents Brian and Susan; my sister Emilie, and my brother Kristian for all their support and encouragement. Along with my cousin Kevin for all the fun we had growing up together.

Finally, I would like to thank God along with my church family and friends for always being there for me through the good times and the bad. In particular, I would like to thank Jia Chen for all her moral support; I can't imagine what life would be like without you in it. Dingeman Wolfert and Joash Kwek for all our long walks in the park discussing God and the troubles of life. Leo Tomita for teaching me the Bible and being such a great mentor. Matthew Robinson for all the great pub conversations and Thomas Collingwood for all the interesting chats over the last 8 years.

Soli Deo Gloria

"And ye shall know the truth and the truth shall make you free."

John VIII-XXXII

CONTENTS

ABBREVIATIONS

AHC	agglomerative hierarchical clustering
ASR	automatic speech recognition
BIC	Bayesian information criterion
BLSTM	bidirectional long short term memory network
CNN	convolutional neural network
CSSM	coherent signal-subspace method
DCT	discrete cosine transform
DER	diarization error rate
DFT	discrete Fourier transform
dLDA	dynamic latent Dirichlet allocation
DNN	deep neural network
DoA	direction of arrival
E-HMM	evolutive hidden Markov model
EKF	extended Kalman filter
EVD	eigenvalue decomposition
F_0	fundamental frequency
FAR	false alarm rate
FA	false alarm
FFV	fundamental frequency variation
FRIDA	finite rate of innovation direction of arrival
GCC	generalized cross-correlation
GLR	generalized likelihood ratio
GMM	Gaussian mixture model
HMM	hidden Markov model
HR	HIT rate

IFB	independent frequency bin
IHM	individual headset microphone
JPDA	joint probabilistic data association
KL	Kullback-Leibler
LOCATA	localization and tracking
LPC	linear predictive coding
LSP	line spectral pair
MDCC	modulation cepstral coefficient
MFCC	mel-frequency cepstral coefficient
MHT	multiple hypothesis tracking
MH	multi-hit
MOT	multiple object tracking
MSE	mean squared error
MUSIC	multiple signal classification
MWC	maximum weighted clique
NIST	National Institute of Standards and Technology
NLS	non-linear least squares
PCA	principal component analysis
PEFAC	pitch estimation filter with amplitude compression
PEVD	polynomial eigenvalue decomposition
PHAT	phase-transform
PHD	probability hypothesis density
PMHT	probabilistic multiple hypothesis tracking
RAPT	robust algorithm for pitch tracking
RMS	root mean square
RNN	recurrent neural network
S4D	SIDEKIT for diarization
SBR2	sequential best rotation algorithm
SC	speaker change
SDM	single distance microphone
SLAM	simultaneous localization and mapping
SMD	sequential matrix diagonalisation

SNR	signal-to-noise ratio
SP	spatial polynomial
SRP	steered response power
SSP	spatio-spectral polynomial
STFT	short-time Fourier transform
T60	reverberation time
TDE	time-delay estimation
TDoA	time difference of arrival
TOPS	test of orthogonality of projected subspaces
UKF	unscented Kalman filter
VAD	voice activity detection
WAVES	weighted average of signal-subspaces

NOTATIONS

Number sets

$\mathbb{R}$	real space
$\mathbb{C}$	complex space
$\mathbf{a}$	vector (lower case, bold)
$\mathbf{A}$	matrix (upper case, bold)
$\boldsymbol{\mathcal{A}}(z)$	polynomial matrix (upper case, bold calligraphic, parameterized)

Matrices

γ	eigenvalue
$\mathbf{0}$	zero matrix
$\mathbf{I}$	identity matrix
$\boldsymbol{\Gamma}$	eigenvalue matrix
$\mathbf{U}$	eigenvector matrix
$\mathbf{Q}$	unitary matrix

Operators

j	complex number, $\sqrt{-1}$
$*$	convolution
$.^*$	complex conjugate
$.^T$	transpose
$.^H$	Hermitian
$.^P$	para-Hermitian
$.^{-1}$	inverse
$\mathbb{E}$	expectation
$\Im$	imaginary part

$\Re$	real part
$\lvert\cdot\rvert$	magnitude/modulus
$\mathcal{F}$	Fourier transform
$\mathcal{Z}$	z-transform
dim	dimension
$\mathcal{O}\{\cdot\}$	big O notation

Polynomial Matrices

$\boldsymbol{\Gamma}(z)$	eigenvalue polynomial matrix
$\boldsymbol{\mathcal{U}}(z)$	eigenvector polynomial matrix
$\mathbf{R}(\tau)$	space-time covariance matrix
$\mathbf{R}(0)$	instantaneous covariance matrix
$\boldsymbol{\mathcal{R}}(z)$	z-transform of $\mathbf{R}(\tau)$

Signal Processing

F_s	sampling frequency
T_s	sampling period
τ	time delay/lag
$\{.\}_s$	signal (signal-plus-noise) subspace
$\{.\}_v$	noise (noise-only) subspace

LIST OF PUBLICATIONS

[1] A. O. T. Hogg, V. W. Neo, S. Weiss, C. Evers, and P. A. Naylor, "A polynomial eigenvalue decomposition MUSIC approach for broadband sound source localization," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, New Paltz, NY, Oct. 2021, pp. 326–330

[2] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multichannel overlapping speaker segmentation using multiple hypothesis tracking of acoustic and spatial features," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Jun. 2021, pp. 26–30

[3] A. O. T. Hogg, C. Evers, A. H. Moore, and P. A. Naylor, "Overlapping speaker segmentation using multiple hypothesis tracking of fundamental frequency," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 1479–1490, Mar. 2021

[4] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multiple hypothesis tracking for overlapping speaker segmentation," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2019, pp. 195–199

[5] A. O. T. Hogg, P. A. Naylor, and C. Evers, "Speaker change detection using fundamental frequency with application to multi-talker segmentation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2019, pp. 5826–5830

[6] S. W. McKnight, A. O. T. Hogg, V. W. Neo, and P. A. Naylor, "Studying human-based speaker diarization and comparing to state-of-the-art systems," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Nov. 2022

[7] V. W. Neo, S. Weiss, S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Polynomial eigenvalue decomposition-based target speaker voice activity detection in the presence of competing talkers," in *Proc. Int. Workshop on Acoust. Signal Enhancement (IWAENC)*, Sep. 2022, pp. 1–5

[8] S. W. McKnight, A. O. T. Hogg, V. W. Neo, and P. A. Naylor, "A study of salient modulation domain features for speaker identification," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Dec. 2021, pp. 705–712

[9] S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Analysis of phonetic dependence of segmentation errors in speaker diarization," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2020, pp. 1–5

[10] D. Sharma, A. O. T. Hogg, Y. Wang, A. Nour-Eldin, and P. A. Naylor, "Non-intrusive POLQA estimation of speech quality using recurrent neural networks," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2019, pp. 1–5

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

We, as humans, have evolved to become extremely good at undertaking certain tasks; one such task being our ability to identify a speaker just from their voice. However, this ability that comes so naturally to us is extremely hard for a computer to mimic. As a result of this speaker recognition is still an active area of research today.

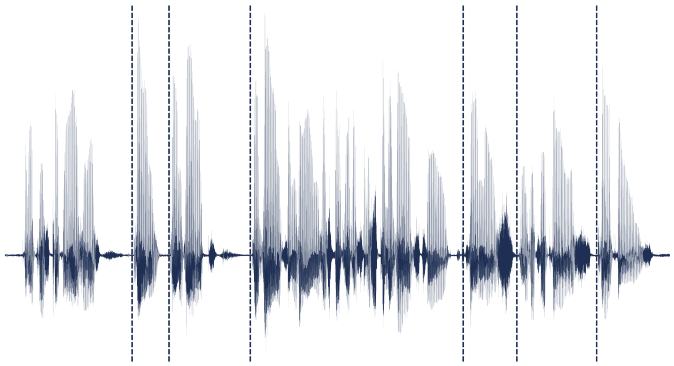
One of the main objectives that are of particular interest is that of speaker diarization. This consists of two tasks, one of which is to classify who is speaking and the other is to identify when they are speaking in an audio stream. It is normally more simply defined as answering the question: "who spoke when?" in an audio recording.

Speaker diarization is also normally complicated due to the fact that it is performed without any prior knowledge of the environment, where both the number of speakers and the amount of speech are unknown. Over the years, a lot of research has been undertaken to improve the performance of speaker diarization systems in different situations and under different conditions.

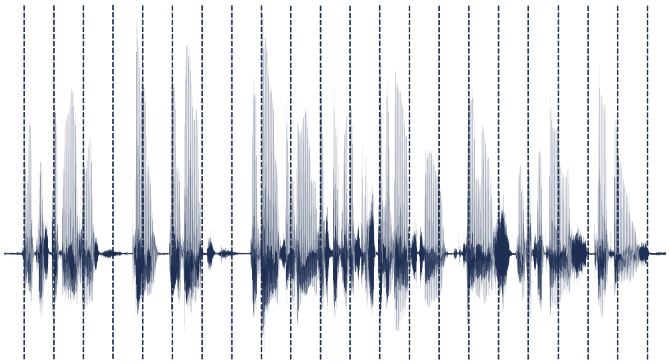
1.1 Motivation and Objectives

The topic of automatic speech recognition (ASR) has been a heavily researched area over the last few decades; a good review has been presented in [11]. This is due to the obvious benefits that come from a machine being able to decipher what a speaker has said. Today, many products and services on the market are taking advantage of this technology ranging from smartphones to smart cars. It appears that being able to talk to devices is seen as being very convenient.

On the other hand, determining the identity of the speaker in any given speech segment does not, on the surface, seem as desirable. This does not mean, however, that it is not extremely pragmatic in certain



(a) Segmentation on speaker change boundaries.



(b) Uniform segmentation.

Figure 1.1: A comparison between speaker change boundary segmentation and uniform segmentation.

situations. The process of speaker diarization can add many favourable benefits to speech technology systems, some of which include:

1. **Aiding transcription and ASR systems**

   It was shown in [12] that the performance of an ASR system can be greatly improved by the presence of only one speaker's speech. This is due to the fact that ASR systems are able to adapt to a single speaker's voice. In an environment of multiple speakers it is, therefore, profitable to separate the speakers in the audio stream so that the ASR system can be run on each speaker individually.

2. **Speaker indexing**

   The most conspicuous benefit of speaker diarization is that of speaker labelling within a multi-speaker speech signal. This is because a labelled transcription allows for ease of data processing and recovery by both machines and humans. For example, this richer level of labelled transcription for speech was recently explored in [13] with a focus given to translation.

3. **Making single speaker-based algorithms a feasible approach for multi-speaker tasks**
   There are many algorithms that have been developed to work well on individual speakers but will perform badly when more than one speaker is present. Diarization, however, makes it possible to use these methods on audio recordings containing multiple speakers. It does this by separating out all of the audio data for each of the speakers.

This thesis will focus on the first task of the diarization process, that being the task of speaker segmentation which can be thought of as trying to determine when speakers change in audio recordings. The segmentation task can often be overlooked, with more research being focused on the task of clustering. This is because uniform segmentation (see Fig. 1.1(b)) before clustering with a post-realignment step has been shown to yield good results in the past, e.g. [14]. However, if uniform segmentation is used the segments have to be small in duration, typically 2.5 s, which makes clustering more difficult as less information is contained in each segment. Correct segmentation on speaker change boundaries (see Fig. 1.1(a)), however, results in much larger segments that, in theory, only contain speech from one speaker making the clustering of the segments a much easier problem. In light of this there is, therefore, a real desire to improve segmentation systems that can accurately detect when speaker changes occur.

1.2 Original Contributions and Overview

The structure of this thesis is as follows:

Chapter 3

Introduces the concept of temporally tracking a single speaker's fundamental frequency (F_0).

Contributions

(a) A detailed investigation revealing that variations in pitch can be used as a reliable indicator of speaker changes in the audio of multi-speaker meetings.

(b) The development of a novel method to extract such speaker changes and test them on a widely available meeting corpus.

Papers

[8] A. O. T. Hogg, P. A. Naylor, and C. Evers, "Speaker change detection using fundamental frequency with application to multi-talker segmentation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2019, pp. 5826–5830

Chapter 4

Expands on Chapter 3 by allowing for the tracking of the F_0 of multiple speakers simultaneously.

Contributions

(a) An analysis that shows how voiced harmonics can be used to propagate the pitch of multiple speakers through periods of overlapping speech and, therefore, how voiced harmonics can be useful in detecting when more than one speaker is talking.

(b) The proposition of two novel systems that are able to track multiple harmonics simultaneously, allowing for the determination of onsets and end-points of a speaker's utterance in the presence of an additional active speaker.

(c) It is shown that the performance of a state-of-the-art speaker segmentation system can be improved if the pitch estimates obtained by one of the proposed systems are used as input features for a neural network.

Papers

[4] A. O. T. Hogg, C. Evers, A. H. Moore, and P. A. Naylor, "Overlapping speaker segmentation using multiple hypothesis tracking of fundamental frequency," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 1479–1490, Mar. 2021

[6] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multiple hypothesis tracking for overlapping speaker segmentation," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2019, pp. 195–199

Chapter 5

Builds on the tracking framework from Chapter 4 by tracking both the F_0 and direction of arrival (DoA) features of multiple speakers simultaneously.

Contributions

(a) A novel method is proposed that uses a multiple hypothesis tracking (MHT) framework to track the F_0 and DoA of multiple speakers simultaneously.

(b) It is shown that MHT of both the DoA and F_0 can lead to significant improvements in speaker segmentation performance over tracking just one of these features alone.

Papers

[3] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multichannel overlapping speaker segmentation using multiple hypothesis tracking of acoustic and spatial features," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Jun. 2021, pp. 26–30

Chapter 6

Develops a polynomial eigenvalue decomposition (PEVD) DoA estimation approach to improve the tracking performance that can be achieved in Chapter 5.

Contributions

(a) The proposal and development of a spatio-spectral polynomial (SSP)-multiple signal classification (MUSIC) approach for noisy reverberant speech, whereas previous work only [15, 16] considered stationary broadband data in a free-space propagation environment, i.e. it only contained the direct-path.

(b) The proposal of modifications to SSP-MUSIC to enhance the DoA estimation of noisy reverberant speech.

(c) An analysis that shows how the temporal, spatial, and spectral decorrelation of SSP-MUSIC can aid the robustness of the PEVD approach towards diffuse noise and reverberation effects.

Papers

[2] A. O. T. Hogg, V. W. Neo, S. Weiss, C. Evers, and P. A. Naylor, "A polynomial eigenvalue decomposition MUSIC approach for broadband sound source localization," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, New Paltz, NY, Oct. 2021, pp. 326–330

Chapter 2

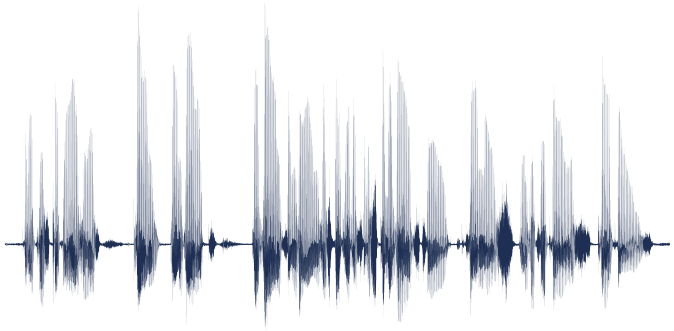
BACKGROUND THEORY AND LITERATURE REVIEW

2.1 Early Work

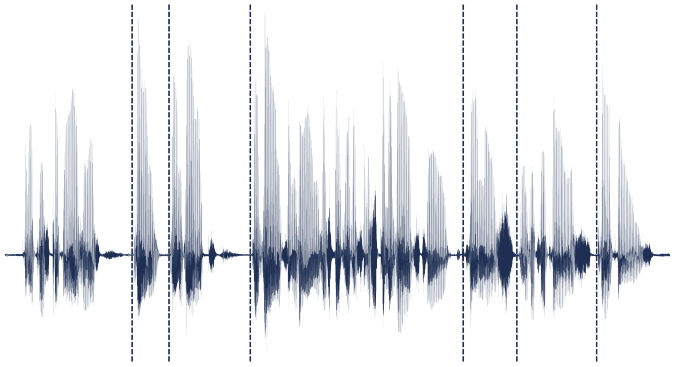
Since the 1960s, work has been carried out to develop systems that can perform speech recognition. At the start, these systems were only able to perform the most basic tasks which were to be able to distinguish speech and non-speech data in an audio recording.

As time went on, advancements in the field allowed some of the first work on speaker diarization to be undertaken in 1997. This was presented in [17] by Matthew A. Siegler *et al.* They developed a system where the Kullback-Leibler (KL) distance [18] was calculated for the means and variances of two consecutive windows in an audio recording. The KL distance would then be compared against a local maximum and would generate a new speaker change boundary if that local maximum was exceeded.

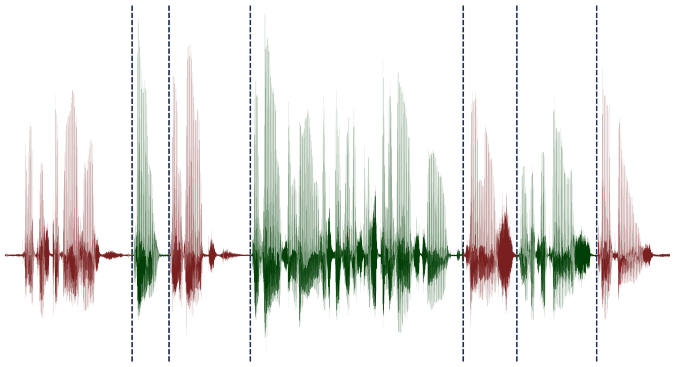
This early work was focused on specific situations where only two speakers needed to be classified, e.g. applications such as broadcast news data which was frequently used for evaluation purposes. A thorough review of the literature in the area of broadcast news data has been synthesised in [19]. More recently the focus has shifted to the domain of meeting room environments where normally more than two speakers are typically present and, hence, overlapping speech is more prevalent. This new domain makes the task of speaker diarization much more challenging as now more obstacles, such as noise and reverberation, have to be overcome. This is as well as having to deal with distant room microphones as the data no longer benefits from being captured in a highly controlled environment involving high-quality, close-talking microphones (which is how broadcast news data is often recorded).



(a) Speech signal.



(b) Segmentation.



(c) Clustering.

Figure 2.1: The diarization process where red and green indicate the two speaker's speech for the given signal.

2.2 Diarization

As stated earlier, speaker diarization can be thought of as answering the question "who spoke when?" [20]. The problem of diarization is challenging because there is no prior knowledge about the number of speakers

or the amount of speech the recording contains. Accurate diarization has become increasingly important in recent years for a multitude of tasks including voice control of smart devices and robot audition [21, 22]. Diarization is also required for applications such as speaker indexing [13], ASR [12] and to enable the use of single speaker-based algorithms in multi-speaker domains [23]. The diarization process is commonly separated into three independent tasks:

1. Segmentation of the speech signal (see Fig. 2.1(a) and (b))
2. Clustering the segments (see Fig. 2.1(c))
3. An optional realign/re-segment step (to refine the segmentation of a diarization system that relies on the clustering of an initial uniform segmentation) [24, 25]

This thesis will focus on the first step of the diarization process, that being the task of segmentation. It should be noted that the segmentation task, i.e. the identification of the onsets and end-points of speakers, is often made more complicated by the presence of overlapping speech where multiple speakers are active at the same time [2, 26]. Overlapping speech commonly occurs in conversational speech due to interruptions and backchannel vocalisations [27]. Environmental factors, such as reverberation [28] and noise [29], also render the task of accurate segmentation difficult to achieve.

2.3 Segmentation

The simplest form of segmentation is that of uniform segmentation [14, 30–32]. This is where the audio stream is divided into a number of segments of equal length which are then clustered together based on acoustic properties (see Fig. 1.1(b) on Page 2). A trade-off arises from wanting the segments to be small so that they only contain one speaker but sufficiently large so that they contain enough information about the speaker to be able to cluster the segment. This leads to methods that try to identify the onsets and end-points of a speaker (see Fig. 1.1(a)) where each segment is the largest it can possibly be while still only containing one speaker [33, 34].

2.3.1 Types of Segmentation

It was common in the past to see segmentation systems assigned to one of three categories: energy-based segmentation, model-based segmentation and metric-based segmentation.

2.3.1.1 Energy-based segmentation

Energy-based segmentation [35, 36] is where segments are generated by cutting the audio stream when silence is detected. This task is much harder than it first appears owing to the fact that most recordings are taken in noisy environments making energy cues for discrimination unreliable.

Voice-based segmentation

Instead of segmenting based on energy, it is now more common to see systems that segment based on the presence of speech. This is very similar to energy segmentation, however, instead of segmenting when silence is detected, these systems segment when no speech is detected. This form of segmentation is often referred to as voice activity detection (VAD) and many algorithms have been proposed for this task including hidden Markov models [37], information entropy [38] and wavelet transform-based methods [39]. Many improvements to this basic implementation have been suggested in the literature.

A typical VAD algorithm is described in [40] in the following way:

1. A speech enhancement or noise reduction algorithm is applied (optional)
2. The audio recording is windowed and feature vectors are calculated for each window of speech
3. Each window of speech is classified as speech or non-speech based on its feature vector

Although, a perfect VAD does not yet exist even today most speaker segmentation systems still use a VAD as part of a pre-processing step [41] to remove any non-voiced regions within the audio stream.

The number one drawback of all energy-based segmentation, including VADs, is that the segment boundaries are not directly related to speaker changes. E.g. how do you know that a silence is due to a change in the speaker or just a brief pause of the active speaker.

2.3.1.2 Model-based segmentation

Model-based segmentation as the name suggests uses a model to classify different parts of an audio recording. For example in [33] Gaussian mixture models (GMMs) are constructed using a training corpus for a fixed set of acoustic classes, such as the speaker, music, etc. The incoming audio stream can be classified by maximum likelihood selection. Segment boundaries are assumed where a change in the acoustic class occurs.

2.3.1.3 Metric-based segmentation

Metric-based segmentation works by first calculating all the distances between neighbouring windows and then by segmenting the audio recording at the maxima of these distances. Examples of metric-based approaches include the KL distance [17] , generalized likelihood ratio (GLR) [42–44] and the Bayesian information criterion (BIC) [34, 45] which has numerous variants [46–49].

2.3.1.4 Feature-based segmentation

In more recent years, segmentation has been achieved using i-vectors [50] and DNN-based embeddings [51, 52]. Other methods that generate the segmentation from the raw audio data are frame-based neural network models, such as deep neural networks (DNNs) [53], convolutional neural networks (CNNs) [54–56] and bidirectional long short term memory networks (BLSTMs) [2, 41, 57].

In 2017, [58] showed that, by using a deep learning architecture, a low segmentation error can be achieved. In [58], a recurrent convolutional neural network is trained and applied to magnitude spectrograms of speech segments. This removes any need to calculate frame-based features.

In [59], another approach is utilised that, instead of removing the need for frame-based features, trains a neural network to generate better features that are optimised for speaker segmentation. This work is motivated by the fact that common features, such as mel-frequency cepstral coefficients (MFCCs), contain a lot of redundant information, such as phonetic information, that is unrelated to the speaker's identity.

In the last few years, vast amounts of data have become available which has resulted in the rise of more machine learning-based approaches [60], however, they are not always the best solution. This is because they are rarely designed to take into account any information that is gathered from conventional speech processing techniques and instead opts to work on the raw speech signal.

2.3.2 Features for Feature-Based Segmentation

It is common for most segmentation systems to use a frame-based approach. This works by first splitting up the audio recording into time frames, typically 10-150 ms, and then calculating a feature vector for each time frame. These feature vectors are then used to make splitting decisions on speaker change boundaries. There are many features that contain information that would be advantageous to use during the segmentation process, where some of these features are more useful than others. The two main

categories that most features are sorted into are acoustic and spatial cues.

2.3.2.1 Acoustic features

Acoustic features, as their name suggests, are derived from the acoustic properties of the signal. Almost all approaches that use acoustic features work on a frame-by-frame basis generating a feature vector for each frame. Some of the more typical acoustic features, that have been used for segmentation in the past, are explored in this section.

Mel-frequency cepstral coefficients (MFCCs)

MFCCs were introduced in [61] by Davis and Mermelstein and are commonplace in the area of ASR systems. Their aim is to try to capture the frequency spectrum of the signal in a small number of coefficients. An important fact to note is that MFCCs were never intended to be used for speaker diarization. This is due to the fact that the spectrum information is mostly related to the phonetic content, i.e. the resonant frequencies or formants in the spectrum, and is concerned with the phone being uttered by the speaker. However, due to the redundant information contained within MFCCs, they are often used and are able to achieve a very good performance.

The MFCCs provide a compact way of representing the envelope of a short-term power spectrum. The basic steps that are necessary to calculate the MFCCs for a portion of speech are as follows:

1. The speech signal is first segmented into frames. This is normally achieved using a window such as the Hanning or Hamming windows.
2. Then for each frame, $x(n)$, the discrete Fourier transform (DFT) is calculated using $X(k) = \sum_{n=0}^{N-1} x(n)e^{\frac{-j2\pi nk}{N}}$ for $k = 0 \cdots N-1$ where N is the number of points used to calculate the DFT.
3. The Mel filter bank is then applied to the resulting power spectrum. The Mel spectrum, $s(n)$ of the magnitude spectrum $X(k)$ is calculated by $s(m) = \sum_{k=0}^{N-1} \left(|X(k)|^2 H_m(k)\right)$ where M is the total number of triangular Mel weighting filters. $H_m(k)$ is the weight given to the kth energy spectrum bin contributing to the mth output band. (It is important to note that Mel frequency, f_{Mel} aims to capture how humans perceive frequency and can be approximated as $f_{\text{Mel}} = 2595 \log_{10}\left(1 + \frac{f}{700}\right)$ where, f is the frequency given in Hertz).
4. The energy in each triangular Mel filter is then calculated.
5. The logarithm of each of these energies is then taken.
6. The discrete cosine transform (DCT) is then performed on these log energies using $c(n) =$

$\sum_{m=0}^{M-1} \log_{10}(s(m)) \cos\left(\frac{\pi n(m-0.5)}{M}\right)$ for $n = 0 \cdots C-1$ where $c(n)$ are the MFCCs, and C is the number of MFCCs.

7. Only the first $C_{\max}$ (e.g. $C_{\max} = 12$) coefficients are kept, the rest are discarded.

These steps are, of course, sometimes modified depending on the application in question.

Fundamental frequency variations (FFVs)

Fundamental frequency variation (FFV) features were introduced in [62] as a way of representing the instantaneous change in F_0. This can be thought of as capturing how much a speaker's F_0 varies in a relatively small number of coefficients, e.g. the FFV spectrum normally consists of 7 coefficients.

The speech signal first undergoes a pre-emphasis before it is split into frames using an overlapping window. Two frequency spectra are calculated for the left and right sides of each frame. Each of these two spectra in turn are then dilated in frequency while the other spectrum is kept constant. A measure of realignment is then obtained from a modified dot product. This result is then passed through a filter bank to isolate the different types of pitch (i.e. quickly falling pitch, slowly falling pitch, flat pitch, slowly rising pitch, and quickly rising pitch) and a couple of filters are also used for normalisation.

The reason why FFV features are informative in the task of speaker segmentation is due to the fact that the variation of a speaker's pitch is very different depending on the speaker.

Modulation cepstral coefficients (MDCCs)

Modulation cepstral coefficients (MDCCs), introduced in [63] and developed in [10], are modulation features that can be used for speaker segmentation. This use of modulation features is motivated by the fact that different speakers' speech has different dynamic properties.

There has been much interest in amplitude modulation domain processing of speech because low-frequency modulations of speech are the fundamental carrier of linguistic information. This representation has been exploited in areas such as speech coding [64], recognition [65, 66], enhancement [67, 68] and speech intelligibility modelling [69, 70]. This approach is also motivated by studies of the human auditory system [71] that point to analysis and separation of different acoustic objects in this domain. A number of modulation-domain methods have been proposed in the fields of emotion detection [72] and speech quality estimation [73, 74]. In the field of speech and acoustics, there are a number of definitions of the amplitude modulation spectrum that differ from the subband decomposition literature. The procedure

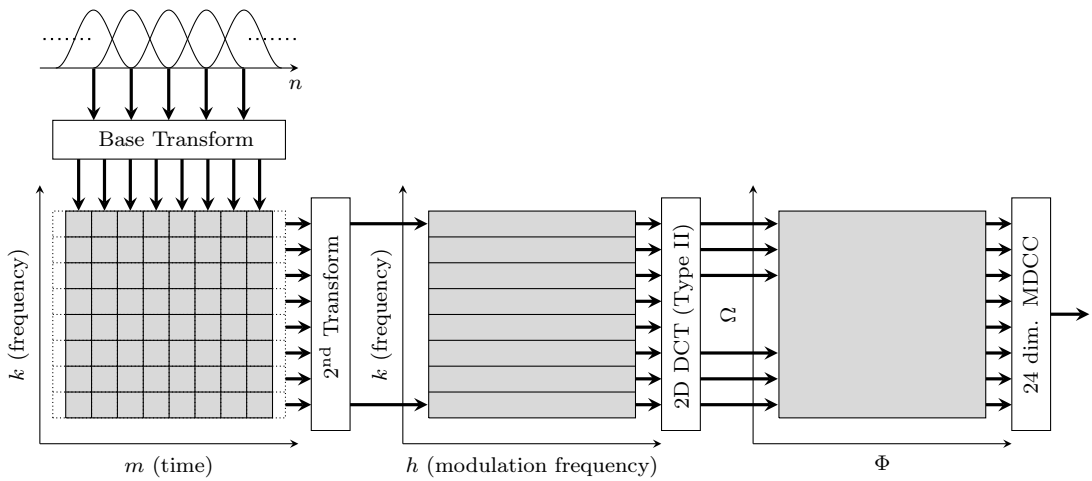


Figure 2.2: Overview of the procedure for computing the modulation spectrogram representation of a speech signal.

used to define the modulation domain representation of the signal is shown in Fig. 2.2. The first transform in the Fig. 2.2 is applied to the time-domain signal to decompose it into subband signals (using linear frequency spacing). The temporal envelope within each band is then computed. Denoting the length N time domain signal as $s(n)$, its short-time Fourier transform (STFT) is calculated as

$$S_k(m) = \sum_{n=0}^{N} s(n) w_a(n - mL) e^{-j\frac{2\pi}{N}k} \ , \tag{2.1}$$

where m is the short-time frame index (typically 30 ms duration), which in the context of modulation domain processing is defined as an 'acoustic frame' [64], $w_a(n)$ is the window applied on each frame and L is the acoustic frame increment in samples. After the first STFT, the temporal envelope of each acoustic frequency band k (also called the modulating signal) is obtained as the magnitude of the transformed signal, $|S(m,k)|$. To obtain the modulation spectrum, a window function $w_m(n)$ is used to segment the amplitude envelope of each frequency bin and a second STFT is performed on each modulation frame, as

$$S_l(k,h) = \sum_{m=0}^{M} |S_k(m)| w_m(m - lL) e^{-j\frac{2\pi}{H}h} \ , \tag{2.2}$$

where H is the number of modulation frequency bins. In the following description the index of the modulation frame, l, is omitted for clarity because the features are extracted independently for each modulation frame. The modulation spectrogram is, therefore, given by $P(k,h) = |S(k,h)|^2$. In order to

compress the information in the modulation spectrum, a two-dimensional DCT-II (2D-DCT) can be used on the modulation spectrogram $P(k,h)$ to produce a set of DCT coefficients

$$D(\Omega,\Phi) = \sum_{k=0}^{K-1}\sum_{h=0}^{H-1} WP(k,h)\cos\left[\frac{\pi}{K}\left(k+\frac{1}{2}\right)\Omega\right]\cos\left[\frac{\pi}{H}\left(h+\frac{1}{2}\right)\Phi\right], \tag{2.3}$$

where $W = \sqrt{(1/K)}\sqrt{(1/H)}$ for $\Omega = 0, ..., K-1$ and $\Phi = 0, ..., H-1$. From empirical experiments, it has been found that only a few coefficients from the upper triangle of $D(\Omega,\Phi)$ are sufficient for capturing most of the variation in the modulation spectrum. For example, 24 MDCCs can be taken from the following set, $[D(1,1:21), D(2,1:3)]$.

2.3.2.2 Spatial features

If there is more than one microphone available, e.g. most voice-controlled home assistants currently possess two or more microphones, it is also possible to exploit spatial features as well as acoustic ones for the task of speaker segmentation. This is of particular interest when the speakers in question are located in different positions within a room.

Time difference of arrival (TDOA)

The most straightforward spatial feature that can be utilised is the time difference of arrival (TDoA). TDoA features measure the delay of the signal at multiple microphones with regard to a reference microphone. Therefore, depending on where the speaker is relative to the microphone array, the TDoA will be different.

The most common approach to solve this problem is an algorithm called the generalized cross-correlation (GCC)-phase-transform (PHAT). This method was first proposed by Knapp and Carter in [75] where they defined the GCC-PHAT as

$$\mathrm{R}_{\mathrm{PHAT}}(\tau) = \frac{1}{2\pi}\int_{-\infty}^{+\infty}\frac{X_i(f)X_j^*(f)}{|X_i(f)X_j^*(f)|}e^{jf\tau}df\ , \tag{2.4}$$

where $X_i(f)$ and $X_j(f)$ are the signals recorded at two different microphones in the Fourier domain; f is the angular frequency, $[.]^*$ stands for the conjugate operation. The TDoA between these two microphones is then estimated as

$$\hat{d}_{\mathrm{PHAT}}(i,j) = \arg\max_{\tau}\left(\mathrm{R}_{\mathrm{PHAT}}(\tau)\right)\ . \tag{2.5}$$

Direction of arrival (DoA)

DoA estimation is the process of calculating the azimuth or angle of a given sound source relative to an array of microphones. This can be done in a number of ways and the most simple of which is to use the TDoA information. This is because if the microphone array geometry is known then the TDoA information can be used to compute the DoA estimate. Many DoA estimation approaches have been proposed including time-delay estimation (TDE)-based, beamformer-based and subspace-based methods [22]. The TDE-based method [75] first computes the TDoA for different microphone pairs and uses *a priori* information about the microphone positions to compute the DoAs. Beamformer-based methods [76, 77] scan the acoustic environment by focusing the microphone array directional pick-up pattern in the directions corresponding to the highest sound intensities. Another common way to calculate the DoA is to use the steered response power (SRP)-PHAT algorithm [78] which works by finding the angle that maximizes the output of a steered delay-and-sum beamformer [79]. Many other algorithms have also been proposed in the past including: MUSIC [80], coherent signal-subspace method (CSSM) [81], weighted average of signal-subspaces (WAVES) [82], test of orthogonality of projected subspaces (TOPS) [83] and finite rate of innovation direction of arrival (FRIDA) [84].

2.3.3 Overlapping Speaker Segmentation

One of the most challenging aspects of the segmentation task is that of overlapping speakers; that is to say when a new speaker starts before the previous speaker has stopped. In the past, the issues surrounding overlapping speaker segmentation were not considered, however, more recently various methods have been proposed to solve this problem. These methods include hidden Markov model (HMM)-based methods that use MFCCs, linear predictive coding (LPC) and root mean square (RMS) energy features [85]. Methods that use long-term conversational features [86] have also been put forward along with multimodal techniques that use multiple microphone and camera systems [87]. Deep learning approaches have also become increasingly prevalent [57, 88–90] but they often require large amounts of labelled training data. As a result, overlapping speaker segmentation is still very much an open problem due to its complex nature and will be a major focus of this thesis.

2.3.4 Temporal Tracking of Features for Speaker Segmentation

It has also been shown in [91–93] that temporal tracking of features can be advantageous when solving the speaker segmentation task. This work on speaker tracking has almost exclusively been applied to

the problem of online (real-time) speaker segmentation. In this thesis, however, this work on speaker tracking will be extended to explore the benefits of using speaker tracking for overlapping speaker segmentation. More formally, it will ultimately explore how the prediction error given by a Kalman filter (see Section 2.3.4.1) tracking multiple features in a MHT framework (see Section 2.3.4.2) can be used to determine speaker change boundaries.

2.3.4.1 Kalman filter

Kalman filters were first proposed in the 1960s by Rudolf E. Kálmán [94] and can be thought of as a dynamic least square approach. Kalman filters are now commonplace in a multitude of applications due to them being optimal estimators in Gaussian noise [95]. This makes Kalman filters useful as a data fusion algorithm that allows them to track multiple noisy features. They work by using a prediction from a model and an observation at each time frame where the Kalman gain combines these two estimates based on which is more reliable. Kalman filters are used in this work to track multiple speech features from different speakers.

2.3.4.2 Multiple hypothesis tracking

MHT is presented in [96] and was revisited in [97]. The method has been shown to be popular in both visual and radar target tracking [98]. It works by generating a list of possible track hypotheses for each target highlighting the inherent data association problem. The likelihood of each track hypothesis is then evaluated and unlikely tracks are removed, consequently, only leaving the tracks that provide the best possible solution. Since the entire track hypothesis can be considered when computing the likelihood, MHT is able to effectively exploit higher-order information, e.g. as the long-term motion of the target. Speaker segmentation can be thought of as finding the onset and end-point of each speaker. Therefore, MHT can be used to track the acoustic and spatial features of different speakers where the start and end of all the tracks provide the complete speaker segmentation.

2.3.5 Evaluation For Segmentation

2.3.5.1 Performance metrics

There are many ways to measure the performance of a segmentation system which are often classified in two ways: categorical or statistical. The metrics used in this thesis are discussed below.

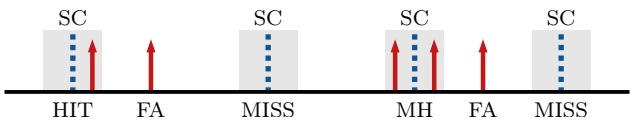


Figure 2.3: Visual representation of the evaluation framework used where the dashed lines represent the oracle speaker change (SC) boundaries and the grey regions correspond to the given collar. The red arrows indicate the detected outputs generated from the segmentation algorithm.

