School of Electrical Engineering, Computing and Mathematical Sciences

Online Audio-Visual Multi-Source Tracking and Separation: A Labeled Random Finite Set Approach

Jonah Ong Soon Xuan
0000-0002-8019-0099

This thesis is presented for the Degree of
Doctor of Philosophy
of
Curtin University

October 2021

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgment has been made. All copyright material has been reproduced with permission granted by means of credit notice to the publisher, as per IEEE dissertation reuse policy.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Human Ethics (For projects involving human participants for audio-visual recording) The research presented and reported in this thesis was conducted in accordance with the National Health and Medical Research Council National Statement on Ethical Conduct in Human Research (2007) – updated March 2014. The proposed research study received human research ethics approval from the Curtin University Human Research Ethics Committee (EC00262), Approval Number HRE2019-0208.

Human Ethics (For projects involving human participants for subjective listening tests) The research presented and reported in this thesis was conducted in accordance with the National Health and Medical Research Council National Statement on Ethical Conduct in Human Research (2007) – updated March 2014. The proposed research study received human research ethics approval from the Curtin University Human Research Ethics Committee (EC00262), Approval Number HRE2020-0679.

_____

Jonah Ong Soon Xuan
(5 October 2021)

Abstract

Online audio-visual source separation entails the fusion of measurements from multiple modalities to decompose the input mixture signals into the individual source signals. Online operation requires the output signals to be produced synchronously with the input signals up to a fixed delay. Separation is typically challenging for an unknown and time-varying number of moving sources. One approach to this problem is a three-stage process of detection, tracking, and (spatial) filtering (DTF). The construction of a spatial filter to separate each source signal and suppress interfering sources requires knowledge of each source position and its identity. However, audio-visual measurements from the detection stage are typically unlabeled and subject to noise, spurious, and missing measurements. Further, both modalities reside in different observation spaces, even though they observe the same scene. These factors give rise to the multi-modal space-time permutation problem, which hampers separation.

The central aspect of the DTF approach is to solve the permutation problem and estimate the source trajectories (tracking) to facilitate source separation. The labeled random finite set (RFS) framework provides a principled mechanism for fusing multiple modalities in a statistically consistently manner. The framework offers the capability of specifying a stochastic model that characterizes the unknown and time-varying number nature of moving sources, and a joint audio-visual stochastic model that encapsulates the uncertainties in the audio and visual measurements, as well as their respective physical relationships to the sources. Based on these RFS models, a principled and tractable Bayesian recursion can be derived to solve the multi-modal space-time permutation and jointly estimate the source positions and labels in an online fashion.

In support of building an audio-visual system for multi-source separation, this dissertation first explores the difficulties of audio-only separation using the DTF approach. The RFS framework specifies an audio likelihood model that encapsulates the uncertainties in the audio measurements from multiple arrays and their physical relationships with the source states. This dissertation subsequently explores the challenges of 3D multi-object tracking using multiple cameras. The RFS framework specifies a visual likelihood model that describes the uncertainties in the measurements, including the physical relationship between the 2D monocular detections and the 3D object states based on a camera model. Additionally, a tractable 3D detection model that is amenable to Bayesian multi-object tracking is introduced to handle visual occlusions in crowded

scenarios.

Finally, the RFS framework provides a systematic and principled mechanism for combining both audio and visual measurement models. Under the same framework, a tractable online Bayesian tracking algorithm is formulated to facilitate the joint estimation of the source positions and labels. By knowing the number of sources and their respective whereabouts for each time frame, a time-varying set of spatial filters can be constructed to separate each individual source and suppress other interfering sources. Results indicate that the fusion of audio-visual data yields better tracking and separation performances compared to audio data only.

Acknowledgements

First, I would like to express my sincere gratitude to my supervisor, Prof. Ba Tuong Vo. It has been an absolute pleasure working with one of the leading researchers in multi-object tracking. His strong understanding of tracking theory and proficiency in coding speak to his passion and enthusiasm for research. His constant support, guidance, and willingness to share his knowledge and experience tremendously contributed toward every milestone of my Ph.D. journey, making my postgraduate education a very rewarding experience.

I would like to thank Prof. Sven Nordholm for his astute knowledge in signal processing. His teaching and supervision had a huge positive influence on my demeanor and attitude to research. I thank him for his helpful technical advice in all the audio-related experiments I conducted during my research. I am also extremely thankful for his continued dedication and commitment to mentoring a young and budding researcher.

I would also like to thank Prof. Ba-Ngu Vo, a known pioneer in the stochastic geometric approach to multi-object tracking system. His outstanding mathematical and statistical knowledge greatly inspired and motivated me in my learning and research. I particularly thank him for his generous teaching and feedback on the mathematical derivations in this dissertation.

Special thanks to Dr. Du Yong Kim, Dr. Diluka Moratuwage, and Dr. Changbeom Shim for their valuable collaborations. Dr. Kim provided insights and suggestions on visual-related topics. Dr. Moratuwage and Dr. Shim assisted in editing my paper and conducting the experiments in Chapter 5.

I gratefully acknowledge my host institution, the School of Electrical Engineering, Computer and Mathematical Science of Curtin University. I also want to thank the Australian Research Council and Curtin International Postgraduate Research Scholarship (CIPRS) for their financial and administrative support during my Ph.D.

In addition, I thank Dr. Manora Caldera for proofreading this dissertation. Capstone Editing has also provided copyediting and proofreading services according to the guidelines laid out in the university-endorsed national "Guidelines for Editing Research Theses." Finally, my heartfelt appreciation goes to my wonderful partner Grace Tiong, my beloved parents Eng Keng Ong and Shin Lin Yeam, and my loving sisters Jolyn Ong and Joann Ong, for their support and encouragement throughout the course of my studies.

Statement of Contribution

Journal Articles

The following papers have been included in this dissertation:

1. **J. Ong**, B. T. Vo and S. Nordholm, "Blind Separation for Multiple Moving Sources With Labeled Random Finite Sets," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 2137-2151, 2021, doi: 10.1109/TASLP.2021.3087003.

   The author's contribution includes theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper. The co-authors contributed by way of making theoretical developments, drafting, and editing the paper, suggesting the design of the experiments, surveying for suitable existing techniques for comparisons, and providing insights on the evaluation of the source separation results. The paper is the basis for Chapter 3 of the dissertation.

2. **J. Ong**, B. -T. Vo, B. -N. Vo, D. Y. Kim and S. Nordholm, "A Bayesian Filter for Multi-view 3D Multi-object Tracking with Occlusion Handling," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, Issue: 5, pp. 2246-2263, 2022, doi: 10.1109/TPAMI.2020.3034435.

   The author's contribution includes theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper. The co-authors contributed by way of making theoretical developments, drafting and editing the paper, documenting the novel occlusion model and tracking filter, providing insights into state-of-the-art visual detection and tracking algorithms and surveying related datasets for experimental comparisons. The paper is the basis for Chapter 4 of the dissertation.

3. **J. Ong**, B. T. Vo, S. Nordholm, B. -N. Vo, D. Moratuwage and C. Shim, "Audio-Visual Based Online Multi-Source Separation," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30, pp. 1219-1234, 2022, doi: 10.1109/TASLP.2022.3156758.

   The author's contribution includes theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper. The co-authors

contributed by way of making theoretical developments, drafting and editing the paper, providing insights into state-of-the-art visual detection and planning the experiments. The paper is the basis for Chapter 5 of the dissertation, reprinted with permission from IEEE.

Conference Papers

The following papers are related to the research but not included in this dissertation:

1. **J. Ong**, D. Y. Kim and S. Nordholm, "Multi-sensor Multi-target Tracking Using Labelled Random Finite Sets with Homography Data," 2019 International Conference on Control, Automation and Information Sciences (ICCAIS), 2019, pp. 1-7, doi: 10.1109/ICCAIS46528.2019.9074716.

   The author's contribution includes theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper. The co-authors contributed by way of editing the paper and proposing ideas for the experiments.

2. **J. Ong**, D. Y. Kim and C.T. Do, "A Tractable Multi-Target Detection Model for Line-of-Sight Measurements," 2021 International Conference on Control, Automation and Information Sciences (ICCAIS), 2021, doi: 10.1109/ICCAIS52680.2021.9624664.

   The author's contribution includes theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper. The co-authors contributed by way of editing the paper and proposing ideas for the experiments.

To Whom It May Concern,

I Jonah Ong Soon Xuan, contributed to the above listed publications as indicated therein.

_____

(Jonah Ong Soon Xuan)

_____

(Supervisor: Prof. Ba Tuong Vo)

Contents

List of Figures

List of Tables

Nomenclature

$1.(\cdot)$	Inclusion function
F_s	Sampling frequency
$G(\cdot)$	Generalized Sidelobe Canceller weight
$P_D^{(c)}(\cdot)$	Probability of detection of sensor index c
$P_S(\cdot)$	Probability of survival
$R^{(a,b)}(\cdot)$	Generalized cross-correlation function between microphones of indexes a and b
$S_n(\cdot)$	Frequency-domain source signal n
$W(\cdot)$	Frequency-domain beamforming weight
X_k	Unlabeled finite set of object states
$Y_k^{(q,m)}(\cdot)$	Frequency-domain segmented received signal at frame k of microphone m and array q
$Z_k^{(c)}$	Finite set of measurements of sensor index c
Z_k	Finite set of measurements
$\Delta(\cdot)$	Distinct label indicator
Γ	Space of association hypotheses
$\Phi_{\text{PSD}}(\cdot)$	Power spectral density matrix of the microphone array signals
Θ_k	Space of all association mappings at time k
$\Upsilon^{(c)}(\cdot)$	Object state (ellipsoid) to measurement (bounding box) transformation function of camera c
Ξ	Space of association map histories
α	Position vector of an object

$\boldsymbol{X}_k$	Labeled finite set of object states
$\boldsymbol{\pi}_{0:k}(\cdot \mid \cdot)$	Labeled posterior density
$\boldsymbol{\pi}_k(\cdot \mid \cdot)$	Labeled filtering density at time step k
$\boldsymbol{f_B}(\cdot)$	Labeled single-object/multi-object probability density of newly born objects
$\boldsymbol{f_S}(\cdot \mid \cdot)$	Labeled single-object/multi-object Markov transition density of surviving objects
$\boldsymbol{f}_{k\mid k-1}(\cdot \mid \cdot)$	Labeled single-object/multi-object Markov transition density
$\boldsymbol{x}_k$	Object labeled state vector at time step k
$\delta_{\cdot}(\cdot)$	Dirac delta function
$\delta_{\cdot}[\cdot]$	Kronecker delta function
ℓ	A unique discrete label
$\eta(\cdot \mid \cdot)$	Mode/class transition function
γ	Association hypothesis
$\kappa(\cdot)$	Intensity function of a Poisson random finite set
λ	Frequency bin index
λ_p	Rate parameter of a Poisson distribution
$\mathbb{B}$	Space of discrete labels for newly born objects
$\mathbb{L}$	Space of discrete labels
$\mathbb{L}^n$	Space of n-dimensional discrete labels
$\mathbb{N}$	Space of natural numbers
$\mathbb{R}$	Space of real numbers
$\mathbb{R}^n$	Space of n-dimensional real numbers
$\mathbb{X}$	State space
$\mathbb{Z}$	Observation/measurement space
$\mathcal{F}(\cdot)$	Space of finite subsets
$\mathcal{L}(\cdot)$	The label extractor

$\mathcal{M}(\cdot)$	Time-frequency mask
$\mathcal{N}(\cdot; m, \mathrm{P})$	Gaussian density with mean m and covariance P
$\mathcal{P}_{A,k}^{(q)}(\cdot)$	Steered-response power function of array q at frame k
$\mathcal{P}_{V}^{(c)}(\cdot)$	Projection transformation of camera c
$\mathcal{S}^{(c)}(\boldsymbol{x})$	Shadow region of object $\boldsymbol{x}$ relative to camera c
$\mathcal{U}(\cdot)$	A uniform distribution
$\mathcal{Z}(\cdot)$	Conic to bounding box transformation
F_{k-1}	Transition transformation matrix at $k-1$
H_k	Measurement transformation matrix at k
K_k	Kalman gain at k
$\mathrm{P}_{3\times4}^{(c)}$	Camera matrix of camera c
Q_{k-1}	Covariance matrix of process noise at $k-1$
R_k	Covariance matrix of measurement noise at k
S_k	Innovation covariance at k
ω	Angular frequency
$\omega^{(I,\xi)}$	Weight of a GLMB hypothesis or component
ϕ	Sampling period
$\pi_{0:k}(\cdot \mid \cdot)$	Posterior density
$\pi_k(\cdot \mid \cdot)$	Filtering density at time step k
$\psi(\cdot)$	Likelihood ratio
$\rho(\cdot)$	Cardinality distribution
σ	Standard deviation
$\tau(\alpha, u^{(a)})$	Time delay between positions α and $u^{(a)}$
$\tau^{(a,b)}$	Time-difference-of-arrival between two microphones of indexes a and b
θ_k	Association map that maps labels to measurements at time k

υ	Process noise
$\Phi(\cdot)$	Frequency-dependent weighting function
$\varphi(\cdot)$	Room impulse response
$\varpi_T(\cdot)$	Selected window function of length T
ξ	Association map history
$_+$	Subscript used to indicate the next time step or frame
$a(\cdot,\cdot)$	Arbitrary nonlinear function of a dynamic model
c	Index of camera
c_s	Speed of sound propagation
$d(\cdot)$	Frequency-domain steering vector
$f_B(\cdot)$	Single-object/multi-object probability density of newly born objects
$f_S(\cdot \mid \cdot)$	Single-object/multi-object Markov transition density of surviving objects
$f_{k\mid k-1}(\cdot \mid \cdot)$	Single-object/multi-object Markov transition density
$g_k^{(c)}(\cdot \mid \cdot)$	Single-object/multi-object measurement likelihood of sensor index c
$g_k(\cdot \mid \cdot)$	Single-object/multi-object measurement likelihood
$h(\cdot,\cdot)$	Arbitrary nonlinear function of a measurement model
k	Time step or frame
l	Selection vector
o	Discrete mode or class
$p^{(\xi)}(\cdot,\ell)$	Probability density function of the state of object ℓ in label set I for the association map history ξ
q	Index of a microphone array
$r_B(\cdot)$	Probability of object birth
$s_n(\cdot)$	Time-domain source signal n
$u^{(a)}$	Position vector of sensor index a
$v(\cdot)$	Intensity function

w	Measurement noise
x_k	Object state vector at time step k
$y^{(q,m)}(\cdot)$	Time-domain received signal at microphone m of array q
$y_k^{(q,m)}(\cdot)$	Time-domain segmented received signal at frame k of microphone m and array q
z_k	Measurement vector at time step k