Categorical metrics

Categorical metrics can be thought of as metrics that use a binary decision in their evaluation of the segmentation performance. In this work, the following categorical metrics have been defined. A 'HIT' is when a speaker change has been detected once. A 'MISS' is when a speaker change has not been detected and a 'multi-hit (MH)' is when a speaker change has been detected multiple times within a time collar applied around every ground-truth speaker change in order to account for possible inaccuracies [9]. A 'false alarm (FA)' is when a detection falls outside of any speaker change collars. The HIT rate (HR) is given by

$$\frac{\text{HITs} + \text{MHs}}{\text{HITs} + \text{MHs} + \text{MISSs}} \text{ expressed as } \%. \tag{2.6}$$

The MISS rate is given by the complement percentage of the HIT rate. The false alarm (FA) rate is given by

$$\frac{\text{FAs}}{\text{HITs} + \text{MHs} + \text{FAs}} \text{ expressed as } \%. \tag{2.7}$$

The MH rate is given by

$$\frac{\text{MHs}}{\text{HITs} + \text{MHs}} \text{ expressed as } \%. \tag{2.8}$$

An illustration is given in Fig. 2.3 where the scores would be as follows: HIT rate: 50%, MISS rate: 50%, MH rate: 50%, FA rate: 50%.

Statistical metrics

Statistical metrics arise due to the fact that categorical metrics by design throw away a lot of performance information. This can be seen by considering a simple example where two systems share the same HR but one system is far more accurate, that is to say, one system has HITs that are much closer to the speaker change boundaries. Statistical metrics, therefore, try to capture some of this lost information.

Mean squared error

In this work, to determine the accuracy of the performance, the mean squared error (MSE) in the

time-domain is calculated for all HITs and the closest MH detections compared to the ground-truth speaker changes, which are provided by hand labelling.

Coverage and purity

Another two statistical metrics that are calculated: 1) the segment-wise coverage, which is the ratio of the duration of the intersection with the most co-occurring hypothesis segment and the duration of the reference segment; 2) the purity, which would be the same as the coverage if the reference and hypothesis segments were to switch roles and indicates how pure the hypothesis is for each segment. It should, therefore, be noted that the purity and the coverage are complementary measures. More formally the coverage [99] is defined as

$$\text{coverage}(N_r, N_h) = \frac{\sum_{r \in N_r} \max_{h \in N_h} |r \cap h|}{\sum_{r \in N_r} |r|} , \tag{2.9}$$

where $|r|$ is the duration of segment r within the set of reference segments N_r,and where $r \cap h$ is the intersection of segments r and segments h within the set of hypothesis segments N_h.

The results presented in Section 4.4.2 are a duration-weighted average over each segment. In this work, the implementation of such metrics from `pyannote.metrics` [100] was utilised.

2.3.5.2 Meeting room data

To evaluate the performance of the different systems, audio data from different corpora are exploited. The big advantage of using corpora is that it makes the comparison with other algorithms that have been developed in the literature easier.

AMI meeting corpus

The AMI corpus [101], is used for meeting room speaker segmentation and diarization. It contains 100 hours of meeting room data captured from three rooms, that differ in shape and construction, located across three different sites. This makes the corpus data generalisable as each meeting room recording possesses very different acoustic properties. The recordings are of meeting scenarios where the participants role-play a design team from an electronics company that is developing a new type of remote control. The meetings were all conducted in English with mostly non-native speakers.

The rooms were set up to record both close-talking and far-field audio using multiple microphones. In practice, this consisted of headset condenser microphones and omnidirectional lapel microphones for the

close-talking audio. For the far-field audio, circular microphone arrays were used that consisted of four or eight miniature omnidirectional electret microphones.

In this thesis, 24 meeting room recordings from the AMI corpus are considered for evaluation each consisting of a length of approximately 30 mins. In Chapter 3 and Chapter 4, a mixed-down stream of the individual headset microphones (IHMs) is used to benchmark performance against a single distance microphone (SDM). This SDM audio was captured from a single microphone located on a circular array placed in the centre of the table which the participants are sitting around. In Chapter 5 and Chapter 6, multichannel signal processing is exploited and the audio from all 8 channels of the same circular array is utilised. The corpus also provides suggested partitions for training, validation and testing which are used for this thesis.

The corpus has also been labelled using an energy-based technique [102] with a human validation step, so an accurate ground truth is available as a reference [103]. This, therefore, makes the AMI corpus ideal for diarization testing.

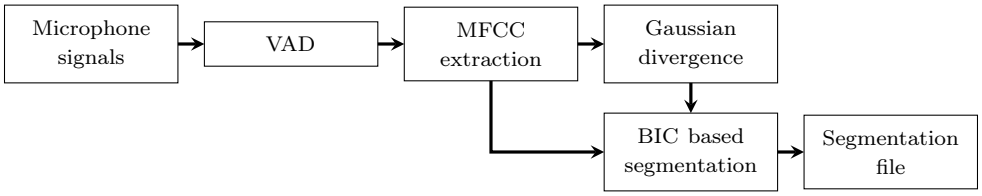
LOCATA corpus

The localization and tracking (LOCATA) corpus [22] was created to provide a novel framework for evaluating and benchmarking sound source localization and tracking algorithms. It contains recordings from four different microphone arrays in static and dynamic scenarios. It also provides the ground-truth positions and orientations for all sources and sensors, hand-labelled voice activity information, and close-talking microphone signals as a reference.

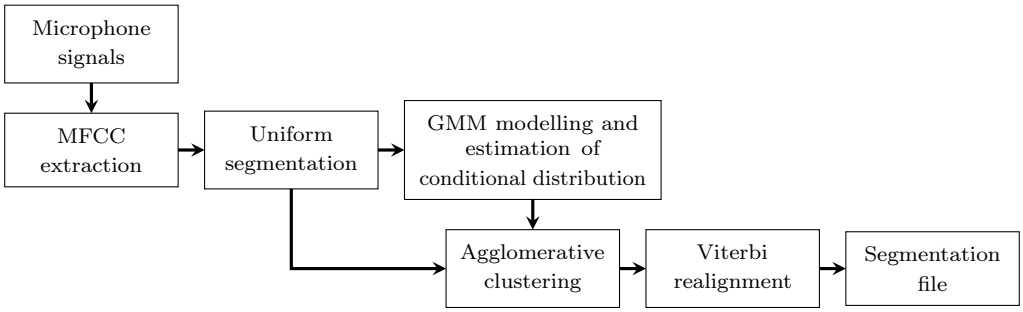
The LOCATA corpus was not designed to evaluate speaker segmentation, however, it is still a useful corpus as it can effectively evaluate an algorithm's spatial and acoustic tracking performance of a single speaker.

TIMIT corpus

The TIMIT corpus [104] was designed for the development and evaluation of ASR systems. The corpus contains broadband recordings of 630 speakers from eight major dialects of American English, where each speaker reads ten phonetically rich sentences. Consequently, the TIMIT corpus provides a way to evaluate speaker segmentation algorithms for a large number of different speakers. The anechoic recordings are also ideal for use in simulated environments.



(a) The 'SIDEKIT' segmentation system.



(b) The 'DiarTK' segmentation system.

Figure 2.4: Segmentation system architecture comparison.

2.3.5.3 Toolkits

There are many toolkits available for segmentation (as well as diarization) and each is specialised for a particular situation or for a particular type of feature.

Pyannote

Pyannote [41] is an open-source toolkit written in Python for speaker segmentation and speaker diarization. At its core, Pyannote is a machine learning framework based on PyTorch and provides a set of trainable end-to-end neural building blocks that can be utilised to build a speaker segmentation system. In this work, Pyannote is used as a baseline system as it comes with a number of pre-trained speaker segmentation models that can be considered state-of-the-art.

SIDEKIT for diarization (S4D)

SIDEKIT for diarization (S4D) is an open-source speaker diarization extension package of SIDEKIT [105] which was written in 2015 and is a Python toolkit that provides the whole chain of tools required to perform speaker diarization. The aim of SIDEKIT was to unify the implementations of the most common methods in one place in order to make it easier to carry out speech processing projects.

The S4D system in particular consists of three steps, shown in Fig. 2.4(a). The first step merges all the voice active regions of the VAD output that are in close proximity to each other. The second step performs Gaussian divergence segmentation where the MFCCs are used to segment the voice active regions that contain multiple speakers. Lastly, linear BIC based segmentation is performed which also uses the MFCCs to fuse consecutive voice active regions of the same speaker.

DiarTK

There are many features that can be used for diarization: acoustic features; spatial features and visual features to list a few. All of these features have different dimensionalities and statistical properties as a result, therefore, most diarization systems can only cope with a subset of these features.

DiarTK detailed in [14] was written in 2012 using C++ and was the first tool to be explicitly designed to be able to use more than one feature stream. A system overview is shown in Fig. 2.4(b).

LIUM

The LIUM toolbox [106] focuses on the task of broadcast news diarization. It works by first computing MFCC features along with a corresponding energy parameter. A two-phase speaker segmentation is then utilised that is based on GLR to calculate the speaker change boundaries and a BIC distance metric is used for the fusion of neighbouring segments belonging to the same speaker. The next step is a BIC hierarchical clustering approach that is deployed to merge the closest clusters until the BIC distance becomes positive. Finally, a Viterbi realignment step is performed to improve the segmentation.

ALIZÉ

The ALIZÉ [107] platform was designed for the task of speaker recognition. It, however, contains a speaker diarization approach which works by first using a VAD to classify the audio segments into the following pre-defined categories of speech, music, music plus speech or telephone speech. Next, speaker segmentation and clustering are achieved by using evolutive hidden Markov models (E-HMMs). An additional segmentation step is then performed in order to refine the initial speaker segmentation and to remove irrelevant speakers, i.e. speakers with a low number of frames assigned to them.

Chapter 3

SPEAKER SEGMENTATION USING FUNDAMENTAL FREQUENCY

3.1 Introduction

It is often desirable to keep records of speech, for example, during conference calls and at meetings. To store these discussions in a more useful manner, automatic speech recognition (ASR) can be used to generate transcripts. Although, ASR addresses the question of what is said, it cannot answer the question of who spoke at any given time. Accurate knowledge of the identity of the speaker is typically required for speaker indexing [13]; improvement in ASR performance [12] and to bring single speaker-based algorithms into multi-speaker domains. The task of identifying a speaker within an audio recording or stream is often referred to as diarization, which has the end goal of answering the question: "who spoke when?" [20]. The process of audio diarization consists of two tasks: the first task is segmentation which establishes when a new talker starts speaking and the current speaker stops; the second task is clustering, where every segmented part of the audio containing speech is assigned to an individual speaker. This whole process is also often complicated by the presence of reverberation [28] and noise [29].

In the past, systems have been proposed for diarization, some of the most commonly used have been discussed in Section 2.3.5.3 including LIUM [106], DiarTk [14], ALIZÉ [107] and SIDEKIT for diarization (S4D) [105]. These systems all contain different segmentation subsystems which can be grouped together into a set of categories. The first type uses mel-frequency cepstral coefficients (MFCCs) [61] to perform Bayesian information criterion (BIC) segmentation [105,106]. The second type uses a uniform segmentation [14]. The last type performs a one-step segmentation and clustering algorithm in the form of an evolutive hidden Markov model (E-HMM) [107]. Other segmentation algorithms have also been proposed in the literature [46, 49, 108].

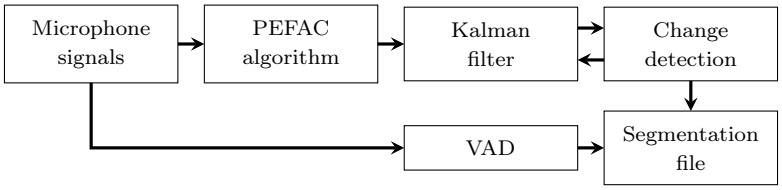


Figure 3.1: Proposed fundamental frequency (F_0)-based segmentation system.

In this chapter, the S4D system, shown in Fig. 2.4(a) on Page 20, which is an open-source speaker diarization extension package of SIDEKIT will be used as a baseline so that the proposed method can be compared against a typical segmentation approach. It is important to note that none of these popular systems, including S4D, use the F_0 of voiced speech to perform segmentation. The F_0 is ordinarily only utilised as a feature [62] to improve the performance of the clustering component in a diarization system. In the past, methods have been proposed that use the F_0 to improve the segmentation process. In [93] the F_0 is used alongside line spectral pair (LSP) [109] and MFCC features to calculate a divergence distance threshold to detect speaker change boundaries. There have also been methods developed for real-time diarization which use the F_0 as the sole feature [110, 111]. None of these previous methods, however, attempt to model the F_0, an approach which has two major advantages. First, the model can be exploited to remove errors in the F_0 estimates. Second, the errors in the F_0 prediction given by the model can be utilised to detect speaker changes instead of using the delta F_0, being the change between two frames [110, 111].

This chapter will focus on the task of segmentation and show why temporal variation in the F_0 can be used advantageously for speaker change detection. It will also present a novel method using the F_0 to improve the segmentation process. Fig. 3.1 shows the novel system that is proposed in this chapter which takes advantage of F_0 modelling when performing segmentation. This method uses a Kalman filter to predict the future F_0 of the speaker. Kalman filters have been used in the past to perform F_0 estimation, for example [112–114]. In contrast, the proposed system only uses the Kalman filter for future F_0 prediction and not F_0 estimation. Hence, it could be used in conjunction with any F_0 estimator; for the purposes of this chapter the pitch estimation filter with amplitude compression (PEFAC) [115] is used as the F_0 estimator. The main idea behind this method is that the F_0 prediction made by the Kalman filter can be used to decide if there has been a change in speaker. It does this by assuming that the F_0 of a speaker should be predictable whereas, if the F_0 cannot be predicted, then a speaker change may be the cause.

It is shown in this chapter that the proposed Kalman filter prediction error-based approach performed well

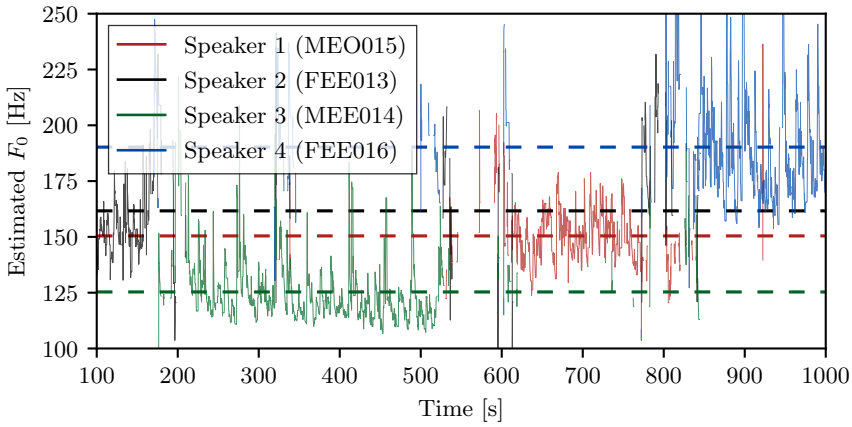
when compared against S4D, a MFCC-based method. It can also be seen in Table 3.2 that the speaker change detection increases from 58.0% to 80.4% on the AMI corpus. This work was also published in the following paper [5].

3.1.1 Fundamental Frequency Estimation

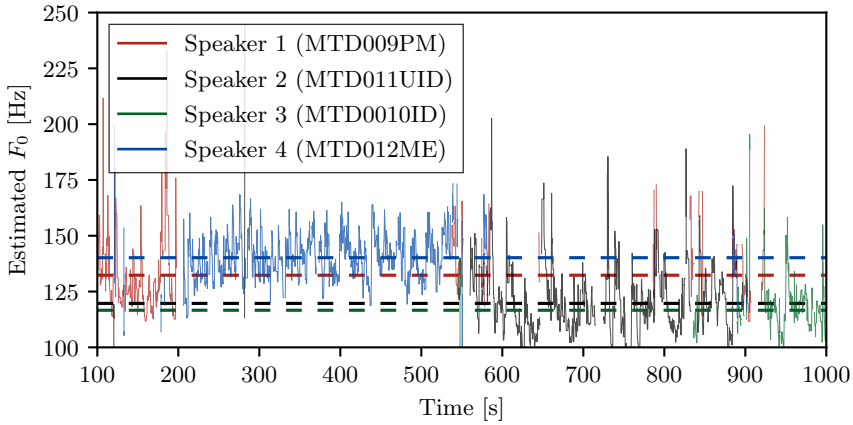
The PEFAC algorithm [115] works by convolving the signal's power spectral density in the log-frequency domain with a filter that sums the energy of the F_0 harmonics and is one of many F_0 estimators. Other estimators include the robust algorithm for pitch tracking (RAPT) [116] algorithm that is frame-based and uses the normalised cross correlation to generate a F_0 estimate which is then refined by dynamic programming. Another popular approach used in the past was the YIN [117] algorithm which uses a normalised difference function based on the autocorrelation function as well as a number of optimisation steps to produce the F_0 estimate. More recently in [118] a fast algorithm which considerably reduces the computational complexity of a non-linear least squares (NLS) estimator was proposed. In the work described in this thesis, the PEFAC algorithm is exploited due to it being able to simultaneously estimate the F_0 reliably (even at negative signal-to-noise ratio (SNR) values as it is able to reject additive noise that has a smoothly varying power spectrum) as well as being able to calculate the probability that a frame is voiced. It will be shown in Section 3.2.2 that this voiced speech detection is utilised advantageously in the F_0 trajectory estimation.

3.1.2 Tracking Fundamental Frequency

It is important to first study whether the F_0 of voiced speech is a good indicator of speaker changes in multi-speaker meeting audio. Fig. 3.2 on Page 25 is generated by first running PEFAC on the headset microphone recordings taken from AMI [103]. Then, the Kalman filtering method that will be described in Section 3.2 was applied to the result to generate smooth F_0 estimates. The measurements obtained are the best ground-truth available of the individual speaker F_0 tracks. It is clear to see from Fig. 3.2(a) that the four individuals in AMI meeting 'ES2004b' speak at a very different F_0. However, AMI meeting 'TS3003b', in Fig. 3.2(b), highlights that some individuals speak at a very similar F_0. This is most likely due to the fact that in this particular meeting all the speakers are male. The dotted lines show the mean F_0 of each speaker in AMI where the first letter of the speaker label relates to the gender of the speaker i.e. M: male and F: female. It can also be observed in both figures that the average variation in the F_0 is very similar for most speakers [119]. This result demonstrates that the mean of the F_0 considered in isolation does not contain enough information to identify the speaker.



(a) Kalman filter F_0 tracks where 'ES2004b' is the input.



(b) Kalman filter F_0 tracks where 'TS3003b' is the input.

Figure 3.2: The individual F_0 tracks generated from a pitch estimation filter with amplitude compression (PEFAC) by using the Kalman filter on the individual headset microphones separately. The dashed horizontal lines represent the mean of each speaker. The AMI speaker labels are also given in brackets where the first letter relates to the gender of the speaker, i.e. M: male and F: female.

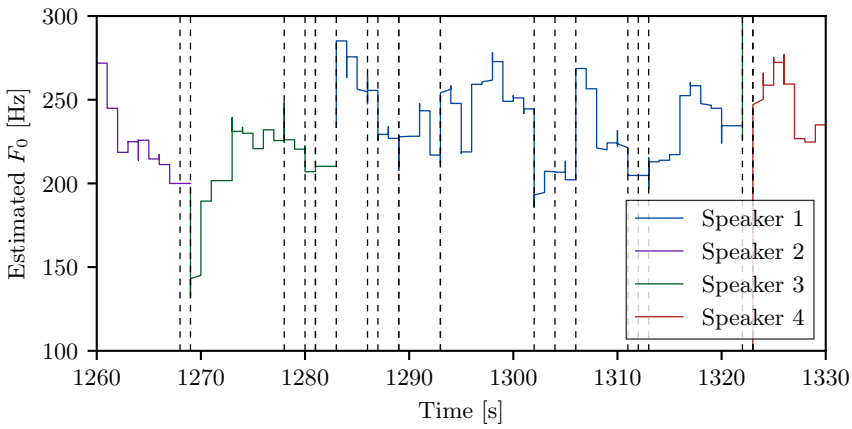


Figure 3.3: Estimated speaker F_0 tracks (solid lines) from 'IS1009b' along with the actual speaker changes (dotted lines).

Meeting	SC | PC
ES2004a	94.49%
ES2004b	89.25%
ES2004c	95.21%
ES2004d	91.85%
IS1009a	96.12%
IS1009b	98.94%
IS1009c	97.67%
IS1009d	98.55%
EN2002a	92.35%
EN2002b	87.01%
EN2002c	79.37%
EN2002d	86.00%
TS3003a	76.54%
TS3003b	76.59%
TS3003c	75.82%
TS3003d	81.34%
Mean	88.57%

Meeting	PC | SC
ES2004a	78.76%
ES2004b	68.60%
ES2004c	70.22%
ES2004d	73.38%
IS1009a	68.91%
IS1009b	64.27%
IS1009c	59.38%
IS1009d	66.60%
EN2002a	88.59%
EN2002b	83.40%
EN2002c	87.70%
EN2002d	81.02%
TS3003a	52.08%
TS3003b	48.46%
TS3003c	56.47%
TS3003d	62.68%
Mean	69.41%

PC | SC The probability that there is a 'F_0 change' given that there is a 'speaker change'

SC | PC The probability that there is a 'speaker change' given that there is a 'F_0 change'

Table 3.1: Speaker and F_0 change analysis for the AMI corpus.

3.1.3 Temporal Variations in Fundamental Frequency

It has been seen in Section 3.1.2 that some speakers do indeed have a very similar mean F_0 for their voice and, therefore, this section will show that even under these conditions, it is still possible to identify when there is a change in the speaker using information about the way in which F_0 varies over time.

It has been previously shown that the F_0 of an individual speaker only varies in a smooth manner due to physiological constraints [120]. Accordingly, it is possible to predict the future F_0 of the speaker based on their current and previous F_0. Thus, if the F_0 cannot be predicted, then this could be an indication that there has been a change in speaker. In this chapter, this prediction is attained by means of a Kalman filter which is described in detail in Section 3.2.

Table 3.1 on Page 26 is generated using the headset microphone recordings taken from AMI and shows the probability that there is a speaker change given that there is a F_0 change and vice-versa. These results demonstrate that if there is a change in speaker, then there is a very high probability that there will be a change in the F_0. Thus, Table 3.1 illustrates that the detection of F_0 changes can be exploited constructively for speaker change detection, however, there can still be a speaker change without a change in the F_0.

Fig. 3.3 on Page 25 provides a visualization example for the results shown in Table 3.1. This plot is generated using the method from Section 3.2 with a single distance microphone (SDM) recording from AMI. It highlights that in this particular meeting, 'IS1009b', when a F_0 change occurs, it always coincides with a change in the speaker. The plot also shows that many speaker changes go undetected.

3.2 Proposed Fundamental Frequency Segmentation Method

A method is now presented that utilises the time-varying properties of the F_0 to detect speaker changes within a multi-speaker scenario. A block diagram of the proposed system is provided in Fig. 3.1.

3.2.1 Fundamental Frequency Estimation

The first step of the proposed method is the F_0 estimator; for this work the PEFAC [115, 121] algorithm was chosen (see Section 3.1.1 on Page 24).

3.2.2 Kalman Filter

The next step is to use a Kalman filter [94] to estimate the F_0 trajectories from PEFAC. F_0, denoted here for reasons of notational consistency as x_t, for time frame t, is modelled here as a random walk with zero-mean, normally distributed increments such that

$$x_t = x_{t-1} + w, \quad w \in \mathcal{N}(0, \sigma_w^2) \ , \tag{3.1}$$

where the F_0, at t deviates from the F_0 at $t-1$ by a process noise term, w, with a variance of σ_w^2. The PEFAC observations, z_t, are modelled as

$$z_t = x_t + v, \quad v \in \mathcal{N}(0, \sigma_v^2) \ , \tag{3.2}$$

where the observation noise, v, in this case, models the errors in the F_0 estimates from PEFAC which is assumed to be an unbiased estimator. The Kalman filter estimates the state of the system and then acquires feedback from noisy observations using a prediction step and an update step. The predicted F_0 estimate, $\hat{x}_{t|t-1}$, and predicted estimate variance, $P_{t|t-1}$, are given by [94, 98]

$$\hat{x}_{t|t-1} = \hat{x}_{t-1|t-1} \ , \tag{3.3}$$

$$P_{t|t-1} = P_{t-1|t-1} + \sigma_w^2 \ . \tag{3.4}$$

The updated F_0 estimate, $\hat{x}_{t|t}$, and updated estimate variance, $P_{t|t}$, are given by

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + K_t(z_t - \hat{x}_{t|t-1}) \ , \tag{3.5}$$

$$P_{t|t} = (1 - K_t)^2 P_{t|t-1} + K_t^2 \sigma_v^2 \ , \tag{3.6}$$

where the innovation variance, S_t, and optimal Kalman gain, K_t, are given by

$$S_t = P_{t|t-1} + \sigma_v^2 \ , \tag{3.7}$$

$$K_t = \frac{P_{t|t-1}}{S_t} \ . \tag{3.8}$$

Thus, the error between observation and prediction follows as

$$\tilde{y}_{t|t} = z_t - \hat{x}_{t|t} \ . \tag{3.9}$$

For the proposed method, two useful outputs from PEFAC are utilised: the F_0 estimate of each frame and the corresponding probability that the frame is voiced.

The prediction step is carried out on every frame, however, the update step is only performed if the frame is voiced. This is considered to be the case if the probability that the frame is voiced is above a threshold ξ. Thus, if an unvoiced frame is observed then the F_0 remains constant in (3.3) with the predicted estimate variance being increased in (3.4). This outcome is desirable as it makes the prediction less reliable as time goes on without a voiced frame being observed. An illustrative example is given in Fig. 3.4 on Page 29 which shows how the variance, $P_{t|t}$, changes over time depending on whether voiced speech is observed.

Given that Kalman gain, K_t, trades-off the measured F_0 against the predicted F_0 for the frame, K_t increases as the time between the update steps increases. This means that as the time elapsed since the last update frame increases, the result will be more influenced by the observation, otherwise, it will be more influenced by the prediction from the model. This is seen in (3.5) if $K_t = 1$ then $\hat{x}_{t|t} = z_t$ (only the observation) else if $K_t = 0$ then $\hat{x}_{t|t} = \hat{x}_{t|t-1}$ (only the prediction).

3.2.3 Speaker Change Detection

The proposed approach for speaker change detection utilises the error between the observation and the prediction in (3.9). If the error is above a threshold, ϕ, then that implies that the error is large and the

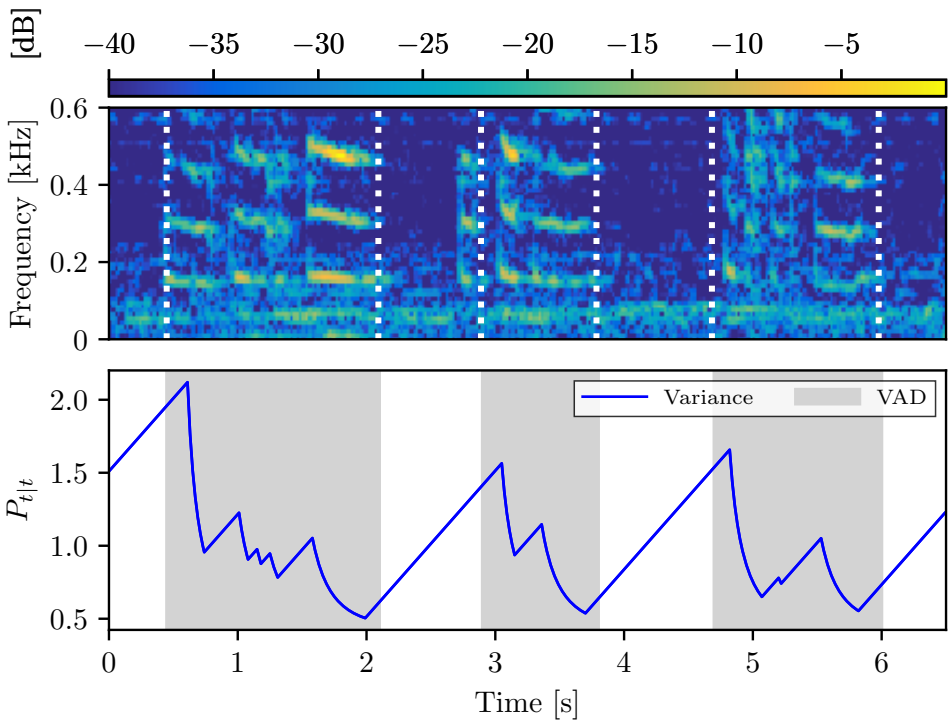


Figure 3.4: Illustrative example of how the variance, $P_{t|t}$, changes over time depending on whether voiced speech is observed.

F_0 could not be predicted. A threshold, ϕ, found through experimentation is acceptable in this case, as the F_0 of different speakers can be easily anticipated. Therefore, this change detection approach works by attributing a large prediction error to a change in the speaker.

A Kalman filter is initialised and tracks the first speaker. Subsequently, when $\tilde{y}_{t|t}$ in (3.9) exceeds a threshold of ϕ, a new Kalman filter is initialised to track the second speaker. On detection of the next speaker change, the observation of the F_0 is compared with all previously generated Kalman filter F_0 tracks to find the track closest to the current observation of the F_0. If the difference between the current observation and the last F_0 value of the closest Kalman F_0 track is below a threshold of ρ, the previous Kalman filter is continued. If, on the other hand, the closest Kalman filter to the observation does not satisfy this threshold, a new Kalman filter would be generated.

The reasoning behind this Kalman filter birthing approach is that if the speakers do indeed have a different mean F_0, e.g. AMI meeting 'ES2004b' shown in Fig. 3.2(a), then different Kalman filter F_0 tracks should correspond to the different speakers in the audio recording of the meeting.

Meeting	Proposed F_0 Segmentation					MFCC Segmentation (S4D)				
	HIT	MISS	MH	MSE	FA	HIT	MISS	MH	MSE	FA
EN2002a	72.82%	27.18%	9.06%	0.0520	48.8%	45.30%	54.70%	8.71%	0.0349	49.0%
EN2002b	76.24%	23.75%	10.21%	0.0606	55.1%	48.22%	51.78%	9.74%	0.0393	58.0%
EN2002c	77.13%	22.88%	8.67%	0.0530	54.6%	50.43%	49.57%	9.01%	0.0330	52.4%
EN2002d	64.86%	35.14%	5.74%	0.0480	51.9%	40.54%	59.46%	7.43%	0.0333	52.2%
ES2004a	84.80%	15.20%	12.00%	0.0334	67.4%	64.00%	36.00%	13.60%	0.0287	67.2%
ES2004b	84.81%	15.20%	9.92%	0.0431	69.6%	60.60%	39.39%	13.85%	0.0337	65.7%
ES2004c	72.73%	27.27%	8.08%	0.0409	68.3%	50.00%	50.00%	7.58%	0.0404	54.5%
ES2004d	76.40%	23.61%	7.30%	0.0379	60.1%	58.80%	41.20%	12.02%	0.0270	50.0%
IS1009a	72.10%	27.91%	6.98%	0.0442	68.4%	37.21%	62.79%	2.33%	0.0487	69.6%
IS1009b	77.48%	22.53%	4.95%	0.0543	74.6%	47.80%	52.20%	10.99%	0.0219	68.1%
IS1009c	83.33%	16.67%	8.02%	0.0452	72.0%	46.29%	53.70%	8.02%	0.0280	66.1%
IS1009d	75.42%	24.58%	13.56%	0.0558	68.1%	49.15%	50.85%	12.71%	0.0335	62.8%
TS3003a	85.71%	14.29%	9.52%	0.0247	71.2%	71.43%	28.57%	33.33%	0.0237	67.9%
TS3003b	97.24%	2.76%	10.14%	0.0362	70.1%	89.41%	10.60%	30.88%	0.0249	62.8%
TS3003c	95.08%	4.92%	18.56%	0.0406	72.8%	88.63%	11.36%	30.68%	0.0290	70.4%
TS3003d	89.92%	10.08%	15.97%	0.0404	61.4%	80.25%	19.75%	23.95%	0.0317	59.0%
Std Dev	8.55%	8.55%	3.50%	0.0090	7.9%	15.97%	15.97%	9.26%	0.0066	7.2%
Mean	80.38%	19.62%	9.92%	0.0444	64.7 %	58.00%	42.00%	14.68%	0.0320	61.0%

Table 3.2: Performance of both the proposed system and S4D on multi-speaker meetings in the AMI corpus. (*A graphical representation of these results is also given in Fig. 3.5 on Page 31.*)

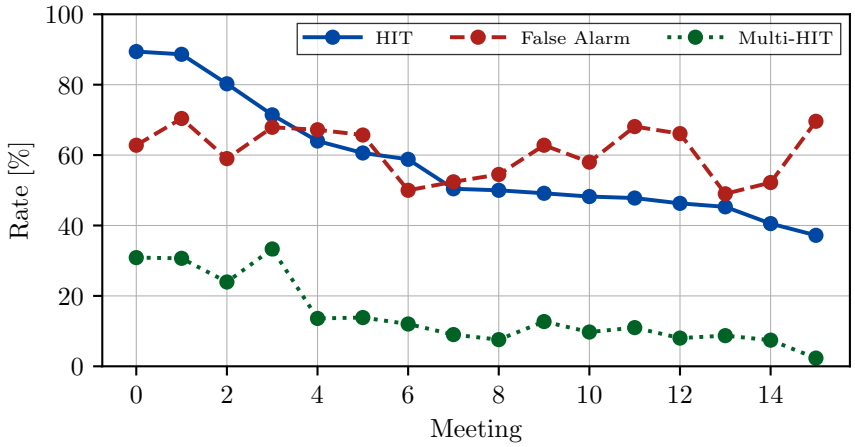
3.2.4 Voice Activity Detection

To generate the final segmentation, the detected speaker changes from the F_0 are merged with the results from voice activity detection (VAD) [122]. As part of a pre-processing step, the VAD output detects active speech regions and if these regions have small pauses between them then they are merged together. Subsequently, both the onsets of speech detected by the VAD and the speaker changes detected by the F_0 are concatenated. If a VAD onset and a detected speaker change are within threshold ζ of each other, then only the detected speaker change is included in the segmentation file.

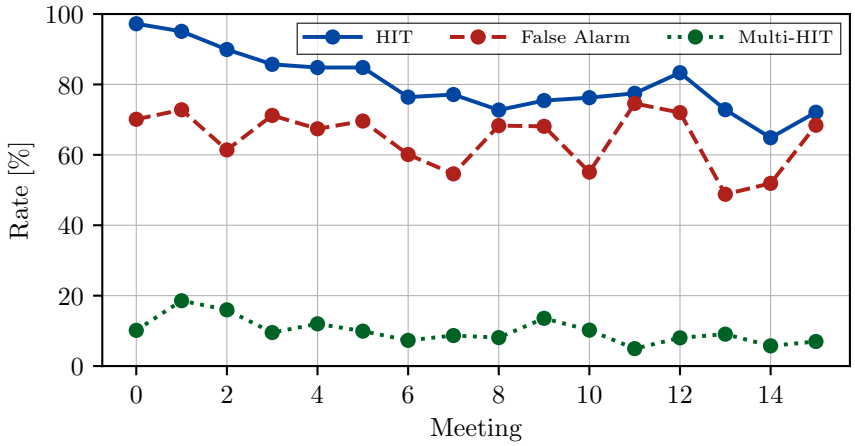
3.3 Comparative Evaluation

In order to evaluate the performance of this newly proposed system shown in Fig. 3.1 on Page 23, it is compared against a typical segmentation system S4D [105] illustrated in Fig. 2.4(a) on Page 20. The code for both the proposed method and the S4D baseline can be found here [123].

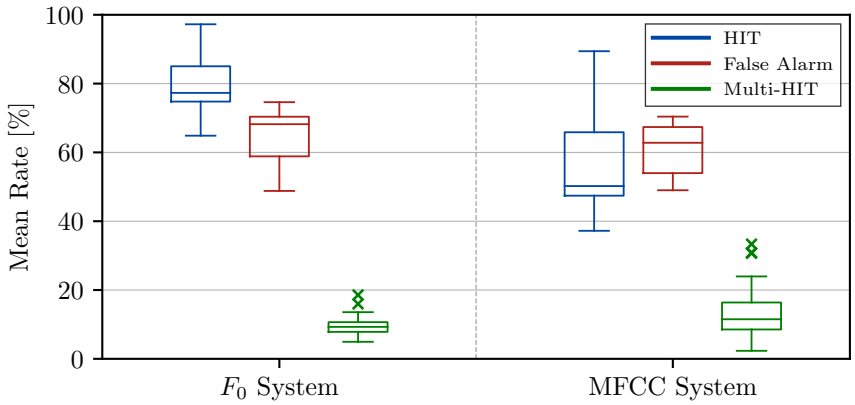
The accuracy and the reliability of speaker change detection are compared for both the proposed system and S4D with the results shown in Table 3.2. The HIT rate (HR) is defined as the number of speaker changes that are detected by a single detection. In contrast, the miss rate is given by the number of speaker changes that go undetected and the multi-hit (MH) rate is specified as the number of speaker changes that are detected multiple times. When evaluating segmentation performance, it is common practice to apply a time collar around every ground-truth speaker change in order to account for possible



(a) Benchmark system evaluation.



(b) Proposed system evaluation.



(c) Comparison of the mean rates overall meetings.

Figure 3.5: Comparative evaluation of the two systems.
(*A tabular form of these results is also given in Table 3.2 on Page 30.*)

ξ	w	v	ζ	ϕ	ρ
95%	20	0.01	5 ms	10 Hz	50 Hz

Table 3.3: Parameter Setting.

inaccuracies. The results in Table 3.2, therefore, incorporate a collar of 50 ms applied to each ground-truth speaker change.

The parameters selected for the proposed system are given in Table 3.3 where initialisation uses both physiological constraints of w: process noise, v: observation noise and empirical tuning of ξ: voiced frame threshold, ζ: VAD region merge threshold, ϕ: speaker change threshold, ρ: continue previous speaker track threshold. This empirical tuning was achieved through an exhaustive grid search [124] on the development set of the AMI corpus.

It can be seen in Table 3.2 that the percentage of speaker changes that are detected increases from 58.0% for S4D to 80.4% for the proposed system. Thus, the proposed F_0 system is far more likely to detect a speaker change within the given 50 ms collar. It is important to note for both systems that increasing the collar decreases the miss rate, increases the MH rate and does not change the HR.

The mean squared error (MSE) in time was also calculated in Table 3.2 for all the HITs and the closest MH detections to the oracle speaker changes, against the ground-truth given by the label files from AMI. The results show that when a speaker change is detected by both systems, the use of MFCCs in SIDEKIT gives slightly more accurate temporal segmentation (MSE = 32 ms) compared to the use of F_0 (MSE = 44 ms).

To realise the significance of this improvement, the whole diarization process should be considered. In a typical diarization system after the segmentation process, clustering is performed and then Viterbi alignment is exploited as previously reported in [125]. Consequently, mediocre performance in the segmentation system is tolerated. However, if the clustering algorithm is given a better segmentation, where almost all segments just contain one speaker, then it will achieve a far better clustering result; improving the performance of the given diarization system which is highly desirable. This is verified in [126] where an evaluation is undertaken which shows that improving the segmentation performance leads to better diarization accuracy and a lower diarization error rate.

3.4 Conclusion

A study of meetings in the AMI corpus has shown that a F_0 change is a strong indicator of a speaker change. This finding motivates the use of F_0 as a feature - possibly combined with other features - in speaker segmentation as used, for example, in the first step of speaker diarization. It was also verified that the F_0 from an individual speaker is smoothly varying and can be predicted by a Kalman filter. Therefore, in this chapter, a Kalman filtering approach was proposed to identify speaker change boundaries based on a model of the temporal variation of the F_0.

The proposed Kalman filter prediction error-based approach performed well when compared against a previous MFCC-based method. An evaluation on the AMI corpus showed a speaker change detection increase from 58.0% to 80.4%. This work was also published in the following paper [5].

Chapter 4

OVERLAPPING SPEAKER SEGMENTATION USING FUNDAMENTAL FREQUENCY

4.1 Introduction

This chapter focuses on the segmentation task which determines the time instant when a new speaker starts talking and when the current speaker stops. Specifically, a novel approach to the segmentation of overlapping speech, i.e. the onset of a new speaker before the end-point of an active speaker, is proposed. Overlapping speech was shown, e.g. in [26], to lead to severe performance degradation of diarization systems.

Existing approaches to the segmentation of overlapping speech include [85] where a hidden Markov model (HMM)-based method, that used MFCC, linear predictive coding (LPC) and root mean square (RMS) energy features, was proposed. It has also been shown in [86] that the segmentation performance can be improved if long-term conversational features are utilised. Other methods include multimodal techniques that use multiple microphone and camera systems [87]. More recently, machine learning methods have also been put forward that rely on bidirectional long short term memory networks (BLSTMs) [57].

In the past, methods have been proposed that use the F_0 to improve the segmentation process as F_0 information is a fast and effective discriminator between male and female speakers. However, the F_0 is not often used as the sole feature, e.g. in [93] the F_0 is fused with MFCC and LPC coefficients and a divergence distance threshold is used to detect speaker change boundaries.

There have also been methods developed for real-time segmentation which use the F_0 as the sole feature [110, 111] where the delta in F_0 over time is compared with a self-adaptable threshold, therefore, [110, 111] make no attempt to model the F_0. In other methods such as, [127] the jitter, which captures variations in

the F_0 of the speaker's voice, is exploited for agglomerative hierarchical clustering (AHC). Similarly, the fundamental frequency variation (FFV) spectrum [128, 129] has also been used in the past as an input feature for AHC approaches, however, none of [127–129] track temporal variations over time. Methods have been proposed that attempt to track F_0, e.g. [130], however, these have been applied to the task of speaker recognition and have never been utilised for speaker segmentation.

In contrast to these previous methods, the proposed method, presented in this chapter, attempts to model the F_0 over time, unlike [127–129]. This modelling approach has two major advantages 1) the model can be exploited to remove errors in the F_0 estimates. 2) the errors in the F_0 prediction given by the model can be utilised to detect speaker changes instead of using the delta F_0, i.e. the difference in F_0 between two frames [110, 111].

The proposed method also relies solely on the F_0 of voiced speech unlike [93]. The proposed method operates by taking advantage of the harmonic nature of voiced speech to determine the periods of time when more than one speaker is active. It is well known that voiced speech contains strong harmonic components, e.g. [131]. The harmonic characteristic has been exploited by many F_0 estimators in order to produce reliable F_0 estimates [114, 115]. These systems, however, assume that only one speaker is active at any given time. Multi-F_0 tracking methods that estimate the F_0 of multiple periodic signals have been proposed in [132–137] but these methods have never been exploited for the task of overlapping speaker segmentation which will be explored in this chapter.

To perform this multi-F_0 tracking a multiple hypothesis tracking (MHT) [96] approach is used. In the past, MHT approaches have been used for radar target tracking [98] and more recently, MHT methods have been shown to work well for visual tracking [97, 138, 139]. As a result, it has been used effectively for the task of robot audition [140, 141]. This work, however, is the first time an MHT method has been proposed for the task of speaker segmentation.

It is shown that the proposed method outperforms a BLSTM approach [41], in terms of HR, by 12.9% on the AMI corpus SDM stream. It is also shown that the F_0 estimates produced by the proposed system can be used as input features, in addition to MFCC features, for neural networks improving the segmentation performance of the baseline BLSTM by 1.21% in terms of coverage and 2.45% in terms of purity. This work was also published in the following papers [3, 4, 142].

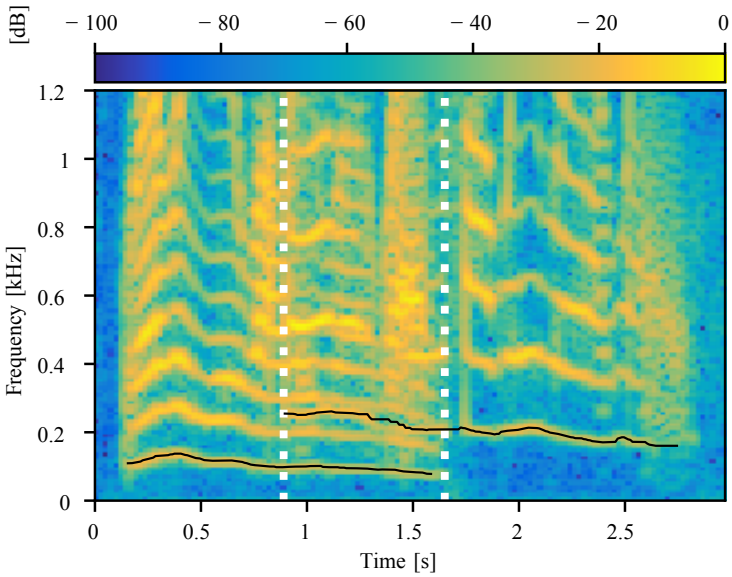


Figure 4.1: F_0 tracks of a male and female speaker from the TIMIT corpus in black. The two speakers overlap between the two white dashed vertical lines.

4.1.1 Voiced and Unvoiced Speech

In using multi-F_0 tracking to address overlapping speaker segmentation, one clear difficulty arises; that of processing both voiced and unvoiced speech [143]. The intervals of unvoiced speech in a recording do not possess an F_0 or any significant harmonic characteristics [144]. Solutions to this problem have been proposed in the past [145] when F_0 features have been used for diarization. This chapter novely intends to deal with the problem of unvoiced regions of speech by using a tracking approach. This allows for the spanning of short unvoiced intervals by continuing tracks for short periods even when no observations are detected. This does not, however, solve the problem caused by the presence of unvoiced speech either at the onset or end-point of a given speaker which will have the effect of delaying or shortening a F_0 track. In this work, these errors will be safely ignored as they will be less than 50 ms [146] which is smaller than most collars used to account for human annotation imprecision [147]. The temporal aspect makes the approach different from [62] which utilises the FFV spectrum for speaker identification [128, 129].

4.1.2 Harmonic Structure of Fundamental Frequency

To extend the study in Section 3.1.3, an analysis of overlapping speech segments is presented. F_0 estimation is not feasible during overlapping speech using single speaker F_0 estimation algorithms [26]. Therefore, this chapter proposes to exploit the harmonic structure of speech for F_0 estimation from overlapping speech signals. Fig. 4.1 shows two speakers from the TIMIT corpus [104] that have been overlapped in

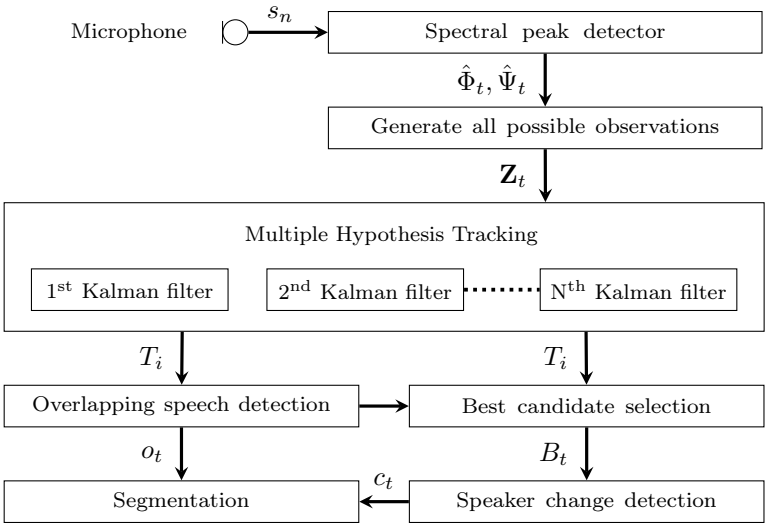


Figure 4.2: Proposed System-1 architecture with s_n: input signal, $\hat{\Phi}_t$: peak detections, $\hat{\Psi}_t$: detection reliabilities, $\mathbf{Z}_t$: generated observations, T_i: selected track hypotheses, o_t: overlapping speech onsets, B_t: strongest candidate track and c_t: speaker change onsets.

the interval between [0.91, 1.58] s; the spectrogram is the resulting mixture with the F_0 tracks, obtained from the individual recordings, being overlaid in black. This illustrative example suggests that, if multiple F_0s can be tracked reliably, then overlapping speakers can be segmented. Fig. 4.1 also shows that the signals from different speakers differ not only in terms of the mean F_0 values of each track but also in the temporal shape of their trajectories. Overlapping speakers may, therefore, be segmented by identifying the onsets and end-points of each speaker's utterance.

4.2 System Model and Method

In this chapter two novel systems are presented shown in Fig. 4.2 and Fig. 4.6.

4.2.1 Proposed System-1 Architecture

Proposed System-1 architecture, shown in Fig. 4.2, is as follows: (i) Estimates of the harmonic frequencies are first obtained using a spectral peak detector (see Section 4.2.1.1). (ii) Subsets (see Section 4.2.1.2) of the detected peaks are used to track the voiced F_0 of multiple, overlapping speakers using a Kalman filter (see Section 4.2.1.3). However, the association of subsets of peak detections with the F_0 of a specific speaker is unknown *a priori*. This problem is exacerbated by some subsets being related to false alarm (FA)[1].

[1] A FA in this context is when an observation/subset does not relate to the F_0 of a speaker.

(iii) A MHT [96, 98, 139] method (see Section 4.2.1.4) that probabilistically updates the F_0 tracks using physically feasible subsets of peak detections is, therefore, proposed. (iv) To reduce the number of track hypotheses and, therefore, the computational complexity, a maximum weighted clique (MWC) method is also deployed (see Section 4.2.1.4). (v) Segments of overlapping speech are identified as the onsets and end-points of multiple, uncorrelated F_0 tracks (see Section 4.2.1.5). (vi) The complete segmentation (see Section 4.2.1.7) is obtained from the union of overlapping speech onsets with the speaker changes detected (see Section 4.2.1.6) based on a model of the temporal variation of the F_0 [5].

4.2.1.1 Spectral peak detector

The harmonics of voiced speech are estimated using a spectral peak detector that generates a set of P_t peaks, $\Phi_t = \{\phi_{t,1}, \cdots, \phi_{t,P_t}\}$ in the short-time Fourier transform (STFT) of the microphone signal at every time frame, t. Each peak, $\phi_{t,p}$, for $p \in \{1, \cdots, P_t\}$ is associated with a peak amplitude, $\psi_{t,p}$, which captures the detection reliability. A peak, $\phi_{t,p}$, is deemed to be reliable if $\psi_{t,p}$ is greater than a threshold, ξ, where ξ can be determined experimentally. Only the reliable peak detections, $\hat{\Phi}_t = \{\hat{\phi}_{t,1}, \cdots, \hat{\phi}_{t,Q_t}\}$, where $Q_t \leq P_t$, are retained for each time frame, t. The cardinality of $\hat{\Phi}_t$ is, therefore, less than or equal to the cardinality of Φ_t and varies over time.

4.2.1.2 Generate all possible observations

Due to the harmonic nature of voiced speech, subsets of elements in $\hat{\Phi}_t$ may correspond to integer multiples of the F_0 of a speaker. For brevity, the remainder of this chapter refers to peak detections corresponding to integer multiples of F_0 as 'harmonically related' detections. For multiple speakers, the subsets of elements in $\hat{\Phi}_t$ corresponding to harmonically related detections are unknown *a priori*. The association between peaks and F_0 of a speaker needs to be resolved in order to track the F_0 of multiple speakers simultaneously. This is further complicated by the fact that the F_0 of two speakers may correspond to integer multiples of each other. Therefore, the association between peak detections and the F_0 of each speaker may be ambiguous in some time frames. The task is particularly problematic if the audio signal contains reverberation and noise as they corrupt the spectrogram structure of the speech signal.

To address the association problem, a probabilistic perspective is adopted. To determine the unknown subsets of harmonically related detections, all possible subsets of $\hat{\Phi}_t$, corresponding to integer multiples of F_0, within a tolerance of F_{tol}, are computed. Each resulting subset is denoted as an 'observation', $\mathbf{z}_{t,n}$. The generation of these subsets is of a combinatorial nature. To reduce the computational complexity of the problem, an F_0 estimate is only considered if it satisfies $F_{\min} \leq F_0 \leq F_{\max}$, which defines the physical

Algorithm 1 Generation of observations from F_0 measurements.

```
for (φ, ψ) in (Φ, Ψ) do                          ▷ inputs: Φ and Ψ; output: Z_t
    if ψ > ξ then                                ▷ remove unreliable measurements
        Φ̂.append(φ)
Z = {}                                           ▷ all observations, Z_t, for a given time frame t
for φ̂ in Φ̂ do
    n = 1;  F_0 = φ̂                              ▷ initialisation
    while F_min < F_0 < F_max do                 ▷ ignore F_0 if unrealistic
        F_0 = φ̂/n                                ▷ needed if F_0 not contained in Φ̂
        n = n + 1
        z = {}                                   ▷ possible subset, z_{t,n}, for a given time frame t
        for φ̂ in Φ̂ do
            if |(round(φ̂/F_0) * F_0) − φ̂| < F_tol then   ▷ ±F_tol
                z.append(φ̂)
        if z not in Z and length(z) > 1 then
            Z.append(z)                          ▷ remove z_{t,n} containing 1 harmonic
```

range of the human speech production system.

Consider the following illustrative example for a given frame: $P_t = 5$, $\Phi_t = \{100, 200, 350, 400, 450\}$ and $\Psi_t = \{6.3 \times 10^7, 4.5 \times 10^7, 4.9 \times 10^6, 2.3 \times 10^6, 8.2 \times 10^4\}$ where $\Psi_t = \{\psi_{t,1}, \cdots, \psi_{t,P_t}\}$ and $\Phi_t = \{\phi_{t,1}, \cdots, \phi_{t,P_t}\}$. If $\xi = 1 \times 10^6$ then $\hat{\Phi}_t = \{100, 200, 350, 400\}$. Then there are three possible associations that are feasible if $F_{\min} = 50$ Hz and $F_{\max} = 300$ Hz. In the first case, the observations could be interpreted such that F_0 is 50 Hz and $\mathbf{z}_{t,0} = [100, 200, 350, 400]^T$. In the second case, the observations could be interpreted such that F_0 is 100 Hz and $\mathbf{z}_{t,0} = [100, 200, 400]^T$. In the third and final case, the observations could be interpreted such that F_0 is 200 Hz and $\mathbf{z}_{t,1} = [200, 400]^T$. Algorithm 1 outlines the generation of the observations from the peak detections.

4.2.1.3 Kalman filter for F_0 tracking

The F_0 of active speakers is tracked by multiple Kalman filters [94] at each time frame, t. The input observations, $\hat{\mathbf{z}}_{t,n}$, contain subsets relating to possible harmonically related detections. The Kalman filters track all possible F_0 trajectories, $\mathbf{x}_t = \{x_{1,t}, \ldots, x_{I,t}\}$, where $x_{i,t}$ corresponds to the F_0 of the ith speaker, $i \in \{1, \ldots, I\}$. The F_0 trajectory, denoted $x_{i,t}$, for the ith speaker at time frame, t, is modelled as

$$x_{i,t} = x_{i,t-1} + w_{i,t}, \quad w_{i,t} \in \mathcal{N}(0, \sigma_w^2) \ , \tag{4.1}$$

where the F_0 at t deviates from the F_0 at $t-1$ by a process noise term, $w_{i,t}$, with a variance of σ_w^2. The observations, $\hat{\mathbf{z}}_{t,n}$, associated with speaker, i, are modelled conditionally on $x_{i,t}$ as

$$\begin{aligned} \hat{\mathbf{z}}_{t,n} = \mathbf{h}_{t,n} x_{i,t} + \mathbf{v}_t, \quad \mathbf{v}_t \in \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \ , \\ \mathbf{R}_t = \mathrm{diag}(\sigma_v^2, \cdots, \sigma_v^2) \ , \end{aligned} \tag{4.2}$$

where the covariance, $\mathbf{R}_t \in \mathbb{R}^{N_t \times N_t}$, and variance, σ_v^2, model the uncertainty in the observations, and $\mathbf{h}_{t,n}$ is a column vector with elements containing integer multiples of F_0 that maps the current state to the harmonic components contained in the current observation. For example, if $\hat{\mathbf{z}}_{t,0} = [100, 200, 400]^T$ (as in the example of Section 4.2.1.2) where F_0 is 100 Hz then $\mathbf{h}_{t,0} = [1, 2, 4]^T$. The Kalman filter operates by estimating the state of the system and then acquiring feedback from the noisy measurements using a prediction step and an update step. The predicted F_0 estimate, $\hat{x}_{i,t|t-1}$, and predicted estimation variance, $p_{i,t|t-1}$, are given by

$$\hat{x}_{i,t|t-1} = \hat{x}_{i,t-1|t-1} \ , \tag{4.3}$$

$$p_{i,t|t-1} = p_{i,t-1|t-1} + \sigma_w^2 \ . \tag{4.4}$$

The updated F_0 estimate, $\hat{x}_{i,t|t}$, and updated estimation variance, $p_{i,t|t}$, are given by

$$\hat{x}_{i,t|t} = \hat{x}_{i,t|t-1} + \mathbf{k}_{i,t}(\hat{\mathbf{z}}_{t,n} - \mathbf{h}_{t,n}\hat{x}_{i,t|t-1}) \ , \tag{4.5}$$

$$p_{i,t|t} = (1 - \mathbf{k}_{i,t}\mathbf{h}_{t,n})^2 \, p_{i,t|t-1} + \mathbf{k}_{i,t}\mathbf{R}_t\mathbf{k}_{i,t}^T \ . \tag{4.6}$$

The optimal Kalman gain, $\mathbf{k}_{i,t}$, is a row vector given by [94, 98]

$$\mathbf{k}_{i,t} = p_{i,t|t-1}\mathbf{h}_{t,n}^T\mathbf{S}_{i,t}^{-1} \ , \tag{4.7}$$

where innovation variance, $\mathbf{S}_{i,t}$, is a matrix given by

$$\mathbf{S}_{i,t} = \mathbf{h}_{t,n} p_{i,t|t-1}\mathbf{h}_{t,n}^T + \mathbf{R}_t \ . \tag{4.8}$$

Therefore, the error between measurement and prediction follows as

$$\mathbf{e}_{i,t|t} = \hat{\mathbf{z}}_{t,n} - \mathbf{h}_{t,n}\hat{x}_{i,t|t} \ . \tag{4.9}$$

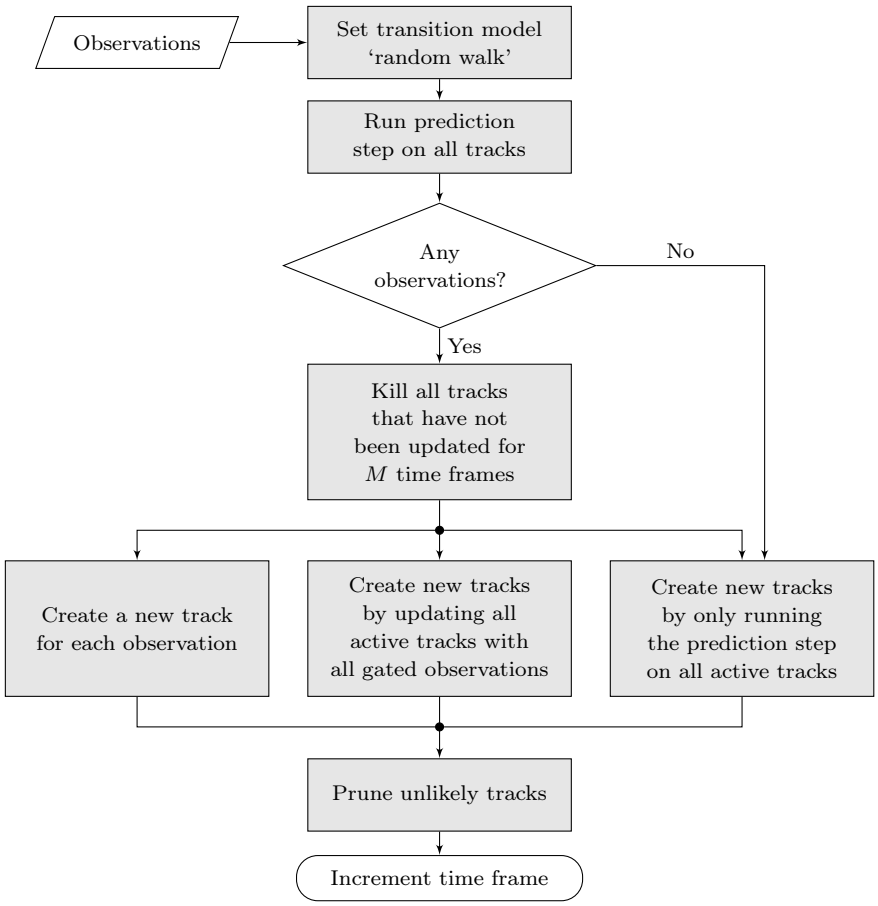
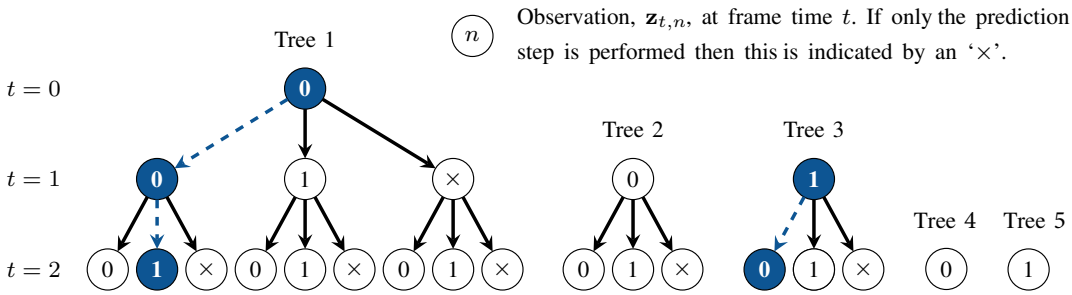


Figure 4.3: Multiple hypothesis tracking (MHT) procedure at each time frame.

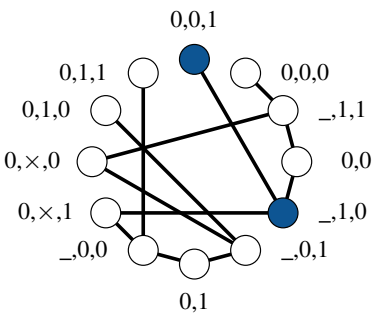
4.2.1.4 Multiple hypothesis tracking

The flowchart in Fig. 4.3 on Page 41 demonstrates the MHT process at each time frame. At $t = 0$, all observations, $\hat{\mathbf{Z}}_0$, are used to generate N_0 new active Kalman filter tracks. For $t > 0$, each observation, $\hat{\mathbf{z}}_{t,n}$, could be interpreted as one of three alternatives: (i) a FA; (ii) the start of a new track or (iii) related to a currently active track.

To resolve this uncertainty, all possibilities are expanded and MHT is utilised [97]. Fig. 4.4(b) on Page 42 shows how new tracks are generated from active tracks and observations. In order to reduce the computational complexity of the problem, gating [98] is also applied to each observation when it is interpreted as being related to an active track. This gating is required in order to stop observations with a low probability of belonging to the active track being used to update the track and instead generate a new track hypothesis. This gating is dependent on the error between the measurement and the prediction,



(a) Possible track hypotheses.



(b) Maximum weighted clique (MWC) in blue.

Figure 4.4: Multiple hypothesis tracking (MHT) illustration (a) Track hypotheses generated. (b) An undirected graph, G, where each node is a track hypothesis and each edge connects two tracks that are not conflicting. The nodes are indexed using the observations that make up each track.

$\mathbf{e}_{i,t|t}$, whenever an update is performed. Thus, gating is applied when a track is only updated by an observation if $\tilde{e}_{i,t|t}$ is below a threshold, ζ, otherwise the update is rejected. $\tilde{e}_{i,t|t}$ is defined as the mean of the absolute values of the estimation error $\mathbf{e}_{i,t|t}$ for time frame t. This is because if the observation is too far from the predicted estimate, it is considered unlikely to have originated from the active track.

Maximum weighted clique

The number of generated tracks grows exponentially as new observations become available, as illustrated in Fig. 4.4(a), for practicality, therefore, pruning, e.g. [98], is required to reduce exponential growth in the number of track hypotheses. Not all tracks can be valid as they may conflict with other tracks, e.g. when more than one track uses one or more of the same observations in their history. The MWC method [148, 149] is, therefore, used to find the most likely set of tracks that contain no conflicts. An undirected graph, $G = (V, E)$, is shown in Fig. 4.4(b) where each hypothesis track, T_i, is represented by the node-set $V = \{T_0, T_1, \cdots, T_L\}$, and the set of edges is $E \subseteq V \times V$, consisting of M edges. A clique is a subgraph of G with pairwise adjacent vertices, meaning that all pairwise vertices T_i and T_j are connected

by an edge (i, j). To find the MWC, each node is assigned a score, w_i, which is calculated by taking the average value of all previous estimation errors, $\mathbf{e}_{i,t|t}$, evaluated at each update. The MWC solution is the clique that maximises the following optimisation problem

$$\begin{aligned} \max\ f(q) &= \sum_{i=0}^{L} w_i q_i \ , \\ \text{s.t. } q_i + q_j &\leq 1, \forall (i,j) \in \bar{E} \ , \\ q_i &\in \{0,1\}, \text{ for } i \in \{0,1,\cdots L\} \ , \end{aligned} \tag{4.10}$$

where $q_i = 1$ if the node T_i belongs to the clique and $q_i = 0$ otherwise. In this case, $\bar{E}$ denotes the edge set of the complementary graph of G. The pruning technique then operates by calculating the MWC after k time frames and discarding all other tracks.

4.2.1.5 Overlapping speech detection

Any tracks remaining after pruning are used to determine onsets and end-points of overlapping speech by identifying uncorrelated tracks that indicate the activity of multiple, simultaneously active speakers. This is accomplished by first detecting any changes in the track cardinality. In the illustrative example in Fig. 4.5, the track cardinality changes from 3 to 5 at the time frame, o_t. Between changes in the track cardinality, the harmonic relation between tracks as well as the similarity of the estimated trajectories are compared in a two-stage process to confirm overlapping speech.

In the first stage, candidates of harmonically related tracks are identified as those tracks whose mean frequency corresponds, within a tolerance of $\pm F_{\text{tol}}$, to an integer multiple of any other tracks. Candidates that are not harmonically related indicate overlapping speech such as $\{100, 124, 197, 259, 299\}$ Hz in Fig. 4.5. For each set of harmonically related candidates, e.g. $\{102, 199, 303\}$ Hz in Fig. 4.5, the corresponding trajectories are compared by evaluating the MSE between all pairs in the set. Pairs corresponding to a MSE above a preset threshold, η, indicate a speaker overlap. The time of this overlapping speech onset, o_t, can also be considered as an onset of a new speaker prior to the previous speaker stopping.

The advantage of this method is that it is robust to the situation where one speaker generates multiple tracks since all the tracks generated by the same speaker will have the same trajectory and will also be harmonically related.

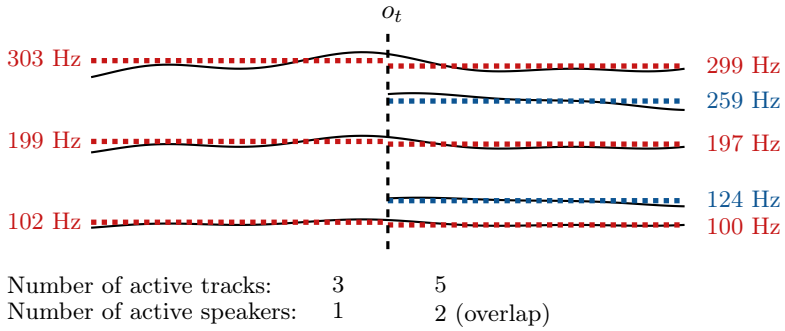


Figure 4.5: Overlapping speech detection.

4.2.1.6 Speaker change detection

The detection of overlapping speech onsets, o_t, provides the speaker changes relating to the onset of a new speaker before the end-point of a previous speaker. To obtain the full segmentation, it is also necessary to detect speaker changes, c_t, when the onset of a new speaker happens after the end-point of the previous speaker.

To accomplish this, the previous method presented in Chapter 3 and in [5], which operates using a single track, is further developed here. Speaker change detection is achieved by exploiting the temporal variations in the F_0. Whereas the previous version of the algorithm operates on only a single track, it has been shown above that multiple tracks can be generated for the same speaker. Therefore, in this work, the tracks are pruned to leave only the track that corresponds to F_0. To achieve this pruning, the $\hat{\psi}_{t,p}$ values related to all the measurements, $\hat{\phi}_{t,p}$, that correspond to a given track are summed for the periods where multiple tracks are active. These results are then used to select the strongest candidate, defined here as the track with the largest sum, B_t.

4.2.1.7 Proposed System-1 segmentation

The complete speaker segmentation is finally determined as the union of the sets of speaker changes, c_t, with the overlapping speech onsets, o_t.

4.2.2 Proposed System-2 Architecture

The number of possible F_0 tracks generated by the proposed System-1 (described in Section 4.2.1) grows exponentially, due to the MHT, as new observations become available. As a result, the computational complexity of the proposed System-1 is high, making it an inefficient implementation. A second system architecture (System-2) is, therefore, proposed which aims at lowering the computational complexity by

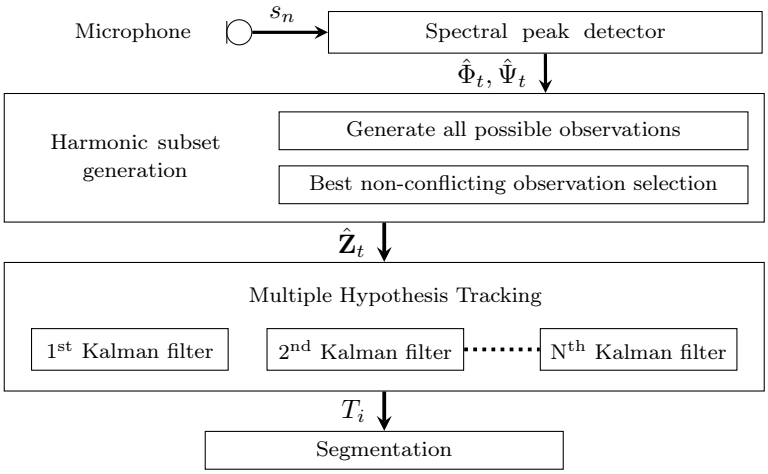


Figure 4.6: Proposed System-2 architecture with s_n: input signal, $\hat{\Phi}_t$: peak detections, $\hat{\Psi}_t$: detection reliabilities, $\hat{\mathbf{Z}}_t$: selected observations, T_i: selected track hypotheses.

the use of pre-processing to reduce the number of new observations at each frame. A computational complexity comparison of both systems is given and explored in detail in Section 4.3.1.2 on Page 47.

The proposed System-2 (shown in Fig. 4.6), therefore, shares many components with the proposed System-1 architecture; the only addition being that of a pre-processing step to select the best non-conflicting observations (which is described in Section 4.2.2.1). The proposed System-2 pre-processing also removes the need for any post-processing, used in System-1, such as overlapping and speaker change detection.

The complete System-2 architecture includes the following components: (i) Measurements of the harmonic frequencies are first obtained using a spectral peak detector (see Section 4.2.1.1). (ii) Subsets of the detected peaks are used to generate possible harmonically related observations (see Section 4.2.1.2). (iii) The best non-conflicting observations (see Section 4.2.2.1) are used to track the voiced F_0 of multiple, overlapping speakers using a Kalman filter (see Section 4.2.1.3). However, the association of subsets of peak detections with the F_0 of a specific speaker is unknown *a priori*. This problem is exacerbated by some subsets being related to FAs[1]. (iv) A MHT method [96, 98, 139] is, therefore, proposed (see Section 4.2.1.4) that probabilistically updates the F_0 tracks using physically feasible subsets of peak detections. (v) To reduce the number of track hypotheses and, therefore, the computational complexity, a MWC method is also deployed (see Section 4.2.1.4). (vi) The complete segmentation is obtained from the resulting tracks where the start of a track corresponds to the onset of a speaker and the end of a track corresponds to the end-point of a speaker (see Section 4.2.2.2).

4.2.2.1 Best non-conflicting observation selection

At each time frame, N_t observations are generated, such that $\mathbf{Z}_t \triangleq \{\mathbf{z}_{t,0}, \mathbf{z}_{t,1}, \cdots, \mathbf{z}_{t,N_t}\}$. A particular problem arises when the observations result in multiple tracks for a single speaker at harmonics or subharmonics of F_0. The subsequent detection errors are normally classified as harmonic or octave errors, which are also common in F_0 estimation algorithms [150]. An F_0 observation is said to conflict if it is a harmonic or subharmonic of any other observation. To reduce such errors in the proposed method, only one F_0 observation is tracked if multiple F_0 observations conflict. At each time frame, t, an iterative selection process is utilised to select the best non-conflicting observations. An empty set, $\hat{\mathbf{Z}}_t$, is first initialised. Then, at each iteration, the observation composed of the most measurements, that is the observation vector, $\hat{\mathbf{z}}_{t,0}$, of longest length is appended to $\hat{\mathbf{Z}}_t$ and all observations conflicting with $\hat{\mathbf{z}}_{t,0}$ are removed. If two or more observation vectors have the same length then the one associated with the highest F_0 is appended. This iterative process continues until all observations are either appended to $\hat{\mathbf{Z}}_t$, or removed. The $M_t \leq N_t$ selected observations, $\hat{\mathbf{Z}}_t = \{\hat{\mathbf{z}}_{t,0}, \hat{\mathbf{z}}_{t,1}, \cdots, \hat{\mathbf{z}}_{t,M_t}\}$, at each time frame are then used to form tracks.

4.2.2.2 Proposed System-2 segmentation

The result of the proposed System-2 is a complete segmentation, obtained from the analysis of the individual tracks. The onsets and end-points of a speaker correspond to times of initialisation and termination respectively of the corresponding tracks.

4.3 Experimental Setup

This section summarises two experiments that are carried out to evaluate the performance of the two proposed systems. A baseline is introduced for performance and complexity comparison. Code for the proposed methods is available at [142].

4.3.1 Exp-1: Full Segmentation using the Proposed System-1 and System-2

Exp-1 evaluates the performance of the two proposed methods as complete segmentation systems. System-2 is compared against System-1 and the baseline, a state-of-the-art deep learning approach presented in [41].

To make the comparison more valid Pyannote provides a model that was pre-trained [151] on the AMI corpus. It was this pre-trained model that was used as the baseline for this experiment. In Section 4.3.1.3

P_t	ξ	$F_{\min}$, $F_{\max}$, F_{tol}	σ_w, σ_v	ζ	k	η
20	1.0×10^6	70, 300, 5	10, 400	25	1	7

Table 4.1: Parameter Setting.

the method used to train this baseline model, provided by Pyannote, is described in detail. The code for the two proposed methods can be found here [152].