Acronyms

DTF Detection, Tracking and Filtering

MOT Multi-Object Tracking

RFS Random Finite Set

FISST Finite Set Statistics

BSS Blind Source Separation

TDOA Time-Difference-Of-Arrival

STFT Short-Time Fourier Transform

RIR Room Impulse Response

GCC Generalized Cross-Correlation

SRP Steered-Response Power

PHAT Phase Transform

SRC Stochastic Region Contraction

ICA Independent Component Analysis

SCA Sparse Component Analysis

NMF Non-negative Matrix Factorization

DAS Delay-And-Sum

LCMV Linearly Constrained Minimum Variance

MVDR Minimum Variance Distortionless Response

GSC Generalized Sidelobe Canceller

SOI Signal Of Interest

ACF Aggregated Channel Features

CNN Convolutional Neural Network

RCNN Region Convolutional Neural Network

RPN Region Proposal Network

CRF Conditional Random Field

DBT Detection-Based Tracking

DFT Detection-Free Tracking

TBD Track-Before-Detect

YOLO You Only Look Once

DSFD Dual-Shot Face Detector

LoS Line of Sight

MCD Multi-Camera Detection

NCV Nearly-Constant-Velocity

NCT Nearly-Constant-Turn

EAP Expected a Posteriori

MAP Maximum a Posteriori

EKF Extended Kalman Filter

UKF Unscented Kalman Filter

SMC Sequential Monte Carlo

NN Nearest Neighbor

GNN Global Nearest Neighbor

MHT Multiple Hypothesis Tracking

JPDA Joint Probabilistic Data Association

MCMC Markov Chain Monte Carlo

i.i.d independently and identically distributed

PHD Probability Hypothesis Density

CPHD Cardinalized Probability Hypothesis Density

CBMeMBer Cardinality Balanced Multi-Bernoulli

LMB Labeled Multi-Bernoulli

GLMB Generalized Labeled Multi-Bernoulli

MS-GLMB Multi-Sensor Generalized Labeled Multi-Bernoulli

MV-GLMB-OC Multi-View GLMB with OCclusion modeling

OSPA Optimal Sub-Pattern Assignment

OSPA(2) Optimal Sub-Pattern Assignment (OSPA-on-OSPA)

Chapter 1

Introduction

1.1 Motivation and Scope

One of the key features of human perception is the ability to fuse information from multiple modalities — chiefly audio and vision senses — to make decisions [1]. Human perception has inspired significant research effort on autonomous systems with audio and visual sensors with the aim of developing a system capable of utilizing data from heterogeneous sensors synergistically to achieve tasks like source/speech separation [2–4] and multi-source (or object) tracking [5–8]. However, in the literature, computer audition (signal processing) and computer vision (image processing) have traditionally proceeded as two independent research fields.

In the field of computer audition, the tasks of signal-based perceptual system are blind source separation (BSS), acoustic source localization, and tracking [4, 8]. BSS is the decomposition of observed mixture signals into individual source signals with little information about the mixing process [4]. BSS has been extensively utilized in wireless communications for speech enhancement and noise suppression [4]. One of the limitations of conventional BSS algorithms is that they require the number of sources, and assume that sources are static. Since, in practical scenarios, sources are typically moving and the number is unknown, source localization and tracking approaches are needed in advance for more sophisticated BSS algorithms [4]. Source localization provides observations of the source positional information, and tracking exploits these observations from past to present to infer the individual source trajectories [8]. The ability to localize and track acoustic sources provides machines with awareness of the surrounding environment, which underpins applications such as acoustic scene analysis [9, 10], spatial filtering for source separation [11, 12], and automatic speech recognition [13].

In the field of computer vision, object detection and multi-object tracking (MOT) have been two ongoing topics of research over the past decade [6, 7, 14]. The task of object detection is to pinpoint object instances in digital images [14]. Object detection is foundational to other important computer vision tasks such as image recognition and

object tracking [15]. Due to the rapid growth of deep learning techniques, state-of-the-art object detectors have now reached real-time speed and accuracy, leading to many real-world applications (e.g., robot vision, smartphone cameras, and video surveillance) [14]. MOT differs from object detection in that it involves filtering detections with respect to the objects, retaining their labels/identities, and generating their individual trajectories [7, 16]. Given an input video, online MOT algorithms attempt to accurately track multiple objects in the least amount of time possible. Real-time implementation is imperative for applications like robot navigation, security surveillance, and autonomous driving [7]. Aside from tracking speed, further MOT difficulties are the occurrences of appearance changes and occlusions, which may lead to falsely terminated tracks and identity switches. There has been much effort in the literature to improve MOT against such adverse conditions [7].

While the separate fields of audio and visual processing have their respective challenges, the processing of both audio and visual data is important for building a more robust perceptual system since many machines are now equipped with both microphones and cameras [1]. A fusion algorithm for multiple microphones and cameras enables a joint analysis of a scene (e.g., audio-visual source tracking and separation, which underpins the development of conferencing applications with multi-modal interfaces, meeting analysis, smart homes and intelligent vehicles systems) [1, 17, 18]. The development of a suitable fusion strategy for the widespread deployment of audio-visual systems is a nontrivial task that requires disciplinary expertise [1]. To this end, several multi-modal fusion strategies have been proposed for audio-visual systems [1, 17, 18].

This dissertation focuses on developing an online audio-visual source separation solution that can handle multiple moving sources whereby the number of sources is time-varying and unknown. The approach taken is based on the process of detection, tracking, and filtering (DTF). The underlying theme of this dissertation is that the proposed solutions are online and model-centric based on the labeled random finite set (RFS) framework [19–22]. Our model-centric approach relies on deriving stochastic and physical models to characterize the problems. In practice, online model-centric approaches are suitable for time-critical applications like audio-visual (online) conferencing, meeting analysis, and smart homes, where the constraints of synchronization, latency and explainability are vital. The following sections detail the research problems, objectives, and contributions of this dissertation.

1.2 Audio-Visual Source Separation

Source separation is the task of decomposing a mixture of signals received at sensor elements into isolated sounds corresponding to individual sources [23, 24]. Source separation approaches can be classified into batch (or offline) processing and online processing. In the context of source separation, an online algorithm is a sampled sys-

tem whereby the analogue mixture (input) and individual source (output) signals are synchronized up to a fixed delay. In contrast, batch or offline algorithms process the entire history of signal samples before producing a decomposition.

Batch Processing

Conventional audio-related separation methods include the independent component analysis (ICA) [25], sparse component analysis (SCA) [26], and non-negative matrix factorization (NMF) [27]. Conventional methods assume a mixing model with a fixed and known number of static sources, and utilize different statistical properties of the source signals to achieve separation [4]. A common theme amongst these methods is the use of a batch processing operation to optimize a cost function based on a particular separation criterion (e.g., independence, sparseness, or non-negativity of the individual signal components in the mixture) [4].

Audio-related data-centric approaches are based on training a deep neural network to achieve separation [28, 29]. The difficulty of training a deep neural network in a scenario of multiple talkers is the label ambiguity problem as there is no information on how to provide a correct reference to the corresponding output layers. The method in [29] proposed a novel permutation invariant training criterion that determines the best output-reference association and then minimizes the error given the association. The method in [28] uses a trained deep network to produce spectrogram embeddings that are discriminative for segmenting and separating the source signals. Both audio-based data-centric approaches have been shown to separate up to three non-moving speakers.

Due to the proliferation of audio-visual applications, data-centric approaches have incorporated visual data in combination with audio data to create a more robust and improved source separation algorithm. Recent works have incorporated deep learning architectures to effectively fuse information from audio and visual data to perform separation [30, 31]. These techniques are inspired by the way humans employ both audio and visual senses to hone in on any speaker of interest in a loud and noisy environment. By exploiting the audio-visual association between lip movements and speech utterances, data-centric approaches rely on training a neural network to learn such audio-visual features to achieve source separation [30].

Though the data-centric approaches have typically shown promising separation results, they require offline training on a large amount of data to function desirably. One of the drawbacks is that training tends to be computationally expensive and restrictive in cases where training data are unavailable. Further, data-centric approaches can be very sensitive to environmental changes and may require constant retraining when there is a model mismatch. Consequently, the abovementioned batch separation approaches are not directly applicable to time-critical applications such as audio-visual online conferencing, particularly where outputs are required to be produced on the fly.

Online Processing

In contrast to batch processing, online separation methods aim to produce the decomposed output signals synchronously with the input mixture signals up to a fixed delay. The online requirement is typically incompatible with batch separation methods which exploit certain statistical properties captured over the entire length of the data. Consequently, online separation approaches are more appealing due to their suitability for real-time applications such as audio-visual online conferencing.

In audio source separation, several online separation approaches have been proposed in the literature [32–35]. Most online separation methods are variations of the conventional (batch) methods through a recursive update of the model parameters. For example, the batch ICA method can be formulated as an incremental estimation of the independent components from online data [32]. In [33], an incremental NMF method based on volume constraint is proposed for online separation. In [35], the author proposes an alternative to the Fourier transform called polyphase subband decomposition that also permits an online implementation. Further recursive algorithms that rely on source independence and non-stationarity to separate the source signals have been proposed in [36–38].

Recent online data-centric approaches are the DANet [39] and TasNet [40]. The DANet [39] method achieves separation via an attractor network that solves the label ambiguity problem without knowing the number of sources. The TasNet [40] method uses an encoder-decoder framework to estimate the source masks, which are then are applied to mixture weights to separate the source signals. Note that while these methods have been shown to meet real-time implementation speed, they still require offline training and are not amenable to applications where training data are unavailable. For audio-visual conferencing applications, it is necessary to use online processing since there are strict requirements on synchronization and latency.

Unknown and Time-Varying Number of Moving Sources

The aforementioned online separation approaches assume sources to be static with a fixed and known number. In a real multi-source scenario with more than one audio and visual sensors, sources are moving, and the number of sources is time-varying and unknown. In this case, conventional separation techniques that assume time-invariant mixing are unsuitable [41, 42]. In addition, it is not clear whether data-centric approaches for either audio or audio-visual data can be extended to accommodate for unknown movement, appearance, and disappearance of sources. Before delving into the approach and challenges, a use-case example is given to show the importance of developing an online separation algorithm that can handle these problems.

Figure 1.1: Online Audio-visual Teleconferencing

Use-case Example

Teleconferencing is an online audio-visual meeting with people who are physically and remotely present. An audio-visual teleconferencing room is typically set up with a control panel, connectivity module, high definition display screens, cameras and microphones as shown in Fig. 1.1. In an online teleconference, a large group of people in an organization can conveniently communicate and exchange collaborative information worldwide. However, audio-visual teleconferencing is not yet as natural as in-person meeting, especially during a collaborative discussion where people may move around and engage in conversations. In this scenario, remote participants may have difficulty listening and understanding multiple concurrent speakers. Moreover, undesired background noise in the room may further corrupt the audio stream. While the number of participants in the meeting is usually known, the number of concurrent speakers is unknown and time-varying throughout the meeting. To this end, having a voice separation feature designed to handle an unknown number of speakers that are moving, enables online users to listen in on any person of interest without interference from other speakers. In addition, it is also advantageous to have an on-screen tracking system that locates and tracks any person of interest as they move, as this helps online users to easily identify the person who is talking.

The Online Model-Centric DTF Approach

To solve the online multi-source separation problem, several audio-related works have applied a tracking algorithm on audio measurements obtained from standard localization algorithms to estimate the trajectories of the sources (i.e., the positions and identities of the sources), and then use adaptations of standard source separation techniques to perform source separation [42–44]. These approaches are based on the model-centric three-step process of detection, tracking, and filtering (DTF), which relies on the physical models of the dynamics of the sources as well as the characteristics of the sensors (cameras and microphones) and their measurements for the estimation of the positions and identities (or labels) of the sources. The labels are important for determining the respective trajectories of the sources. Knowledge of each source trajectory underpins the construction of a model-based separation filter capable of isolating the source signals and suppressing interference from other sources. The model-centric DTF approach has the salient features of being capable of operating online, amenable to audio and visual data, and able to cater for a time-varying and unknown number of moving sources without the need of constant training/retraining.

Adoption of model-centric DTF approaches to audio-visual separation is relatively uncommon in the literature. More importantly, there is currently no online solution for audio-visual separation of a time-varying and unknown number of moving sources. Online solutions are more versatile and applicable to time-critical applications such as online conferencing [17, 18, 45] and meeting analysis [1, 46, 47], where both audio and visual modes are readily available and likely to be more effective than using audio data alone. While there are a few works on online multi-source tracking using either audio-only [42, 44] and audio-visual data [48, 49], these Bayesian tracking algorithms do not estimate the positions and labels of the sources in a statistically consistent manner, as they resolve each source trajectory individually using (heuristic) track management techniques. From a tracking perspective, it is more complete to have a Bayes-optimal solution that jointly estimates the source positions and labels in a principled manner, as it has the potential to improve tracking performance and thus separation performance.

Key Challenges of the Online DTF Approach

Online audio-visual separation of a time-varying and unknown number of moving sources is a challenging problem because the estimation of individual source signals needs to be done for a dynamic scene. In addition, since it is an online process, the generation of the separated signals must be done synchronously with the input signals with minimal delay. The DTF approach has the capability of operating online by utilizing knowledge of the scene (i.e., the positions and labels of the sources) at every time frame to design a time-varying set of separation filters for signal separation and interference suppression. To achieve this, the concept of online detection and tracking is applied to inform

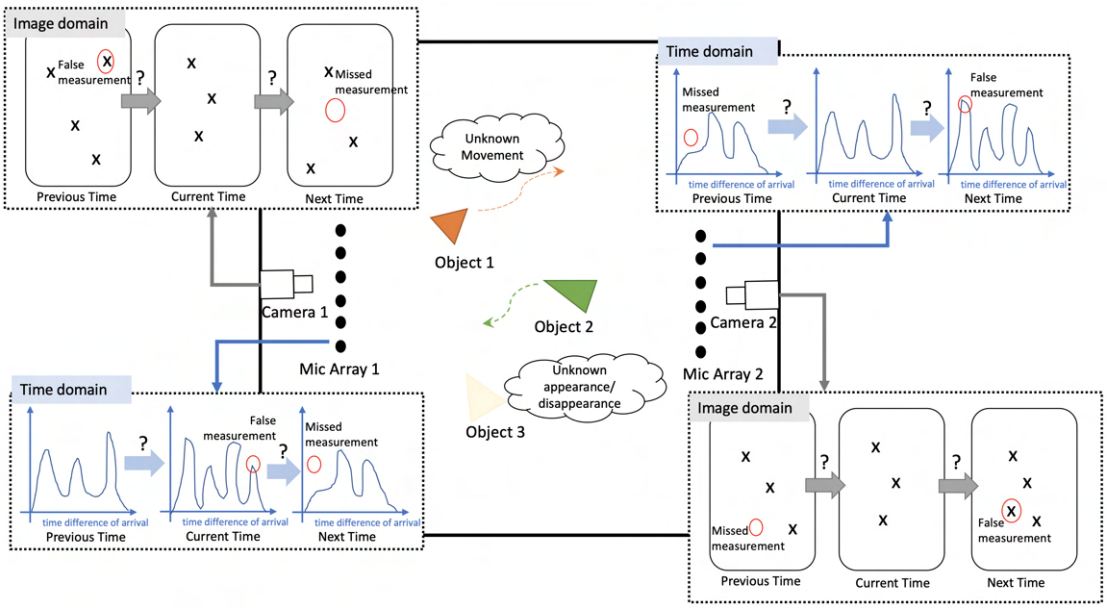


Figure 1.2: Problem Illustration

a separation algorithm.