4.3.1.1 Parameters of System-1 and System-2

In Exp-1 and Exp-2, a modified version of PEFAC [115, 121] was developed as the spectral peak detector for both System-1 and System-2. The modification removes the restriction, in PEFAC, that the filter used to detect the harmonics is centred in the limited range of 0.9 - 1.1 times the F_0. The parameters selected for System-1 and System-2 in the experiments, as shown in this chapter, are given in Table 4.1 where initialisation uses both physiological constraints of σ_w: process noise variance, σ_v: observation noise variance, $F_{\min}$: minimum allowed F_0, $F_{\max}$: maximum allowed F_0, F_{tol}: allowed tolerance of harmonic components and empirical tuning of P_t: maximum number allowed spectral peaks, ξ: voiced frame threshold, ζ: VAD region merge threshold, k: the interval of time frames before discarding less likely tracks using the MWC, η: speaker overlap threshold. This empirical tuning was achieved through an exhaustive grid search [124] on the development set of the AMI corpus.

4.3.1.2 Computational complexity of System-1 and System-2

The proposed System-1 approach is computationally demanding as all generated subset information needs to be kept and tracked. In contrast, the proposed System-2 approach has a lower computational complexity making it a more efficient implementation. To highlight the efficiency of the proposed System-2 approach, consider the computational complexity for a given time frame, t, with A_t active tracks and N_t observations. If all tracks are updated with all observations, then $A_t \times N_t$ possible new tracks are created. It is also possible that all N_t observations are wrong and, therefore, for each active track only the prediction step is performed, creating A_t further tracks. Assuming in this analysis that no tracks are terminated and no observations are discarded due to gating, the total number of possible new tracks is $(A_t \times N_t) + A_t$. The MWC is then computed using the Bron-Kerbosch algorithm [153], an enumeration algorithm for finding MWCs in an undirected graph, where each possible track represents a node. Calculating the MWC at each time step is the most expensive operation and, therefore, the Bron-Kerbosch algorithm dominates the complexity O-number. In the worst case, the time complexity for the Bron-Kerbosch algorithm is

$O(3^{\frac{L}{3}})$ for an L-node graph. Accordingly, the complexity for each time frame is

$$f = O(3^{(A_t N_t + A_t)/3}) \ . \tag{4.11}$$

Therefore, the number of tracks, A_t, and observations, N_t, at each time frame determine the computational complexity. The reduced computation of the proposed System-2 method over the proposed System-1 is achieved due to the early pruning in order to reduce the number of observations at each stage. Moreover, the proposed approach benefits from not requiring a post-processing step since each speaker corresponds to exactly one track.

4.3.1.3 Baseline in Exp-1

The baseline is a state-of-the-art deep learning approach [41]. This task can be formulated as a sequence labelling task where the input is the sequence of feature vectors

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T\} \ , \tag{4.12}$$

where $\mathbf{X}$ is a sequence of frame features extracted on a short overlapping sliding window and T is the total number of frames. The output is denoted by the corresponding sequence of labels

$$\mathbf{y} = \{y_1, y_2, \cdots, y_T\} \in \{0, 1\}^T \ . \tag{4.13}$$

The `pyannote.audio` [41] framework was used to train a neural network $f : \mathbf{X} \rightarrow \mathbf{y}$ that matches a feature sequence $\mathbf{X}$ to the corresponding label sequence $\mathbf{y}$. If there is a speaker change at frame t then $y_t = 1$ otherwise $y_t = 0$.

Data: The partitions used for training, validation and testing in this experiment were the partitions suggested by the AMI corpus [101].

Feature extraction: The waveform is used directly where $\mathbf{x}_t$ is SincNet learnable features [147].

Network architecture: The model stacks 2 BLSTMs and a multi-layer perceptron, each with 128 units in both forward and backward directions, and a final classification layer (2 units, softmax activation). This was to match the architecture from the original paper [41, 147].

Training: The network was trained for 1000 epochs on the AMI database using the training set given

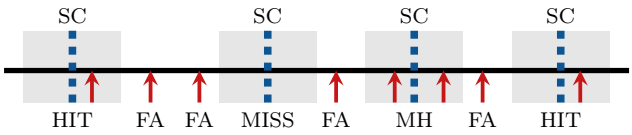


Figure 4.7: Illustrative example of the evaluation framework used in Exp-1 where the blue dashed lines represent the oracle speaker change boundaries and the grey regions correspond to the given collar. A 'HIT' is where a speaker change has been detected once. A 'MISS' is when a speaker change has not been detected and a multi-hit, 'MH', is where a speaker change has been detected multiple times within its collar. A false alarm (FA) is when a detection falls outside of any speaker change collars.

in the 'full-corpus partition of meetings' [103]. The training configuration is the same as the original paper [41, 147]. To address the class imbalance problem and to account for human annotation imprecision, a 50 ms collar (which is the same collar length used in Chapter 3) is used around each speaker change event. The training is implemented using the Keras toolkit [154].

Targets: The speaker change labels are obtained from the ground-truth AMI annotation files.

4.3.1.4 Evaluation framework used for Exp-1

To evaluate the performance of the proposed systems, the following metrics have been defined (more details are given in Section 2.3.5.1 on Page 16). A 'HIT' is when a speaker change has been detected once. A 'MISS' is when a speaker change has not been detected and a 'MH' is when a speaker change has been detected multiple times within a time collar applied around every ground-truth speaker change in order to account for possible inaccuracies [9]. A FA is when a detection falls outside of any speaker change collars.

The HR is given by

$$\frac{\text{HITs} + \text{MHs}}{\text{HITs} + \text{MHs} + \text{MISSs}} \text{ expressed as } \%. \tag{4.14}$$

The MISS rate is given by complement percentage of the HR. The false alarm (FA) rate is given by

$$\frac{\text{FAs}}{\text{HITs} + \text{MHs} + \text{FAs}} \text{ expressed as } \%. \tag{4.15}$$

The MH rate is given by

$$\frac{\text{MHs}}{\text{HITs} + \text{MHs}} \text{ expressed as } \%. \tag{4.16}$$

These detection types are defined graphically in Fig. 4.7 where the scores would be as follows: HR: 75%,

59	**MFCC Features**
19	Cepstral coefficients
19	Delta cepstral coefficients
19	Delta delta cepstral coefficients
1	Delta energy coefficients
1	Delta delta energy coefficients

Table 4.2: MFCC Features.

MISS rate: 25%, MH rate: 33%, false alarm rate (FAR): 57%. A standard collar of 250 ms is used for Exp-1. The MSE in time is also calculated for all the HITs and the closest MH detections to the ground truth given by the AMI annotation files.

4.3.2 Exp-2: Full Segmentation using the F_0 Tracks as Features in F_0-BLSTM

4.3.2.1 Proposed F_0-BLSTM model

Exp-2 evaluates the performance of the proposed method when used as an input feature to the F_0-BLSTM, the same BLSTM used for the baseline. This section shows how the F_0 can be used as an input feature to a deep neural network (DNN) along with standard MFCC features.

Data: The partitions used for training, validation and testing in this experiment were the same partitions that were used for Exp-1.

Feature extraction: Two features are extracted for use in Exp-2. A 59-dimensional MFCC feature is extracted using `librosa` [155] as shown in Table 4.2, and a 26-dimensional F_0 feature is extracted using the proposed method. To extract this F_0 feature using the proposed System-2 at each frame, 26 bins are created at intervals of 10 Hz in the range from 50 to 300 Hz. Each bin is assigned a value of '1' if a track is contained in that bin and '0' otherwise. This makes it possible to create a feature vector of a fixed length that captures the number of F_0 tracks and their frequency information at each frame. A comparison is made by performing segmentation using the 59-dimensional MFCC feature alone and in conjunction with the 26-dimensional F_0 feature.

Network architecture: The F_0-BLSTM uses the same network architecture as the baseline from Exp-1 (see Section 4.3.1.3).

Training: The network was trained for 200 epochs on the same AMI training set used in Exp-1. The same training configuration as Exp-1 was also used.

Targets: The same targets as Exp-1 are used which were obtained from the ground-truth AMI annotation files.

4.3.2.2 Exp-2: Evaluation framework

To evaluate the performance of the F_0-BLSTM when trained on different features, `pyannote.metrics` [100] was utilised. Two metrics are calculated: 1) the segment-wise coverage, which is the ratio of the duration of the intersection with the most co-occurring hypothesis segment, and the duration of the reference segment; 2) the purity, which would be the same as the coverage if the reference and hypothesis segments were to switch roles and indicates how pure the hypothesis is for each segment. The results presented in Section 4.4.2 are a duration-weighted average over each segment.

4.4 Experimental evaluation

In this section, the performance of Exp-1 and Exp-2 will be evaluated on the AMI corpus.

4.4.1 Exp-1 Evaluation

Exp-1 is evaluated (see Section 4.3.1.4) on AMI and the statistical results are given in Table 4.3. An illustrative example is also provided in Fig. 4.8 on Page 52 and Fig. 4.9 on Page 53 to compare the proposed System-1 method against the proposed System-2.

4.4.1.1 Illustrative example on AMI

To illustrate the operation of the proposed System-1, shown in Fig. 4.8, and System-2, shown in Fig. 4.9, a speech segment from meeting 'TS3003b' in the AMI corpus [103] was selected. Fig. 4.8(a) and Fig. 4.9(a) show $\hat{\Phi}_t$ generated from PEFAC. Fig. 4.8(b) shows the observations, $\mathbf{Z}_t$, generated from the F_0 measurements, $\hat{\Phi}_t$, (see Section 3.2). Fig. 4.8(c) highlights how multiple tracks, T_i, can be generated for the same speaker when System-1 is used. Fig. 4.8(c) also shows how all these tracks are still harmonically related to each other. Fig. 4.9(b) shows the effect of choosing the best non-conflicting observation used in the proposed method. Lastly, Fig. 4.9(c) demonstrates how post-processing is not needed as the single speaker no longer has multiple tracks associated with it. To compare the performance of the proposed System-1, Fig. 4.8(d), and proposed System-2, Fig. 4.9(d), the results for both methods are shown.

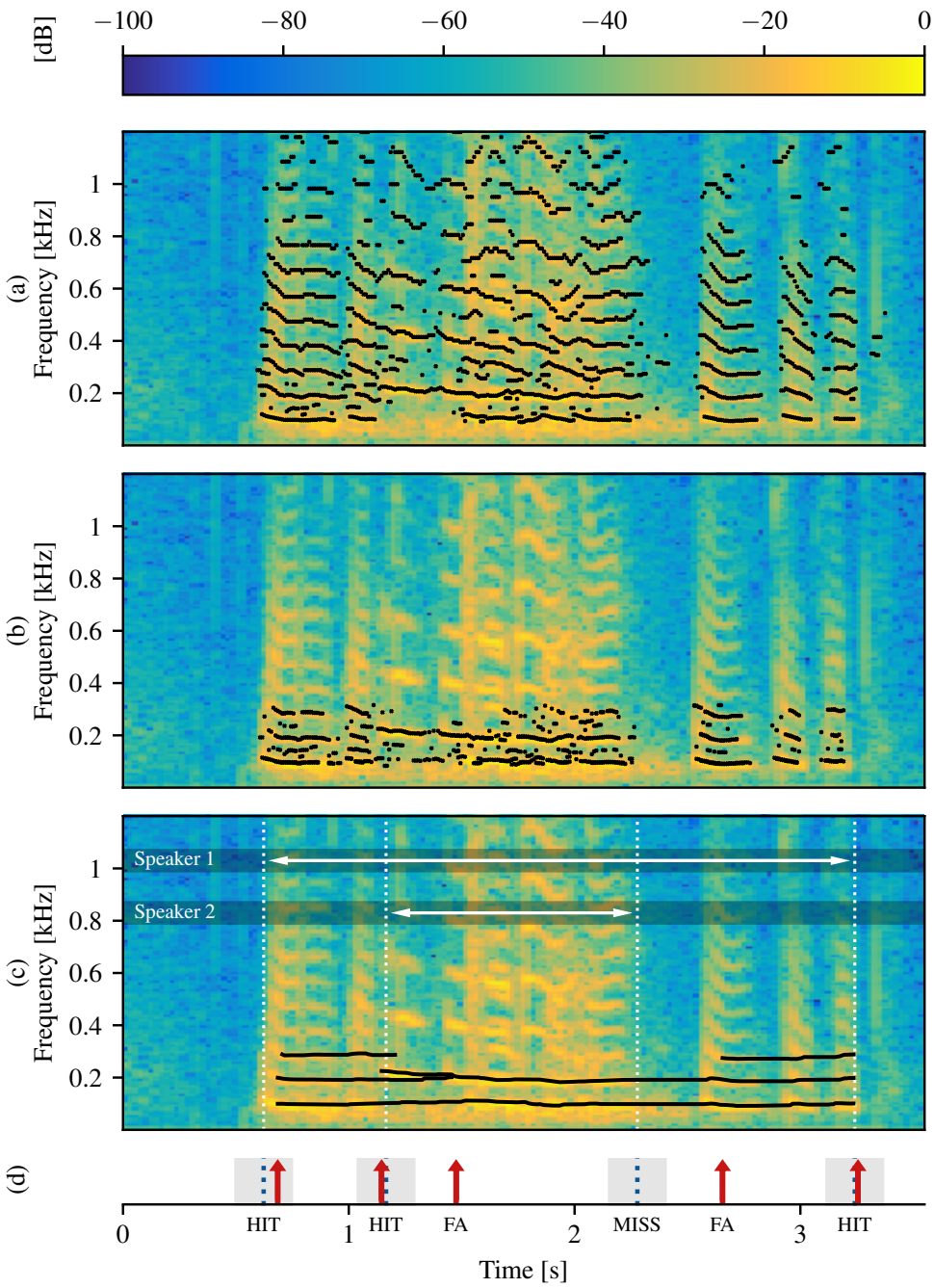


Figure 4.8: Illustrative AMI example using the proposed System-1. (a) PEFAC output, (b) generated observations where the black crosses show the F_0 value for each observation, (c) generated tracks from all possible observations (System-1 before post-processing) and (d) proposed System-1 performance.

The complete segmentation given by System-1, shown in Fig. 4.8(d), results in 3 errors (1 MISS and 2 FA) caused by the F_0 track of Speaker 2 being very similar to the first harmonic of Speaker 1. This highlights a disadvantage of System-1 where all the observations are tracked. The complete segmentation given by System-2, shown in Fig. 4.9(d), contains two errors (1 MH and 1 FA) due to an unvoiced region of

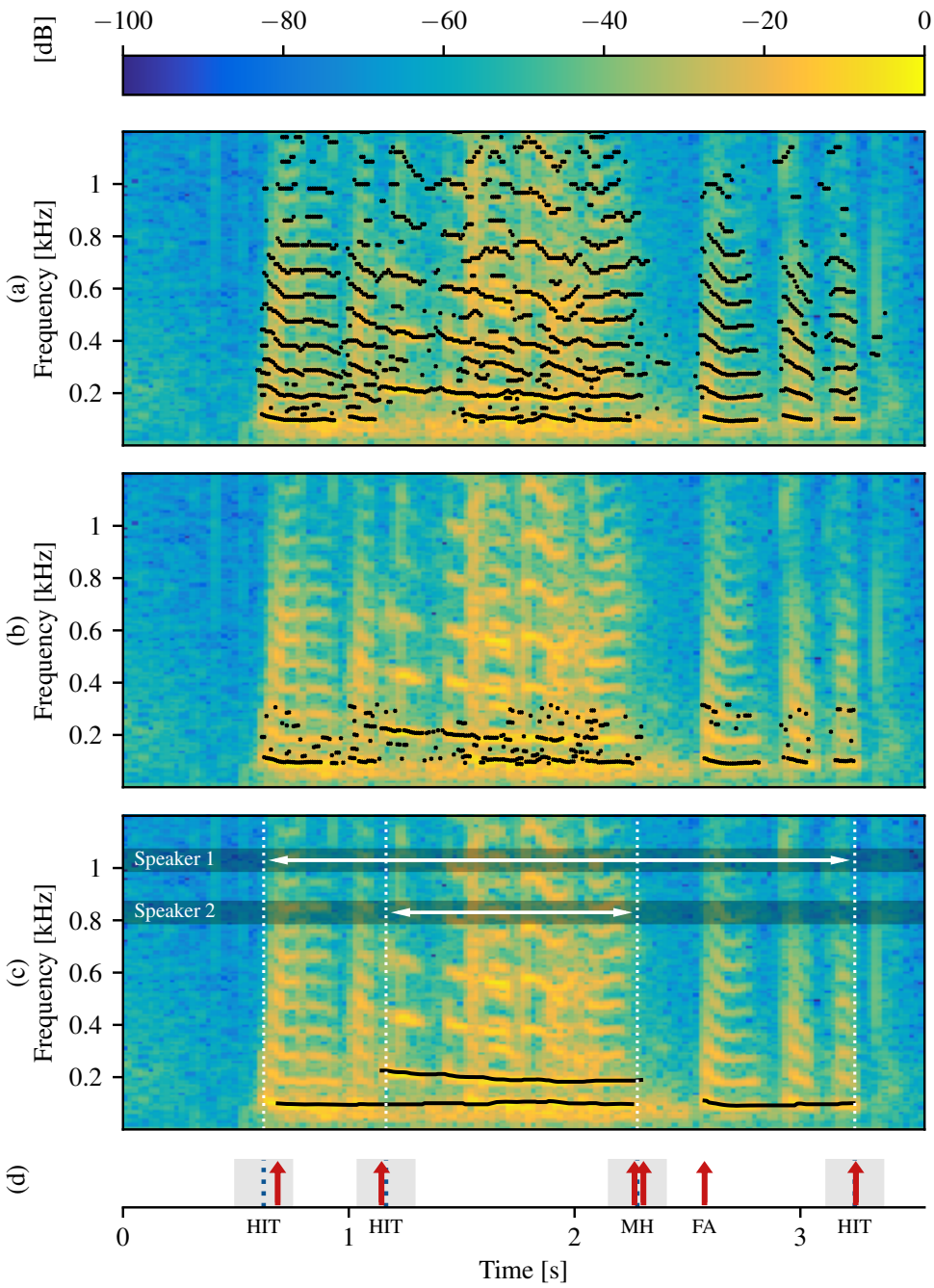


Figure 4.9: Illustrative AMI example using the proposed System-2. (a) PEFAC output, (b) best non-conflicting observations, (c) generated tracks from best non-conflicting observations and (d) proposed System-2 performance.

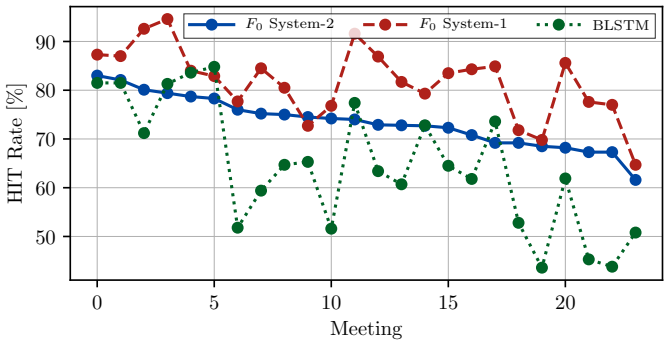
speech from Speaker 1 between [2.28,2.58] s. The System-2 method attempts to deal with this problem of unvoiced speech by continuing tracks even after the F_0 is no longer detectable. This can be seen earlier in the example between [1.21, 1.45] s where there are gaps in the detection of the F_0 track from Speaker 1.

Meeting	Proposed F_0 System-2									
	IHM Mixed-Down Stream					SDM Stream				
	HIT	MISS	MH	MSE	FA	HIT	MISS	MH	MSE	FA
EN2002a	78.7%	21.3%	52.1%	0.014	64.9%	80.0%	20.0%	53.9%	0.014	64.7%
EN2002b	83.0%	17.0%	58.4%	0.014	68.5%	80.2%	19.9%	55.2%	0.015	70.4%
EN2002c	82.1%	17.9%	57.8%	0.014	71.5%	80.0%	20.0%	54.0%	0.015	73.8%
EN2002d	78.3%	21.7%	54.4%	0.014	66.6%	80.8%	19.1%	54.5%	0.015	66.9%
ES2004a	72.8%	27.2%	48.9%	0.015	73.6%	75.0%	25.0%	53.0%	0.015	76.1%
ES2004b	70.8%	29.2%	42.1%	0.015	81.0%	75.8%	24.1%	51.7%	0.014	81.3%
ES2004c	72.9%	27.1%	43.2%	0.016	78.2%	77.2%	22.9%	49.4%	0.015	79.5%
ES2004d	75.2%	24.8%	49.1%	0.014	71.0%	76.2%	23.8%	50.6%	0.013	74.7%
ES2014a	61.6%	38.4%	33.7%	0.015	85.3%	61.6%	38.4%	33.7%	0.015	85.3%
ES2014b	67.3%	32.7%	42.3%	0.015	83.4%	67.3%	32.7%	42.3%	0.015	83.4%
ES2014c	67.3%	32.7%	38.7%	0.016	82.0%	67.3%	32.7%	38.7%	0.016	82.0%
ES2014d	69.2%	30.8%	43.6%	0.015	77.1%	69.2%	30.8%	43.6%	0.015	77.1%
IS1009a	72.7%	27.3%	44.9%	0.014	77.0%	73.5%	26.5%	49.6%	0.014	77.0%
IS1009b	69.2%	30.8%	41.0%	0.016	79.2%	72.2%	27.8%	44.0%	0.016	80.7%
IS1009c	74.5%	25.5%	43.5%	0.014	84.5%	70.3%	29.7%	40.6%	0.015	86.3%
IS1009d	72.3%	27.7%	45.0%	0.014	75.8%	76.2%	23.8%	48.7%	0.015	76.0%
TS3003a	68.5%	31.5%	39.5%	0.015	85.2%	70.7%	29.3%	43.0%	0.015	88.4%
TS3003b	76.0%	24.0%	43.2%	0.015	83.8%	80.7%	19.3%	53.5%	0.016	85.8%
TS3003c	74.2%	25.8%	42.8%	0.015	86.4%	76.8%	23.2%	49.5%	0.015	89.0%
TS3003d	75.0%	25.0%	43.2%	0.016	74.7%	80.8%	19.2%	57.9%	0.014	75.1%
TS3007a	68.2%	31.9%	39.4%	0.015	77.5%	68.2%	31.9%	39.4%	0.015	77.5%
TS3007b	80.1%	19.9%	57.9%	0.014	84.9%	80.1%	19.9%	57.9%	0.014	84.9%
TS3007c	74.0%	26.0%	50.0%	0.014	76.6%	74.0%	26.0%	50.0%	0.014	76.6%
TS3007d	79.4%	20.6%	57.9%	0.014	68.2%	79.4%	20.6%	57.9%	0.014	68.2%
Std Dev	5.1%	5.1%	6.8%	0.001	6.3%	5.3%	5.3%	6.6%	0.001	6.5%
Mean	73.5%	26.5%	46.4%	0.015	77.4%	74.7%	25.3%	48.9%	0.015	78.4%

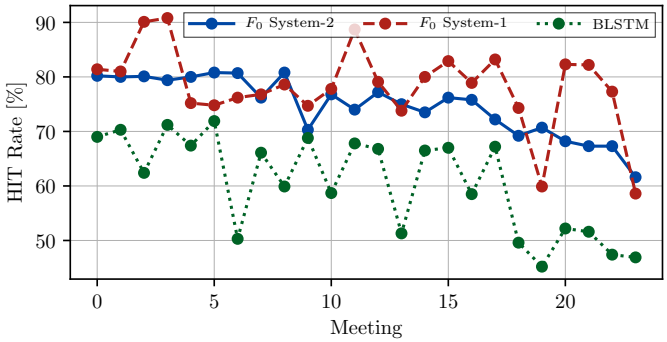
Meeting	Proposed F_0 System-1									
	IHM Mixed-Down Stream					SDM Stream				
	HIT	MISS	MH	MSE	FA	HIT	MISS	MH	MSE	FA
EN2002a	84.0%	16.0%	70.0%	0.009	71.4%	75.2%	24.9%	59.8%	0.014	74.3%
EN2002b	87.3%	12.7%	73.6%	0.009	76.3%	81.4%	18.6%	67.1%	0.014	77.5%
EN2002c	87.0%	13.0%	72.7%	0.009	81.2%	81.0%	19.0%	65.3%	0.015	82.6%
EN2002d	82.9%	17.1%	70.2%	0.009	74.4%	74.8%	25.2%	60.2%	0.015	76.7%
ES2004a	81.7%	18.3%	68.4%	0.009	81.5%	73.8%	26.2%	60.4%	0.014	83.8%
ES2004b	84.3%	15.7%	70.8%	0.009	88.0%	78.9%	21.1%	62.4%	0.016	90.0%
ES2004c	86.9%	13.1%	74.4%	0.009	87.2%	79.1%	20.9%	63.5%	0.015	89.1%
ES2004d	84.5%	15.5%	71.1%	0.009	82.3%	76.8%	23.2%	61.9%	0.014	83.9%
ES2014a	64.7%	35.3%	49.9%	0.010	90.6%	58.6%	41.4%	43.0%	0.015	91.3%
ES2014b	77.6%	22.4%	62.4%	0.009	87.1%	77.3%	22.7%	60.1%	0.015	87.7%
ES2014c	77.0%	23.0%	60.8%	0.009	86.0%	82.2%	17.8%	68.2%	0.012	84.5%
ES2014d	71.8%	28.2%	54.2%	0.010	84.7%	74.3%	25.7%	55.2%	0.016	84.3%
IS1009a	79.3%	20.6%	66.5%	0.009	83.5%	80.0%	20.0%	64.1%	0.016	83.7%
IS1009b	84.9%	15.1%	70.3%	0.009	86.8%	83.2%	16.8%	69.9%	0.014	88.2%
IS1009c	72.7%	27.3%	55.6%	0.010	91.2%	74.7%	25.3%	60.2%	0.014	92.0%
IS1009d	83.5%	16.4%	68.6%	0.008	84.3%	82.9%	17.1%	70.7%	0.014	84.5%
TS3003a	69.8%	30.2%	57.0%	0.010	93.3%	59.9%	40.1%	43.3%	0.018	93.4%
TS3003b	77.7%	22.3%	63.4%	0.013	92.8%	76.2%	23.8%	60.7%	0.019	92.2%
TS3003c	76.8%	23.2%	63.7%	0.012	93.3%	77.8%	22.2%	63.8%	0.018	93.1%
TS3003d	80.5%	19.4%	67.8%	0.010	85.7%	78.6%	21.4%	63.3%	0.016	85.7%
TS3007a	85.6%	14.4%	74.8%	0.009	84.2%	82.3%	17.7%	70.5%	0.014	84.2%
TS3007b	92.6%	7.4%	82.3%	0.007	87.8%	90.1%	9.9%	78.6%	0.012	87.0%
TS3007c	91.6%	8.4%	82.5%	0.007	82.5%	88.7%	11.3%	77.9%	0.012	82.8%
TS3007d	94.6%	5.4%	85.9%	0.006	74.1%	90.8%	9.2%	79.3%	0.012	75.1%
Std Dev	7.1%	7.1%	8.7%	0.001	5.9%	7.4%	7.4%	8.7%	0.002	5.4%
Mean	81.6%	18.3%	68.2%	0.009	84.6%	78.3%	21.7%	63.7%	0.015	85.3%

Meeting	BLSTM Baseline System									
	IHM Mixed-Down Stream					SDM Stream				
	HIT	MISS	MH	MSE	FA	HIT	MISS	MH	MSE	FA
EN2002a	83.6%	16.4%	38.1%	0.008	47.2%	67.4%	32.6%	67.4%	0.011	55.6%
EN2002b	81.5%	18.5%	36.0%	0.008	53.6%	69.0%	31.0%	69.0%	0.011	56.3%
EN2002c	81.5%	18.6%	35.3%	0.007	52.2%	70.3%	29.7%	70.3%	0.012	61.2%
EN2002d	84.8%	15.2%	40.0%	0.008	51.1%	71.9%	28.1%	71.9%	0.012	55.3%
ES2004a	60.7%	39.3%	22.5%	0.010	54.0%	51.3%	48.7%	51.3%	0.015	65.2%
ES2004b	61.8%	38.2%	19.3%	0.010	54.8%	58.5%	41.5%	58.5%	0.014	73.6%
ES2004c	63.4%	36.6%	24.3%	0.008	45.0%	66.8%	33.2%	66.8%	0.012	72.9%
ES2004d	59.4%	40.6%	20.0%	0.010	48.4%	66.1%	33.9%	66.1%	0.012	61.5%
ES2014a	50.8%	49.2%	21.4%	0.014	61.9%	46.9%	53.1%	15.7%	0.012	81.8%
ES2014b	43.8%	56.2%	14.9%	0.010	61.7%	47.4%	52.6%	15.3%	0.013	80.0%
ES2014c	45.3%	54.7%	15.0%	0.012	61.0%	51.6%	48.4%	17.5%	0.013	79.5%
ES2014d	52.8%	47.2%	17.6%	0.013	61.4%	49.6%	50.4%	12.9%	0.013	76.5%
IS1009a	72.8%	27.1%	31.3%	0.008	52.1%	66.5%	33.5%	66.5%	0.008	64.7%
IS1009b	73.6%	26.4%	29.2%	0.008	47.3%	67.2%	32.8%	67.2%	0.010	67.3%
IS1009c	65.3%	34.7%	22.4%	0.008	63.3%	68.8%	31.2%	68.8%	0.009	76.2%
IS1009d	64.5%	35.5%	23.0%	0.010	49.3%	67.0%	33.0%	67.0%	0.010	63.4%
TS3003a	43.6%	56.4%	11.1%	0.011	71.9%	45.2%	54.8%	45.2%	0.014	81.6%
TS3003b	51.8%	48.2%	23.2%	0.017	51.7%	50.3%	49.7%	50.3%	0.021	79.7%
TS3003c	51.6%	48.4%	14.9%	0.018	68.9%	58.7%	41.3%	58.7%	0.021	81.1%
TS3003d	64.7%	35.3%	24.8%	0.015	48.3%	59.9%	40.1%	59.9%	0.015	61.2%
TS3007a	61.9%	38.1%	22.8%	0.011	49.6%	52.2%	47.8%	16.8%	0.009	75.5%
TS3007b	71.2%	28.8%	22.5%	0.007	41.9%	62.4%	37.6%	14.2%	0.009	80.5%
TS3007c	77.4%	22.6%	33.3%	0.009	51.4%	67.8%	32.2%	23.8%	0.009	73.1%
TS3007d	81.3%	18.6%	34.4%	0.009	46.6%	71.2%	28.8%	22.4%	0.010	67.8%
Std Dev	12.8%	12.8%	7.9%	0.003	7.5%	8.7%	8.7%	22.4%	0.003	8.8%
Mean	64.5%	35.5%	24.9%	0.010	53.9%	60.6%	39.4%	47.6%	0.012	70.5%

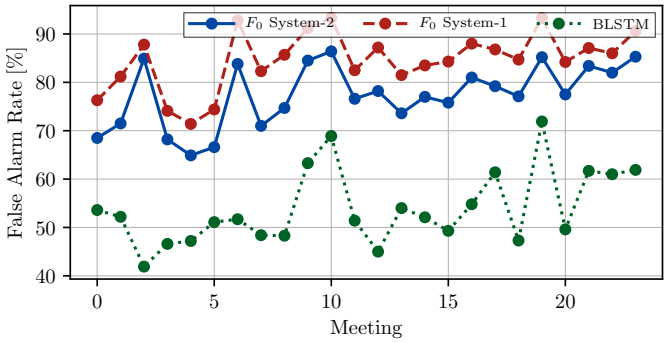
Table 4.3: Performance comparison of both the individual headset microphone (IHM) mixed-down stream and the single distant microphone (SDM) stream on the multi-speaker meetings in the AMI corpus along with the bidirectional long short term memory network (BLSTM) approach (baseline) using a collar of 250 ms. (*A graphical representation of these results is also given in Fig. 4.10 on Page 55 and Fig. 4.11 on Page 56.*)



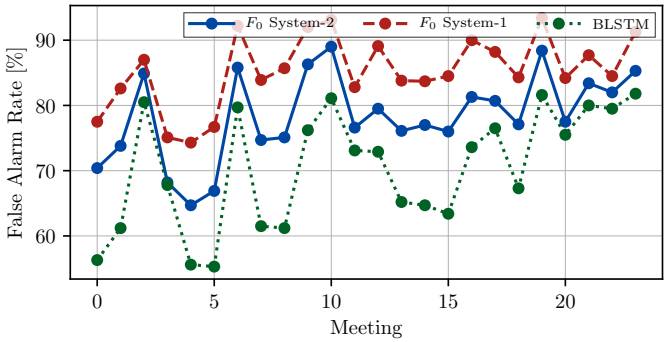
(a) HR comparison on the IHM mixed-down stream.



(b) HR comparison on the SDM Stream.

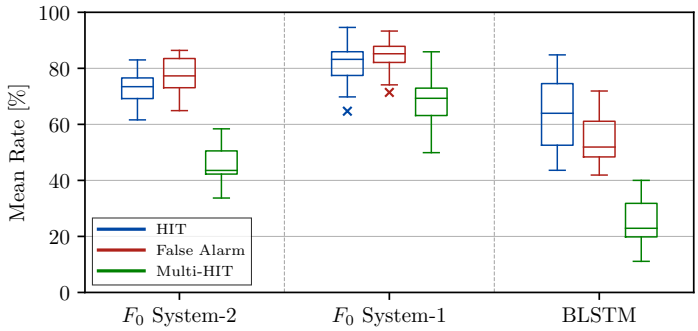


(c) False alarm rate comparison on the IHM mixed-down stream.

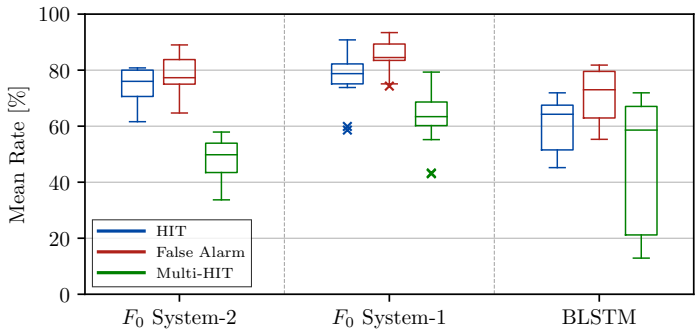


(d) False alarm rate comparison on the SDM Stream.

Figure 4.10: Performance comparison of both the individual headset microphone (IHM) mixed-down stream and the single distant microphone (SDM) stream on the multi-speaker meetings in the AMI corpus along with the bidirectional long short term memory network (BLSTM) approach (baseline) using a collar of 250 ms. (*A tabular form of these results is also given in Table 4.3 on Page 54.*)



(a) Mean rate comparison on IHM mixed-down stream.



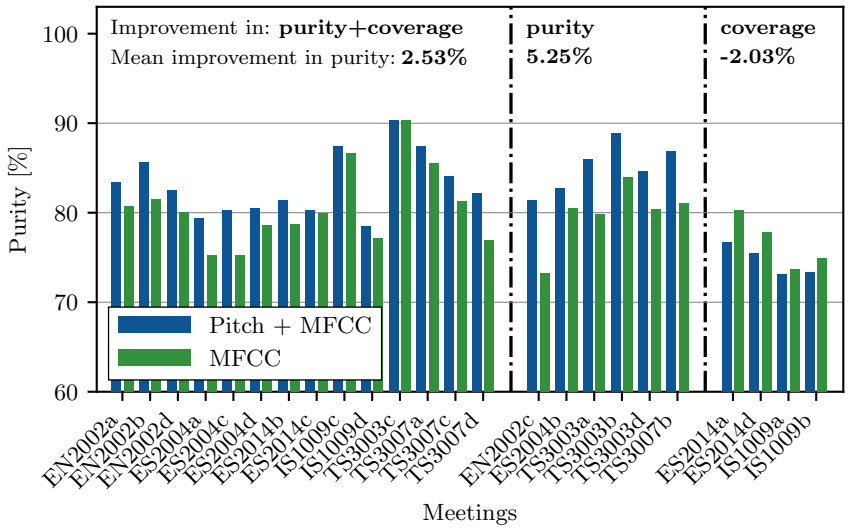
(b) Mean rate comparison on the SDM Stream.

Figure 4.11: Comparison of the mean rates on both the individual headset microphone (IHM) mixed-down stream and the single distant microphone (SDM) stream on the multi-speaker meetings in the AMI corpus. (*A tabular form of these results is also given in Table 4.3 on Page 54.*)

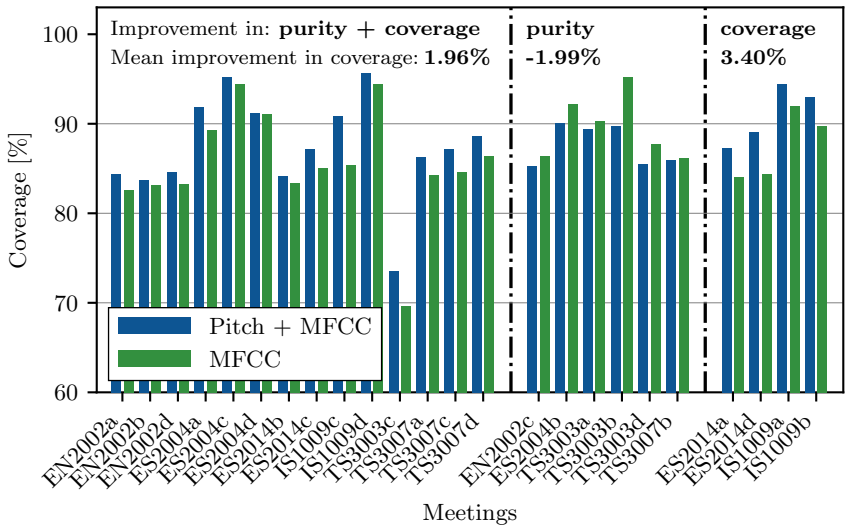
4.4.1.2 Statistical results on AMI

To evaluate the performance, the two proposed methods are compared against a commonly used BLSTM baseline method [41] where the model was trained on the AMI corpus [103]. Two types of microphone inputs were used in the evaluation of 24 meetings in the AMI corpus: (i) mixed-down individual headset microphone (IHM) stream containing the sum of close-talking microphones without significant reverberation or noise; (ii) a SDM containing room reverberation and ambient noise. The results in Table 4.3 on Page 54 show that the two proposed methods achieve a better HR using only the F_0 information compared to a baseline that uses SincNet learnable features extracted from the raw waveform. One explanation for the improvement in the HR seen on the proposed systems is an improvement in detecting overlapping speech events in the spoken backchannel, such as 'um' and 'uh-huh'.