In the detection stage, audio and visual measurements obtained from the standard detection algorithms are unsuitable for the construction of any separation filter due to the following problems:

- Audio measurements from acoustic localization algorithms are usually in the time or space domain, while visual measurements from detection algorithms are typically in the 2D image domain. As a result, both modalities are different physical quantities that do not fall in the same observation space, even though they observe the same physical space (see Fig. 1.2).

- The audio and visual measurements are unlabeled, and subject to noise, false measurements and missing measurements (see Fig. 1.2).

- As sources are subject to unknown movement, appearance and disappearance over time, all abovementioned factors give rise to the *multi-modal space-time permutation problem*, since it is not known how measurements are associated across domains, and generated by which sources across space and time, if any at all (see Fig. 1.2).

For these reasons, an online Bayesian multi-source tracking system that encapsulates the abovementioned problems and jointly estimates the source labels and positions using the obtained audio and visual measurements at each time frame is required. This solves the inherent multi-modal space-time permutation problem. Based on the tracking estimates at each time frame, a set of spatial filters (beamformers) can be constructed to achieve separation and suppression, all in an online fashion.

1.3 Objectives and Contributions

The dissertation proposes an online and model-centric solution via the DTF approach for audio-visual separation, which is useful for audio-visual conferencing-type applications where the constraints of synchronization and latency are vital. Since audio and visual measurements obtained from standard detection algorithms give rise to the inherent multi-modal space-time permutation problem, an online tracking algorithm plays an integral part in solving the permutation problem in order to achieve separation.

To achieve audio-visual multi-source tracking, a stochastic model that captures the time-varying and unknown nature of multiple sources and a stochastic model that describes the relationship between the multi-modal measurements and source (labeled) states are needed. It is crucial that the stochastic model for the multi-modal measurements captures the following three components:

- the respective physical relationships between the audio and visual measurements, and the source states
- the noise or perturbations in the audio and visual measurements due to hardware and environmental factors
- the inherent multi-modal space-time permutation issue, since it is unknown which measurements in both modes are associated to which sources across space and time, if any at all.

The thesis of this dissertation is that the labeled RFS framework offers a principled mechanism for the modeling and fusion of multiple measurement modalities, thereby enabling online solutions for audio-visual separation problems. To support the thesis, this dissertation first explores the difficulties in tracking and separation using audio data only, and proposes a labeled RFS-based stochastic model for characterizing the nature of the audio measurements and the physical relationship between the audio measurements and source states. Subsequently, the dissertation explores the difficulties in multi-object tracking using visual data only, and proposes a labeled RFS-based stochastic model for characterizing the nature of the visual measurements and the physical relationship between the visual measurements and source states. Finally, given that both modalities observe the same scene, the labeled RFS provides a systematic and principled mechanism for the fusion of both audio and visual measurement models. Under the same framework, an online Bayesian tracking algorithm is constructed to recursively estimate the labels and positions of the sources jointly. Given the online tracking estimates, a time-varying set of spatial filters can then be constructed to separate each source signal and suppress the inference from other sources.

1.3.1 Audio Multi-Source Tracking and Separation

The dissertation first explores the challenges in online tracking and separation for multiple moving sources (i.e., the number of sources is time-varying and unknown) using audio data only. Standard acoustic localization algorithms are used to acquire audio measurements, which are unlabeled, and subject to false negatives, false positives and noise. Moreover, audio measurements are typically based on some form of nonlinear transformation corresponding to the source position, which result in low observability. Multiple microphone arrays are used to mitigate the observability issue since independent sets of audio measurements generated by the arrays can be used collectively to infer the positions of the sources. However, the main challenge is the space-time permutation ambiguity problem, as associations between the multi-array measurements and the sources across space and time are unknown.

The labeled RFS tracking framework provides a principled way of addressing the aforementioned problems and facilitates a joint estimation of the labels and positions of the sources. The framework enables specifications of statistically consistent stochastic models for capturing the physical relationship between the multi-array (audio) measurements and the sources, including the measurement uncertainties. A multi-sensor multi-source Bayes filter is applied to recursively estimate the trajectories of the sources (i.e., the source labels and positions), thereby solving the permutation ambiguity problem. Knowledge of the positions and labels of the sources enables the construction of a time-varying set of spatial filters capable of separating each source signal and suppressing interference from other sources. The overall algorithm is online and scales linearly with the number of arrays.

The proposed online solution is evaluated using real data in mild reverberation and simulated data in low, mild, and high reverberation. To show that the method can handle multiple moving sources that are time-varying and unknown, the experiment is designed to first have an active source moving in the scene, followed by two other sources appearing and moving at different times. The times when the sources appear and disappear are unknown to the system. The Optimal Sub-Pattern Assignment (OSPA) metric [50] is used to evaluate the quality of the multi-array measurements. Subsequently, the tracking results are evaluated using the OSPA$^{(2)}$ metric [51], which is an extension of the OSPA metric capable of penalizing labeling errors in the tracks. Finally, the separation results for both real and simulated data are evaluated using the ITU-T P.835 based listening tests [52]. This contribution has been published in [53] and is the basis for Chapter 3 of this dissertation.

1.3.2 Visual Multi-Object Tracking with Occlusion Handling

The dissertation further explores the task of designing an online multi-camera 3D multi-object tracking (MOT) algorithm with occlusion handling. The interest of visual track-

ing is to estimate the labels and positions of the objects based on visual measurements (e.g., bounding boxes) obtained from standard object detectors. The visual measurements are typically unlabeled with respect to the objects, and are not perfect for state estimation and track management due to noise, missing measurements and false measurements (or clutter). The occurrence of missing and false measurements are typically caused by failure of the detector to correctly register the objects of interest, and by occlusions when objects are obstructed by other objects in the camera. These imperfections in the measurements may potentially give rise to identity switching, track fragmentation, and track loss. Consequently, occlusion handling is an important element of MOT to improve tracking performance.

In a multi-view (or multi-camera) setting, an object that is visually obstructed in one camera may be visible by other cameras, therefore complementary multi-camera data can be exploited to resolve occlusions [54]. Further, the deployment of multiple cameras can potentially improve the overall tracking performance, as data from multiple cameras reduce the uncertainty on the states of the objects. However, the implementation of multi-view MOT is challenging, mainly due to the high-dimensional space-time permutation problem between the objects and visual measurements across different cameras. To date, the best solutions for multi-view MOT are batch algorithms [55–57]. These methods rely on a data-centric multi-camera detector to obtain the object measurements, which are processed to produce tracking estimates in the 2D ground plane. Such multi-camera detectors require training in a high-dimensional input space (as the number of combinations across multiple cameras is large) [58], and retraining when there is a reconfiguration or extension in the multi-camera system.

This dissertation proposes an online multi-view MOT solution that relies on monocular detector training, thereby avoiding any training process when there is an extension or a reconfiguration in the multi-camera system. The solution is a model-centric Bayesian MOT filter derived using the labeled RFS framework to handle the subtasks of state estimation, track management, occlusion handling, and clutter rejection. The framework permits a statistically consistent manner of characterizing the false/missing measurements, random appearance/disappearance of the objects, and the multi-dimensional space-time permutation problem. Further, a novel 3D detection model is proposed to handle occlusions. The proposed detection model is incorporated in the RFS Bayesian tracking filter, which enables the filter to retain occluded tracks correctly.

The proposed multi-view MOT algorithm operates in the 3D world coordinates. The relationship between the objects in 3D and the detections in 2D is established via the camera matrix, which can be obtained using standard camera calibration techniques. Operating in the 3D world frame allows tracking people falling and jumping, which is suitable for applications like school environment monitoring, age care, and sports analytics. Further, the proposed algorithm has a linear complexity in the total number

of measurements (from all cameras), which is desirable for scalability and online implementation. The proposed method is validated on the latest WILDTRACKS dataset and show comparable results with its best-performing method [57]. To evaluate the 3D tracking performance of the proposed method, a new dataset with varying degrees of crowded scenarios is developed. Based this new dataset, an experiment on multi-camera reconfiguration is performed to show that the proposed method is able to operate uninterrupted. Lastly, the proposed method is evaluated on a dataset with people jumping and falling. This contribution has been published in [59] and is the basis for Chapter 4 of this dissertation.

1.3.3 Audio-Visual Multi-Source Tracking and Separation

Lastly, this dissertation achieves online tracking and separation of multiple moving sources (i.e., the number of sources is time-varying and unknown) using audio and visual data. In audio-only tracking, multiple microphone arrays are needed to compensate for the observability issue. By exploiting the complementarity of visual assistance, the inclusion of a camera device with only a single microphone array has the capability to achieve desirable tracking performance and, subsequently, some degree of source separation. The main challenge in fusing both modalities is that they fall in different observation spaces. For example, 2D bounding-box detections from images are in pixel coordinates, while audio measurements obtained via acoustic localization algorithms can be in the form of Cartesian coordinates, time-delays, time-difference-of-arrivals or direction-of-arrivals. In addition, both modalities are susceptible to noise, false measurements and missing measurements. Consequently, these factors give rise to the multi-modal space-time permutation problem as the associations between sources and the multi-modal measurements across space and time are unknown.

To achieve separation, an online multi-source tracking solution is required to address the space-time permutation issue. The proposed tracking solution is a dynamic Bayesian estimation approach formulated under the labeled RFS framework to fuse both audio and visual data in a principled manner for tracking multiple sources. The RFS approach enables the development of a stochastic model that describes the dynamics of the time-varying and unknown number of sources and their physical motions over time, in addition to the development of a stochastic model that captures the physical relationships between the measurements and the sources including the abovementioned uncertainties. The tracking filter is capable of estimating the labels and positions of the sources jointly, thereby addressing the multi-modal space-time permutation problem. The tracking estimates inform the construction of a time-varying set of generalized sidelobe cancellers (GSCs) [60] for achieving source separation and interference suppression.

Experimental verification is carried out on real data with live human speakers. The

algorithm is tested with live human speakers in a near-field scenario and a far-field scenario. In addition, an ablation study for each scenario is presented whereby the measurements, tracking and separation are performed using the audio data only. This is undertaken to demonstrate the improvement in performance due to the combination of audio and visual data. The OSPA [50], OSPA$^{(2)}$ [51], and ITU-T P.835 based listening tests are applied for the evaluation of the audio measurements, tracking, and separation results [52], respectively. This contribution has been submitted for publication [61] and is the basis for Chapter 5 of this dissertation.

Chapter 2

Background

In this dissertation, the detection, tracking and filtering (DTF) approach is applied to online audio-visual separation for multiple moving sources where the number of sources is unknown and changing over time. The online requirement is that output signals are synchronous with the input signals up to a fixed delay. The construction of an online separation filter to isolate a source signal and suppress the interference from other sources requires knowledge of each source position and its unique label for each time frame. However, audio and visual measurements acquired from detection algorithms at each time frame give rise to the multi-modal space-time permutation problem. The central aspect of this approach is to solve the permutation problem and estimate the source trajectories (tracking) in a recursive manner to facilitate the construction of a time-varying set of separation filters. The labeled random finite set (RFS) framework is incorporated to address the problem for audio and visual data separately and jointly.

This chapter lays the foundation for the development of the proposed solutions to audio, visual and audio-visual data. Section 2.1 reviews the techniques for blind source separation (BSS), acoustic source localization and tracking, and spatial filtering. Section 2.2 reviews the evolution of visual object detections, techniques on occlusion-handling, and multi-object tracking (MOT). Section 2.3 reviews related works on audio-visual source separation. Section 2.4 provides a detailed account of the classical Bayesian state estimation or filtering techniques and their principled extension to multiple objects via the RFS system.

2.1 Microphone Array Signal Processing

A microphone array is a set of microphones arranged in specific geometry to sample an acoustic field [62]. Microphone array processing involves the blind extraction (or separation) of source signals and the localization of sources using the (mixture) signal information at the input of the array [62].

The objective of the BSS problem is the decomposition of mixture signals into the

respective source signals with minimal knowledge about the mixing process [4, 23, 24, 63]. This is a complicated problem because the mixture signals are convolutive mixtures of the clean source signals with a corresponding acoustic channel (e.g., a room impulse response (RIR)) [64–66]. Traditional BSS methods are generally constructed based on the assumption that source signals are statistically independent and that sources are static with a known and fixed number [24]. However, in reality, this assumption is often violated, especially in applications like speech separation in meeting room where the speakers are subject to move freely. Accordingly, recent works have been geared toward solving BSS for multiple sources where the number of sources is time-varying and unknown [42, 44].

Source localization is an important aspect in the blind separation of multiple moving sources. In acoustics, localization is achieved by estimating intermediary parameters (e.g., the direction-of-arrival and the time-difference-of-arrival), which are essential cues for inferring the source location via triangulation, multilateration or tracking [8, 67–69]. Alternatively, the use of region search algorithms enables a direct form of estimating the source positions without any intermediary steps [70]. In practice, approaches to source localization are complicated by estimation errors (noise), missing estimates of sources (missing measurements), and false estimates (spurious measurements), which are caused by reverberation, interference, dynamic variations in the source-sensor geometries, and source (speech) inactivity [8].

For the abovementioned reasons, localization algorithms are coupled with specialized tracking algorithms to achieve acoustic source tracking [69, 71–73]. The ability to localize and track multiple acoustic sources informs the construction of a set of separation filters to isolate the respective sources and suppress interference [42, 44]. The following subsections elaborate the aforementioned methods (BSS approaches, acoustic source localization, tracking, and spatial filtering).

2.1.1 Blind Source Separation

Conventional BSS algorithms can be categorized into four groups, namely Independent Component Analysis (ICA) [24, 74–76], Non-negative Matrix Factorization (NMF) [24, 77, 78], Sparse Component Analysis (SCA) [3, 24, 79, 80], and Bounded Component Analysis (BCA) [24, 81–83]. These BSS approaches are typically based on a mixing model that is solved via optimizing a particular cost function based on a separation criterion, e.g., independence, sparseness, or non-negativity [4]. The optimization requires the number of sources to be known in advance and is usually a batch process since the algorithm exploits certain statistical properties and the non-stationarity of the source signals [4].

The ICA method is one of the first BSS solutions that aims to segregate the mixture signal into its additive source signals (subcomponents) by estimating the linear trans-

formation (mixing matrix) via maximizing the independence and non-Gaussianity of source signals [2]. As ICA assumes that the mixing matrix is invertible, it is only optimal for determined and overdetermined cases (where the number of sources is equal or less than the number of microphones) [2, 25]. In addition, under a convolutive mixture model, frequency-domain ICA suffers from the permutation ambiguity problem, whereby the frequency bins may not be properly aligned so that the source signals are correctly reconstructed [84]. To counteract some of ICA limitations, recent extensions of the ICA method, such as the independent vector analysis (IVA) [85] and the independent low-rank matrix analysis (ILRMA) [86, 87], have been proposed. These methods sidestep the inherent permutation ambiguities by defining dependence between multivariate components of the source signals [85] and using nonnegative matrix factorization to model the power spectrograms of the source signals [88].

NMF methods decompose the mixture spectrogram into its constituents in a non-destructive manner based on non-negative constraints [89]. This decomposition can be achieved using several optimization techniques in a supervised or unsupervised manner [90]. The supervised solution of NMF leverages prior knowledge of the kinds of sources present, while the unsupervised solution makes use of the spatial covariance matrix formed by mixtures from distributed (multichannel) microphones [42]. Note that spatial covariance matrix methods are sensitive to initialization, hence, in practice, they require careful fine-tuning in the learning/training stage for good results [41].

SCA methods exploit the sparsity of source signals to achieve separation. The sparseness assumption suggests that significant time-frequency points are likely to be dominated by only one source. This means that there is no overlapping in the time–frequency representations of different sources, which is likely the case for speech signals [91]. One of the well-known SCA methods is the degenerate unmixing estimation technique (DUET), which uses time-frequency masking to separate the sources [91]. The masks are constructed from stereo sensor observations using the clustering of relative attenuation and phase information [26]. DUET has stirred a plethora of demixing methods [92, 93], culminating in the extension of DUET to anechoic and multiple sensors with arbitrary arrangement [94].

The BCA method exploits the geometric boundedness property of source signals as opposed to the source independence condition, for the decomposition of the mixture signal [81]. BCA extracts each individual source component based on convex support and Cartesian decomposition of the mixture signals [4, 81, 83]. To date, BCA has been widely applied to the fields of wireless communication, and is regarded as an emerging technique of BSS [4].

In summary, conventional BSS approaches operate in the batch mode, as they assume static sources and exploit certain statistical properties of the source signals to achieve separation. For an unknown number of sources, several BSS solutions have been proposed in [95, 96]. Online BSS methods are typically variation of the batch

methods through a recursive update of the model parameters [32, 33]. Further online BSS algorithms that rely on source independence and non-stationarity have been proposed in [36–38, 97, 98]. Note that these online separation algorithms have not been demonstrated in a dynamic acoustic environment where sources are moving and are subject to unknown appearance and disappearance.

2.1.2 Acoustic Source Localization

The premise of acoustic localization is that sources at different positions exhibit different relative delays between pairs of spatially separated microphones i.e., the time-difference-of-arrivals (TDOAs), from the source position to the respective microphones in the array. In the presence of multiple arrays, this property enables techniques of triangulation and multilateration to determine the actual source positions [62, 99].

A TDOA of a signal generated by a source at position $\alpha \in \mathbb{R}^3$ between two microphones of indices 1 and 2 located at $u^{(1)} \in \mathbb{R}^3$ and $u^{(2)} \in \mathbb{R}^3$ respectively, is given by:

$$\tau^{(1,2)} = \tau(\alpha, u^{(1)}) - \tau(\alpha, u^{(2)}), \tag{2.1}$$

where

$$\tau(\alpha, u^{(m)}) = \frac{||\alpha - u^{(m)}||}{c_s}, \tag{2.2}$$

is the time-of-travel or time-delay from source position α and microphone position $u^{(m)}$, c_s is the propagation speed of sound, and $\|\cdot\|$ represents the Euclidean norm.

Signal Model

Each source in an acoustic environment is indexed by $n \in \{1, ..., N\}$, where N denotes the number of sources. The source situated at position $\alpha_n \in \mathbb{R}^3$ emits a signal that is denoted by s_n. The source signals impinge on an array of M microphones and the signal received by each microphone element m is contaminated with noise $v^{(m)}$. In a reverberant condition where the multipath effect is present, the received signal at each microphone is modeled as:

$$y^{(m)}(t) = \sum_{n=1}^{N} (s_n * \varphi_{\alpha_n}^{(m)})(t) + v^{(m)}(t), \tag{2.3}$$

where $\varphi_{\alpha_n}^{(m)}$ is the room impulse response (RIR) that captures the signal's multipath and direct-path propagations.

Estimating the TDOAs based on the signal model above is an extremely difficult

problem, hence it is simpler to assume the (direct-path) signal model below [62]:

$$y^{(m)}(t) \approx \sum_{n=1}^{N} \frac{s_n\left(t - \tau(\alpha_n, u^{(m)})\right)}{4\pi||\alpha_n - u^{(m)}||} + v^{(m)}(t). \tag{2.4}$$

To estimate the delay parameter, acoustic localization approaches exploit the cross-correlation between signals of various microphone pairs. In the literature, traditional approaches to acoustic source localization can be categorized into two main classes: indirect and direct [100]. Additionally, there has been a recent development of source localization using deep learning as detailed below.

Indirect Localization Approach

The indirect approach has two steps. First, estimate the TDOAs between a pair of microphones in the array. Then, based on the geometry of the array and the estimated delays, estimate the source positions [101]. Consider the direct-path model with only two microphones in an array (i.e., $M = 2$), the estimation of the TDOA can be achieved via the generalized cross-correlation (GCC) function, which is the inverse Fourier transform of a cross spectral density of two observed signals in the frequency domain along with a frequency-dependent weighting function $\Phi(\omega)$ [102]:

$$R^{(1,2)}(\tau) = \int_{-\infty}^{\infty} \Phi(\omega) Y^{(1)}(\omega) Y^{(2)^*}(\omega) e^{j\omega\tau} d\omega, \tag{2.5}$$

where $Y^{(m)}(\omega)$ denotes the Fourier transformed observed signal $y^{(m)}$ from microphone of index m and the asterisk $*$ denotes the complex conjugate. One common weighting function that is purposed to pre-whiten the correlated speech signals is known as the phase transform (PHAT) [103]:

$$\Phi(\omega) = \frac{1}{|Y^{(1)}(\omega) Y^{(2)^*}(\omega)|}. \tag{2.6}$$

In a single source scenario (i.e., $n = N = 1$), the TDOA estimate for the source can be obtained by maximizing the GCC-PHAT function $\hat{\tau}^{(1,2)} = \arg\max_\tau R^{(1,2)}(\tau)$ [102]. Subsequently, the estimated TDOA is used to determine the source directions, for example, using triangulation [104–106], or multi-dimensional lookup tables [107].

In the presence of multiple sources, the GCC function comprises the cross-correlations of various paths due to the respective sources, yielding multiple peaks in the GCC function. In this case, multiple TDOAs are estimated from the GCC function via picking more than one peak in (2.5) [72]. However, in a realistic acoustic environment, the source signal is immersed in ambient noise and subject to a multipath propagation. The noise and multipath propagation effect cause perturbed and spurious peaks in the GCC function, which give rise to noisy, false and missing TDOA measurements.

Direct Localization Approach

Direct approaches perform the estimations of TDOAs and source positions all in a single step by searching over the space of source location and choosing the most probable source location based on significant sound intensity [108–110]. One example of the direct approach is the steered response power phase transform (SRP-PHAT) localization algorithm. The SRP-PHAT is interpreted as a spatial filtering technique that relies on the array's ability to focus on signals originating from a particular position or direction in space [109].

Given a microphone array of M microphone elements, the SRP-PHAT is given by [103]:

$$\mathcal{P}(\alpha) = \sum_{a=1}^{M-1} \sum_{b=a+1}^{M} \int_{-\infty}^{\infty} \frac{Y^{(a)}(\omega) Y^{*(b)}(\omega)}{\left| Y^{(a)}(\omega) Y^{*(b)}(\omega) \right|} e^{j\omega\left(\tau(\alpha,u^{(b)}) - \tau(\alpha,u^{(a)})\right)} d\omega, \tag{2.7}$$

where $\alpha \in \mathbb{R}^3$ is the source position and $Y^{(i)}(\omega)$ denotes the Fourier transformed observed signal $y^{(m)}$ from microphone m.

Assuming that there is only a single source, the SRP-PHAT algorithm aims to search for the source position estimate $\hat{\alpha}$ by maximizing (2.7), i.e., $\hat{\alpha} = \arg\max_{\alpha} \mathcal{P}(\alpha)$. Unlike the GCC-PHAT, localization using the SRP-PHAT approach is computationally expensive due to a large dimensional search in the SRP space [103]. Effective optimization algorithms, for example, the coarse-to-fine region contraction and the stochastic region contraction (SRC) have been shown to be capable of reducing the computational cost without loss of accuracy [70, 111]. Further, the work in [109] has proposed a modified SRP-PHAT approach that performs an exploration search over sampled spaces produced by a GCC function [109].

Similar to the GCC-PHAT approach, in the presence of multiple sources, the position estimates can be obtained via peak-picking (2.7) with a certain threshold. Nonetheless, the presence of ambient noise and significant multipath in highly reverberant environments give rise to noisy, false, and missing source position candidates or measurements.

Deep Learning Approach

In the recent years, there has been an active research on deep learning strategies for acoustic source localization [112]. Most deep-learning based localization approaches typically rely on different kinds of input features that are extracted from the microphone signals [113]. These extracted features are generally low-level signal representations like spectrograms, waveforms, or adopted from traditional signal processing methods, e.g., GCC-PHAT and SRP-PHAT. In standard machine learning fashion, these input features are eventually fed into a deep neural network (DNN) which estimates the source locations or direction-of-arrivals (DOAs) based on different output strategies [113].

In general, there are two output strategies in a DNN to estimate the locations or

DOAs of the sources, i.e., classification and regression [113]. In a classification paradigm, the localization search space is divided into equal size grids that are associated to different classes, and the DNN generates a probability value to each class. Such a classification strategy has been applied to localizing multiple sources, as the DNN is trained to determine the probability of a source activity without prior knowledge of the number of sources [113]. In contrast, a regression strategy directly estimates the values of source positions or DOAs, which are usually in the spherical or Cartesian coordinates. However, to build a robust DNN model with high estimation accuracy, a sufficiently diverse training dataset is generally required [113].

Convolutional neural networks (CNN) have been widely adopted in source localization, due to their property of being translation invariant [114]. The works in [115, 116] are some of the early research papers that have shown to use CNNs to estimate the azimuth angles of multiple speakers under high reverberation. The approach in [115] cast a source DOA estimation into a deep CNN classification problem. The idea is to first transform the received microphone signals into the STFT domain, then utilizing the phase component of the STFT coefficients to learn relevant features for the estimation of source DOAs. As an extension to this work, the authors have used the W-disjoint orthogonality (WDO) assumption [26] on speech signals to achieve a CNN that learns from phase correlations between adjacent microphone signals [116]. More recent deep-learning based localization techniques have shown to directly input raw multichannel waveforms into a CNN to predict the Cartesian coordinates of the sources [117, 118].

Consequently, DNNs have shown to be powerful models that are able to learn the relationships between raw microphone signals and source locations under reverberant and noisy environment. To achieve great results, training a DNN model for source localization requires a sufficiently large and discriminative amount of training examples [112]. As a result, one of the major drawbacks of neural network localization methods is the lack of generality because a network that is trained for a particular sensor configuration may not produce desirable results once the setting has changed [113].

2.1.3 Acoustic Source Tracking

Acoustic source localization approaches only provide estimates of the source TDOA or position, without the dependence of past observations [8]. These obtained source TDOAs and positions are referred to as acoustic (or audio) measurements that are unlabeled and cannot be easily joined with measurements from the past to get the sources' trajectories [8]. Moreover, localization systems are prone to missing and spurious candidates, as well as localization errors due to reverberation and noise in real scenarios. Specialized tracking algorithms are commonly applied to address the abovementioned issues.

Bayesian state estimation algorithms apply a two-stage process that uses past in-

formation to predict the source locations via a dynamic motion model and corrects the predicted estimates based on the measurements/candidates produced by the localization system via a likelihood model [8]. Classical dynamic state estimation technique (e.g., particle filtering) has been shown to track a single source in [69, 71, 119]. For scenarios where the number of sources is changing over time and unknown, tracking using the classical single-object Bayesian tracking framework is not suitable because it does not incorporate the uncertainty in the number of sources [8].

The tracking of multiple sources entails solving the inherent space-time permutation problem because the associations between the sources and audio measurements across space and time are unknown. This problem also includes the possible appearance and disappearance of the sources, which are unknown to the system. In the literature, specialized online multi-object tracking algorithms, for example, the Rao-Blackwellised particle filter (RBPF) [42, 120], the probabilistic data association (PDA) filter [121], and the probabilistic multiple hypothesis tracker (PMHT) [44] have been demonstrated for tracking multiple moving sources.

The random finite set (RFS) framework [122] provides a principled mechanism for accommodating a time-varying and unknown number of sources, and is directly applicable to acoustic tracking [8]. Online RFS Bayesian methods have been applied to tracking multiple acoustic sources in [72, 123, 124]. Further, the probability hypothesis density (PHD) filter [10, 43, 125, 126], the cardinalized PHD (CPHD) filter [127, 128], the cardinality-balanced multi-object multi-Bernoulli (CBMeMBer) filter [129], and the RFS particle filter [130], have also been demonstrated for tracking multiple sources. These RFS-based filtering algorithms are elaborated on in Section 2.4.

While the above online multi-object tracking algorithms are capable of handling multiple sources, they have not provided a statistically consistent approach for estimating the source positions and identities (or labels) of the sources, which are essential for determining the respective trajectories of the sources. Instead, they rely on additional post-processing (e.g., a track management scheme) to obtain the source trajectories. Further, when multiple microphone arrays are used to observe the tracking scene, the number of measurements acquired in a multi-array system poses a highly complex (multidimensional) data association issue. From a tracking perspective, it is desirable to have a tracking solution that jointly estimates the source positions and labels in a principled manner, as tracking performance can potentially be improved. In addition, a tracking solution that scales linearly in the number of microphone arrays (sensors) and measurements provides the benefits of extensibility and scalability.