The proposed System-2 method also achieves an improvement of 1.2% on the HR on the SDM stream compared against the performance on the IHM stream. The improvement in HR does, however, need to be traded off against a degradation of 1.0% in the FAR. What should be noted is that the 1.2%



(a) Purity performance improvement where the mean improvement for each subgroup is also given. The mean improvement for all meetings is 2.45%.



(b) Coverage performance improvement where the mean improvement for each subgroup is also given. The mean improvement for all meetings is 1.21%.

Figure 4.12: Performance comparison of the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream. The meetings are ordered alphabetically in three subgroups. The first group sees an improvement in both metrics; the second group only sees an improvement in purity and the last group only sees an improvement in coverage.

improvement in HR and a slight increase in FAR does show that the proposed System-2 method is not massively affected by the presence of noise and reverberation. The BLSTM baseline is more affected by these degradations which can be seen from a drop in HR of 3.9% and an increase in the FAR by 16.6% between the IHM and SDM stream.

Meeting	F_0 + MFCC Features					MFCC Features				
	HIT	MISS	MH	MSE	FA	HIT	MISS	MH	MSE	FA
EN2002a	80.9%	19.1%	60.1%	0.015	24.4%	80.7%	19.3%	64.3%	0.016	27.6%
EN2002b	81.0%	18.9%	54.3%	0.015	28.7%	80.1%	19.9%	61.9%	0.015	31.8%
EN2002c	74.2%	25.8%	50.6%	0.014	29.5%	69.0%	30.9%	51.0%	0.017	34.5%
EN2002d	81.8%	18.2%	60.3%	0.016	23.4%	81.7%	18.3%	64.2%	0.018	27.4%
ES2004a	59.1%	40.9%	36.6%	0.021	29.9%	57.5%	42.5%	43.8%	0.020	32.9%
ES2004b	59.2%	40.8%	36.9%	0.017	24.8%	53.9%	46.1%	32.0%	0.016	29.4%
ES2004c	53.0%	47.0%	29.5%	0.020	24.4%	52.4%	47.6%	30.9%	0.021	27.9%
ES2004d	63.3%	36.7%	38.4%	0.016	27.0%	61.1%	38.9%	39.9%	0.019	30.9%
ES2014a	47.3%	52.7%	25.4%	0.023	44.1%	60.5%	39.5%	50.0%	0.020	41.5%
ES2014b	49.7%	50.3%	31.1%	0.020	40.7%	48.0%	52.0%	34.9%	0.023	47.6%
ES2014c	57.2%	42.8%	33.7%	0.023	33.9%	60.9%	39.1%	42.2%	0.020	37.0%
ES2014d	44.5%	55.5%	23.7%	0.019	36.2%	53.3%	46.7%	36.8%	0.018	41.6%
IS1009a	45.5%	54.5%	25.1%	0.022	33.6%	45.0%	55.0%	22.8%	0.016	39.6%
IS1009b	46.2%	53.8%	22.0%	0.020	25.5%	50.1%	49.9%	30.1%	0.021	28.9%
IS1009c	50.1%	49.9%	32.0%	0.016	34.5%	40.0%	60.0%	22.9%	0.016	60.4%
IS1009d	46.5%	53.5%	27.3%	0.016	23.9%	38.6%	61.4%	18.4%	0.025	35.3%
TS3003a	42.2%	57.8%	24.9%	0.026	58.8%	28.5%	71.5%	16.7%	0.020	64.1%
TS3003b	48.1%	51.9%	32.1%	0.019	45.6%	27.2%	72.8%	15.6%	0.026	52.8%
TS3003c	65.1%	34.9%	41.2%	0.018	60.3%	60.1%	39.9%	39.1%	0.020	65.4%
TS3003d	64.0%	36.0%	39.7%	0.018	38.1%	55.3%	44.7%	35.5%	0.018	34.5%
TS3007a	67.5%	32.5%	44.6%	0.016	26.2%	66.6%	33.4%	51.9%	0.015	28.9%
TS3007b	67.9%	32.1%	43.1%	0.014	23.9%	53.9%	46.1%	34.2%	0.018	32.1%
TS3007c	73.9%	26.1%	48.0%	0.018	28.2%	72.0%	28.0%	54.7%	0.017	32.6%
TS3007d	78.1%	21.9%	53.8%	0.018	23.0%	73.1%	26.9%	52.0%	0.019	30.4%
Std Dev	12.8%	12.8%	11.4%	0.003	10.3%	14.6%	14.6%	14.3%	0.003	11.4%
Mean	60.3%	39.7%	38.1%	0.018	32.9%	57.1%	42.9%	39.4%	0.019	38.1%

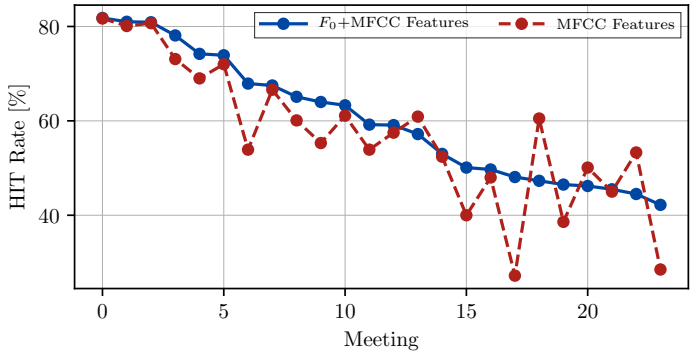
Table 4.4: Performance comparison of the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream with a collar of 250 ms. (*A graphical representation of these results is also given in Fig. 4.13 on Page 59.*)

4.4.2 Exp-2 Evaluation

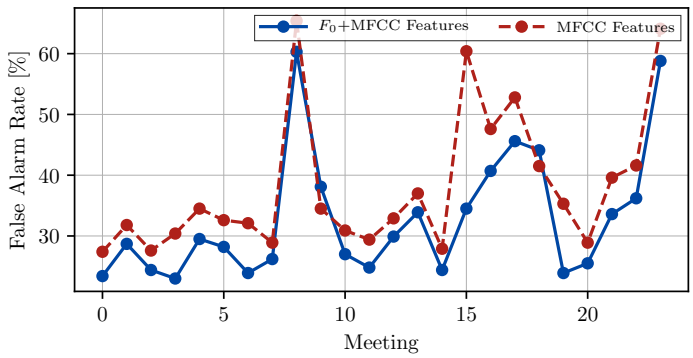
In its simplest, direct form, the proposed System-2 only relies on F_0 information. This motivates consideration of the proposed method as part of a multimodal approach. Such a multimodal approach can be formulated by using the proposed System-2 to calculate F_0 features and using those features in conjunction with other audio features as an input to the F_0-BLSTM system. The network architecture and feature extraction of the F_0-BLSTM system along with a description of the data used for training is elaborated on in Section 4.3.2.

4.4.2.1 Statistical results on AMI

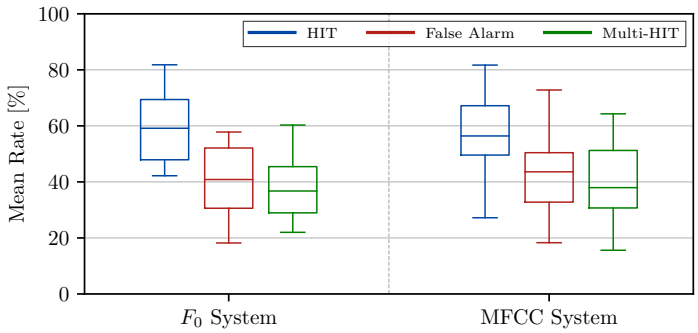
The performance is evaluated on the AMI SDM stream and the results for the coverage and purity can be seen in Fig. 4.12. The increase in performance by including F_0 features generated from the proposed



(a) HR comparison of the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream.



(b) False alarm rate comparison of the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream



(c) Comparison of mean rates on the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream

Figure 4.13: Performance comparison of the F_0-BLSTM system using F_0 and MFCCs as input features on the SDM stream with a collar of 250 ms. (*A tabular form of these results is also given in Table 4.4 on Page 58.*)

System-2 is evident. 20 out of 24 meetings have an improvement in purity; the largest improvement being 5.51% for meeting 'IS1009c'. The coverage is also improved by incorporating the F_0 with improvements seen in 18 out of 24 meetings; the largest improvement is in meeting 'EN2002c', which increased by 8.15%. In 14 out of 24 meetings, both the purity and the coverage are improved by incorporating the proposed F_0 features. Furthermore, there are no meetings where the performance is worse for both measures.

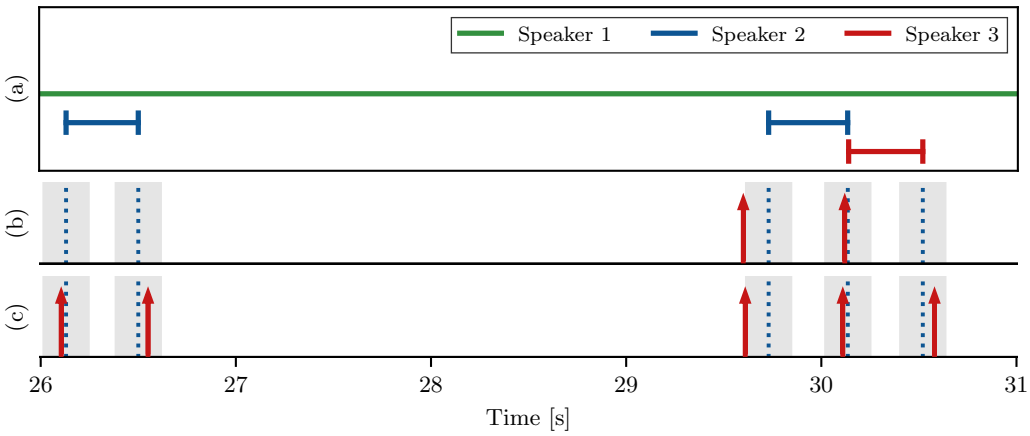


Figure 4.14: AMI meeting 'IS1009c' between [18:26, 18:31] mins. (a) Reference given by the AMI labels where Speaker 1 is 'FIO084', Speaker 2 is 'FIO089' and Speaker 3 is 'FIE088'. (b) Segmentation generated from F_0-BLSTM using only MFCCs as input features. (c) Segmentation generated from F_0-BLSTM using both MFCCs and F_0, extracted from the proposed method, as input features.

The explanation for this improvement is that it is likely due to better overlapping speech detection. To illustrate this for meeting 'IS1009c', an example is given in Fig. 4.14 that shows the improvements to overlapping speech detection when F_0 features from the proposed System-2 are utilised. In this example, Speaker 1 is active for the entire 6 seconds shown while Speakers 2 and 3 voice the phatic expressions of 'mm-hm' to signify they are listening.

'TS3003b' is an example of a meeting where including the F_0 features generated by the proposed System-2 does not improve the performance in terms of coverage (see Fig. 4.12). One possible explanation is that FAs can be caused by hesitation markers (e.g. 'um', 'er', or 'uh') which are words that are spoken in conversation by the active speaker to indicate that they have not finished speaking.

The performance is also evaluated using the metrics for Exp-1 and the results are given in Table 4.4. It can be seen that the four 'TS3003' meetings give a poor performance when only MFCC or SincNet learnable features (see Table 4.3) are used. This performance can be improved, however, by the addition of F_0 features.

An example of these FA errors can be seen between [10:24, 10:35] mins in meeting 'TS3003b' which contains the utterance "you can put uh a lot uh of uh functions uh in one uh yeah" from speaker 'MTD011UID'. These hesitation markers are the cause of FAs due to the speaker voicing them at a different F_0 from the rest of his utterance. The proposed System-2 incorrectly identifies the 'uh' hesitations as a different speaker. These FA detections result in errors being present in the generated F_0 feature.

The inclusion, therefore, of the proposed F_0 features in a segmentation system improves the overlapping speech detection. It is, however, worth bearing in mind that the inclusion of the proposed F_0 features does result in more FAs caused by hesitation markers. Another way to view the impact of including the proposed F_0 features is to say that not including F_0 features will lead to mostly 'under segmentation' errors and the inclusion of F_0 features will likely lead to 'over segmentation' errors. The results (see Fig. 4.12) show that in the AMI corpus, F_0 inclusion leads to better overall performance in most cases.

It could also be argued that in a complete diarization system 'over segmentation' errors are easier to handle as two segments belonging to the same speaker can be merged back together. If an audio recording is 'under-segmented', however, it becomes a much more challenging problem to solve.

4.5 Conclusion

This chapter has shown that the harmonic structure of voiced speech can be exploited for the task of speaker segmentation. An investigative study on a well-established corpus of conversational speech showed how changes in F_0 and changes in speaker are related. A novel method has been proposed that relies on a MHT framework to track multiple speakers even when they are talking simultaneously. The proposed method outperformed a BLSTM approach, in terms of HR, by 12.9% on the AMI corpus SDM stream. It has also been shown that the F_0 estimates obtained by the proposed System-2 can be used as input features for neural networks. This chapter showed that the segmentation performance of the baseline BLSTM can be improved by 1.21% in terms of coverage and 2.45% in terms of purity, by incorporating the F_0 estimates as input features, in addition to MFCCs. This work has also been published in the following papers [3, 4, 142].

Chapter 5

OVERLAPPING SPEAKER SEGMENTATION USING FUNDAMENTAL FREQUENCY AND DIRECTION OF ARRIVAL ESTIMATION

5.1 Introduction

As has been discussed in the preceding chapters, it is often desirable to detect which person is speaking at any given time in an audio recording [20] by the process of diarization. Accurate diarization is increasingly important for a multitude of applications including voice-controlled smart devices [21], speaker indexing [13], ASR [12] and multi-speaker separation [23]. In this chapter, the focus will be on segmentation which is one important aspect of the diarization process. Speaker segmentation is the process of detecting the onsets and end-points of each speaker utterance. One of the main challenges presented by speaker segmentation is that of segmenting overlapping speech, where a new speaker begins to speak before a previous speaker has stopped [26]. Overlapping speech is not the only complication, however, when it comes to speaker segmentation. Noise [29] and reverberation [28] also make the task more difficult.

A number of approaches were proposed to solve the problem of speaker segmentation. Most of these methods rely on features that fall into three separate categories: acoustic features [5, 85], spatial features [156] and linguistic features [157]. More recently data-driven, deep learning approaches have become popular [57, 89, 90]. However, these approaches often require large amounts of labelled training data.

The tracking of the temporal variations of a speaker's F_0 has already been shown to be useful in overlapping speaker segmentation [3] (see Chapter 4). The main problem with F_0-only tracking is the issue of unvoiced speech [143] which does not contain an F_0 or any significant harmonic characteristics [144, 145]. Likewise, direction of arrival (DoA) [80] information has also been exploited in the past for speaker

segmentation within the diarization task [158] and is commonly used in simultaneous localization and mapping (SLAM) [22].

In the past methods have been proposed that used DoA information to improve segmentation performance by enhancing the clustering step of uniformly segmented systems (see Fig. 1.1). These include [159] where MFCC coefficients are concatenated with time difference of arrival (TDoA) values to give a single vector which is passed to a HMM-based AHC framework where the clusters are merged in successive iterations to finally reach the optimal number of clusters [47]. Likewise, [160] shows that using DoA features alone can achieve a good segmentation and speaker clustering performance but also shows that DoA features are complimented by the use of acoustic features. The limitation of these methods is that they attempt to solve the segmentation task by using a clustering and re-alignment step and, therefore, do not take advantage of spatially varying temporal information.

It has also been shown that DoA information can be used as the sole input feature to a given segmentation system. For example, in [161] the DoA estimator proposed in [162] is used to estimate both the azimuth of the sound source and its power. A probabilistic model called dynamic latent Dirichlet allocation (dLDA) is then used for speaker clustering which is able to automatically infer the number of clusters. More recently in [163] spatial information was exploited through robust statistical Gaussian mixture model (GMM) modelling of TDoA estimates obtained using pairs of microphones. These methods all suffer from clustering mistakes introduced by the unreliable DoA estimates that could be overcome by using acoustic features.

Another problem with DoA-only segmentation methods is that they often estimate just one DoA per frame making them not suitable for overlapping speech for which two or more speakers at different DoAs are simultaneously active. Methods have been proposed that attempt to solve the overlapping speaker segmentation task using DoA information, such as, [164] which uses a multi-look [165] approach to calculate several look directions covering the panorama. This multi-look strategy allows the proposed system to automatically learn specific spatial information. However, this still does not solve the problem that arises when multiple speakers are located at the same azimuth making the speakers impossible to differentiate.

In this chapter, a method has been proposed that attempts to overcome the limitations of these previous methods by exploiting the varying temporal information, unlike [159, 160]. The proposed method will also track both an acoustic and spatial feature, in contrast to [161, 163] and will, therefore allow for

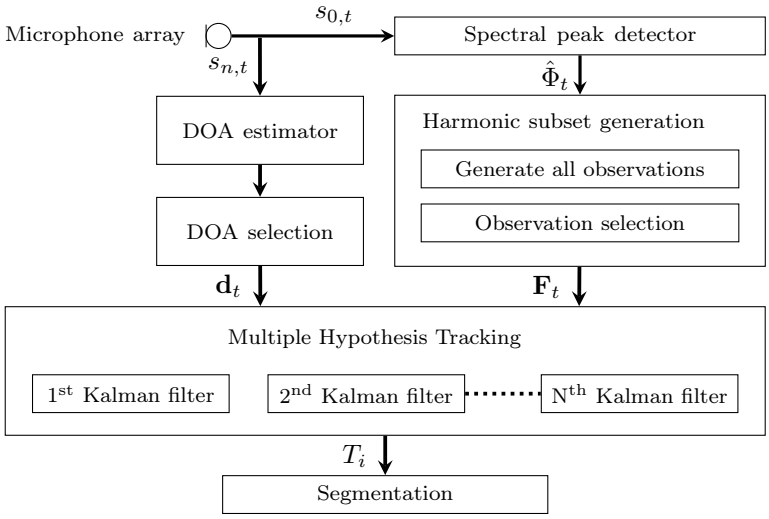


Figure 5.1: Proposed system architecture with $s_{n,t}$: input signals, $\hat{\Phi}_t$: reliable peak detections, $\mathbf{d}_t$: selected DoA observations, $\mathbf{F}_t$: selected F_0 observations, T_i: selected track hypotheses for the i-th speaker.

the tracking of multiple speakers even when they are located at the same azimuth which is not possible using [164]. The proposed method builds on the work from Chapter 4 by using a Kalman filter [94] and MHT [97] to simultaneously track both the F_0 and spatial DoA features with the aim of achieving more accurate speaker segmentation even in the context of overlapping speech. The multiple signal classification (MUSIC) algorithm [80, 166] is used to estimate the DoA features that will be passed to the MHT framework (see Section 5.2.3). The MUSIC algorithm was chosen as it has been shown to work well for the task of speaker segmentation [167, 168] in the past. However, [167, 168] still suffer from the limitation that different speakers cannot share the same azimuth.

This proposed method of simultaneous tracking of F_0 and spatial DoA features approach is also advantageous as it removes a drawback of F_0-only methods. The drawback is that tracks generated by F_0-only methods are not able to span over unvoiced regions. However, with DoA tracking this becomes possible as it can track speakers even when the F_0 is absent. The proposed approach is summarised in Fig. 5.1, and shows how F_0 and DoA information is exploited simultaneously for tracking.

It is shown that when the proposed approach is evaluated on 12 meetings of the AMI corpus using a circular microphone array, that consisted of eight miniature omnidirectional electret microphones, placed in the centre of the meeting participants, it outperforms a BLSTM approach by 14.1% in terms of HR. It has also been shown that the proposed method achieves a reduction of 3.9% in the FAR and 16.3% in the MH rate when compared against a MHT approach that relies on only the F_0 estimates. This work was also published in the following paper [2].

5.2 Proposed Method

5.2.1 Spectral Peak Detector

In this work, all the harmonics of voiced speech are tracked so that overlapping speech can be considered. This is in contrast to existing work, e.g. [5], where only the most predominant F_0 is tracked. The harmonic estimates of voiced speech are obtained using a spectral peak detector to generate Y peak detections, $\Phi_t = \{\phi_{t,1}, \cdots, \phi_{t,Y}\}$, for each time frame, t, from the STFT of the input signal. Only the reliable peak detections, $\hat{\Phi}_t = \{\hat{\phi}_{t,1}, \cdots, \hat{\phi}_{t,J}\}$, are retained where the reliability is determined by thresholding each peak's amplitude by ξ as described in Section 4.2.1.1 on Page 38.

5.2.2 Harmonic Subset Generation

The method in [3, 4] is used to create subsets, $\mathbf{f}_{t,m}$, which are vectors containing reliable peak detections, $\hat{\phi}_t$, that are harmonically related. Multiple subsets are considered to allow tracking of more than one F_0 track in the presence of overlapping speech. It is still possible, however, for subsets to conflict if the F_0 of a subset is a harmonic or subharmonic of the F_0 of any other subset. Thus, at each time frame, an iterative selection process is utilised to select the best non-conflicting subsets. This selection process will stop tracks from being created for the subharmonic or for integer multiples of the F_0. M selected F_0 observations, $\mathbf{F}_t = \{\mathbf{f}_{t,0}, \mathbf{f}_{t,1}, \cdots, \mathbf{f}_{t,M}\}$, at each time frame, t, are then used to form tracks. However, the association of an observation with the F_0 of a specific speaker is unknown *a priori*.

5.2.3 Direction of Arrival Estimator

The MUSIC algorithm [80, 166] is a popular super-resolution technique for narrowband DoA estimation. There currently exists a strongly developed framework for the algorithm due to its popularity and for this reason, it is used in this work to estimate the DoA of multiple speakers. At each time frame, t, DoA observations are generated along with their relative peak amplitude.

5.2.4 Direction of Arrival Selection

The first J DoA detections, $\mathbf{d}_t = \{d_{t,0}, d_{t,1}, \cdots, d_{t,V}\}$, with the largest peak amplitudes are considered. Then, of these detections, only those with an amplitude greater than a threshold are considered reliable measurements. The threshold is chosen as β per cent of the mean value of the next B DoA detections in descending order of amplitude.

5.2.5 Kalman Filter

At each time frame, t, every DoA observation, $d_{t,v}$, is appended to every F_0 observation, $\mathbf{f}_{t,m}$, to create a new combined observation,

$$\mathbf{z}_{t,n} = [\mathbf{f}_{t,m}, d_{t,v}]^T \ , \tag{5.1}$$

for every possible combination. The Kalman filter state for the i-th speaker, modelled by the F_0 and the DoA using x_f and x_d respectively, is

$$\mathbf{x}_{i,t} = \mathbf{x}_{i,t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \ , \tag{5.2}$$

where

$$\mathbf{x}_i = \left[x_f, x_\mathrm{d}\right]^T, \quad \mathbf{Q}_t = \mathrm{diag}\left(\sigma^2_{w_f}, \sigma^2_{w_d}\right) \ . \tag{5.3}$$

The state at t evolves from the state at $t-1$ by a process noise term with a covariance $\mathbf{Q}_t \in \mathbb{R}^{2\times 2}$, and variances, $\sigma^2_{w_f}$ and $\sigma^2_{w_d}$ for the F_0 and DoA respectively. The observation, $\mathbf{z}_{t,n}$, associated with the i-th speaker is modelled conditional on $\mathbf{x}_{i,t}$ as

$$\begin{aligned} \mathbf{z}_{t,n} = \mathbf{H}_{t,n}\mathbf{x}_{i,t} + \mathbf{v}_t, \quad & \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \ , \\ & \mathbf{R}_t = \mathrm{diag}\left(\sigma^2_{v_f}, \cdots, \sigma^2_{v_f}, \sigma^2_{v_d}\right) \ , \end{aligned} \tag{5.4}$$

where the covariance, $\mathbf{R}_t \in \mathbb{R}^{N_t \times N_t}$, and the variances, $\sigma^2_{v_f}$ and $\sigma^2_{v_d}$ for the F_0 and DoA respectively, model the uncertainty in the observations, and

$$\mathbf{H}_{t,n} = \begin{bmatrix} h_0 & h_1 & \cdots & h_K & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T \ , \tag{5.5}$$

where h_k is the harmonic order of the associated harmonic observation plus one.

In the Kalman filter, a prediction step is performed where the state, $\hat{\mathbf{x}}_{i,t|t-1}$, predicted for time frame, t is given by

$$\hat{\mathbf{x}}_{i,t|t-1} = \hat{\mathbf{x}}_{i,t-1|t-1} \ , \tag{5.6}$$

$$\mathbf{P}_{i,t|t-1} = \mathbf{P}_{i,t-1|t-1} + \mathbf{Q}_t \ . \tag{5.7}$$

The updated F_0 and DoA estimate, $\hat{\mathbf{x}}_{i,t|t}$, and updated estimation covariance, $\mathbf{P}_{i,t|t}$, are given by

$$\hat{\mathbf{x}}_{i,t|t} = \hat{\mathbf{x}}_{i,t|t-1} + \mathbf{k}_{i,t}(\mathbf{z}_{t,n} - \mathbf{H}_{t,n}\hat{\mathbf{x}}_{i,t|t-1}) \ , \tag{5.8}$$

$$\mathbf{P}_{i,t|t} = (\mathbf{I} - \mathbf{k}_{i,t}\mathbf{H}_{t,n})^2 \mathbf{P}_{i,t|t-1} + \mathbf{k}_{i,t}\mathbf{R}_t\mathbf{k}_{i,t}^T \ . \tag{5.9}$$

The Kalman gain, $\mathbf{k}_{i,t}$, is a row vector given by [94, 98]

$$\mathbf{k}_{i,t} = \mathbf{P}_{i,t|t-1}\mathbf{H}_{t,n}^T\mathbf{S}_{i,t}^{-1} \ , \tag{5.10}$$

where innovation variance, $\mathbf{S}_{i,t}$, is a matrix given by

$$\mathbf{S}_{i,t} = \mathbf{H}_{t,n}\mathbf{P}_{i,t|t-1}\mathbf{H}_{t,n}^T + \mathbf{R}_t \ . \tag{5.11}$$

Therefore, the error between measurement and prediction follows as

$$\mathbf{e}_{i,t|t} = \mathbf{z}_{t,n} - \mathbf{H}_{t,n}\hat{\mathbf{x}}_{i,t|t} \ . \tag{5.12}$$

5.2.6 Multiple Hypothesis Tracking

At each time frame, t, the prediction step is always executed. However, the update step is only performed when new observations emerge. Depending on whether F_0-only, DoA-only or F_0 and DoA observations are observed at time frame, t, three possible tracks can be generated: a track-only containing the F_0 observation; a track-only containing the DoA observation and a track containing both the F_0 and DoA observations. Accordingly, there are also three possible update steps to be considered on all existing tracks: updating only the F_0; updating only the DoA and updating both the DoA and F_0 of a track. There is also the possibility that all of the observations are FAs and, therefore, for each track, only the prediction step needs to be executed.

To reduce the computational complexity, gating is applied where a track is only updated by an observation if $\tilde{e}_{i,t|t}$ is below a threshold, ζ, otherwise the update is rejected. $\tilde{e}_{i,t|t}$ is defined as the mean of the absolute values of estimation error, $\mathbf{e}_{i,t|t}$ for time frame, t. This is because the observation is considered to be too far from the predicted estimate and, therefore, unlikely to have originated from the active track. As there are many possible ways to update the tracks with the same observations, a MHT approach is utilised [3, 97] to resolve the uncertainty and generate hypothesis tracks, T_i.

Y	J	β	B	ξ	σ_{w_f}, σ_{w_d}	σ_{v_f}, σ_{v_d}	ζ
20	2	0.85	20	1.0×10^6	0.001, 300	0.001, 100	30

Table 5.1: Parameter Setting for the proposed 'DoA-&-F_0' method; the 'F_0-Only' baseline and the 'DoA-Only' baseline.

5.2.6.1 Maximum weighted clique

The MHT approach uses a MWC method [3, 148, 149], as previously described in Section 4.2.1.4 on Page 42, to find the most likely set of tracks that contain no conflicts, i.e. no two tracks contain the same observation. An undirected graph, $G = (V, E)$, is created after each time frame, t, where the L hypothesis tracks are represented by the node set $V = \{T_0, T_1, \cdots, T_L\}$, and the set of edges is $E \subseteq V \times V$. A clique is a subgraph of G with pairwise adjacent vertices, meaning that all pairwise vertices T_i and T_j are connected by an edge (i, j). Each node is assigned a score, w_i, which is the average of $\tilde{e}_{i,t|t}$ for all previous estimation errors. The MWC solution is the clique that maximises the following optimisation problem

$$\begin{aligned} \max\ f(q) &= \sum_{i=0}^{L} w_i q_i \ , \\ \text{s.t. } q_i + q_j &\leq 1, \forall (i,j) \in \bar{E} \ , \\ q_i &\in \{0,1\}, \text{ for } i \in \{0, 1, \cdots L\} \ , \end{aligned} \tag{5.13}$$

where $q_i = 1$ if the node T_i belongs to the clique and $q_i = 0$ otherwise. In this case $\bar{E}$ denotes the edge set of the complementary graph of G. The pruning technique then operates by calculating the MWC at every time frame and discarding all other tracks.

5.3 Comparative Evaluation

5.3.1 Experimental Setup

The proposed 'DoA-&-F_0' method is compared against three baselines. The first two baselines employ simplified versions of the proposed method to track only the F_0 and DoA separately, and are called 'F_0-Only' and 'DoA-Only' respectively. The parameters used in the experiments described in this Section are given in Table 5.1 for the 'DoA-&-F_0' method, the 'F_0-Only' and the 'DoA-Only' baseline approaches. The selection of these parameters used both physiological constraints of σ_{w_f} and σ_{w_d}: process noise variances (for the F_0 and DoA respectively), σ_{v_f} and σ_{v_d}: observation noise variances (for the F_0 and DoA respectively) and empirical tuning of Y: maximum number allowed spectral peaks, J: number of

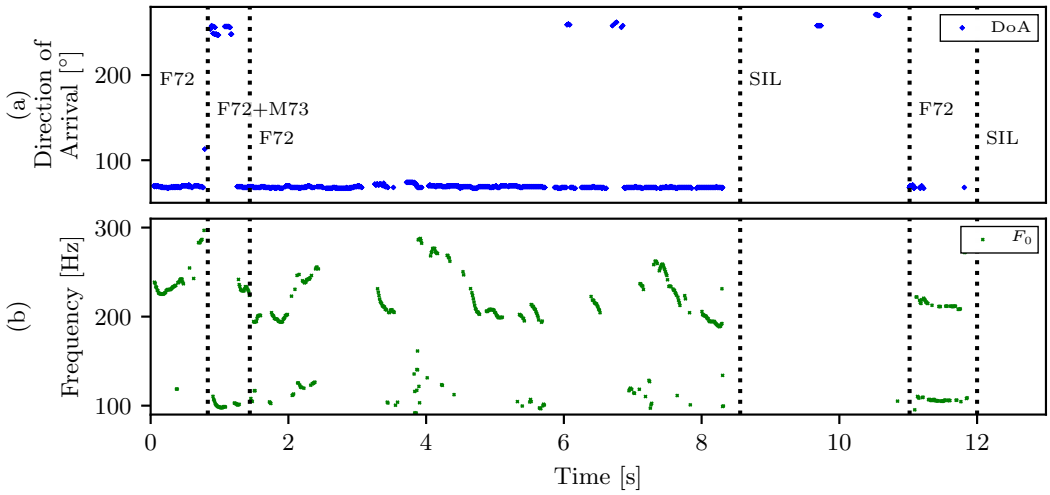


Figure 5.2: An illustrative example of part of a meeting from the AMI corpus. (a) DoA estimates from a circular array using MUSIC. (b) F_0 estimates from a single distance microphone (SDM) using the method in [3].

DoA detections selected, β: DoA reliability threshold, B: number of DoA detections compared using the β threshold, ξ: voiced frame threshold, ζ: VAD region merge threshold. This empirical tuning was achieved through an exhaustive grid search [124] on the development set of the AMI corpus.

The third baseline method is the BLSTM in [41], which represents a state-of-the-art deep learning approach. The BLSTM network was trained for 1000 epochs on the AMI corpus [103] using SincNet learnable features with the configuration published in [147]. To make this comparison as valid as possible, a model that was pre-trained on the AMI corpus was used as a baseline for this experiment. The code to run this pre-trained model, provided by Pyannote [151], can be found here [152]. The baseline used for this experiment is the same as the one used in Chapter 4 and the training process is described in Section 4.3.1.3.

5.3.2 Evaluation Metrics

To evaluate the performance of the 'DoA-&-F_0' method, the following metrics have been defined in Section 4.3.1.4 on Page 49. A 'HIT' is when a speaker change has been detected once. A 'MISS' is when a speaker change has not been detected and a 'MH' is when a speaker change has been detected multiple times within a time collar applied around every ground-truth speaker change in order to account for possible inaccuracies [9]. Finally, a FA is when a detection falls outside of any speaker change collars. The HR is given by (4.14), the false alarm (FA) rate is given by (4.15) and the MH rate is given by (4.16). An illustration of these metrics is given in Fig. 4.7 on Page 49.

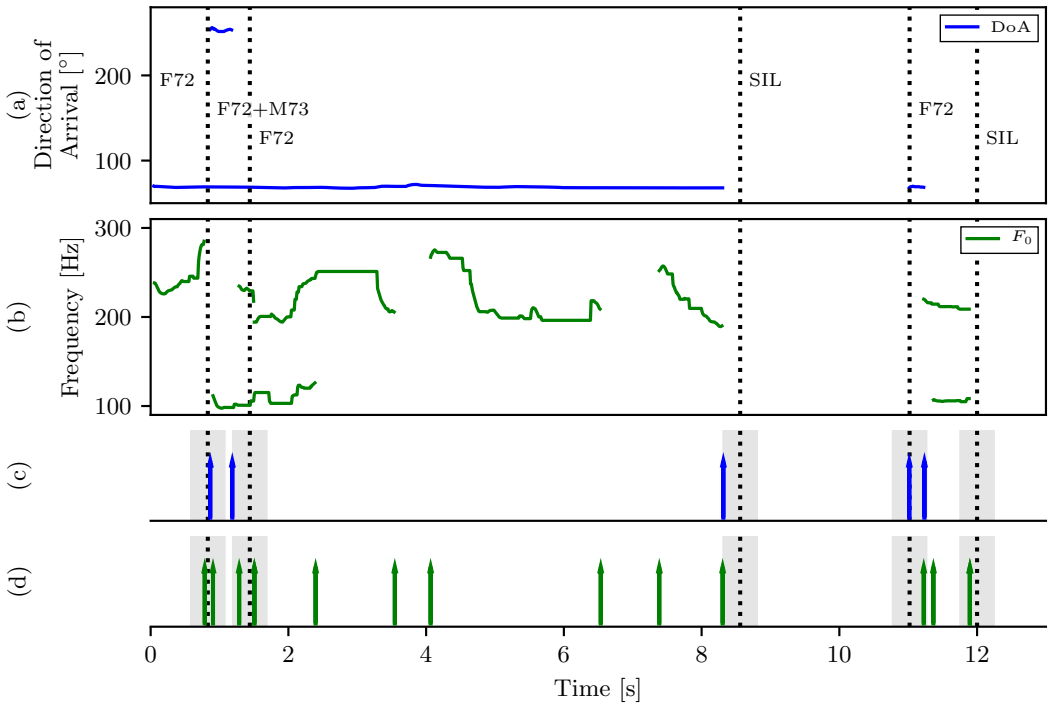


Figure 5.3: An illustrative example of part of a meeting from the AMI corpus. (a) DoA tracked alone and (b) F_0 tracked alone both using [3]. (c) 'DoA-Only' segmentation [HIT: 80%, MISS: 20%, MH: 20%, FA:0%] and (d) 'F_0-Only' segmentation [HIT: 100%, MISS: 0%, MH: 60%, FA:55%] where each arrow marks the start or end of a track.

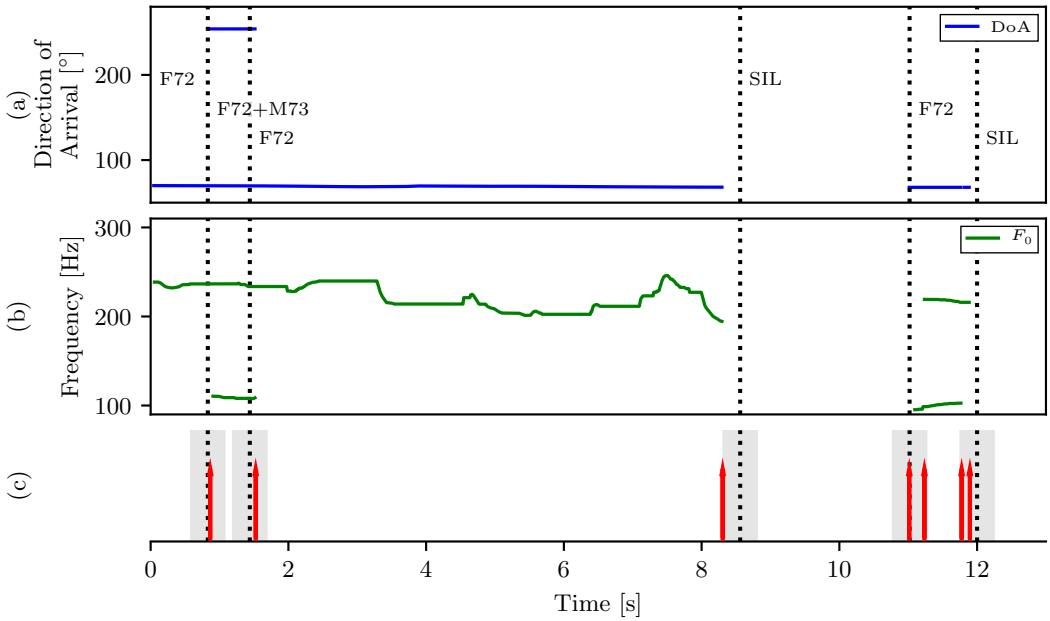


Figure 5.4: An illustrative example of part of a meeting from the AMI corpus. (a) and (b) The proposed method where both the DoA and F_0 are tracked together. (c) Proposed 'DoA-&-F_0' segmentation [HIT: 100%, MISS: 0%, MH: 40%, FA:0%].