2.1.4 Spatial Filtering for Moving Source Separation

Conventional BSS approaches based on time-invariant mixing (i.e., for static and fixed number of sources) are not directly suitable for conditions where sources are moving

with respect to the microphones [42]. In a short-time block/frame, the assumption of time-invariant mixing can be achieved by treating moving sources as stationary within that time block [42]. Block-wise ICA approaches in [131–133] separate moving sources by propagating a block-wise mixing matrix that adapts and preserves source ordering in successive blocks. In [41, 89], a multichannel NMF method has been demonstrated for separating moving sources but the overall algorithm requires an additional state-of-the-art separation technique to assist the initialization in a blind setting. Alternatively, blind moving source separation can be achieved by using spatial filtering.

Spatial filtering or beamforming is a technique used in microphone array processing for separating a target source signal by forming a beam directed at a desired location [63]. The operation of spatial filtering consists of the following steps, synchronization and weight-and-sum [62]. In the first step, each microphone element signal is delayed by a certain amount of time to synchronize the signal components coming from a desired direction. In the second step, the aligned signals are weighted and added to form a beamformed output [62].

Delay-and-Sum Beamformer

A standard delay-and-sum (DAS) beamformer is classified as a fixed beamforming method that does not depend on statistical properties of the observed data [60]. The important aspect of a DAS beamformer is the filter weight design. In the frequency domain, these weights represent a phase correction term that corresponds to time-aligning the signals from all microphone elements in the time domain, which are then added up to form a single beamformed signal.

Based on the signal model in Section 2.1.2, let $Y^{(m)}(\omega)$ be the Fourier transform of an observed signal $y^{(m)}$ at microphone of index m, the DAS beamformer is expressed as [62]:

$$S^{(\mathrm{DAS})}(\omega) = \sum_{m=1}^{M} \left(W^{(m)}(\omega)\right)^{H} Y^{(m)}(\omega), \tag{2.8}$$

where ^{H} is the Hermitian transpose, M is the number of microphone elements, $W^{(m)}(\omega)$ is the beamformer weight for each element, and $S^{(\mathrm{DAS})}(\omega)$ is the beamformed signal in the frequency domain. One of the drawbacks of a DAS beamformer is the directivity issue at lower frequencies [62]. Moreover, the presence of significant off-axis pick-ups (i.e., the beamformer side-lobes) causes leakages in the beamformed signal. Therefore, estimating the beamformer weights in an adaptive manner using statistical characteristics of the signals and noise has the potential to improve the performance of the beamformer [60].

Minimum Variance Distortionless Response

The minimum variance distortionless response (MVDR) beamformer is one of the most widely used optimal beamformers whose goal is to minimize the variance of the recorded signal [62]. Given that Φ_{PSD} is the power spectral density matrix of the microphone array input signals, MVDR seeks to minimize the output power to acquire the filter weights W [62]:

$$\min_{W} W(\omega)^T \Phi_{\text{PSD}}(\omega) W(\omega) \qquad \text{subject to} \quad W(\omega)^T d(\omega) = 1, \tag{2.9}$$

where

$$d(\omega) = \left[e^{j\omega\left(\tau(\alpha, u^{(1)})\right)}, \ldots, e^{j\omega\left(\tau(\alpha, u^{(M)})\right)} \right]^T \tag{2.10}$$

is the representation of the delays or travel times (refer to (2.2)) in the frequency domain which depend on the actual geometry of the array, i.e., the positions of individual microphones $u^{(1)}, \ldots, u^{(M)}$ and the position of the source signal α. The use of Lagrange multipliers has been demonstrated to solve (2.9) [62]. Assuming that noise and the desired signal are independent, the sum of the variances of noise and the desired signal is the variance of the observed signal [62]. Thus, the MVDR beamformer aims to minimize this sum, which keeps the desired signal while mitigating the effect of the noise.

Linearly Constrained Minimum Variance

The linearly constrained minimum variance (LCMV) beamformer aims to keep the desired signal of interest while allowing for addition constraints to suppress any known interfering signals (or jamming signals) [60]. This requires not only the desired signal position to be known but also the positions of interfering (undesired) signals. The LCMV seeks to minimize the same expression as MVDR expressed in (2.9), but with different constraints designed to keep the desired signal and cancel any known jammer directions [60, 134]:

$$\text{subject to} \quad \mathrm{D}(\omega)^H W(\omega) = l_N(n), \tag{2.11}$$

$$\mathrm{D}(\omega) = \begin{bmatrix} e^{j\omega\left(\tau(\alpha_1, u^{(1)})\right)} & \ldots & e^{j\omega\left(\tau(\alpha_N, u^{(1)})\right)} \\ \vdots & \ddots & \vdots \\ e^{j\omega\left(\tau(\alpha_1, u^{(M)})\right)} & \ldots & e^{j\omega\left(\tau(\alpha_N, u^{(M)})\right)} \end{bmatrix}, \tag{2.12}$$

where $\tau(\alpha_i, u^{(j)})$ is the time of travel from source position α_i to microphone position $u^{(j)}$, l_N is a selection vector whose dimension varies depending on the estimated number of sources N, i.e., $l_N(n) = [\delta_1[n], \ldots, \delta_N[n]]^T$ such that $\delta_a[b]$ equals to one if $a = b$, otherwise it equals to zero. An example of designing the LCMV to keep a desired signal while suppressing the other interfering source signals is by selecting $n = 1$ as the signal of

interest with its location as α_1, and the interfering sources as $n = 2, ..., N$ with locations $\{\alpha_n\}_{n=2}^{N}$. The solution to (2.11) can be obtained using Lagrange multipliers [60, 134].

Generalized Sidelobe Canceller

A generalized sidelobe canceller (GSC) is an unconstrained version of an LCMV whose objective is to steer a beam toward the desired signal and suppress interference by nulling the interfering signals [60]. The GSC composes of an upper beamformer that keeps the desired signal and a blocking system that allows any signal other than the desired signal from entering the canceller [60]. The upper part of the GSC, in its simplest form, can be the DAS beamformer (2.8), and the bottom part comprises a blocking matrix $\mathrm{B}(\omega)$ given by[60]:

$$\mathrm{B}(\omega) = \mathrm{I} - W(\omega)\left[(W(\omega))^H W(\omega)\right]^{-1}(W(\omega))^H, \tag{2.13}$$

where $W(\omega) = [W^{(1)}(\omega), ..., W^{(M)}(\omega)]^T$ is the upper beamformer weights, and I is an identity matrix. The weight vector of the GSC $G(\omega)$ is given by [60]:

$$G(\omega) = W(\omega) - \mathrm{B}(\omega)V(\omega). \tag{2.14}$$

The weights $V(\omega)$ are computed by minimizing the spectral density error [60]:

$$\epsilon(\omega) = (W(\omega) - \mathrm{B}(\omega)V)^H Y(\omega),$$

where $Y(\omega) = [Y^{(1)}(\omega), ..., Y^{(M)}(\omega)]^T$ is the frequency-domain observation signals from all microphone elements. The output of the GSC is therefore given by [60]:

$$S^{(\mathrm{GSC})}(\omega) = (G(\omega))^H Y(\omega). \tag{2.15}$$

In summary, the GSC is derived based on the fact that the weights of the LCMV beamformer can be decomposed into two parts, one part fulfilling the constraint and the other part responsible for minimization. One of the advantages of the GSC is that it can be solved in an unconstrained manner which makes it more efficient than solving the abovementioned MVDR and LCMV beamformers [135]. Furthermore, it has also been shown that the GSC is capable of producing good suppression under strong interference from concurrent signals. However, a slight disadvantage of a GSC filter is that signal distortions can be difficult to control since it is inevitable to have a mismatch between design model and real-life scenario [60]. Based on these properties and the relative advantages, the GSC beamformer has been adopted as the spatial filter for the proposed online separation algorithms discussed in this dissertation.

Application of Spatial Filtering to Multi-Source Separation

The use of beamformers for source separation requires some knowledge of the sources (i.e., the source spatial positions or direction-of-arrivals) [66]. Because of this, complementary algorithms for estimating the beamformer parameters are usually applied to achieve source separation in a blind setting. For example, in [136], a blind approach for learning the beamformer parameters was proposed based on the speakers' TDOAs. Alternatively, blind moving source separation can be achieved using a Bayesian tracking algorithm to estimate the source trajectory (the position and label) based on acoustic measurements obtained from an acoustic localization technique, then using the information to construct a beamformer. Salient features of the Bayesian approach are that possible outliers are incorporated into the problem formulation and the source location is sequentially predicted and corrected with respect to time via a Markov transition model and a likelihood model, respectively [8].

The strategy of acoustic detection, tracking and filtering (DTF) is adopted in [42, 44, 137] for blind separation of multiple moving sources where the number of sources is unknown and subject to change over time. In [44], the PMHT is used to facilitate the construction of an MVDR filter for source separation, while in [42], the RBPF is used to inform a NMF separation filter. In the context of spatial filtering, it is important to point out that knowledge of the source positions and their unique labels (trajectories), are necessary for constructing a beamformer capable of separating a source and suppressing the other interfering sources. However, the above multi-source tracking algorithms do not jointly estimate the source positions and labels in statistically consistent manner. Instead, they rely on track management (post-processing) techniques to resolve each source trajectory individually. For online conferencing applications, it is desirable to have a tracking system that jointly estimates the source positions and labels for each time frame, so that a time-varying set of spatial filters can be constructed to perform separation and suppression, all in an online fashion.

2.2 Computer Vision and Image Processing

Computer vision is the study of extracting useful information from images [138]. Researchers in the field of computer vision have been developing techniques to detect and recognize objects, and to reconstruct their properties, such as shape, illumination, and color distribution [139]. Vision sensors observe things in a 3D world, and when computers attempt to analyze objects in 3D space, the visual sensors (cameras) usually give 2D images due to projective geometry. This projection to a lower dimension incurs an enormous loss of information [139]. Consequently, we seek to recover some unknowns and details of the 3D world, given information from a 2D image, to fully specify a solution [139].

The last 20 years have seen rapid progress in the field of computer vision due to improved processing power, storage capacity, and random-access-memory of state-of-the-art computers. A few commonly known visual applications include object detections, multi-object tracking, 3D modeling, medical imaging, optical character recognition, and surveillance [139]. As a portion of this dissertation is focused on visual multi-object tracking (MOT), the following subsections review visual object detection, occlusion-handling techniques, and MOT.

2.2.1 Visual Object Detection

The aim of object detection is to devise computational techniques for determining the class of an object in the image and returning the spatial location and extent of that object if present [140]. In the past two decades, researchers have oriented toward solving the following difficulties in object detection: object localization accuracy, object rotation and scale changes, detection speed (capable of real-time use), and object occlusions [14]. Over years of development, object detectors have been through two evolutionary periods: the traditional period and deep learning period. [14].

Traditional Object Detectors

In the traditional object detection period, Viola and Jones invented the first human face detector using the concept of sliding windows that slide through all locations and scales in the image [141]. The histogram of oriented gradients (HOG) feature descriptor [142] was then proposed as a significant improvement for scale invariant feature transform (SIFT) [143, 144] and shape contexts [145], primarily for pedestrian detection. The HOG detector inspired the development of the deformable part-based model (DPM) detector [146–149].

A typical DPM detector consists of a root-filter that is equivalent to a HOG model with image pyramid for capturing the object boundary, and a number of part-filters or the so-called the deformable parts, which capture finer resolution edges and details of the object [146]. A sliding window strategy is adopted by the detector to run a classifier over the entire image [146]. For improving detection accuracy, the DPM has incorporated hard negative mining and bounding box regression, both of which are valuable insights that inspire many subsequent object detectors in the literature [14].

The downside of HOG and DPM detectors is that the computation of features at every scale of an image is very slow [14]. This is mainly due to the use of the classical sliding window strategy, which generally involves a large number of windows that do not scale linearly with the number of image pixels [15]. Further, the HOG and DPM detectors require searching over multiple aspect ratios and image scales to achieve robust detection [15]. Because of this, improved detection accuracy is always accompanied by increased computational cost.

The Aggregated Channel Features (ACF) object detector has subsequently been proposed to counteract slow detection speed without accuracy loss by extracting features directly as pixel values in extended channels, instead of direct feature computation [150]. The types of channels include color, gradient magnitude, and gradient histogram [151, 152]. ACF not only has the advantage over accelerated detection speed but also over richer representation, and more accurate localization of objects [150].

Deep Learning based Detectors

The aforementioned object detectors rely on handcrafted features and they have become saturated in terms of detection performances and speed [14]. This is because handcrafted features are sophisticated feature representations due to the lack of effective image representation and computing resources 25 years ago [140]. As opposed to traditional handcrafted features, a more robust and information-rich feature representation of the image can be achieved via a deep convolutional neural network (CNN) [153]. CNN are superior to handcrafted features because they are able to exploit properties like compositional hierarchies, translation invariance, and local connectivity [15].

A typical CNN is designed to adaptively learn object features through a hierarchical structure composed of multiple convolutional layers, pooling layers, and fully connected layers [153]. The operation of a CNN begins with a convolution that is performed on an image with N 2D convolutional kernels (or filters) to obtain feature maps. Each convolution is subject to a bias term and a nonlinear operation, which is referred to as rectified linear unit (ReLU). Finally, the process of pooling downsamples/upsamples the feature maps. The sequential operations of convolution, ReLU, and pooling are illustrated in Fig. 2.1. Note that every subsequent step results in a reduction of resolution/dimension up to the fully connected layer. The fully connected layer is capable of recognizing global patterns and providing a compact representation of features for classification/detection [15].

Training a CNN requires optimizing an objective function (e.g., a mean squared error loss) using stochastic gradient descent [153]. The accuracy of a deep CNN detector is dependent on the different feature extraction networks, which are called the "engines" or the backbones of a detector. Over the years, many well-known backbones of a CNN detector have been proposed, e.g., GoogLeNet [154], AlexNet [153], VGG [155], and ResNet [156]. Despite the attractive qualities of a CNN detector, the implementation is prohibitively expensive in high-resolution and large-scale images, or more importantly, CNNs cannot straightforwardly segment images into multiple objects [140]. Therefore, the construction of effective and efficient detection algorithms is central to reducing this computational cost. Milestone approaches proposed to overcome the computational bottlenecks can be categorized into two classes: a two-stage detection framework that has a separate object (or region) proposal module for detection, and a one-stage detection framework that unifies the process of object proposal and detection [15].

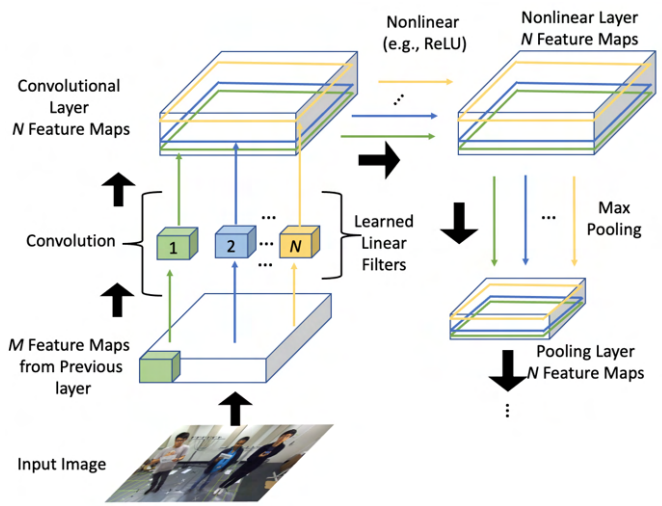


Figure 2.1: The three repeated operations of a CNN: convolution with linear filters, nonlinearities (e.g., ReLU), and local pooling (e.g., max pooling). N filters are applied to M feature maps from previous layer. The result of this operation is passed into a nonlinear function (e.g., ReLU) and pooled to obtain feature maps at a lower resolution.

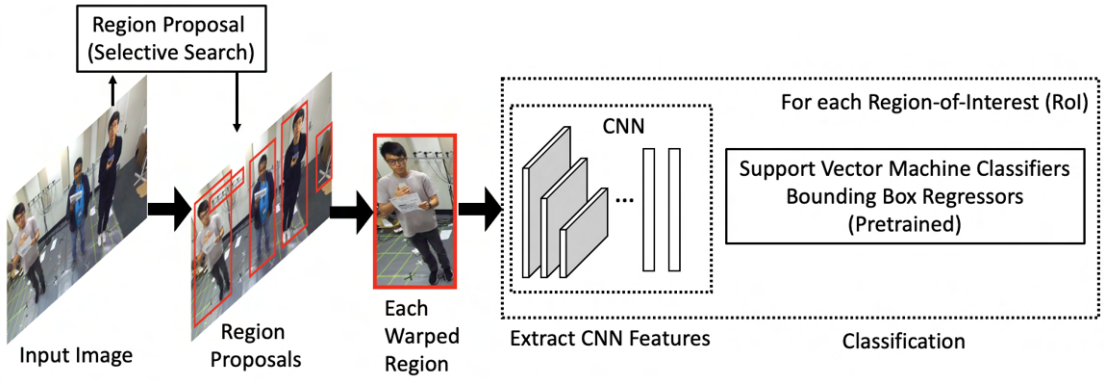


Figure 2.2: RCNN Pipeline

Two-Stage RCNN

Region CNN (RCNN) incorporates a region proposal algorithm in a CNN, which is responsible for generating a set of candidate regions containing an object [157]. This strategy modularizes the detection pipeline so that computational bottleneck is alleviated and overall system efficiency improved [157]. Training an RCNN consists of the following steps [15]. First is the use of selective search [158] to propose candidate regions (bounding boxes) that might contain objects. Then, these regions are warped into fixed standard size images and used for training a CNN model [153]. Lastly, features extracted by the CNN are used to train the back-end classifiers, while the bounding box regressors are trained on each object class to refine the localization of bounding boxes [15]. These steps are illustrated in Fig. 2.2.

While RCNN has been shown to achieve significant performance and speed improvements over its predecessor, there are several shortcomings [15]. First, the training

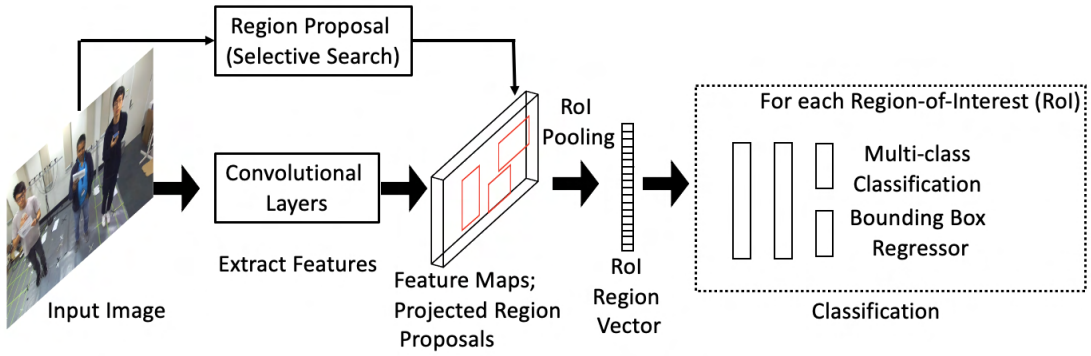


Figure 2.3: Fast RCNN Pipeline

process of the classifier and bounding box regressor are not scalable with the number object proposals in each image, making it expensive both in memory and time for large-scale detection [15]. Second, the training process is a multistage pipeline, which is computationally expensive and hard to optimize [15]. Lastly, the testing process is slow because the CNN features are selected from every region proposal of the image without computation sharing [15].

Two-Stage Fast RCNN

Fast RCNN [159] trains the classifiers and class-specific bounding box regressors simultaneously, as opposed to separately in RCNN [15]. The detector computes the convolution across region proposals efficiently (by computation sharing) and includes a region of interest (RoI) layer in the network [15] (see Fig. 2.3). Pooling is performed on the RoI layer to generate a feature vector for each region proposal [15]. These features are fed into a series of fully-connected layers which then lead into the classifiers and bounding box regressors for object category prediction and refinement, respectively [15]. Compared to RCNN, Fast RCNN trains three times faster and is ten times faster in testing with higher detection accuracy [15]. Nonetheless, the approach still relies on an external region proposal module, which remains as a computational bottleneck.

Two-Stage Faster RCNN

Faster RCNN [160] improves detection speed by introducing a region proposal network (RPN) in place of the conventional selective search. Reference boxes (or anchors) of different scales are initialized by the RPN in the convolutional feature map [15]. Each anchor is mapped into a vector of low dimension and fed into two sibling fully connected layers (i.e., the bounding box regression layer and the classification layer), as shown in Fig. 2.4 [15]. The RPN shares convolutional features, thereby enabling a highly efficient region proposal computation. In summary, the RPN has been shown to efficiently generate region proposals with multiple scales and aspect ratios without much computational bottleneck [160].

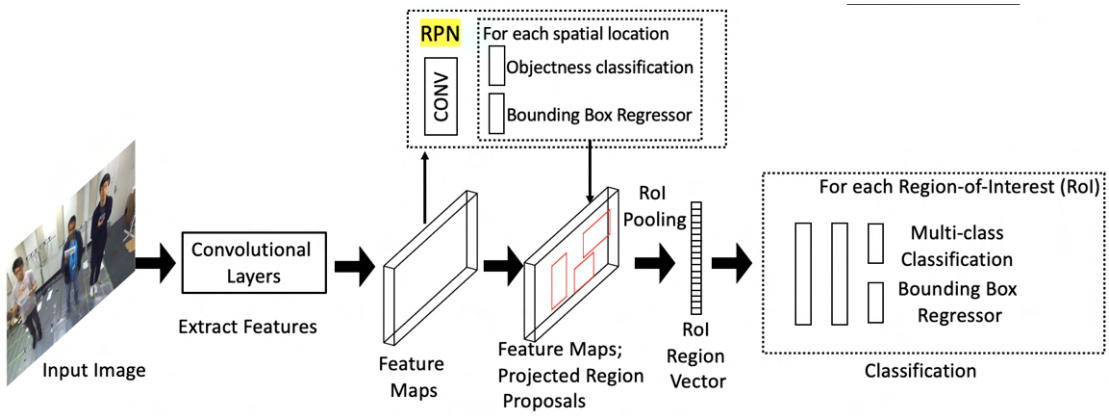


Figure 2.4: Faster RCNN Pipeline

Two-Stage Mask and Mesh RCNN

An alternative way of proposing object candidates is the use of instance segmentation [161, 162]. Unlike region proposal methods, segmentation is a more challenging task that requires the predictions of all objects along with their per-pixel segmentation masks [163]. The work in [161] introduces segment proposals using a deep network called DeepMask, which is responsible of predicting a segmentation mask and an associated objectness score corresponding to how likely an object is present. The author later introduced a second version of this network called SharpMask that can efficiently incorporate rich spatial and strong semantic information to generate the object masks [162]. Since segmentation is more information-rich than bounding box proposal, several works have used instance segmentation to improve the performance of object detection [15]. A recent technique that has taken advantage of object instance segmentation is Mask RCNN, which is an extension of Faster RCNN by including an operation of generating object masks alongside the existing operation of bounding box generation [164]. As a follow-up, the recently proposed Mesh RCNN augments Mask RCNN with a mesh prediction operation, allowing the inference of 3D object shapes [165].

One-Stage YOLO

The aforementioned region-based (two-stage) approaches have demonstrated reliable detection performance, however the pipeline of a region-based detector is slow and each individual component requires to be trained separately [15]. Instead of training a complex region-based neural network, researchers have investigated a unified detection pipeline for high-speed detectors. The state-of-the-art object detector You Only Look Once (YOLO) has adopted the unified detection pipeline wherein object bounding boxes and class probabilities are predicted directly from the images with a single CNN, without the use of a region proposal module, as shown in Fig. 2.5 [15].

A YOLO detector divides the input image into grids, where each grid is responsible for predicting a number of bounding boxes and the confidence scores for those boxes

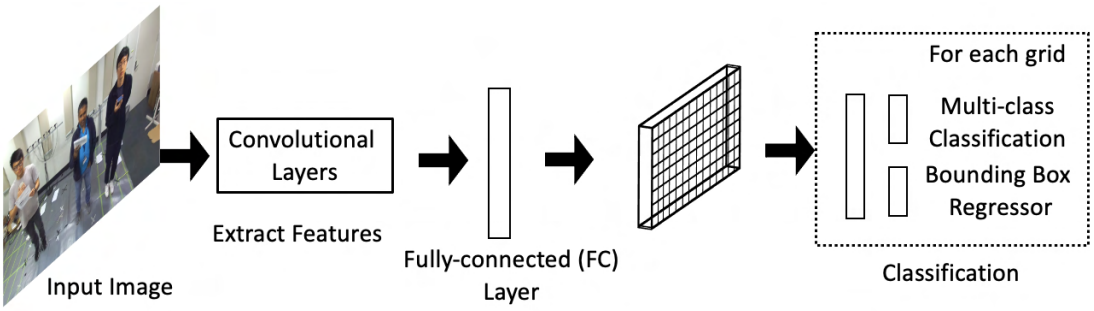


Figure 2.5: YOLO Pipeline

[166]. The prediction of each bounding box has the following parameters: the center positions, width, height, and the confidence. Each grid cell also predicts several class probabilities for object classification [166]. YOLO is based on GoogLeNet [154] that has a network composed of 24 convolutional layers and two fully connected layers [166]. The network has been trained on the ImageNet 1000-class competition dataset [167] at half the input image resolution [166].

The first version of YOLO limits the bounding box prediction such that each grid cell can only predict two boxes with only a single class. This spatial constraint limits the detection of nearby objects and causes high missed detections on objects that are small and appear in groups [168]. To overcome these issues, the inventors of YOLO made modifications to the CNN, the prediction of bounding boxes, and training techniques. These changes led to the release of the second and third versions of YOLO, named YOLOv2 [168] and YOLOv3 [169], respectively.

In YOLOv2, the network is replaced with DarkNet19, which has 19 convolutional layers, six max pooling layers and without the fully connected layers [168]. Compared to YOLO, DarkNet19 has been shown to be less complex but outperforms YOLO in terms of accuracy and speed [168]. In YOLOv3, DarkNet19 is substituted by a more powerful DarkNet53, which has 53 convolutional layers [169]. As per reported in [169], DarkNet53 achieves the highest measured floating point operations per second, which indicates that the network is more efficient to evaluate and hence faster.

One of the salient features of YOLOv2 and YOLOv3 is the use of anchor boxes (inspired by Faster RCNN [160]) to predict the extent of the bounding box as offsets from the centroid instead of the actual coordinates of bounding boxes [168]. These anchor boxes are object candidates of various aspect ratios and sizes. The selection of these boxes is done via k-means clustering to get good priors for the training model [168]. Moreover, YOLOv2 and YOLOv3 both adopt a multi-scale training scheme, where instead of fixing the size of the input image, the neural network randomly chooses a new image size from multiples of 32 (i.e., $\{320, 352, ..., 608\}$; the smallest image size being 320x320 and the largest being 608x608) [168]. This scheme enables the network to predict accurately across various input dimensions.

2.2.2 Occlusion-Handing Detectors

The aforementioned monocular (single-view) object detectors are reliable and robust against object rotation and scale changes, object localization accuracy, and detection speed, but they remain limited in handling dense and occluded objects. Although deep layers of CNNs have rich semantics, they are not effective in detecting dense objects [14]. One way of solving this is to design loss functions by considering the attraction of objects and the repulsion of other surrounding objects [170]. Previous works have used the ensemble of part detectors [171, 172] and attention mechanism [173] to improve occluded pedestrian/people detection. In practice, a more effective way of handling object occlusion is the use of multiple cameras [174].

In [175], a multi-camera pedestrian detector is proposed. The method is based on a multi-view Bayesian network that models the occlusion relationship, ground locations and the geometrical constraints across all camera views. The work in [174] accounts for occlusion by utilizing information from multiple views jointly using the probabilistic of occupancy map (POM) technique. This method uses background subtraction to extract foreground (moving) objects from their background, then estimate the occupancy probabilities using mean-field inference. A variant of this method uses a modified optimization scheme to leverage time consistency [176], while further modifications and improvements have been made in [177, 178]. The main issue with background subtraction preprocessing is the cause of ambiguity when foreground segmented blobs are interconnected with the background. This limits the performance under crowded scenarios, in addition to erroneously segmenting objects in the scene that are not of interest.

Recently, researchers have investigated the integration of deep CNN into a multi-camera object detection architecture. The work in [55] uses monocular and multi-view data to train multi-camera detectors (MCDs). An extension of this work that uses mean field variational inference and conditional random field (CRF) modeling has shown to achieve remarkable performance in a crowded scenario [56]. While deep MCD approaches have been shown to outperform monocular detectors in challenging scenarios, they require training on a large dataset, which is expensive due to a high-dimensional input space [58].

2.2.3 Visual Multi-Object Tracking

MOT is another computer vision task that aims to process and analyze stream of images (video) to associate a unique identity to each object of one or more categories (e.g., pedestrians, animals, vehicles) without any prior knowledge of the appearance and number of objects [6]. In the literature, MOT is fundamental to many computer vision applications such as behavior analysis [179], human-computer interaction [180], action recognition [181], pose estimation [182], visual surveillance [183], and virtual reality [184].