5.3.3 Illustrative Example

An illustrative example taken from the AMI meeting 'EN2002c' between 140 and 153 s, is shown in Fig. 5.2, on Page 69, Fig. 5.3 on Page 70 and Fig. 5.4 on Page 70. Fig. 5.4(c) highlights the performance improvements of the 'DoA-&-F_0' method against the 'DoA-Only' and the 'F_0-Only' shown in Fig. 5.3(c) and (d) respectively. The improvement of the 'DoA-&-F_0' method against the 'F_0-Only' is a reduced FAR, although the HR is 100% for both approaches. The improvement in the FAR is a result of the 'DoA-&-F_0' method enabling tracks to span over missing F_0 detections, during for example unvoiced speech activity, while the DoA can still be detected. The proposed method does, on the other hand, improve the HR when compared against the 'DoA-Only'. These missed detections are caused by the early termination of the 'DoA-Only' tracks, e.g. in Fig. 5.3(a) the DoA track between 11.02 and 11.24 s ends earlier than the speaker change at 12.0 s.

5.3.4 Statistical Results

The proposed method was compared against the three baselines across 12 meetings in the AMI corpus (see Section 2.3.5.2) using speech signals from multi-speaker meetings captured using a circular array in naturally reverberant rooms with ambient noise. The circular array was placed in the centre of a table that the participants were sitting around and consisted of eight miniature omnidirectional electret microphones. The results are shown in Table 5.2.

5.3.4.1 Improvements in HIT rate (HR)

Improvements in the HR of 4.7% and 14.1% on average can be seen when the proposed method is compared against the 'DoA-Only' and BLSTM baseline respectively. The 'F_0-Only', however, does perform slightly better in terms of HR when compared against the proposed method. This slight improvement, however, does come at the cost of a high FAR which is to be expected given the illustrative example shown in Fig. 5.2, Fig. 5.3 and Fig. 5.4. It is interesting to note that the BLSTM baseline performs the worst in terms of HR. This may be due to other tracking approaches performing better at detecting overlapping speech events in the spoken backchannel, such as 'um' and 'uh-huh'.

5.3.4.2 Improvements in false alarm rate (FAR)

The results show that an improvement of 3.9% in the FAR can be achieved when compared against the 'F_0-Only'. This is likely due to a reduction in errors caused by unvoiced speech. Unvoiced speech lacks

Meeting	Proposed DoA-&-F_0 System				F_0-Only Baseline System				DoA-Only Baseline System				BLSTM Baseline System			
	HIT	MISS	MH	FA	HIT	MISS	MH	FA	HIT	MISS	MH	FA	HIT	MISS	MH	FA
EN2002a	86.0%	14.0%	47.2%	52.6%	80.3%	19.7%	51.6%	55.4%	80.0%	20.0%	53.9%	64.7%	89.3%	10.7%	71.1%	36.1%
EN2002b	88.2%	11.8%	50.2%	56.8%	85.6%	14.4%	56.2%	60.4%	80.2%	19.9%	55.2%	70.4%	70.0%	30.0%	50.5%	34.5%
EN2002c	81.5%	18.5%	37.2%	65.6%	85.1%	14.9%	53.0%	74.0%	80.0%	20.0%	54.0%	73.8%	72.6%	27.4%	52.7%	45.7%
EN2002d	88.4%	11.6%	51.6%	53.0%	83.0%	17.0%	54.1%	56.3%	80.8%	19.1%	54.5%	66.9%	78.4%	21.6%	61.5%	34.2%
ES2004a	70.6%	29.4%	27.5%	58.5%	77.6%	22.4%	46.8%	68.8%	75.0%	25.0%	53.0%	76.1%	30.9%	69.1%	17.2%	31.5%
ES2004b	75.2%	24.8%	26.8%	73.8%	82.5%	17.5%	49.3%	76.8%	75.8%	24.1%	51.7%	81.3%	31.7%	68.3%	20.3%	41.7%
ES2004c	85.3%	14.7%	34.4%	70.6%	85.2%	14.8%	57.2%	73.8%	77.2%	22.9%	49.4%	79.5%	35.2%	64.8%	15.4%	55.0%
ES2004d	73.8%	26.2%	29.7%	61.2%	81.6%	18.4%	50.9%	65.2%	76.2%	23.8%	50.6%	74.7%	54.4%	45.6%	37.0%	47.4%
IS1009a	81.3%	18.7%	28.2%	68.9%	81.3%	18.7%	57.0%	68.8%	73.5%	26.5%	49.6%	77.0%	86.7%	13.3%	67.2%	45.9%
IS1009b	82.7%	17.3%	38.7%	73.3%	85.1%	14.9%	50.8%	79.8%	72.2%	27.8%	44.0%	80.7%	88.0%	12.0%	70.4%	44.6%
IS1009c	84.9%	15.1%	34.3%	80.5%	78.2%	21.8%	50.7%	85.2%	70.3%	29.7%	40.6%	86.3%	84.3%	15.7%	66.4%	62.1%
IS1009d	76.7%	23.3%	26.2%	68.8%	81.1%	18.9%	49.9%	73.2%	76.2%	23.8%	48.7%	76.0%	83.9%	16.1%	63.4%	45.3%
Std Dev	5.6%	5.6%	8.8%	8.5%	2.6%	2.6%	3.1%	12.1%	3.2%	3.2%	4.2%	5.9%	22.0%	22.0%	20.5%	8.6%
Mean	81.2%	18.8%	36.0%	65.3%	82.2%	17.8%	52.3%	72.0%	76.5%	23.6%	50.4%	75.6%	67.1%	32.9%	49.4%	43.7%

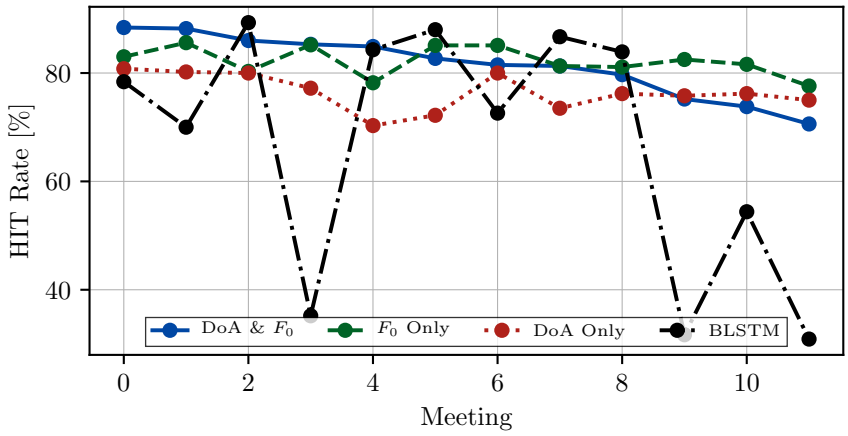
Table 5.2: Performance comparison of the proposed method on 12 multi-speaker meetings in the AMI corpus using a collar of 300 ms compared against the performance achieved by only using DoA or F_0 features alone as well as machine learning BLSTM approach. (*A graphical representation of these results is also given in Fig. 5.5 on Page 73.*)

harmonic structure and causes gaps in the F_0 observations which can be bridged by considering the DoA. Against the 'DoA-Only' method a 10.3% improvement in the FAR can also be seen because the DoA tracks often end early leading to a 'MISS' along with a FA. The process of tracking both the DoA and F_0 features simultaneously reduces the likelihood of tracks ending early. This is due to the F_0 track continuing even when the DoA track has already ended.

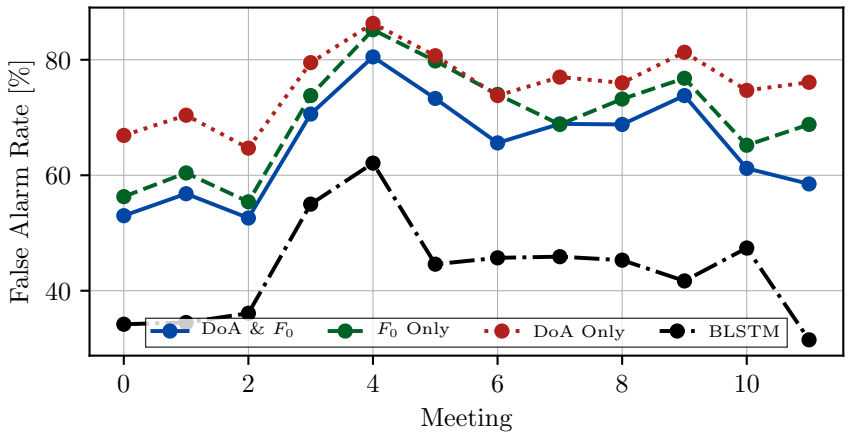
The BLSTM baseline also highlights the important trade-off between the HR and the number of FAs detected. This is mainly due to sensitivity, the more sensitive the system is to detect the speaker changes, the more likely it will be triggered by noise or other common degradations in the recording giving rise to an increase in the FAR. The reverse is also true for a less sensitive system. In this case, the BLSTM baseline does outperform the proposed system in terms of the FAR but also suffers from a 14.1% lower HR. This trade-off, therefore, needs to be considered whenever a system is selected for a particular application.

5.3.4.3 Improvements in the multi-hit rate

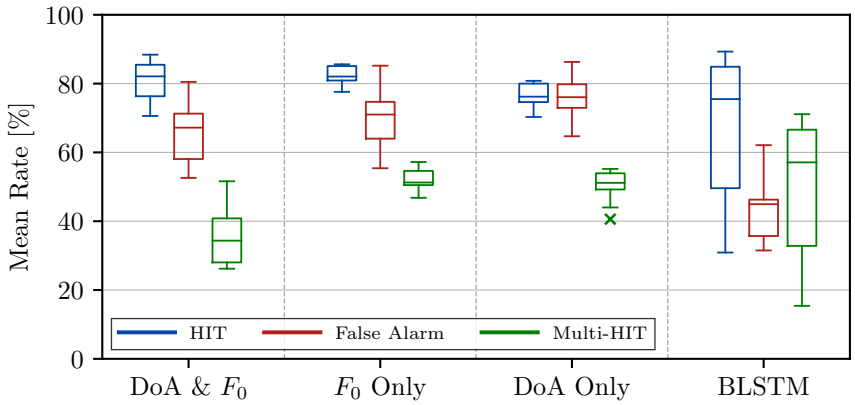
An improvement in the MH rate is achieved by the proposed method over all three baselines. The mean improvements of 16.3% over the 'F_0-Only' and 14.4% over the 'DoA-Only' are likely due to a reduction in the number of tracks being generated. By tracking both the F_0 and DoA together, the proposed system reduces the number of spurious tracks being generated from noise in the measurement data and leads to a lower MH rate. An improvement of 13.4% in the MH rate is also achieved over the state-of-the-art BLSTM baseline.



(a) Comparison of HRs on 12 multi-speaker meetings in the AMI corpus.



(b) Comparison of FARs on 12 multi-speaker meetings in the AMI corpus.



(c) Comparison of mean rates on 12 multi-speaker meetings in the AMI corpus.

Figure 5.5: Performance comparison of the proposed method on 12 multi-speaker meetings in the AMI corpus using a collar of 300 ms compared against the performance achieved by only using DoA or F_0 features alone as well as machine learning BLSTM approach. (*A tabular form of these results is also given in Table 5.2 on Page 72.*)

5.4 Conclusion

In this chapter, it has been shown that MHT of both the DoA and F_0 can lead to an improved speaker segmentation performance over tracking just one of these features alone. A novel method has been proposed that uses a MHT framework to track the F_0 and DoA of multiple speakers simultaneously. The proposed method was evaluated on a well-known AMI corpus of conversational speech and outperformed a BLSTM approach by 14.1% in terms of HR. It has also been shown that the proposed method achieves an improvement of 3.9% in the FAR and 16.3% in the MH rate when compared against a MHT approach that relies on only the F_0 estimates. This work has also been published in the following paper [2].

Chapter 6

A POLYNOMIAL EVD MUSIC APPROACH TO OVERLAPPING SPEAKER SEGMENTATION

6.1 Introduction

Sound source localization is an important task for a multitude of applications, including robot audition [169] and voice-controlled smart devices. DoA estimates are essential in providing angular positional information for localization. In real-world environments, where acoustic scenes are complex and dynamic, DoA estimation can be a challenging problem to solve because of background noise, reverberation, interference and sound source inactivity.

Many DoA estimation approaches have been proposed including time-delay estimation (TDE)-based, beamformer-based and subspace-based methods [22]. The TDE-based method [75] first computes the TDoA for different microphone pairs and uses *a priori* information about the microphone positions to compute the DoAs. However, TDE approaches such as generalized cross-correlation (GCC)-phase-transform (PHAT) cannot cope with multiple sources in reverberant environments [22]. Beamformer-based methods [76, 77] scan the acoustic environment by focusing the microphone array directional pick-up pattern in the directions corresponding to the highest sound intensities. However, [76, 77] require the formation of a large number of steering beam angles for high resolution and is computationally expensive.

In a subspace-based approach such as the MUSIC algorithm [80], the covariance matrix is computed from the received signals. An eigenvalue decomposition (EVD) is then used to decompose the covariance matrix into signal and noise subspaces for DoA estimation. The MUSIC algorithm, however, assumes that the source signals are narrowband and uncorrelated. Consequently, its performance is limited in real-world scenarios involving broadband signals such as speech and correlated sources originating from reverberant environments.

A number of broadband extensions have been proposed for MUSIC [170–172]. Most of these extensions rely on transforming the broadband DoA problem into several narrowband problems. This can be achieved by decomposing the broadband signal into several independent frequency bins [173]. The resulting narrowband signals for each frequency bin or filtered output can then be processed independently, or incoherently. This approach, however, is based on a narrowband signal model and ignores phase coherence across different frequency bins [174] which can lead to errors [175].

When broadband signals such as speech signals are involved, time delays cannot be modelled using phase shifts because e.g. time delays between different microphones need to be explicitly resolved. Consequently, an EVD cannot completely decorrelate the signals and separate the signal and noise subspaces effectively [22]. Instead, the spatio-spectral polynomial (SSP)-MUSIC approach in [15, 16] is based on a broadband signal model. The approach uses polynomial matrices to model the correlations across different microphones and temporal lags, and polynomial eigenvalue decomposition (PEVD) to generate the signal and noise subspaces. SSP-MUSIC has been shown to be robust and effective for non-speech sources in anechoic environments [15, 16].

In this chapter, [15, 16] is extended to sound source localization for speech signals in noisy and reverberant environments. The novel contributions are: (i) proposed enhancements to SSP-MUSIC for sound source localization which include; incorporating a noisy reverberant signal model in the subspace decomposition; modifying SSP-MUSIC to only include the direct-path response in order to reduce the impact of reverberation on localization performance; using SSP-MUSIC to approximate spatial polynomial (SP)-MUSIC for the frequency range of speech; (ii) an analysis on how diffuse noise and reverberation affects the proposed approach; and (iii) a comprehensive evaluation of the proposed method against benchmark algorithms for simulated and real-world recordings.

This chapter shows that a SSP-MUSIC approach for single sound source localization is beneficial at SNR values lower than 5 dB or reverberation time (T60) values larger than 0.7 s as it is more robust to noise and reverberation. It is also shown on real data, taken from the localization and tracking (LOCATA) corpus, that SSP-MUSIC can outperform independent frequency bin (IFB)-MUSIC on real-world signals. This work was also published in the following paper [1].

6.2 Method

In [176], the noisy and reverberant signal, $x_m(n)$, at the m-th microphone for discrete-time sample $n = 0, 1, \ldots, N$, is

$$x_m(n) = \mathbf{h}_m^T \mathbf{s}_0(n) + v_m(n) \ , \tag{6.1}$$

where $\mathbf{h}_m = [h_{m,0}, h_{m,1}, \ldots, h_{m,J}]^T$ is the m-th acoustic channel, which is modelled as a J-th order finite impulse response filter and decomposed into the direct path, $\tilde{\mathbf{h}}_{m,dp}$, early reflections, $\tilde{\mathbf{h}}_{m,er}$, and the late reflections, $\tilde{\mathbf{h}}_{m,lr}$, [28] in the following way

$$\begin{aligned} x_m(n) &= \tilde{\mathbf{h}}_{m,dp}^T \mathbf{s}_0(n) + \tilde{\mathbf{h}}_{m,er}^T \mathbf{s}_0(n) + \tilde{\mathbf{h}}_{m,lr}^T \mathbf{s}_0(n) + v_m(n) \ , \\ &= \tilde{s}_m(n) + \tilde{v}_m(n), \quad m = 1, 2, \ldots, M \ , \end{aligned} \tag{6.2}$$

where $\mathbf{s}_0(n) = [s_0(n), s_0(n-1), \ldots, s_0(n-J)]^T$ is the anechoic speech signal, $v_m(n)$ is additive noise and $[\cdot]^T$ denotes the transpose operator. The noise signals are assumed to be zero-mean, not perfectly coherent with each other and uncorrelated with the source signals [177]. By exploiting the lack of correlation between the late reflections and anechoic speech signal [178] $\tilde{s}_m(n) = \tilde{\mathbf{h}}_{m,dp}^T \mathbf{s}_0(n) + \tilde{\mathbf{h}}_{m,er}^T \mathbf{s}_0(n)$ and $\tilde{v}_m(n) = \tilde{\mathbf{h}}_{m,lr}^T \mathbf{s}_0(n) + v_m(n)$, can be decomposed into the speech and noise components respectively.

6.2.1 Review of Polynomial MUSIC

Assuming direct-path-only propagation in the far-field and a noise-free environment, $v_m(n) = 0$, such that (6.2) simplifies to

$$x_m(n) = f_{\tau_m}(n) * x_0(n) \ , \tag{6.3}$$

where * denotes a linear convolution and $f_{\tau_m}(n)$ is a fractional delay filter [179, 180]. This is required since the m-th relative delay can be fractional, such that

$$f_{\tau_m}(n) = \frac{\sin(\pi(n - \Delta\tau_m))}{\pi(n - \Delta\tau_m)} \ . \tag{6.4}$$

In the narrowband case, $\Delta\tau_m$ is represented by a simple phase shift and those phase shifts are exploited in the MUSIC algorithm. For broadband sources, however, the delays are frequency-dependent phase shifts.

To capture the temporal correlations of the speech signals at different microphones, the space-time covariance matrix is [178]

$$\mathbf{R}_{\mathbf{xx}}(\tau) = \mathbb{E}\{\mathbf{x}(n)\mathbf{x}^T(n-\tau)\} \ , \tag{6.5}$$

where the $(p,q)^{\text{th}}$ element, $r_{pq}(\tau) = \mathbb{E}\{x_p(n)x_q(n-\tau)\}$, is the cross-correlation sequence between microphone p and q for discrete-time shift τ. Concatenating the covariance matrix, $\mathbf{R}_{\mathbf{xx}}(\tau)$, for all

choices of $\tau \in \{-N, \ldots, N\}$, results in a tensor of dimension $M \times M \times (2N+1)$. The z-transform of (6.5) is a polynomial matrix, $\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z) = \sum_{\tau=-\infty}^{\infty} \mathbf{R}_{\mathbf{xx}}(\tau) z^{-\tau}$, which can be decomposed by an iterative PEVD algorithm [181–185] to give

$$\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z) \approx \boldsymbol{\mathcal{U}}(z) \boldsymbol{\Lambda}(z) \boldsymbol{\mathcal{U}}^P(z) \ , \tag{6.6}$$

where the columns of $\boldsymbol{\mathcal{U}}(z)$ are the eigenvectors and the diagonal elements of $\boldsymbol{\Lambda}(z)$ are the eigenvalues. Furthermore, $\boldsymbol{\mathcal{U}}^P(z) = \boldsymbol{\mathcal{U}}^H(1/z^*)$, where $[\cdot]^*$, $[\cdot]^H$ and $[\cdot]^P$ are respectively, the complex-conjugate, Hermitian and para-Hermitian operators.

Thresholding the eigenvalues enables the partitioning of the polynomial matrix into orthogonal signal and noise subspaces, which given appropriate assumptions are associated with $\boldsymbol{\mathcal{U}}_s(z)$ and $\boldsymbol{\mathcal{U}}_v(z)$, respectively. The nullspace of $\boldsymbol{\mathcal{U}}_v(z)$ is probed by the broadband steering vector, which implements fractional delays and is defined as [15]

$$\mathbf{a}_\theta(z) = \begin{bmatrix} A_0(z) & \cdots & A_{M-1}(z) \end{bmatrix}^T , \tag{6.7}$$

for look direction θ. In this case $A_\ell(z)$ is defined as $A_\ell(z) = \sum_{n=-\infty}^{\infty} a_\ell(n) z^{-n}$ where $a_\ell(n) = \text{sinc}((n - \Delta\tau_\ell)T_s)$ and T_s is the sampling period. Generalised from MUSIC, the following quantity,

$$\Gamma_\theta(z) = \mathbf{a}_\theta^P(z) \boldsymbol{\mathcal{U}}_v(z) \boldsymbol{\mathcal{U}}_v^P(z) \mathbf{a}_\theta(z) \ , \tag{6.8}$$

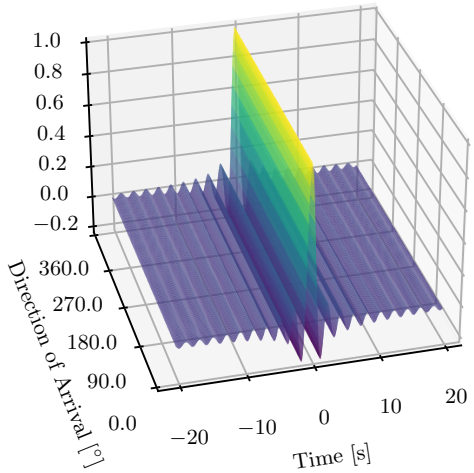
is used to compute the pseudo-spectrogram for SSP-MUSIC [15],

$$\mathbf{P}_{SSP-MU}(\theta, \Omega) = \left. \frac{1}{\Gamma_\theta(z)} \right|_{z=e^{-j\Omega}} , \tag{6.9}$$

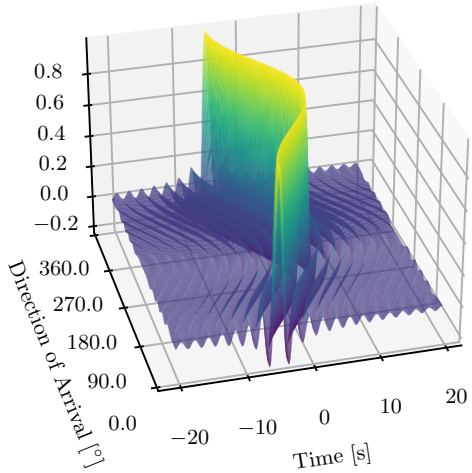
where frequency Ω is obtained by evaluating z on the unit circle. Therefore, the pseudo-spectrogram can localize sources by exploiting the DoAs in the relevant range of frequencies.

6.2.2 Proposed Enhancements for Sound Source Localization

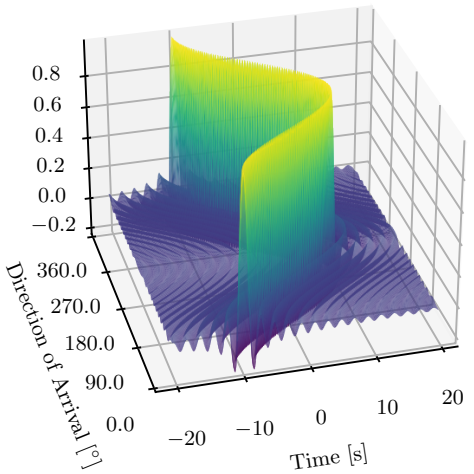
Similar to [178], but unlike [15, 16] which only focuses on non-speech sources in anechoic environments, this work incorporates the noisy reverberant signal model in the subspace decomposition. The difference is that the method in [178] is designed for speech enhancement and incorporates the early reflections that may improve speech intelligibility in some conditions [186, 187]; whereas this chapter focuses on sound source localization which only requires the direct-path component with the greatest amplitude and shortest time delay. Consequently, the largest delay, W, corresponding to the first maximum peak between



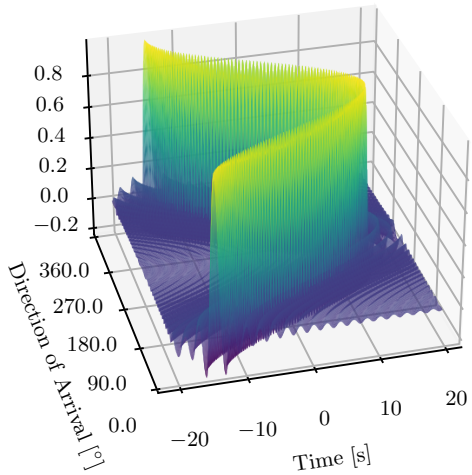
(a) Broadband steering vector for $m = 0$.



(b) Broadband steering vector for $m = 1$.



(c) Broadband steering vector for $m = 2$.



(d) Broadband steering vector for $m = 3$.

Figure 6.1: An illustrative example of a broadband steering vector.

every microphone pair is computed and, therefore, the z-transform of (6.5) is approximated by

$$\tilde{\boldsymbol{\mathcal{R}}}_{\mathbf{xx}}(z) \approx \sum_{\tau=-W}^{W} \boldsymbol{R}_{\mathbf{xx}}(\tau) z^{-\tau} \ , \tag{6.10}$$

using the windowed space-time covariance matrix with dimensions $M \times M \times (2W + 1)$. Furthermore, the introduction of W reduces the number of elements used in PEVD and offers computational improvement. While this window choice includes the largest direct-path propagation delay, some reflections are also

inevitably captured by microphones that are near the sound sources.

Consequently, the PEVD of (6.5) gives [178]

$$\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z) \approx \left[\boldsymbol{\mathcal{U}}_{\tilde{s}}(z) \middle| \boldsymbol{\mathcal{U}}_{\tilde{v}}(z)\right] \left[\begin{array}{c|c} \boldsymbol{\Lambda}_{\tilde{s}}(z) & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{\Lambda}_{\tilde{v}}(z) \end{array}\right] \left[\begin{array}{c} \boldsymbol{\mathcal{U}}^{\boldsymbol{P}}_{\tilde{s}}(z) \\ \hline \boldsymbol{\mathcal{U}}^{\boldsymbol{P}}_{\tilde{v}}(z) \end{array}\right], \tag{6.11}$$

where $\{.\}_{\tilde{s}}$ and $\{.\}_{\tilde{v}}$ represent the orthogonal signal and noise subspace components. The speech subspace comprises anechoic speech convolved with the direct path and some 'leaked' early reflections while the noise subspace contains ambient noise, both early and late reflections associated with the reverberant channel.

To better understand the space-time covariance matrix, $\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z)$, an example is given in Section 6.2.2 on Page 81 for the same frame shown in Fig. 6.7. The resulting eigenvalues, $\boldsymbol{\Lambda}(z)$, generated from the PEVD can be seen in Fig. 6.2(b). Fig. 6.2(b) clearly shows that the rank of $\boldsymbol{\Lambda}_{\tilde{s}}(z)$ is 2 since the eigenvalues are significantly larger than those in $\boldsymbol{\Lambda}_{\tilde{v}}(z)$ which is to be expected given that there are two active speakers in this frame.

To cope with the infinite temporal support of the sinc function in (6.7), tapered windows have been proposed for truncation [180]. An illustrative example of a broadband steering vector is given in Fig. 6.1 on Page 79. In this chapter, the Hamming window defined by

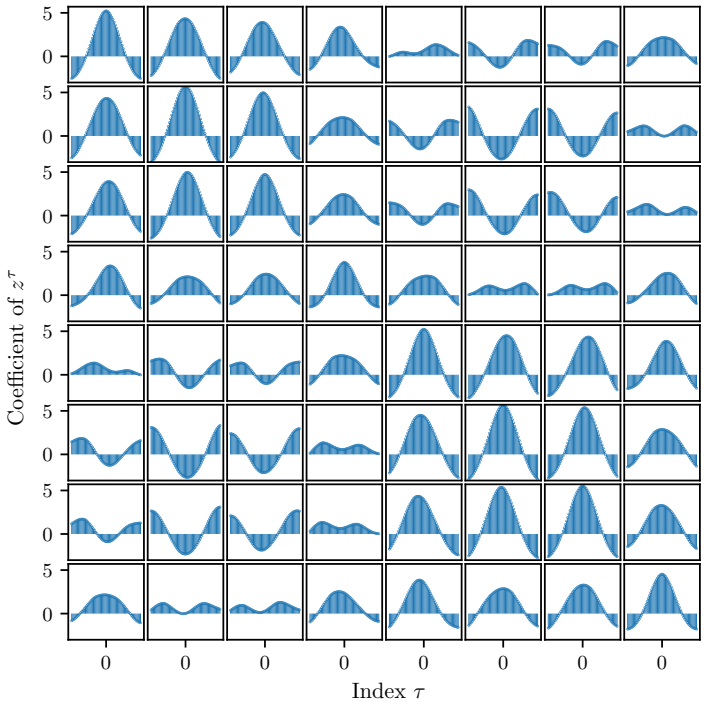
$$w_{L,\text{Hamm}}(n) = (0.54 - 0.46\cos\left(\frac{\pi n}{2L}\right))w_{L,\text{rect}}(n), \quad \text{where} \quad w_{L,\text{rect}}(n) = \begin{cases} 1, & |n| \leq L \\ 0, & |n| > L \end{cases}, \tag{6.12}$$

is used and L is the length of the truncated sinc function.

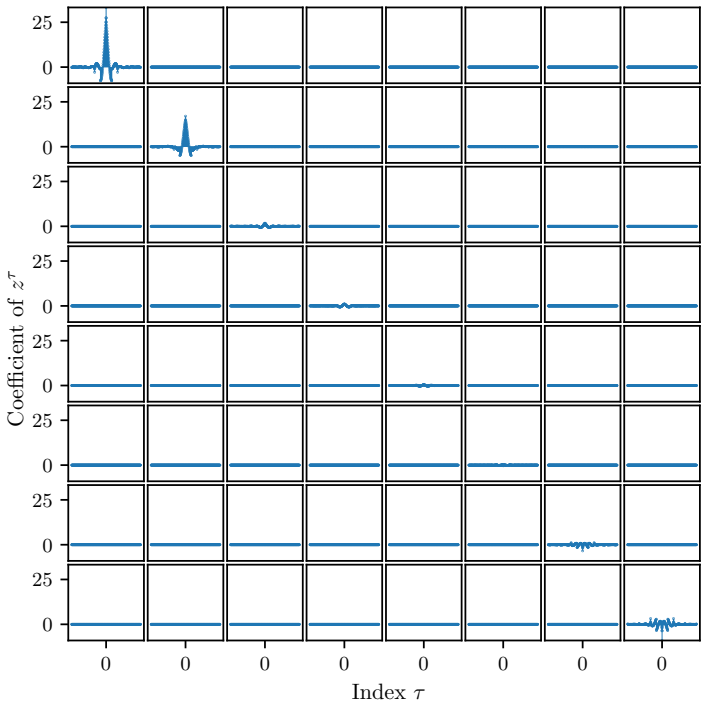
To compute the DoAs only, the pseudo-spectrogram in (6.9) is integrated over Ω. For K discrete points evaluated on the unit circle, the spatial-only pseudo-spectrum is approximated by

$$\hat{\mathbf{P}}_{SSP-MU}(\theta) = \frac{1}{K}\sum\nolimits_{k=0}^{K-1} \mathbf{P}_{SSP-MU}(\theta, \Omega_k)\ , \tag{6.13}$$

where $\Omega_k = \frac{2\pi}{K}k$ is the k-th frequency bin. The whole frequency range is considered in SP-MUSIC [15]. However, in this work, only Ω_k in the relevant frequency range of speech (100 Hz to 4000 Hz) [188] is used in (6.13). A peak detection algorithm [166] is used to estimate the DoAs from (6.13).



(a) A space-time covariance matrix, $\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z)$, that captures the temporal correlations of the speech signals at different microphones.



(b) A rank-2 matrix, $\boldsymbol{\Lambda}(z)$, where the diagonal elements are the eigenvalues

Figure 6.2: $\boldsymbol{\mathcal{R}}_{\mathbf{xx}}(z)$ and $\boldsymbol{\Lambda}(z)$ matrices for illustrative example in Fig. 6.7(a) and (b) on Page 88.

6.3 Experimental Setup

The sequential matrix diagonalisation (SMD) [189] method was used to perform an iterative PEVD as it has been shown to give a higher resolution for SSP-MUSIC than sequential best rotation algorithm (SBR2) [16]. The proposed approach is benchmarked against IFB-MUSIC [166], with MUSIC applied to each frequency bin independently to estimate the DoAs.

6.3.1 Evaluation Metrics

The performance of SSP-MUSIC and IFB-MUSIC is evaluated using the following metrics (more details are given in Section 2.3.5.1 on Page 16). A 'HIT' is when a sound source (speaker) has been detected once within a $\pm\xi$ collar applied around the ground-truth azimuth. A 'MISS' is when a speaker change has not been detected within this collar and a FA is when a detection falls outside of a ground-truth azimuth collar. The HR and FAR are, therefore, defined as,

$$\text{HR} = \frac{\text{HITs}}{\text{HITs} + \text{MISSs}} \text{ expressed as } \%, \tag{6.14}$$

$$\text{FAR} = \frac{\text{FAs}}{\text{HITs} + \text{FAs}} \text{ expressed as } \%. \tag{6.15}$$

respectively wherein this work the collar ξ is set to 15°. To further evaluate the accuracy of the DoA estimates, the absolute errors for all the HITs are shown in the form of boxplots.

6.3.2 Simulated Data Generation

In this work, the performance of SSP-MUSIC is first evaluated on data generated using a simulated room of dimensions $3 \times 3 \times 2$ m [166]. A uniform circular array of 8 microphones, with a diameter of 4.2 cm, is positioned in the centre of the room. Two experiments are run using this scenario. Exp-1 evaluates the performance when a single active speaker is placed at a distance of 1.5 m from the centre of the array at an angle of 50° where the anechoic speech used was a 3 s recording taken from the LOCATA corpus [22] (Task 1, Recording 1). Exp-2 evaluates the performance for two speakers where the anechoic speech is taken from LOCATA (Task 2, Recording 1) and the sources are placed at the same distance as Exp-1 but at angles of 70° and 230°.

It should be noted that, in this work, only DoA estimates for frames that are known to contain speech

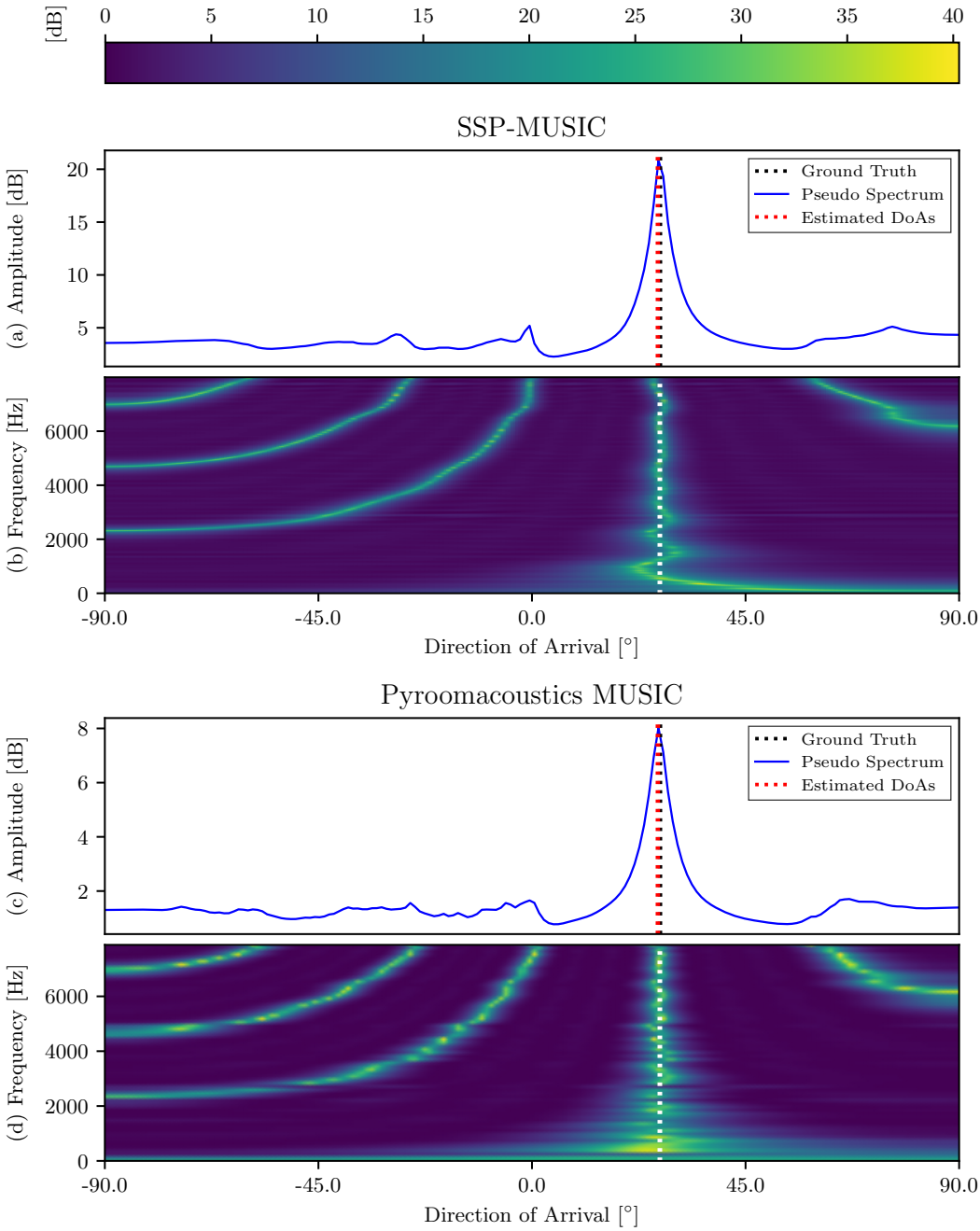


Figure 6.3: Illustrative example using white Gaussian noise as the sound source. (a) pseudo-spectrum of SSP-MUSIC, (b) pseudo-spectrogram of SSP-MUSIC, (c) pseudo-spectrum of IFB-MUSIC, (d) pseudo-spectrogram of IFB-MUSIC.

activity are evaluated. This information is provided by an oracle VAD given in LOCATA. The oracle VAD labelling is also used to determine the number of active speakers for both IFB-MUSIC and SSP-MUSIC.