MOT algorithms have two types of processing modes: online tracking and offline tracking [7]. In online tracking [185–188], images are processed in a step-wise manner (i.e., only past observations available up to the current frame are utilized) and the trajectories of objects are produced on-the-fly for each frame. In offline tracking or batch processing [189–192], observations from all the frames are analyzed and processed jointly to estimate the object trajectories. In principle, batch algorithms typically performs better than online algorithms because all the data is used for estimation [193–197]. However, online algorithms that are designed to synchronously process data up to the current time and output the results are more suited for time-critical applications [198–202].

Most existing MOT works can be classified into detection-based tracking (DBT) and detection-free tracking (DFT) [7]. The DBT approach, which is known as the tracking-by-detection paradigm, relies on object detectors applied to each image frame to obtain object detections that are unlabeled (observation or measurements), then (online or offline) tracking is conducted to join detections together to form consistent trajectories [199, 203, 204]. The DFT approach, which is also known as track-before-detect (TBD), processes the entire image sequentially without requiring any form of pre-trained object detectors [185, 205–207]. However, DFT is often more computationally expensive and does not perform as well as DBT in challenging scenarios [198]. Due to the rapid evolution of object detectors, many DBT approaches for MOT have been developed.

Detection-Based MOT

The goal of detection-based MOT is to determine the trajectories of a time-varying and unknown number of objects from a collection of detections (or measurements) [208–210]. The tasks of a MOT system are state estimation, track management, and occlusion handling. Track management pertains to the identification, termination, and initiation of individual object trajectories, while state estimation determines the state vectors of the trajectories [7]. These MOT tasks are hindered by identity switching, track fragmentation, and track loss, which are typically caused by noise, false negatives (misdetection), and false positives (clutter) in the visual measurements. One of the factors contributing to misdetection and clutter is the occurrence of occlusion when objects are blocked visually from a camera.

Theoretical developments for occlusion handling in MOT are relatively scarce [175]. This is due to the complex relationship between objects and computational tractability since all possible sets of object partitions have to be considered [198]. To date, few heuristic techniques have been proposed to handle occlusion in a single-camera/view setting. One of them is the "part-to-hold" strategy, which exploits the assumption that when occlusion occurs, some part of the object remains visible [147, 185, 192]. This strategy divides a bounding box into several parts, and when the tracker detects an occlusion, the visible parts of the object are utilized for estimation. Another occlu-

sion handling strategy is based on remembering the object states before occlusion and buffering the observations during the occlusion to recover object states after the occlusion [211–213]. Further methods have resolved occlusions by exploiting *a priori* information about the objects [193, 194, 201, 202, 214].

Multi-Camera MOT Approaches

The use of multiple cameras in MOT is an effective way of resolving occlusions since an occluded object may not be may not be occluded in another view [54]. Further, the combined information from multiple cameras has the potential to reduce uncertainty in the object states, which improves tracking performance.

In [197], a hierarchical composition approach is proposed to construct estimates in the ground plane using monocular information from multiple views. Another multi-view approach preprocesses image data from multiple camera views via background subtraction to estimate the occupancies on the ground plane [174]. In [215], an occlusion model is formulated by utilizing 2D visual angles from multiple views. A more sophisticated approach in [175] proposes a multi-view Bayesian network that models the homography correspondence and occlusion relationship between multiple views to obtain detections on the ground plane for tracking. Further, 3D object estimation and tracking using stereo cameras have been demonstrated in [216–218].

Data-centric approaches have also been applied to multi-camera MOT. One example is the use of MCD [55] combined with batch processing to estimate the object trajectories [57]. In [56], the authors have demonstrated remarkable MOT performance in a high (people) density scenario using mean field variational inference and CRF modeling [219]. Subsequent deep learning based multi-view MOT approaches have been proposed in [220, 221]. Model-centric approaches that exploit the characteristics and geometry of the cameras and the physical models of object dynamics have been applied to 3D online MOT with monocular data, using 3D point cloud techniques [5], 3D object proposals [216], and 2D object detections [222]. Further, RFS-based filtering solutions have been applied to visual MOT problems [223–227]. In-depth expositions on model-centric MOT algorithms are provided in Section 2.4 of this chapter.

2.3 Audio-Visual Source Separation

Conventional source separation approaches are based on processing audio data only [25–27, 80]. More recent (audio-only) data-centric source separation approaches have been proposed in [28, 29, 39, 40]. These data-centric approaches require the number of speakers to be known and static during training and testing [31]. The main challenge of these approaches is the permutation ambiguity in distinguishing between different speech signals when the vocal characteristics of the speakers are similar, or when any

underlying assumption of signal independence is violated [30].

Instead of using audio data only to achieve source separation, the synergistic use of both audio and visual data has shown to improve separation performance over audio-only methods [30]. The idea of the approach is to train a neural network that leverages the audio-visual correspondence between human lip movements and speech utterances to address the permutation problem [30]. A different approach in [228] has achieved localization and separation by exploiting the low-rank and sparse representations of audio and visual data to capture the associations between the two modalities. Further, a time domain-based audio-visual separation deep network that captures lip embeddings and phonetic information has been proposed to improve speech separation performance [31]. The training process of the aforementioned data-centric approaches can be computationally intensive and restrictive in cases where training data are unavailable. Furthermore, these approaches have not been demonstrated with moving sources whereby the number of sources is time-varying and unknown.

Alternatively, audio-visual source separation is achieved using the three-step process of detection, tracking, and filtering (DTF). This approach can be designed to operate online without pre-training and to accommodate multiple moving sources. The DTF approach uses the characteristics of the audio and visual measurements acquired from standard detection algorithms and physical models of the source dynamics for the estimation (or tracking) of the source spatial positions and identities (trajectories) over time. The trajectory estimates gathered by such tracking system can then be leveraged for source separation with a model-based separation filter.

The difficulty in tracking multiple sources using both audio and visual data is that measurements from both modalities are unidentified (unlabeled) and do not fall in the same observation space. Moreover, these measurements are subject to missing measurements, false measurements, and noise. These issues give rise to the multi-modal space-time permutation problem: in space, it is unknown how the measurements of different modalities are associated across domains with respect to the sources; and in time, it is unknown how the measurements are associated to which sources, if any at all. Further, the solution must accommodate the unknown disappearance and appearance of moving sources.

So far, a number of audio-visual tracking algorithms have been proposed in the literature [48, 49, 229, 230]. In [48, 49, 229], a Bayesian particle filtering framework that builds on audio-visual likelihood models has been proposed for audio-visual multi-speaker tracking. However, a limitation is that the algorithm is unable to associate the multi-modal measurements to sources. Hence, it invokes an external post-processing mechanism that provides the associations. In [230], the authors use a variational Bayesian inference and an expectation maximization solver to propagate the approximate filtering distribution over time. The method accounts for the observation-to-person association problem without post-processing, but the formulation is based on

functional approximations and the resulting algorithm is not scalable for an extended number of sensors and measurements.

2.4 Advances in Model-Centric Object Tracking

The common theme of this dissertation is the estimation (or tracking) of source positions and labels, which, when combined across space and time, form the source trajectories. This information is central to the development any separation filter for multiple moving sources whereby the number of sources is time-varying and unknown. Whether using audio, visual or audio-visual data, a model-centric object tracking solution is grounded on dynamic state-space estimation. Standard object tracking models are used to capture the evolution of an object state over time and to update the object state using the measurement. The Bayesian paradigm offers a framework for dynamic state-space estimation. Conventional Bayes estimation methods are designed for single-object tracking. This subsection further discusses the extensions of single-object tracking algorithms to MOT. This includes the RFS and the labeled RFS formalisms for Bayesian state estimation, which are the foundations of the proposed tracking algorithms in this dissertation.

2.4.1 Bayesian Estimation

In Bayesian state estimation, a state vector is usually the kinematic characteristics of the object. Let the state of an object be represented by a vector x_k that resides in a state space $\mathbb{X}$, where k denotes the time step. At time k, the object state x_k generates a measurement z_k that resides in an observation space $\mathbb{Z}$. The measurement z_k and object state x_k are treated as realizations of random (vector) variables in their respective vector spaces and that their uncertainties are represented by probability densities [231]. The objective is to compute a probability density $\pi_{0:k}$ of the object states $x_{0:k} \equiv (x_0, ..., x_k)$ (where x_0 is the initial state), given the measurements $z_{1:k} \equiv (z_1, ..., z_k)$ at the start of time to the current time. All available information for the object states is obtainable from the probability density $\pi_{0:k}$.

In the Bayesian paradigm, the density $\pi_{0:k}$ is called the posterior probability density and it is computed recursively over time. The posterior probability density $\pi_{0:k}$ is given by the Bayes rule [232]:

$$\pi_{0:k}(x_{0:k}|z_{1:k}) = \frac{g_k(x_k|z_k) f_{k|k-1}(x_k|x_{k-1}) \pi_{0:k-1}(x_{0:k-1}|z_{1:k-1})}{\int g_k(x_k|z_k) f_{k|k-1}(x_k|x_{k-1}) \pi_{0:k-1}(x_{0:k-1}|z_{1:k-1}) dx_{0:k}}, \tag{2.16}$$

where $\pi_{0:k-1}(\cdot|\cdot)$ is the previous posterior probability density, $f_{k|k-1}(\cdot|\cdot)$ is the Markov transition density, and $g_k(\cdot|\cdot)$ is the likelihood function. The recursive computation of the posterior probability density considers the entire history of object states up to the

current time given the history of measurements. Since the posterior density embodies all statistical information about the object states, it is a complete solution to the estimation problem [231]. However, the computation of the posterior density can become computationally demanding as the dimension begins to grow over several recursions.

A cheaper alternative is to consider the filtering density $\pi_k(x_k|z_{1:k}) \triangleq \int \pi_{0:k}(x_{0:k}|z_{1:k})dx_{0:k-1}$, which is the marginal of the posterior density. The filtering density can be propagated via the Bayes filter [231, 232]:

$$\pi_k(x_k|z_{1:k}) = \frac{g_k(z_k|x_k)\pi_{k|k-1}(x_k|z_{1:k-1})}{\int g_k(z_k|x_k)\pi_{k|k-1}(x_k|z_{1:k-1})dx}, \tag{2.17}$$

where

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \int f_{k|k-1}(x_k|z_{k-1})\pi_{k-1}(x_{k-1}|z_{1:k-1})dx_{k-1}, \tag{2.18}$$

is the prior density or the so-called predicted density that is computed via the *Chapman-Kolmogorov* equation [232].

The implementation of Bayes filter is recursive and can be broken down into two stages per recursion: the prediction and the update. The prediction stage is governed by the *Chapman-Kolmogorov* equation, which is responsible for computing the predicted filtering density $\pi_{k|k-1}$ (the prior (2.18)) using the state transition density $f_{k|k-1}$, where π_{k-1} denotes the filtering density computed at the previous time $k-1$. The update stage is based on the Bayes rule to compute the filtering probability density π_k (i.e., (2.17)) at the current time k using the likelihood g_k. Consequently, at time k, all information about object state x_k given all past observations $z_{1:k}$ is captured in the filtering probability density π_k.

Dynamic Model

The dynamic transition of an object state $x \in \mathbb{R}^{n_x}$ of n_x dimension is characterized by a stochastic model:

$$x_k = a_k(x_{k-1}, \upsilon_{k-1}), \tag{2.19}$$

where υ_{k-1} is the process noise. The nonlinear function a_k describes the mapping of υ_{k-1} and x_{k-1} to x_k. This state dynamic model can also be described by a Markov transition density $f_{k|k-1}(\cdot|\cdot)$, where the probability density that a state x_{k-1} transitions to x_k is

$$f_{k|k-1}(x_k|x_{k-1}).$$

The state of an object moving in the physical world is typically specified by the 3D position and velocity vectors. Object motions are generally classed into two categories: non-maneuver and maneuver [233]. A non-maneuvering motion is described as a constant-velocity level and straight motion. The nearly-constant-velocity model (NCV) is an example of such non-maneuvering motion [233]. Accelerations in respec-

tive axes or Cartesian coordinates are typically modeled as Gaussian noise to account for undesirable modeling errors. It is called "nearly-constant-velocity" because the perturbation in the accelerations is small [233]. NCV dynamic model is suitable for tracking instances where objects follow a straight-line path with minimal sharp turns.

On the other hand, an example of a maneuvering motion model is the nearly-constant-turn (NCT) model [233]. In this model, a turn rate parameter is added for capturing the maneuvering of an object. Constant turn rate means that the object motion follows an approximately constant angular velocity with perturbations captured by the process noise. For most practical applications, the turn rate parameter is included in the object state in order to be estimated based on the latest velocity estimates. The NCT model is commonly used to model the motions of vehicular objects such as aircrafts, cars and trucks. Another maneuvering motion model is the Langevin model which is used to characterize various types of stochastic motions [234]. Motions in each Cartesian coordinate is assumed to be an independent first-order process. This model is commonly used in acoustic source tracking [71, 234].

Measurement Model

The generation of a measurement $z \in \mathbb{R}^{n_z}$ of n_z dimension by an object state x_k is given by:

$$z_k = h_k(x_k, w_k), \tag{2.20}$$

where w_k denotes the noise in the measurement. The nonlinear function h_k describes the mapping of the state vector x_k and measurement noise w_k to the measurement vector z_k. This model can equivalently be represented by the likelihood function denoted by $g_k(\cdot|\cdot)$, where

$$g_k(z_k|x_k)$$

characterizes the probability density that a state x_k generates the measurement vector z_k.

Measurements obtained from a particular live sensor in a sensor coordinate system usually differ from the coordinate system of the object states. For example, object motion is best described in a Cartesian coordinate system, whereas visual detections from cameras fall in the pixel coordinate frame. Audio measurements such as the TDOAs are in the time domain, which has a lower dimension compared to the Cartesian coordinate system. When there are multiple heterogeneous sensors, the obtained measurements are generated in multiple different sensor coordinate systems. The measurement model is a crucial aspect of object motion tracking as it constitutes the relationship between measurements in the sensor coordinates to the states in the Cartesian coordinates. The measurement models used in this dissertation are described in the subsequent chapters. For detailed considerations of the coordinates systems and the respective transformations, the reader is referred to [235].

Bayes Filter State Estimators

The estimate of an object state can be computed optimally from a given filtering density $\pi_k(\cdot|\cdot)$ with respect to a specific criterion. The most common Bayes estimators that have been shown to minimize the Bayes risks [236, 237] are the *maximum a posteriori* (MAP) and *expected a posteriori* (EAP) estimators:

$$\hat{x}_k^{(\mathrm{MAP})} = \underset{x_k}{\operatorname{argsup}} \quad \pi_k(x_k|z_{1:k}), \tag{2.21}$$

$$\hat{x}_k^{(\mathrm{EAP})} = \int x_k \pi_k(x_k|z_{1:k}) dx_k. \tag{2.22}$$

2.4.2 Solutions for the Single-Object Bayes Filter

The implementation of the Bayes recursion is generally intractable in practice due to the normalizing term at each iteration. Traditional grid-based filters have exploited state-space discretization to overcome complexity [232]. However, since the complexity grows exponentially with the state-space dimension, grid-based filters are only feasible in low-dimensional problems [232]. The following subsections provide an outline of notable closed-form solutions and tractable approximations to the Bayes filter.