6.4 Results

To evaluate the performance of SSP-MUSIC, it is compared against the `pyroomacoustics` implementation of IFB-MUSIC [166].

6.4.1 Illustrative Example: One White Gaussian Noise Sound Source

Fig. 6.3 on Page 83 compares SSP-MUSIC against the IFB-MUSIC for a single white Gaussian noise sound source that has been convolved with a broadband steering vector.

6.4.2 Exp-1: One Static Speaker

In this experiment, a comparison is carried out for when only 1 static talker is active. Table 6.1 shows the impact of diffuse white Gaussian noise on the accuracy of the DoA estimates. It can clearly be seen that at SNR conditions lower than 5 dB the performance of IFB-MUSIC has lower HR and a higher FAR when compared to SSP-MUSIC. This is to be expected as it is well known that subspace methods, such as MUSIC, suffer from the so-called 'threshold effect', which results in a degradation both in terms of resolution and precision at low SNR values [190].

Table 6.1 goes on to show the degradation that occurs when the T60 is increased. It can be seen that SSP-MUSIC performs better when the T60 value is large. This robustness to reverberation is likely a result of approximation defined in (6.10) which is one of the proposed enhancements and forces SSP-MUSIC to mainly consider the direct-path component and only allows a few reflections to be additionally captured. Fig. 6.5(a) and (c) show the estimates' accuracy by highlighting the absolute errors of all the estimates considered to be HITs.

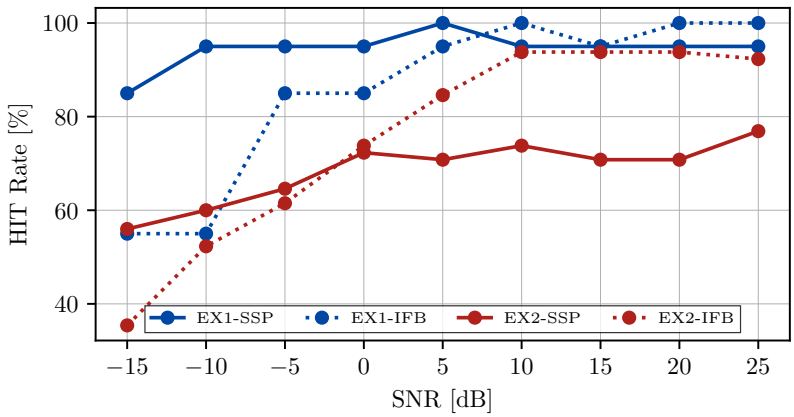
6.4.3 Exp-2: Two Static Speakers

In this second experiment, a comparison is carried out for the situation pertaining to 2 active talkers. In a similar manner to Exp-1, Table 6.1 shows that as the diffuse white Gaussian noise increases, so do the errors in the DoA estimates where IFB-MUSIC is more adversely affected.

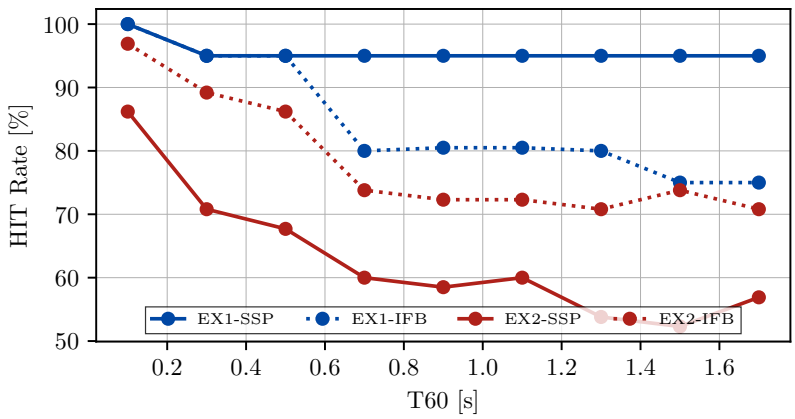
An illustrative example, when the SNR is -15 dB and the T60 is 0.25 s, is also given in Fig. 6.6(a) to highlight the performance improvement of SSP-MUSIC at low SNR values. A second example is also given, where the SNR is 25 dB and the T60 is 1.7 s, in Fig. 6.6(b) to show the performance improvement of SSP-MUSIC at high T60 values.

Task		Exp-1				Exp-2			
Algorithm		SSP-MUSIC		IFB-MUSIC		SSP-MUSIC		IFB-MUSIC	
Metric		HR	FAR	HR	FAR	HR	FAR	HR	FAR
SNR [dB]	-15	85.0	15.0	55.0	45.0	56.9	43.1	35.4	64.6
	-10	95.0	5.0	55.0	45.0	60.0	40.0	52.3	46.9
	-5	95.0	5.0	85.0	15.0	64.6	35.4	61.5	38.5
	0	95.0	5.0	85.0	15.0	72.3	27.7	73.8	26.2
	5	100.0	0.0	95.0	5.0	70.8	29.2	84.6	15.4
	10	95.0	5.0	100.0	0.0	73.8	26.2	93.8	6.2
	15	95.0	5.0	95.0	5.0	70.8	29.2	93.8	6.2
	20	95.0	5.0	100.0	0.0	70.8	29.2	93.8	6.2
	25	95.0	5.0	100.0	0.0	76.9	23.1	92.3	7.7
T60 [s]	0.1	100.0	0.0	100.0	0.0	86.2	13.8	96.9	3.1
	0.3	95.0	5.0	95.0	5.0	70.8	29.2	89.2	10.8
	0.5	95.0	5.0	95.0	5.0	67.7	32.3	86.2	13.8
	0.7	95.0	5.0	80.0	20.0	60.0	40.0	73.8	26.2
	0.9	95.0	5.0	85.0	15.0	58.5	41.5	72.3	27.7
	1.1	95.0	5.0	85.0	15.0	60.0	40.0	72.3	27.7
	1.3	95.0	5.0	80.0	20.0	53.8	46.2	70.8	29.2
	1.5	95.0	5.0	75.0	25.0	52.3	47.7	73.8	26.2
	1.7	95.0	5.0	75.0	25.0	56.9	43.1	70.8	29.2

Table 6.1: Comparison of HRs and false alarm rates for Exp-1 and Exp-2.
(*A graphical representation of these results is also given in Fig. 6.4.*)

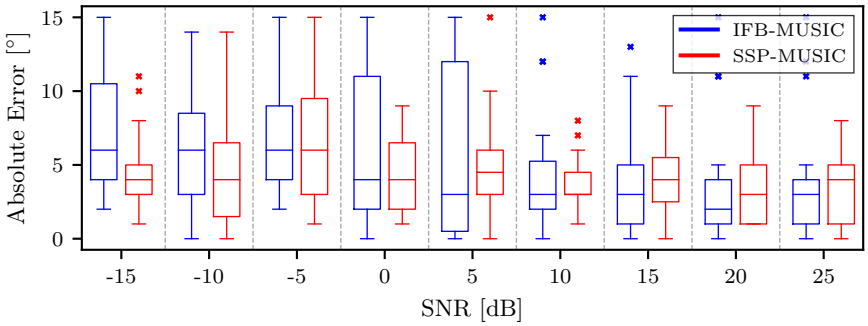


(a) Effect of different SNR values on the HRs for Exp-1 and Exp-2

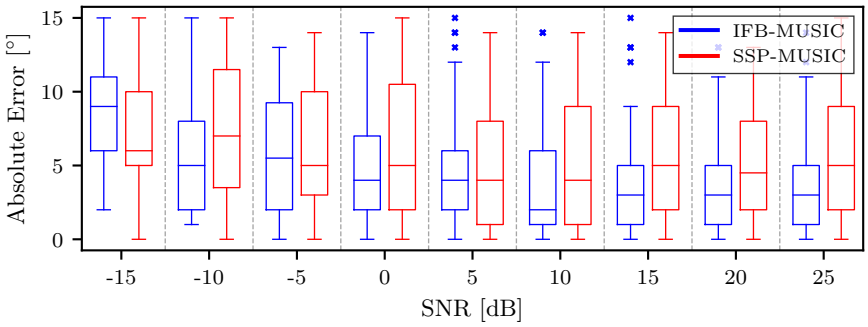


(b) Effect of different T60 values on the HRs for Exp-1 and Exp-2

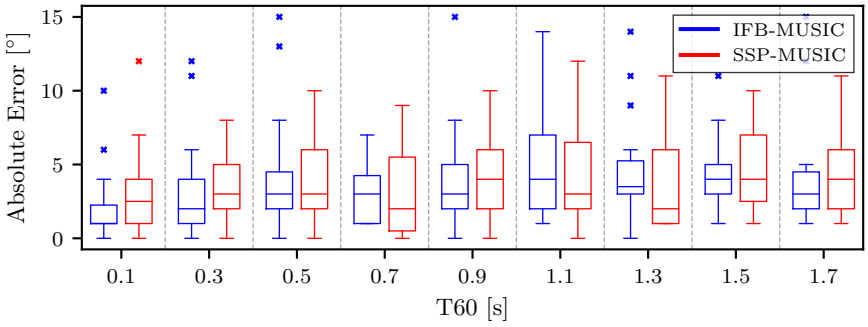
Figure 6.4: Comparison of HRs for Exp-1 and Exp-2.
(*A tabular form of these results is also given in Table 6.1.*)



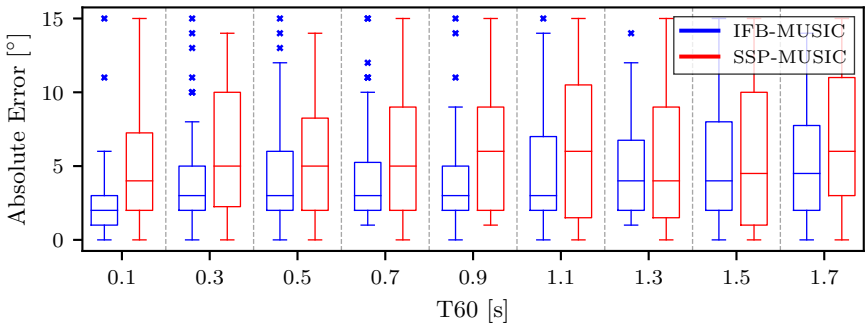
(a) Exp-1 performance when the T60 is 0.25 s and the SNR is varied.



(b) Exp-2 performance when the T60 is 0.25 s and the SNR is varied.

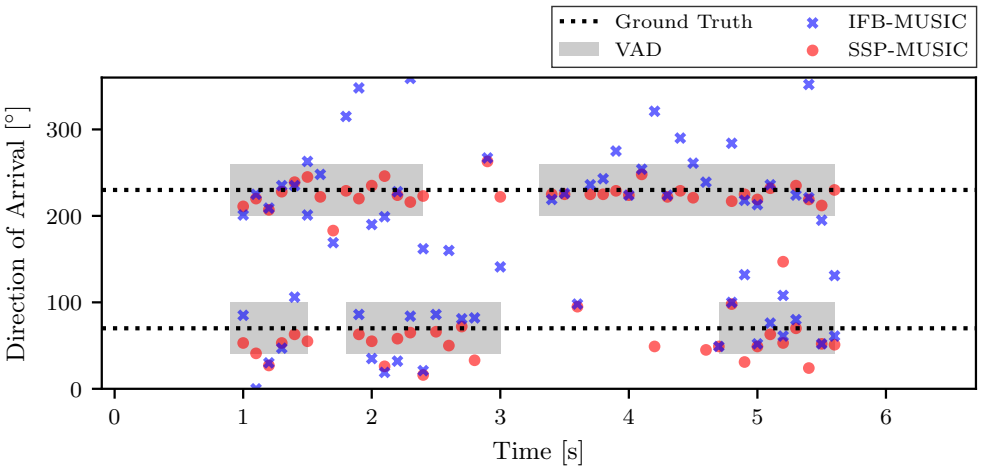


(c) Exp-1 performance when the SNR is 15 dB and the T60 is varied.

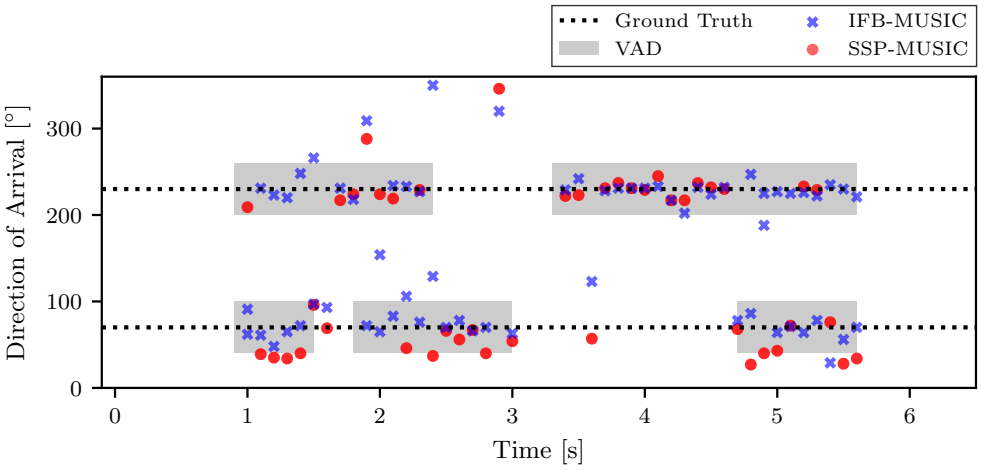


(d) Exp-2 performance when the SNR is 15 dB and the T60 is varied.

Figure 6.5: Comparison of absolute errors of all HITs.



(a) Performance when SNR is -15 dB and the T60 is 0.25 s.



(b) Performance when SNR is 25 dB and the T60 is 1.7 s.

Figure 6.6: Two illustrative examples of 2 active sources in a simulated room.

Fig. 6.7(b) and (d) compares the two pseudo-spectrograms of both SSP-MUSIC and IFB-MUSIC for a single 100 ms frame. The final DoA estimates have an average absolute error of 1.5° and 4° for SSP-MUSIC and IFB-MUSIC respectively and the SSP-MUSIC spectrogram has far more accurate components across frequencies when compared with IFB-MUSIC.

The dynamic range of the amplitudes of IFB-MUSIC is also much smaller than SSP-MUSIC which could lead to numerical issues with the peak detection algorithm when it is run on the pseudospectra of IFB-MUSIC. It can also be observed in Fig. 6.7(a) and (c) that the resolution of the pseudo-spectrum for

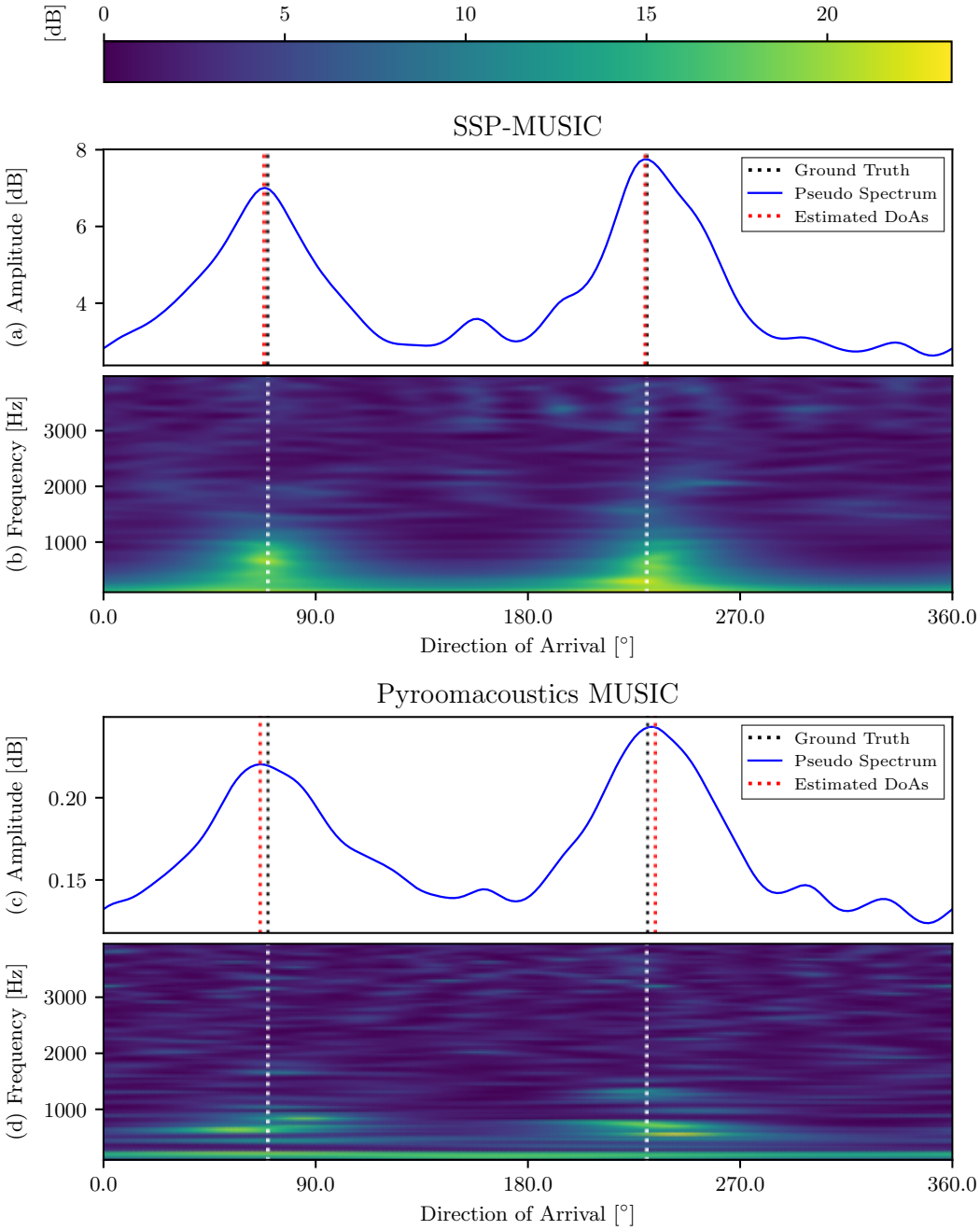


Figure 6.7: Illustrative example of a 100 ms frame from Exp-2 (SNR: -10 dB, T60: 0.25 s). (a) pseudo-spectrum of SSP-MUSIC, (b) pseudo-spectrogram of SSP-MUSIC, (c) pseudo-spectrum of IFB-MUSIC, (d) pseudo-spectrogram of IFB-MUSIC.

SSP-MUSIC was not as sharp as IFB-MUSIC. This is likely due to the fact that fractional delay filters implementing the time delays associated with different angles are not accurate across the entire frequency range [179].

Method		SSP-MUSIC		IFB-MUSIC	
Metric		HR	FAR	HR	FAR
Recording	1	90.0	10.0	95.0	5.0
	2	64.7	35.3	67.6	32.4
	3	95.1	4.9	75.6	24.4

Table 6.2: Comparison of HR and FAR for both SSP-MUSIC against IFB-MUSIC on the first 3 LOCATA recordings for Task 1.

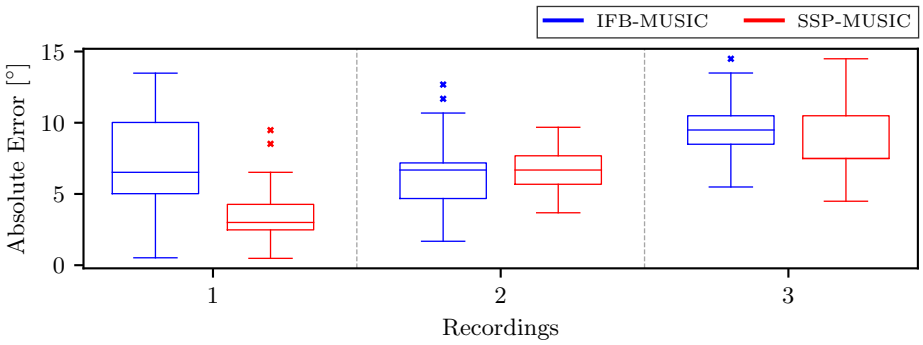


Figure 6.8: Performance comparison of IFB-MUSIC against SSP-MUSIC across Task 1 LOCATA recordings.

6.4.4 Exp-3: LOCATA Task 1

To validate SSP-MUSIC as a good alternative to IFB-MUSIC, both algorithms are compared for 3 recordings from Task 1 of the LOCATA challenge where the ground-truth DoAs are provided by an optical tracking system, *OptiTrac*. It should also be noted that the LOCATA challenge [22] uses IFB-MUSIC as a baseline making it a useful benchmark for this experiment.

In this experiment, real-world audio signals that were captured from an 8-microphone non-uniform circular array, selected from an Eigenmike, were used for the evaluation. As the recordings were measured in a real environment, they contained low-level background noise along with reverberation where the T60 of the room was about 0.5 s.

Table 6.2 shows that on average a 3.9% better HR and a 3.9% lower FAR can be achieved by SSP-MUSIC when compared against IFB-MUSIC on the 3 recordings studied. Fig. 6.8 shows that the accuracy of the DoA estimates given by SSP-MUSIC is better or the same in terms of mean absolute error when compared against IFB-MUSIC. It should be noted that, as the SNR and T60 values are low across all these recordings, it is not expected that great improvements in the performance will be achieved. This result, however, still illustrates the benefits of incorporating a broadband signal model for reverberant speech in the subspace decomposition.

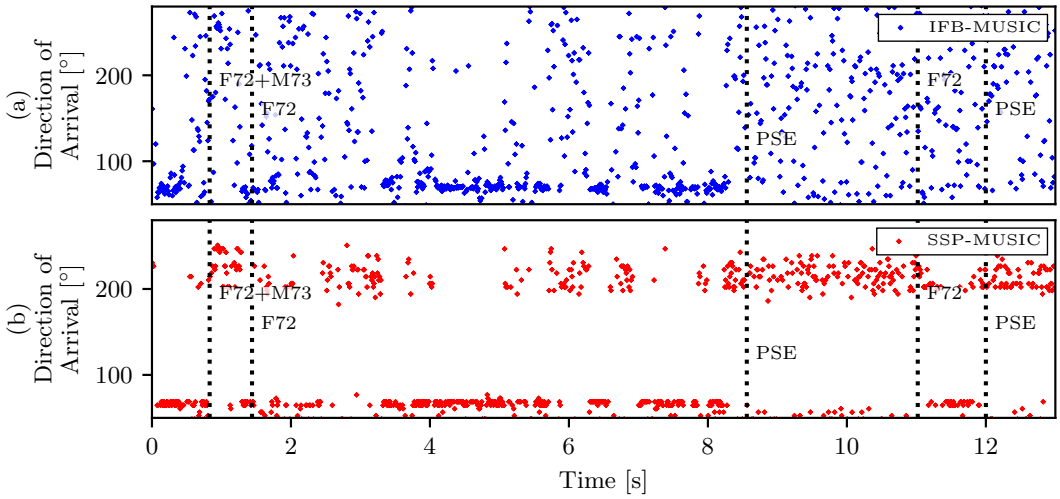


Figure 6.9: An illustrative example of part of a meeting from the AMI corpus. (a) DoA estimates from a circular array using IFB-MUSIC. (b) DoA estimates from a circular array using SSP-MUSIC.

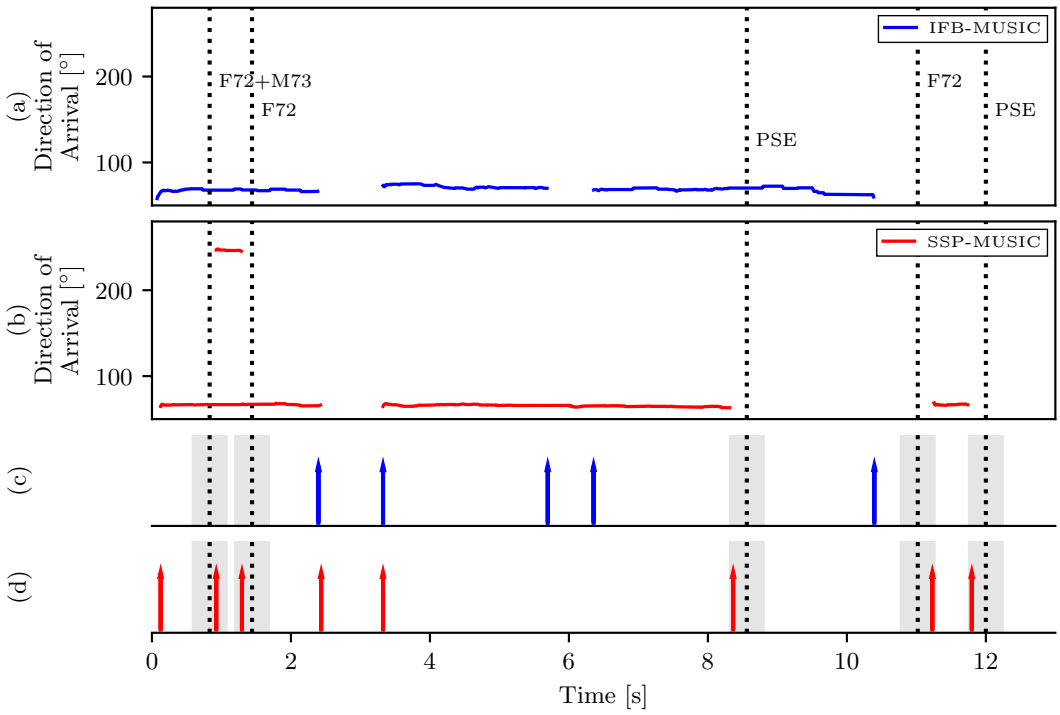


Figure 6.10: An illustrative example of part of a meeting from the AMI corpus. (a) IFB-MUSIC DoA estimates tracked. (b) SSP-MUSIC DoA estimates tracked. (c) IFB-MUSIC segmentation [HIT: 0%, MISS: 100%, MH: 0%, FA:100%] and (d) SSP-MUSIC segmentation [HIT: 100%, MISS: 0%, MH: 0%, FA:29%] where each arrow marks the start or end of a track.

6.5 Overlapping Speaker Segmentation

In this section, SSP-MUSIC is used for overlapping speaker segmentation. It has been shown in Chapter 5 that a MHT framework can be exploited to track the DoA of multiple speakers simultaneously. Instead of using IFB-MUSIC DoA estimates presented in Chapter 5, SSP-MUSIC estimates are used in the MHT framework.

To quantify the improvement obtained using SSP-MUSIC, an illustrative example taken from Section 5.3.3 is evaluated. This example is taken from the AMI meeting 'EN2002c' between 140 and 153 s and is modified by the addition of diffuse Gaussian noise at 0 dB SNR. This additional noise is required as SSP-MUSIC only outperforms IFB-MUSIC at low SNR values, therefore, SNR is set to 0 dB. The aim is to simulate a typical meeting in a noisy environment. The result is shown in Fig. 6.9 on Page 90 and Fig. 6.10 on Page 90. The improvement of the SSP-MUSIC method against the IFB-MUSIC method can most clearly be seen in the HRs. The SSP-MUSIC method achieves a 100.0% HRs whereas IFB-MUSIC achieves a 0.0% HRs. The improvement in HRs for SSP-MUSIC does, however, need to be taken in conjunction with the less than perfect FAR of 29.0%. It should also be noted that this example was selected due to the presence of overlapping speech and, therefore, although IFB-MUSIC performs poorly in terms of the HRs; Fig. 6.9 does show that IFB-MUSIC still produces a reliable DoA estimate trajectory for much of the given signal.

6.6 Conclusion

In this chapter, the potential of SSP-MUSIC, which is a polynomial extension of MUSIC, has been developed and explored. In addition, some enhancements have been proposed for sound source localization. This chapter has highlighted the benefits of using SSP-MUSIC for localization of a single sound source at SNR values lower than 5 dB or T60 values larger than 0.7 s as it is more robust to noise and reverberation. An evaluation was also carried out on real data, taken from the LOCATA corpus, which has shown that SSP-MUSIC can outperform IFB-MUSIC on real-world signals. This work has also been published in the following paper [1].

Chapter 7

MAIN CONCLUSIONS AND FUTURE WORK

7.1 Discussion

This thesis has focused on the segmentation task of the diarization process. It has been shown that the temporal tracking of acoustic and spatial features can be used advantageously to improve the segmentation performance of overlapping speech. Although outside the scope of this work, it has also been shown in the past that diarization systems are greatly affected by the performance of their speaker segmentation component when the classical technique of performing segmentation followed by clustering is used [191, 192].

In [191], the impact of different segmentation systems was evaluated in terms of the diarization error rate (DER) in the context of online (real-time) diarization, where a final resegmentation step would not be possible. The DER is defined by the National Institute of Standards and Technology (NIST) in [193]. Results are given for 1) Uniform segmentation, where the input audio is divided into short segments of equal length. 2) A generalized likelihood ratio (GLR)-based speaker segmentation system described in [194] which uses a two-pass approach. The first pass calculates the GLR distance between two sliding windows over the entire input audio, where a threshold is used to identify speaker change boundaries. Then a second pass is performed where long segments are split based on an algorithm proposed in [194]. Lastly, any segments that are made up of a low percentage of frames containing speech (calculated using an oracle VAD obtained from the reference transcripts) are removed. 3) A convolutional neural network (CNN)-based speaker segmentation algorithm where a CNN is trained on spectrograms obtained using the approach described in [54] where the CNN outputs the probability of a speaker change for every frame. 4) Oracle segmentation based on the reference transcripts where every entry in the transcript is taken to be a single segment.

Segmentation system	Offline DER	Online DER
Uniform	9.23	18.62
GLR-based	11.98	15.04
CNN-based	7.84	15.16
Oracle	6.80	10.98

Table 7.1: Comparison of the impact of different segmentation systems on the complete diarization performance in terms of the DER taken from [191]. The results are given for both an offline diarization system where a resegmentation step is performed and an online diarization system where there is no resegmentation step.

The results in [191] show the effect of accurate segmentation on a typical diarization system that uses an i-vector approach. First, a supervector of statistics [195] is calculated and used to extract i-vectors via factor analysis [50]. Principal component analysis (PCA) is then used to reduce the size of the i-vectors. Finally, the i-vectors are clustered to give the complete diarization output. This clustering is performed using k-means clustering on the cosine distance between i-vectors [196]. The CALLHOME American English corpus of telephone speech [197] is used where only two speakers are present. The results are taken from [191] and shown in Table 7.1. Table 7.1 clearly shows the importance of accurate speaker segmentation for a typical diarization system. In particular, in the context of online diarization, which has no resegmentation step, the DER can be reduced by almost half from 18.62% to 10.98% if oracle speaker segmentation is used instead of uniform segmentation. These findings motivate the need for better segmentation systems such as the approaches described in this thesis.

Furthermore, it has also been shown that these improvements can be mainly accredited to improved handling of overlapping speech [192, 198, 199]. In [192], a diarization system that uses MFCCs as its input feature is tested on 27 recordings of conference meetings, with the results highlighting that overlapping speech and misclassifications in speaker segmentation accounted for an increase of 10.9% in the DER. An analysis undertaken in [198] also showed that at least 40% of the DER, for five different diarization systems, is due to mistakes made in the classification of short segments and segments that fall within 0.5 seconds of a speaker change boundary; both of which can be improved by more accurate segmentation. [199] also produced similar findings showing that some of the main contributing factors to overall DER were the initial speaker segmentation and the consideration of overlapping speech. It was shown for a given system that if one speaker was correctly identified during periods of overlapping speech, then the DER halved from 19.11% to 11.19%. If the second active speaker was also correctly identified during this overlapping period, the DER halved again to 6.56%. These results demonstrate the importance of dealing with overlapping speech when carrying out the speaker segmentation task and, therefore, validates the focus of

this thesis on improving speaker segmentation performance in the presence of overlapping speech.

It should, however, still be noted that in more recent years, with the emergence of DNNs, there has been a shift away from deep learning methods that replace a single module in the pipeline of segmentation followed by clustering to deep learning approaches that are fully end-to-end [60]. That being said, speaker segmentation is still used today in many applications and improving the performance of segmentation systems remains an active area of research.

7.2 Conclusions

7.2.1 Fundamental Frequency Tracking for Overlapping Speaker Segmentation

It has been shown on a well-established corpus of conversational speech that a F_0 change is a strong indicator of a speaker change. A F_0 segmentation system has been proposed that uses a Kalman filter prediction error-based approach based on a model of the temporal variation of the F_0.

This F_0 segmentation system has been extended to track the harmonic structure of voiced speech for the task of overlapping speaker segmentation. This overlapping speaker segmentation system relies on a MHT framework to track multiple speakers even when they are talking simultaneously. It was also shown that the F_0 estimates obtained by the proposed system can be used as an input feature for a neural network which allows for the proposed method to be exploited as part of a multimodal approach along with other features.

7.2.2 Fundamental Frequency and Direction of Arrival Estimation Tracking for Overlapping Speaker Segmentation

It has also been shown that the proposed MHT framework can be exploited to track not only acoustic features, such as F_0, but also spatial features, such as the DoA. A novel MHT method that tracks both the DoA and F_0 of multiple speakers simultaneously has been developed.

It has been shown that this method can lead to an improved speaker segmentation performance over tracking just one of these features alone.

7.2.3 Polynomial MUSIC for Overlapping Speaker Segmentation

A SSP-MUSIC approach, which is a polynomial extension of MUSIC, has been developed and explored. In addition, some enhancements have been proposed for sound source localization. It has been shown that

using SSP-MUSIC over conventional MUSIC for localization is advantageous in terms of its robustness to noise and reverberation. However, in low reverberation and noiseless environments conventional MUSIC achieves a similar performance at a lower computational complexity.

Finally, SSP-MUSIC has been applied to the proposed MHT framework which results in an improved segmentation performance.

7.3 Suggestions for Future Work

7.3.1 Tracking Approaches

The MHT approach has been exploited in Chapters 3, 4 and 6. An interesting topic for future work would be to explore different tracking approaches, including other traditional techniques, for example, joint probabilistic data association (JPDA) filters [200] and the probabilistic multiple hypothesis tracking (PMHT) [201].

This could then be extended to more recent tracking approaches that use DNNs for the task of object tracking. This would include methods such as [202] which uses an end-to-end DNN to directly compute a similarity score between pairs of detections and tracks for online multiple object tracking (MOT). In [202] the main tracker is based on a recurrent neural network (RNN) model that aims to mimic a Bayesian filter algorithm. Another approach introduced in [203] tracks bounding boxes using a Kalman filter and associates each bounding box with its highest overlapping detection in the current frame using bipartite matching. Many more approaches exist and a comprehensive overview has been given in both [204] and [205].

In this thesis, a Kalman filter with a random walk model has also been heavily utilised to perform the task of tracking. However, this is not the only tracking algorithm available. Other algorithms include modifications to the traditional Kalman filtering method such as the extended Kalman filter (EKF) and the unscented Kalman filter (UKF) [206]. More recently, probability hypothesis density (PHD) filters [207] have been prevalent as they are able to recursively estimate both the number and the state of a set of targets from the given observations. This work is developed in [208] where a von Mises distribution, instead of the Gaussian distribution that is exploited in (5.2) from Section 5.2.5 on Page 66, is used to model audio-source DoA estimates in conjunction with the variational expectation-maximisation algorithm.

7.3.2 Deep Learning and Neural Networks

In recent years there have been significant developments in deep learning approaches. To highlight how this work could contribute to this body of research, a pitch feature was conceived in Chapter 4, that could be used as an input to a BLSTM method [41]. In future work, it would be advantageous to explore these types of input features in more detail, for example, creating a DoA feature from the work presented in Chapters 5 and 6.

7.3.3 Polynomial Eigenvalue Decomposition for Speaker Counting

The PEVD MUSIC approach explored in Chapter 6 currently relies on the prior of knowing the number of active speakers in a given recording. This work could, therefore, be extended to incorporate speaker counting by evaluating the energy of each eigenvalue. The energy of eigenvalues corresponding to active speakers should be much larger and, therefore, the rank of the eigenvalue matrix should be equivalent to the number of active speakers in a given frame (an illustrative example showing how the rank of the eigenvalue matrix relates to the number of active speakers can be seen in Fig. 6.2(b)).

BIBLIOGRAPHY

[1] A. O. T. Hogg, V. W. Neo, S. Weiss, C. Evers, and P. A. Naylor, "A polynomial eigenvalue decomposition MUSIC approach for broadband sound source localization," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, New Paltz, NY, Oct. 2021, pp. 326–330.

[2] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multichannel overlapping speaker segmentation using multiple hypothesis tracking of acoustic and spatial features," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Jun. 2021, pp. 26–30.

[3] A. O. T. Hogg, C. Evers, A. H. Moore, and P. A. Naylor, "Overlapping speaker segmentation using multiple hypothesis tracking of fundamental frequency," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 1479–1490, Mar. 2021.

[4] A. O. T. Hogg, C. Evers, and P. A. Naylor, "Multiple hypothesis tracking for overlapping speaker segmentation," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2019, pp. 195–199.

[5] A. O. T. Hogg, P. A. Naylor, and C. Evers, "Speaker change detection using fundamental frequency with application to multi-talker segmentation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2019, pp. 5826–5830.

[6] S. W. McKnight, A. O. T. Hogg, V. W. Neo, and P. A. Naylor, "Studying human-based speaker diarization and comparing to state-of-the-art systems," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Nov. 2022.

[7] V. W. Neo, S. Weiss, S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Polynomial eigenvalue decomposition-based target speaker voice activity detection in the presence of competing talkers," in *Proc. Int. Workshop on Acoust. Signal Enhancement (IWAENC)*, Sep. 2022, pp. 1–5.

[8] S. W. McKnight, A. O. T. Hogg, V. W. Neo, and P. A. Naylor, "A study of salient modulation domain features for speaker identification," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Dec. 2021, pp. 705–712.

[9] S. W. McKnight, A. O. T. Hogg, and P. A. Naylor, "Analysis of phonetic dependence of segmentation errors in speaker diarization," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2020, pp. 1–5.

[10] D. Sharma, A. O. T. Hogg, Y. Wang, A. Nour-Eldin, and P. A. Naylor, "Non-intrusive POLQA estimation of speech quality using recurrent neural networks," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2019, pp. 1–5.

[11] P. Saini and P. Kaur, "Automatic speech recognition: A review," *Int. J. of Engineering Trends and Technol.*, vol. 4, no. 2, pp. 1–5, 2013.

[12] P. Cerva, J. Silovsky, J. Zdansky, J. Nouza, and L. Seps, "Speaker-adaptive speech recognition using speaker diarization for improved transcription of large spoken archives," *Speech Commun.*, vol. 55, no. 10, pp. 1033–1046, Nov. 2013.

[13] M. Sinclair, "Speech segmentation and speaker diarisation for transcription and translation," Ph.D. dissertation, The University of Edinburgh, Jun. 2016.

[14] D. Vijayasenan and F. Valente, "DiarTk : An open source toolkit for research in multistream speaker diarization and its application to meetings recordings," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, 2012, pp. 2170–2173.

[15] M. A. Alrmah, S. Weiss, and S. Lambotharan, "An extension of the MUSIC algorithm to broadband scenarios using a polynomial eigenvalue decomposition," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2011, pp. 629–633.

[16] M. Alrmah, "Broadband angle of arrival estimation using polynomial matrix decompositions," Ph.D. dissertation, University of Strathclyde, Scotland, Oct. 2015.

[17] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proc. DARPA Speech Recognition Workshop*, 1997, pp. 97–99.

[18] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[19] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 14, no. 5, pp. 1557–1565, Sep. 2006.

[20] M. H. Moattar and M. M. Homayounpour, "A review on speaker diarization systems and approaches," *Speech Commun.*, vol. 54, no. 10, pp. 1065–1103, Dec. 2012.

[21] C. Evers and P. A. Naylor, "Acoustic SLAM," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 9, pp. 1484–1498, Sep. 2018.

[22] C. Evers, H. W. Löllmann, H. Mellmann, A. Schmidt, H. Barfuss, P. A. Naylor, and W. Kellermann, "The LOCATA challenge: Acoustic source localization and tracking," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1620–1643, Apr. 2020.

[23] C. Kwan, J. Yin, B. Ayhan, S. Chu, X. Liu, K. Puckett, Y. Zhao, K. C. Ho, M. Kruger, and I. Sityar, "Speech separation algorithms for multiple speaker environments," in *Proc. Int. Joint Conf. on Neural Networks (IJCNN)*, Jun. 2008, pp. 1644–1648.

[24] G. Sell and D. Garcia-Romero, "Diarization resegmentation in the factor analysis subspace," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2015, pp. 4794–4798.

[25] X. A. Miró, "Robust speaker diarization for meetings," Ph.D. dissertation, Universitat Politècnica de Catalunya, Oct. 2006.

[26] E. Shriberg, A. Stolcke, and D. Baron, "Observations on overlap: Findings and implications for automatic processing of multi-party conversation," in *Proc. Eur. Conf. on Speech Commun. and Technol.*, 2001, pp. 1359–1362.

[27] M. Kunešová, M. Hrúz, Z. Zajíc, and V. Radová, "Detection of overlapping speech for the purposes of speaker diarization," in *Proc. Speech and Comput.*, ser. Lecture Notes in Computer Science, A. A. Salah, A. Karpov, and R. Potapova, Eds., vol. 11658. Cham: Springer, Jul. 2019, pp. 247–257.

[28] P. A. Naylor and N. D. Gaubitch, Eds., *Speech Dereverberation.* Springer-Verlag, 2010.

[29] X. A. Miró, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 2, pp. 356–370, Feb. 2012.

[30] Q. Wang, C. Downey, L. Wan, P. A. Mansfield, and I. L. Moreno, "Speaker diarization with LSTM," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2018, pp. 5239–5243.

[31] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, *et al.*, "Diarization is hard: Some experiences and lessons learned for the JHU team in the inaugural DIHARD challenge," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Sep. 2018.

[32] M. Senoussaoui, P. Kenny, T. Stafylakis, and P. Dumouchel, "A study of the Cosine distance-based mean shift for telephone speech diarization," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 1, pp. 217–227, Jan. 2014.

[33] T. Kemp, M. Schmidt, M. Westphal, and A. Waibel, "Strategies for automatic segmentation of audio data," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 3, Jun. 2000, pp. 1423–1426 vol.3.

[34] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *DARPA Broadcast News Transcription and Understanding Workshop*, Jan. 1998, pp. 127–132.

[35] R. Bachu, S. Kopparthi, B. Adapa, and B. Barkana, "Voiced/unvoiced decision for speech signals based on zero-crossing rate and energy," in *Adv. Techniques in Computing Sciences and Softw. Eng.*, K. Elleithy, Ed., 2010, pp. 279–282.

[36] K. Sakhnov, E. Verteletskaya, and B. Simák, "Approach for energy-based voice detector with adaptive scaling factor," *IAENG Int. J. of Comput. Science*, vol. 36, pp. 4–10, 2009.

[37] H. Veisi and H. Sameti, "Hidden-Markov-model-based voice activity detector with high speech detection rate for speech enhancement," *Proc. IET Signal Process.*, vol. 6, no. 1, pp. 54–63, Feb. 2012.

[38] M. Asgari, A. Sayadian, M. Farhadloo, and E. abouie Mehrizi, "Voice activity detection using entropy in spectrum domain," in *Proc. Australasian TeleCommun. Networks and Applicat. Conf. (ATNAC)*, Dec. 2008, pp. 407–410.

[39] J. Stegmann and G. Schroder, "Robust voice-activity detection based on the wavelet transform," in *Proc. IEEE Workshop on Speech Coding*, Sep. 1997, pp. 99–100.

[40] J. Ramirez, J. M. Górriz, and J. C. Segura, "Voice activity detection. fundamentals and speech recognition system robustness," in *Noise Reduction in Speech Appl.*, M. Grimm and K. Kroschel, Eds. InTech, 2007, pp. 1–22.

[41] H. Bredin, R. Yin, J. M. Coria, G. Gelly, P. Korshunov, M. Lavechin, D. Fustes, H. Titeux, W. Bouaziz, and M.-P. Gill, "Pyannote.audio: Neural building blocks for speaker diarization," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2020.

[42] R. Gangadharaiah, B. Narayanaswamy, and N. Balakrishnan, "A novel method for two-speaker segmentation," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, 2004.

[43] J.-F. Bonastre, P. Delacourt, C. Fredouille, T. Merlin, and C. Wellekens, "A speaker tracking system based on speaker turn detection for NIST evaluation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 2, Jun. 2000, pp. II1177–II1180 vol.2.

[44] H. Gish, M.-H. Siu, and R. Rohlicek, "Segregation of speakers for speech recognition and speaker identification," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 1991, pp. 873–876 vol.2.

[45] A. Tritschler and R. Gopinath, "Improved speaker segmentation and segments clustering using the bayesian information criterion," in *Proc. Eur. Conf. on Speech Commun. and Technol.*, 1999.

[46] A. Malegaonkar, A. Ariyaeeinia, P. Sivakumaran, and J. Fortuna, "Unsupervised speaker change detection using probabilistic pattern matching," *IEEE Signal Process. Lett.*, vol. 13, no. 8, pp. 509–512, Aug. 2006.

[47] J. Ajmera, I. McCowan, and H. Bourlard, "Robust speaker change detection," *IEEE Signal Process. Lett.*, vol. 11, no. 8, pp. 649–651, Aug. 2004.

[48] K. Mori and S. Nakagawa, "Speaker change detection and speaker clustering using VQ distortion for broadcast news speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 1, May 2001, pp. 413–416 vol.1.

[49] P. Delacourt and C. J. Wellekens, "DISTBIC: A speaker-based segmentation for audio data indexing," *Speech Commun.*, vol. 32, no. 1, pp. 111–126, Sep. 2000.

[50] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

[51] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2018, pp. 5329–5333.

[52] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2014.

[53] V. Gupta, "Speaker change point detection using deep neural nets," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2015, pp. 4420–4424.

[54] M. Hrúz and Z. Zajíc, "Convolutional neural network for speaker change detection in telephone speaker diarization system," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2017, pp. 4945–4949.

[55] M. Hrúz and M. Kunesova, "Convolutional neural network in the task of speaker change detection," in *Proc. Speech and Comput.*, Aug. 2016.

[56] L. Mateju, P. Cerva, and J. Zdánský, "An approach to online speaker change point detection using DNNs and WFSTs," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Sep. 2019.

[57] R. Yin, H. Bredin, and C. Barras, "Speaker change detection in broadcast TV using bidirectional long short-term memory networks," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Aug. 2017, pp. 3827–3831.

[58] P. Cyrta, T. Trzciński, and W. Stokowiec, "Speaker diarization using deep recurrent convolutional neural networks for speaker embeddings," in *International Conference on Information Systems Architecture and Technology*, 2017, pp. 107–117.

[59] S. H. Yella, A. Stolcke, and M. Slaney, "Artificial neural network features for speaker diarization," in *Proc. IEEE Spoken Language Technol. Workshop (SLT)*, 2014, pp. 402–406.

[60] T. J. Park, N. Kanda, D. Dimitriadis, K. J. Han, S. Watanabe, and S. S. Narayanan, "A review of speaker diarization: Recent advances with deep learning," *Comput. Speech and Language*, 2022.

[61] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[62] K. Laskowski, M. Heldner, and J. Edlund, "The fundamental frequency variation spectrum," in *Proc. FONETIK*, 2008.

[63] D. Dimitriadis, P. Maragos, and A. Potamianos, "Modulation features for speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 1, May 2002, pp. I–377–I–380.

[64] L. Atlas and S. A. Shamma, "Joint acoustic and modulation frequency," *EURASIP J. on Appl. Signal Process.*, vol. 7, pp. 668–675, 2003.

[65] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 4, pp. 578–589, Oct. 1994.

[66] B. E. D. Kingsbury, N. Morgan, and S. Greenberg, "Robust speech recognition using the modulation spectrogram," *Speech communication*, vol. 25, no. 1, pp. 117–132, 1998.

[67] S. So and K. K. Paliwal, "Modulation-domain Kalman filtering for single-channel speech enhancement," *Speech Commun.*, vol. 53, no. 6, pp. 818–829, Jul. 2011.

[68] Y. Wang, K. Han, and D. Wang, "Exploring monaural features for classification-based speech segregation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 2, pp. 270–279, Feb. 2013.

[69] T. Houtgast, H. J. M. Steeneken, and R. Plomp, "Predicting speech intelligibility in rooms from the modulation transfer function. I. General room acoustics," *Acustica*, vol. 46, no. 1, pp. 60–72, 1980.

[70] T. M. Elliott and F. E. Theunissen, "The modulation transfer function for speech intelligibility," *PLoS Computational Biology*, vol. 5, no. 3, Mar. 2009.

[71] B. Kollmeier and R. Koch, "Speech enhancement based on physiological and psychoacoustical models of modulation perception and binaural interaction," *The Journal of the Acoustical Society of America*, vol. 95, no. 3, pp. 1593–1602, Mar. 1994.

[72] J. Wu and X.-L. Zhang, "An efficient voice activity detection algorithm by combining statistical model and energy detection," *EURASIP J. on Advances in Signal Process.*, vol. 18, 2011.

[73] T. H. Falk, C. Zheng, and W.-Y. Chan, "A non-intrusive quality and intelligibility measure of reverberant and dereverberated speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 7, pp. 1766–1774, Sep. 2010.

[74] D. S. Kim and A. Tarraf, "ANIQUE+: A new american national standard for non-intrusive estimation of narrowband speech quality," *Bell Labs Tech. J.*, vol. 12, pp. 221–236, 2007.

[75] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, no. 4, pp. 320–327, Aug. 1976.

[76] M. Wax and T. Kailath, "Optimum localization of multiple sources by passive arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 31, no. 5, pp. 1210–1217, Oct. 1983.

[77] W. Bangs and P. Schultheis, "Space-time processing for optimal parameter estimation," *Signal Process.*, pp. 577–590, 1973.

[78] J. H. Dibiase, "A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays," Ph.D. dissertation, Brown University, Aug. 2000.

[79] M. S. Brandstein and D. B. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications.* Berlin, Germany: Springer-Verlag, 2001.

[80] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[81] H. Wang and M. Kaveh, "Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wide-band sources," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 4, pp. 823–831, Aug. 1985.

[82] E. D. di Claudio and R. Parisi, "WAVES: Weighted average of signal subspaces for robust wideband direction finding," *IEEE Trans. Signal Process.*, vol. 49, no. 10, pp. 2179–2191, Oct. 2001.

[83] Yeo-Sun Yoon, L. Kaplan, and J. McClellan, "TOPS: New DOA estimator for wideband signals," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 1977–1989, Jun. 2006.

[84] H. Pan, R. Scheibler, E. Bezzam, I. Dokmanić, and M. Vetterli, "FRIDA: FRI-based DOA estimation for arbitrary array layouts," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2017, pp. 3186–3190.

[85] K. Boakye, B. Trueba-Hornero, O. Vinyals, and G. Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2008, pp. 4353–4356.

[86] S. H. Yella and H. Bourlard, "Improved overlap speech diarization of meeting recordings using long-term conversational features," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2013, pp. 7746–7750.

[87] V. Rozgic, K. J. Han, P. G. Georgiou, and S. Narayanan, "Multimodal speaker segmentation in presence of overlapped speech segments," in *Proc. IEEE Int. Symp on Multimedia (ISM)*, Dec. 2008, pp. 679–684.

[88] J. Zhong, P. Zhang, and X. Li, "A combined feature approach for speaker segmentation using convolution neural network," in *Advances in Multimedia Inform. Process. – PCM*, ser. Lecture Notes in Computer Science, B. Zeng, Q. Huang, A. El Saddik, H. Li, S. Jiang, and X. Fan, Eds. Cham: Springer International Publishing, 2018, pp. 550–559.

[89] L. Sarı, S. Thomas, M. Hasegawa-Johnson, and M. Picheny, "Pre-training of speaker embeddings for low-latency speaker change detection in broadcast news," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2019, pp. 6286–6290.

[90] R. Wang, M. Gu, L. Li, M. Xu, and T. F. Zheng, "Speaker segmentation using deep speaker vectors for fast speaker change scenarios," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2017, pp. 5420–5424.

[91] M. H. Moattar and M. M. Homayounpour, "Variational conditional random fields for online speaker detection and tracking," *Speech Communication*, vol. 54, no. 6, pp. 763–780, Jul. 2012.

[92] M. Zamalloa, L. J. Rodríguez-Fuentes, G. Bordel, G. Bordel, M. Penagarikano, and J. P. Uribe, "Low-latency online speaker tracking on the AMI corpus of meeting conversations," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2010, pp. 4962–4965.

[93] L. Lu and H.-J. Zhang, "Unsupervised speaker segmentation and tracking in real-time audio content analysis," *Multimedia Systems*, vol. 10, no. 4, pp. 332–343, Apr. 2005.

[94] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. of the ASME J. of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, Mar. 1960.

[95] R. Faragher, "Understanding the basis of the kalman filter via a simple and intuitive derivation [lecture notes]," *IEEE Signal Process. Mag.*, vol. 29, pp. 128–132, Sep. 2012.

[96] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[97] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. on Comput. Vision (ICCV)*, Washington, DC, USA, 2015, pp. 4696–4704.

[98] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems.* Artech House, 1998.

[99] J. Patino, H. Delgado, and N. Evans, "Speaker change detection using binary key modelling with contextual information," in *Statistical Language and Speech Process. (SLSP)*, Oct. 2017.

[100] H. Bredin, "Pyannote.metrics: A toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, 2017.

[101] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, and V. Karaiskos, "The AMI meeting corpus," in *Proc. Int. Conf. on Methods and Techniques in Behav. Research*, vol. 88, 2005, p. 100.

[102] R. Stiefelhagen, "Tracking focus of attention in meetings," in *Proc. IEEE Int. Conf. on the Science of Elect. Eng. (ICSEE)*, Oct. 2002, pp. 273–280.

[103] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, *et al.*, "The AMI meeting corpus: A pre-announcement," in *Proc. Intl. Conf. Machine Learning for Multimodal Interaction (ICMI)*, Berlin, Heidelberg, 2006, pp. 28–39.

[104] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT acoustic-phonetic continuous speech corpus," Linguistic Data Consortium (LDC), Philadelphia, USA, Corpus LDC93S1, 1993.

[105] A. Larcher, K. A. Lee, and S. Meignier, "An extensible speaker identification sidekit in python," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2016, pp. 5095–5099.

[106] M. Rouvier, G. Dupuy, P. Gay, E. Khoury, T. Merlin, and S. Meignier, "An open-source state-of-the-art toolbox for broadcast news diarization," *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Aug. 2013.

[107] J.-F. Bonastre, N. Scheffer, D. Matrouf, C. Fredouille, A. Larcher, A. Preti, G. Pouchoulin, N. Evans, B. Fauve, and J. Mason, "ALIZE/SpkDet: A state-of-the-art open source software for speaker recognition," *Proc. Odyssey IEEE Workshop*, 2008.

[108] R. Li, T. Schultz, and Q. Jin, "Improving speaker segmentation via speaker identification and text segmentation." in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Sep. 2009, pp. 3073–3076.

[109] F. Itakura, "Line spectrum representation of linear predictor coefficients of speech signals," *J. Acoust. Soc. Am.*, vol. 57, p. S35, Jan. 1975.

[110] M. Yang, Y. Yang, and Z. Wu, "A pitch-based rapid speech segmentation for speaker indexing," in *Proc. IEEE Int. Symp on Multimedia (ISM)*, Dec. 2005, p. 6 pp.

[111] B. Abdolali, "A novel method for speech segmentation based on speaker's characteristics," *Signal & Image Process. : An Int. J. (SIPIJ)*, vol. 3, no. 2, pp. 65–78, Apr. 2012.

[112] Ö. Salor, M. Demirekler, and U. Orguner, "Kalman filter approach for pitch determination of speech signals," *Proc. Speech and Comput.*, p. 4, Jun. 2006.

[113] O. Das, J. O. S. Iii, and C. Chafe, "Real-time pitch tracking in audio signals with the extended complex kalman filter," *Proc. Conf. on Digital Audio Effects*, vol. 20, pp. 118–124, Sep. 2017.

[114] L. Shi, J. K. Nielsen, J. R. Jensen, M. A. Little, and M. G. Christensen, "A Kalman-based fundamental frequency estimation algorithm," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, New Paltz, NY, Oct. 2017, pp. 314–318.

[115] S. Gonzalez and D. M. Brookes, "PEFAC - A pitch estimation algorithm robust to high levels of noise," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, pp. 518–530, Feb. 2014.

[116] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds., 1995, pp. 495–518.

[117] A. de Cheveigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Am.*, vol. 111, no. 4, pp. 1917–1930, Apr. 2002.

[118] J. K. Nielsen, T. L. Jensen, J. R. Jensen, M. G. Christensen, and S. H. Jensen, "Fast and statistically efficient fundamental frequency estimation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2016, pp. 86–90.

[119] W. Zhang, Y. Xie, B. Lin, L. Wang, and J. Zhang, "Estimation of the underlying F0 range of a speaker from the spectral features of a brief speech input," *Applied Sciences*, vol. 12, no. 13, p. 6494, Jan. 2022.

[120] Y. Xu and X. Sun, "Maximum speed of pitch change and how it may relate to speech," *J. Acoust. Soc. Am.*, vol. 111, no. 3, pp. 1399–1413, Mar. 2002.

[121] D. M. Brookes, "VOICEBOX: A speech processing toolbox for MATLAB," 1997. [Online]. Available: http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html

[122] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, 1999.

[123] A. O. T. Hogg, 2019. [Online]. Available: https://github.com/ahogg/hogg2019-icassp-paper

[124] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, no. null, pp. 281–305, Feb. 2012.

[125] S. Meignier, D. Moraru, C. Fredouille, J.-F. Bonastre, and L. Besacier, "Step-by-step and integrated approaches in broadcast news speaker diarization," *Comput. Speech and Language*, vol. 20, no. 2, pp. 303–330, Apr. 2006.

[126] S. Cheng, H. Wang, and H. Fu, "BIC-based speaker segmentation using divide-and-conquer strategies with application to speaker diarization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 1, pp. 141–157, Jan. 2010.

[127] A. woudie, J. Luque, and J. Hernando, "Jitter and shimmer measurements for speaker diarization," in *Proc. Int. Conf. on Advances in Speech and Language Technologies for Iberian Languages (IBERSPEECH)*, Nov. 2014, pp. 21–30.

[128] K. Laskowski and Q. Jin, "Modeling instantaneous intonation for speaker identification using the fundamental frequency variation spectrum," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2009, pp. 4541–4544.

[129] Q. Jin, R. Li, Q. Yang, K. Laskowski, and T. Schultz, "Speaker identification with distant microphone speech," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2010, pp. 4518–4521.

[130] A. G. Adami, R. Mihaescu, D. A. Reynolds, and J. J. Godfrey, "Modelling prosodic dynamics for speaker recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 4, 2003, pp. 788–791.

[131] T. Abe, T. Kobayashi, and S. Imai, "Harmonics tracking and pitch extraction based on instantaneous frequency," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 1, May 1995, pp. 756–759.

[132] M. Wu, D. L. Wang, and G. J. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 3, pp. 229–241, May 2003.

[133] M. G. Christensen, A. Jakobsson, and S. H. Jensen, "Multi-pitch estimation using harmonic MUSIC," in *Proc. Asilomar Conf. on Signals, Syst. & Comput.*, 2006, pp. 521–524.

[134] M. G. Christensen, P. Stoica, A. Jakobsson, and S. H. Jensen, "The multi-pitch estimation problem: Some new solutions," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 3, 2007, pp. 1221–1224.

[135] M. G. Christensen, P. Vera-Candeas, S. D. Somasundaram, and A. Jakobsson, "Robust subspace-based fundamental frequency estimation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2008, pp. 101–104.

[136] R. Peharz, M. Wohlmayr, and F. Pernkopf, "Gain-robust multi-pitch tracking using sparse nonnegative matrix factorization," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2011, pp. 5416–5419.

[137] M. Wohlmayr, R. Peharz, and F. Pernkopf, "Efficient implementation of probabilistic multi-pitch tracking," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2011, pp. 5412–5415.

[138] I. J. Cox and S. L. Hingorani, "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 2, pp. 138–150, Feb. 1996.

[139] K. Yoon, Y. Song, and M. Jeon, "Multiple hypothesis tracking algorithm for multi-target multi-camera tracking with disjoint views," *IET Image Process.*, vol. 12, no. 7, pp. 1175–1184, Jun. 2018.

[140] P. E. Rybski and M. M. Veloso, "Prioritized multihypothesis tracking by a robot with limited sensing," *EURASIP J. on Advances in Signal Process.*, vol. 2009, no. 1, pp. 1–17, Dec. 2009.

[141] Y. Sasaki, N. Hatao, K. Yoshii, and S. Kagami, "Nested iGMM recognition and multiple hypothesis tracking of moving sound sources for mobile robot audition," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Syst. (IROS)*, Nov. 2013, pp. 3930–3936.

[142] A. O. T. Hogg, 2021. [Online]. Available: https://github.com/ahogg/Overlapping_speaker_segmentation_using_multiple_hypothesis_tracking_of_fundamental_frequency

[143] D. Wang and G. Hu, "Unvoiced speech segregation," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 5, May 2006.

[144] S. Ahmadi and A. Spanias, "Cepstrum-based pitch detection using a new statistical V/UV classification algorithm," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 3, pp. 333–338, May 1999.

[145] G. Friedland, O. Vinyals, Y. Huang, and C. Muller, "Prosodic and other long-term features for speaker diarization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 17, no. 5, pp. 985–993, Jul. 2009.

[146] B. Atal and L. Rabiner, "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, no. 3, pp. 201–212, Jun. 1976.

[147] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with SincNet," in *Proc. IEEE Spoken Language Technol. Workshop (SLT)*, Dec. 2018, pp. 1021–1028.

[148] D. J. Papageorgiou and M. R. Salpukas, "The maximum weight independent set problem for data association in multiple hypothesis tracking," in *Optimization and Cooperative Control Strategies.* New York, NY, USA: Springer, 2009, pp. 235–255.

[149] P. R. J. Östergård, "A new algorithm for the maximum-weight clique problem," *Nordic Journal of Computing*, vol. 8, no. 4, pp. 424–436, Dec. 2001.

[150] W. Hess, *Pitch Determination of Speech Signals.* Springer-Verlag, 1983.

[151] R. Yin, H. Bredin, and C. Barras, "Neural speech turn segmentation and affinity propagation for speaker diarization," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Sep. 2018, pp. 1393–1397.

[152] H. Bredin, 2022. [Online]. Available: https://github.com/pyannote/pyannote-audio

[153] C. Bron and J. Kerbosch, "Algorithm 457: Finding all cliques of an undirected graph," *Commun. ACM*, vol. 16, no. 9, pp. 575–577, Sep. 1973.

[154] F. Chollet, "Keras," 2015. [Online]. Available: https://keras.io

[155] B. McFee, V. Lostanlen, M. McVicar, A. Metsai, S. Balke, C. Thomé, C. Raffel, A. Malek, D. Lee, F. Zalkow, *et al.*, "Librosa/librosa: 0.7.2," Jan. 2020. [Online]. Available: https://zenodo.org/record/6759664

[156] S. E. Chazan, H. Hammer, G. Hazan, J. Goldberger, and S. Gannot, "Multi-microphone speaker separation based on deep DOA estimation," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2019.

[157] F. Valente, D. Vijayasenan, and P. Motlicek, "Speaker diarization of meetings based on speaker role n-gram models," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2011, pp. 4416–4419.

[158] S. Araki, M. Fujimoto, K. Ishizuka, H. Sawada, and S. Makino, "A DOA based speaker diarization system for real meetings," in *Proc. Joint Workshop on Hands-free Speech Commun. and Microphone Arrays (HSCMA)*, May 2008, pp. 29–32.

[159] J. Pardo, X. Anguera, and C. Wooters, "Speaker diarization for multiple-distant-microphone meetings using several sources of information," *IEEE Trans. Comput.*, vol. 56, no. 9, pp. 1212–1224, Sep. 2007.

[160] E. C. W. Koh, H. Sun, T. L. Nwe, T. H. Nguyen, B. Ma, E.-S. Chng, H. Li, and S. Rahardja, "Speaker diarization using direction of arrival estimate and acoustic feature information: The I2R-NTU submission for the NIST RT 2007 evaluation," in *Multimodal Technologies for Perception of Humans*, ser. Lecture Notes in Computer Science, R. Stiefelhagen, R. Bowers, and J. Fiscus, Eds. Berlin, Heidelberg: Springer, 2008, pp. 484–496.

[161] K. Ishiguro, T. Yamada, S. Araki, and T. Nakatani, "A probabilistic speaker clustering for DOA-based diarization," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2009, pp. 241–244.

[162] S. Araki, H. Sawada, R. Mukai, and S. Makino, "DOA estimation for multiple sparse sources with normalized observation vector clustering," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, vol. 5, May 2006, pp. V–V.

[163] P. P. Parada, D. Sharma, T. van Waterschoot, and P. A. Naylor, "Robust statistical processing of TDOA estimates for distant speaker diarization," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2017, pp. 86–90.

[164] N. Zheng, N. Li, J. Yu, C. Weng, D. Su, X. Liu, and H. Meng, "Multi-channel speaker diarization using spatial features for meetings," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2022, pp. 7337–7341.

[165] X. Ji, M. Yu, J. Chen, J. Zheng, D. Su, and D. Yu, "Integration of multi-look beamformers for multi-channel keyword spotting," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, May 2020, pp. 7464–7468.

[166] R. Scheibler, E. Bezzam, and I. Dokmanić, "Pyroomacoustics: A Python package for audio room simulation and array processing algorithms," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Apr. 2018, pp. 351–355.

[167] Y. Huang, T. Otsuka, and H. G. Okuno, "A speaker diarization system with robust speaker localization and voice activity detection," in *Contemp. Challenges and Solutions in Appl. Artificial Intell.*, ser. Studies in Computational Intell., M. Ali, T. Bosse, K. V. Hindriks, M. Hoogendoorn, C. M. Jonker, and J. Treur, Eds. Heidelberg: Springer, 2013, pp. 77–82.

[168] Y. Wakabayashi, K. Inoue, H. Yoshimoto, and T. Kawahara, "Speaker diarization based on audio-visual integration for smart posterboard," in *Asia-Pacific Signal and Inform. Process. Assoc. Annual Summit and Conf. (APSIPA)*, Dec. 2014, pp. 1–4.

[169] C. Evers and P. A. Naylor, "Optimized self-localization for SLAM in dynamic scenes using probability hypothesis density filters," *IEEE Trans. Signal Process.*, vol. 66, no. 4, pp. 863–878, Feb. 2018.

[170] J. Dmochowski, J. Benesty, and S. Affes, "Direction of arrival estimation using the parameterized spatial correlation matrix," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 4, pp. 1327–1339, 2007.

[171] H. L. Van Trees, *Optimal Array Processing. Part IV of Detection, Estimation, and Modulation Theory.* John Wiley & Sons, 2002.

[172] B. Friedlander, "The root-MUSIC algorithm for direction finding with interpolated arrays," *Signal Process.*, vol. 30, no. 1, pp. 15–29, Jan. 1993.

[173] M. Alrmah, S. Weiss, S. Redif, S. Lambotharan, and J. G. McWhirter, "Angle of arrival estimation for broadband signals: A comparison," in *Proc. IET Int. Conf. on Intelligent Signal Process.*, Jan. 2013, pp. 1–6.

[174] A. Rao and R. Kumaresan, "On decomposing speech into modulated components," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 3, pp. 240–254, May 2000.

[175] S. Weiss and I. K. Proudler, "Comparing efficient broadband beamforming architectures and their performance trade-offs," in *Proc. IEEE Int. Conf. Digital Signal Process. (DSP)*, Jul. 2002, pp. 417–424.

[176] V. W. Neo, C. Evers, and P. A. Naylor, "Speech dereverberation performance of a polynomial-EVD subspace approach," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2020, pp. 221–225.

[177] Y. Huang, J. Benesty, and J. Chen, "Analysis and comparison of multichannel noise reduction methods in a common framework," *IEEE Trans. Audio, Speech, Language Process.*, vol. 16, no. 5, pp. 957–968, Jul. 2008.

[178] V. W. Neo, C. Evers, and P. A. Naylor, "PEVD-based speech enhancement in reverberant environments," in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, 2020, pp. 186–190.

[179] T. I. Laakso, V. Valimaki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay [FIR/all pass filters design]," *IEEE Signal Process. Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.

[180] J. Selva, "An efficient structure for the design of variable fractional delay filters based on the windowing method," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3770–3775, Aug. 2008.

[181] J. G. McWhirter, P. D. Baxter, T. Cooper, S. Redif, and J. Foster, "An EVD algorithm for para-Hermitian polynomial matrices," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 2158–2169, May 2007.

[182] S. Weiss, J. Pestana, and I. K. Proudler, "On the existence and uniqueness of the eigenvalue decomposition of a para-Hermitian matrix," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2659–2672, May 2018.

[183] S. Weiss, I. K. Proudler, and F. K. Coutts, "Eigenvalue decomposition of a para-Hermitian matrix: Extraction of analytic eigenvalues," *IEEE Trans. Signal Process.*, vol. 69, pp. 722–737, 2021.

[184] V. W. Neo, C. Evers, and P. A. Naylor, "Speech enhancement using polynomial eigenvalue decomposition," in *Proc. IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2019, pp. 125–129.

[185] S. Redif, S. Weiss, and J. G. McWhirter, "An approximate polynomial matrix eigenvalue decomposition algorithm for para-hermitian matrices," in *Proc. Int. Symp. on Signal Process. and Inform. Technol. (ISSPIT)*, 2011, pp. 421–425.

[186] J. S. Bradley, H. Sato, and M. Picard, "On the importance of early reflections for speech in rooms," *J. Acoust. Soc. Am.*, vol. 113, no. 6, pp. 3233–3244, Jun. 2003.

[187] H. Kuttruff, *Room Acoustics*, 4th ed. London: Taylor & Francis Ltd., 2000.

[188] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals.* Englewood Cliffs, New Jersey, USA: Prentice Hall, 1978.

[189] S. Redif, S. Weiss, and J. G. McWhirter, "Sequential matrix diagonalisation algorithms for polynomial EVD of para-Hermitian matrices," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 81–89, Jan. 2015.

[190] J. K. Thomas, L. L. Scharf, and D. W. Tufts, "The probability of a subspace swap in the SVD," *IEEE Trans. Signal Process.*, vol. 43, no. 3, pp. 730–736, 1995.

[191] M. Kunešová, Z. Zajíc, and V. Radová, "Experiments with segmentation in an online speaker diarization system," in *Int. Conf. on Text, Speech and Dialogue (ICTSD)*, ser. Lecture Notes in Computer Science, K. Ekštein and V. Matoušek, Eds. Cham: Springer, 2017, pp. 429–437.

[192] M. Huijbregts, D. Van Leeuwen, and C. Wooters, "Speaker diarization error analysis using oracle components," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, pp. 393–403, Feb. 2012.

[193] J. G. Fiscus, N. Radde, J. S. Garofolo, A. Le, J. Ajot, and C. Laprun, "The rich transcription 2005 spring meeting recognition evaluation," in *International Workshop on Machine Learning for Multimodal Interaction*, 2005, pp. 369–389.

[194] Z. Zajíc, M. Kunešová, and V. Radová, "Investigation of segmentation in i-vector based speaker diarization of telephone speech," in *Proc. Speech and Comput.*, ser. Lecture Notes in Computer Science, A. Ronzhin, R. Potapova, and G. Németh, Eds. Cham: Springer International Publishing, 2016, pp. 411–418.

[195] Z. Zajíc, L. Machlica, and L. Müller, "Initialization of fMLLR with sufficient statistics from similar speakers," in *Proc. Int. Conf. on Text, Speech and Dialogue*, ser. Lecture Notes in Computer Science, I. Habernal and V. Matoušek, Eds. Berlin, Heidelberg: Springer International Publishing, 2011, pp. 187–194.

[196] S. Shum, N. Dehak, E. Chuangsuwanich, D. A. Reynolds, and J. R. Glass, "Exploiting Intra-Conversation Variability for Speaker Diarization," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, Aug. 2011, pp. 945–948.

[197] A. Canavan, D. Graff, and G. Zipperlen, "CALLHOME American English speech," 1997. [Online]. Available: https://catalog.ldc.upenn.edu/LDC97S42

[198] M. Knox, N. Mirghafori, and G. Friedland, "Where did I go wrong?: Identifying troublesome segments for speaker diarization systems," in *Proc. Conf. of Int. Speech Commun. Assoc. (INTERSPEECH)*, vol. 1, Jan. 2012, pp. 486–489.

[199] M. Sinclair, "Where are the challenges in speaker diarization?" in *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Process. (ICASSP)*, Oct. 2013, pp. 7741–7745.

[200] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, and I. Reid, "Joint probabilistic data association revisited," in *Proc. IEEE Int. Conf. on Comput. Vision (ICCV)*, Dec. 2015, pp. 3047–3055.

[201] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multi-hypothesis tracking," in *Proc. SPIE*, vol. 2235, 1994, pp. 5–7.

[202] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *Proc. of the Conf. on Artificial Intell.*, Feb. 2017, pp. 4225–4232.

[203] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. Int. Conf. Image Process.*, Sep. 2017, pp. 3645–3649.

[204] S. K. Pal, A. Pramanik, J. Maiti, and P. Mitra, "Deep learning in multi-object detection and tracking: State of the art," *Appl. Intell.*, vol. 51, no. 9, pp. 6400–6429, Sep. 2021.

[205] G. Ciaparrone, F. Luque Sánchez, S. Tabik, L. Troiano, R. Tagliaferri, and F. Herrera, "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, Mar. 2020.

[206] Q. Li, R. Li, K. Ji, and W. Dai, "Kalman filter and its application," in *Proc. Int. Conf. on Intell. Networks and Intell. Syst. (ICINIS)*, Nov. 2015, pp. 74–77.

[207] I. Marković, J. Ćesić, and I. Petrović, "Von Mises mixture PHD filter," *IEEE Signal Process. Lett.*, vol. 22, no. 12, pp. 2229–2233, Dec. 2015.

[208] Y. Ban, X. Alameda-Pineda, C. Evers, and R. Horaud, "Tracking multiple audio sources with the von Mises distribution and variational EM," *IEEE Signal Process. Lett.*, vol. 26, no. 6, pp. 798–802, Jun. 2019.