Kalman Filter

The Kalman filter is a closed-form solution to the Bayes filter given that the measurement and dynamic models, expressed in (2.20) and (2.19), respectively, are linear transformations with additive Gaussian noise [238]:

$$z_k = \mathrm{H}_k x_k + w_k, \tag{2.23}$$

$$x_k = \mathrm{F}_{k-1} x_{k-1} + \upsilon_{k-1}, \tag{2.24}$$

where w_k is an independent zero-mean Gaussian noise with covariance matrix R_k, and H_k is the measurement matrix, υ_{k-1} is an independent zero-mean Gaussian noise with covariance matrix Q_{k-1}, and F_{k-1} is a transition matrix, .

In the following, the notation $\mathcal{N}(\cdot; m, \mathrm{P})$ is used to denote a Gaussian density function with mean m and covariance matrix P. Under the linear Gaussian assumption above, the measurement likelihood and transition density are expressed as:

$$g_k(z_k|x_k) = \mathcal{N}(z_k; \mathrm{H}_k x_k, \mathrm{R}_k), \tag{2.25}$$

$$f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x_k; \mathrm{F}_{k-1} x_{k-1}, \mathrm{Q}_{k-1}). \tag{2.26}$$

At time $k-1$, if the filtering density is Gaussian:

$$\pi_{k-1}(x_{k-1}|z_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1}, \mathrm{P}_{k-1}), \tag{2.27}$$

then the predicted density is also a Gaussian [232]:

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, \mathrm{P}_{k|k-1}), \tag{2.28}$$

where

$$m_{k|k-1} = \mathrm{F}_{k-1} x_{k-1}, \tag{2.29}$$
$$\mathrm{P}_{k|k-1} = \mathrm{Q}_{k-1} + \mathrm{F}_{k-1} \mathrm{P}_{k-1} \mathrm{F}_{k-1}^T. \tag{2.30}$$

According to the Bayes rule, at time k, the updated filtering density is also a Gaussian [232]:

$$\pi_k(x_k|z_{1:k}) = \mathcal{N}(x_k; m_k, \mathrm{P}_k), \tag{2.31}$$

where

$$m_k = m_{k|k-1} + \mathrm{K}_k(z_k - \mathrm{H}_k x_{k|k-1}), \tag{2.32}$$
$$\mathrm{P}_k = \mathrm{P}_{k|k-1} - \mathrm{K}_k \mathrm{S}_k \mathrm{K}_k^T, \tag{2.33}$$
$$\mathrm{S}_k = \mathrm{R}_k + \mathrm{H}_k \mathrm{P}_{k|k-1} \mathrm{H}_k^T, \tag{2.34}$$
$$\mathrm{K}_k = \mathrm{P}_{k|k-1} \mathrm{H}_k^T \mathrm{S}_k^{-1}. \tag{2.35}$$

In summary, if both dynamic and measurement models assume a linear Gaussian form, then the filtering density can be propagated via the Kalman recursion, which is a tractable solution to the single-object Bayesian filter.

Extended Kalman Filter

The extended Kalman filter (EKF) is a technique that accommodates for mildly nonlinear dynamic and measurement models. Consider the nonlinear dynamic function (2.19), where the process noise υ_{k-1} is modeled by a Gaussian with covariance matrix Q_{k-1}, and the measurement function (2.20), where the measurement noise w_k is modeled by a Gaussian with covariance matrix R_k. Note that the Gaussian vectors of the process and measurement noises are zero-mean and independent. The EKF assumes a linear approximation to the Kalman filter via localized approximations [232]. Considering the leading terms of the Taylor series expansion on $a_{k-1}(\cdot,\cdot)$ and $h_k(\cdot,\cdot)$ results in the following local linearizations:

$$\hat{\mathrm{F}}_{k-1} = \frac{\partial a_{k-1}(x,0)}{\partial(x)} \Big|_{x=m_{k-1}}, \qquad \hat{\mathrm{Q}}_{k-1} = \frac{\partial a_{k-1}(m_{k-1},\upsilon)}{\partial(\upsilon)} \Big|_{\upsilon=0}, \tag{2.36}$$
$$\hat{\mathrm{H}}_k = \frac{\partial h_k(x,0)}{\partial(x)} \Big|_{x=m_{k|k-1}}, \qquad \hat{\mathrm{R}}_k = \frac{\partial h_k(m_{k|k-1},w)}{\partial(w)} \Big|_{w=0}. \tag{2.37}$$

At time $k-1$, let the filtering density be a Gaussian:

$$\pi_{k-1}(x_{k-1}|z_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1}, \mathrm{P}_{k-1}). \tag{2.38}$$

At time k, the predicted and filtering densities are, respectively, as follows [232]:

$$\pi_{k|k-1}(x_k|z_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, \mathrm{P}_{k|k-1}), \tag{2.39}$$
$$\pi_k(x_k|z_{1:k}) = \mathcal{N}(x_k; m_k, \mathrm{P}_k), \tag{2.40}$$

where

$$m_{k|k-1} = a_{k-1}(m_{k-1}, 0), \tag{2.41}$$
$$\mathrm{P}_{k|k-1} = \hat{\mathrm{Q}}_{k-1}\mathrm{Q}_{k-1}\hat{\mathrm{Q}}_{k-1}^T + \hat{\mathrm{F}}_{k-1}\mathrm{P}_{k-1}\hat{\mathrm{F}}_{k-1}^T, \tag{2.42}$$
$$m_k = m_{k|k-1} + \mathrm{K}_k(z_k - h_k(m_{k|k-1}, 0)), \tag{2.43}$$
$$\mathrm{P}_k = \mathrm{P}_{k|k-1} - \mathrm{K}_k\mathrm{S}_k\mathrm{K}_k^T \tag{2.44}$$
$$\mathrm{S}_k = \hat{\mathrm{R}}_k\mathrm{R}_k\hat{\mathrm{R}}_k^T + \hat{\mathrm{H}}_k\mathrm{P}_{k|k-1}\hat{\mathrm{H}}_k^T, \tag{2.45}$$
$$\mathrm{K}_k = \mathrm{P}_{k|k-1}\hat{\mathrm{H}}_k^T\mathrm{S}_k^{-1}. \tag{2.46}$$

Since the EKF is based on first-order approximation, it will perform poorly under severe nonlinearities in the dynamic and measurement models.

Unscented Kalman Filter

The unscented Kalman filter (UKF) approximates the Kalman filter using the principles of sampling according to the unscented transform (UT) [239, 240]. The UT seeks to generate a set of weighted sample points such that they encompass the mean and the covariance of the density from which they are sampled [239, 240]. For nonlinear transition and measurement functions (i.e., (2.19) and (2.20), respectively), the UKF propagates the first and second moments of the filtering density using the UT.

The process noise of the dynamic model υ_{k-1} is a zero-mean independent Gaussian vector of n_υ dimension with covariance matrix Q_{k-1}, and the noise of the measurement model w_k is a zero-mean independent Gaussian vector of n_w dimension with covariance matrix R_k. At time $k-1$, let the filtering density be a Gaussian:

$$\pi_{k-1}(x_{k-1}|z_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1}, \mathrm{P}_{k-1}). \tag{2.47}$$

Let C_k and μ_k and be the augmented covariance and mean, respectively defined as:

$$\mathrm{C}_k = \mathrm{diag}(\mathrm{P}_{k-1}, \mathrm{Q}_{k-1}, \mathrm{R}_k), \tag{2.48}$$
$$\mu_k = [\ m_{k-1}^T \quad 0^T \quad 0^T\ ]^T, \tag{2.49}$$

the set of weighted sample points $\{(\bar{x}_k^{(i)}, \omega^{(i)})\}_{i=1}^{E}$ are generated via the following criteria [232]:

$$\bar{x}_k^{(i)} = \mu_k \qquad \omega^{(i)} = \frac{\varsigma_U}{n_U + \varsigma_U} \qquad i = 0, \tag{2.50}$$

$$\bar{x}_k^{(i)} = \mu_k + \left[\sqrt{(n_U + \varsigma_U)\mathrm{C}_k}\right]_i \qquad \omega^{(i)} = \frac{1}{2(n_U + \varsigma_U)} \qquad i = 1, ..., n_U, \tag{2.51}$$

$$\bar{x}_k^{(i)} = \mu_k - \left[\sqrt{(n_U + \varsigma_U)\mathrm{C}_k}\right]_i \qquad \omega^{(i)} = \frac{1}{2(n_U + \varsigma_U)} \qquad i = n_U + 1, ..., E, \tag{2.52}$$

where $E = 2n_U + 1$, $n_U = n_x + n_\upsilon + n_w$, ς_U denotes a scaling parameter on the condition that $n_U + \varsigma_U \neq 0$, and $[\cdot]_i$ denotes the ith row of the matrix. For $i = 0, ..., E$, each sample point can thus be partitioned into:

$$\bar{x}_k^{(i)} = [(x_{k-1}^{(i)})^T, (\upsilon_{k-1}^{(i)})^T, (w_k^{(i)})^T]^T. \tag{2.53}$$

In the prediction step, all sample points $x_{k-1}^{(i)}$ and $\upsilon_{k-1}^{(i)}$ are inserted into the nonlinear function to obtain $x_{k|k-1}^{(i)} = a_k(x_{k-1}^{(i)}, \upsilon_{k-1}^{(i)})$. These individual points are then used to compute the predicted Gaussian density with mean $m_{k|k-1}$ and covariance $\mathrm{P}_{k|k-1}$:

$$m_{k|k-1} = \sum_{i=0}^{E} \omega^{(i)} x_{k|k-1}^{(i)}, \tag{2.54}$$

$$\mathrm{P}_{k|k-1} = \sum_{i=0}^{E} \omega^{(i)} [x_{k|k-1}^{(i)} - m_{k|k-1}][x_{k|k-1}^{(i)} - m_{k|k-1}]^T. \tag{2.55}$$

In the update step, all sample points $x_{k|k-1}^{(i)}$ and $w_k^{(i)}$ are inserted into the nonlinear function to obtain $z_{k|k-1}^{(i)} = h_k(x_{k|k-1}^{(i)}, w_k^{(i)})$. These individual points are then used to compute the updated filtering density at time k with mean m_k and covariance $\mathrm{P_k}$:

$$m_k = m_{k|k-1} + \mathrm{K}_k(z_k - z_{k|k-1}), \tag{2.56}$$

$$\mathrm{P}_k = \mathrm{P}_{k|k-1} - \mathrm{P}_{k,xz}\mathrm{P}_{k,zz}\mathrm{P}_{k,xz}^T, \tag{2.57}$$

where

$$z_{k|k-1} = \sum_{i=0}^{E} \omega^{(i)} z_{k|k-1}^{(i)}, \tag{2.58}$$

$$\mathrm{K}_k = \mathrm{P}_{k,xz}\mathrm{P}_{k,zz}^{-1}, \tag{2.59}$$

$$\mathrm{P}_{k,zz} = \sum_{i=0}^{E} \omega^{(i)} [z_{k|k-1}^{(i)} - z_{k|k-1}][z_{k|k-1}^{(i)} - z_{k|k-1}]^T, \tag{2.60}$$

$$\mathrm{P}_{k,xz} = \sum_{i=0}^{E} \omega^{(i)} [x_{k|k-1}^{(i)} - m_{k|k-1}][z_{k|k-1}^{(i)} - z_{k|k-1}]^T. \tag{2.61}$$

Particle Filter

The particle filter is a random sampling-based approximation to the densities in Bayes filter [232]. Based on the concept of SMC, the particle filter is applicable to more generic probability densities that are non-Gaussian to which variations of the Kalman filter cannot be applied. Readers can refer to detailed tutorial articles for comprehensive treatments in [231, 241, 242]. A basic description of the particle filter framework is given below.

Given an arbitrary probability density $\pi(\cdot)$, if N independent samples denoted by $\{x^{(i)}\}_{i=1}^{N}$, can be drawn from the density, then the density $\pi(\cdot)$ can be approximated by [232]:

$$\pi(x) \approx \frac{1}{N}\sum_{i=1}^{N}\delta_{x^{(i)}}(x), \tag{2.62}$$

where $\delta.(\cdot)$ is the Dirac delta function. Convergence analysis of the above approximation can be found in [243, 244].

In the case that a density $\pi(\cdot)$ is only known up to a normalizing constant, importance sampling can be used to propagate the filtering density. The basic idea of importance sampling is to draw samples from a proposal density $q(\cdot)$, which is selected to be close to the actual density $\pi(\cdot)$. These samples are then weighted accordingly to obtain a Monte Carlo approximation to $\pi(\cdot)$ [232]. The proposal density $q(\cdot)$ has to be chosen such that its support must contain the support of the actual density $\pi(\cdot)$ [232]. The density $\pi(\cdot)$ can therefore be approximated by a weighted point mass representation given by [232]:

$$\pi(x) \approx \sum_{i=1}^{N}\tilde{\omega}^{(i)}\delta_{x^{(i)}}(x), \tag{2.63}$$

where

$$\tilde{\omega}^{(i)} = \frac{\omega(x^{(i)})}{\sum_{j=1}^{N}\omega(x^{(j)})}, \tag{2.64}$$

$$\omega^{(i)} = \frac{p(x^{(i)})}{q(x^{(i)})}. \tag{2.65}$$

In the Bayesian paradigm, sequential importance sampling (SIS) is used to compute a point mass approximation of the filtering density. At time $k-1$, let the filtering density $\pi_{k-1}(\cdot)$ be represented by a large set of weighted samples denoted by $\left\{(x_{k-1}^{(i)}, \omega_{k-1}^{(i)})\right\}_{i=1}^{N}$, i.e.,

$$\pi_{k-1}(x_{k-1}|z_{1:k-1}) \approx \sum_{i=1}^{N}\omega_{k-1}^{(i)}\delta_{x_{k-1}^{(i)}}(x_{k-1}) \qquad \text{where} \sum_{i=1}^{N}\omega_{k-1}^{(i)} = 1. \tag{2.66}$$

Based on the proposal density $q_k(\cdot|x_{k-1}^{(i)}, z_k)$, a new set of weighted particles $\left\{(x_k^{(i)}, \omega_k^{(i)})\right\}_{i=1}^{N}$

that approximates the filtering density $\pi_k(\cdot)$ at time k is computed [232]:

$$\pi_k(x_k|z_{1:k}) \approx \sum_{i=1}^{N} \tilde{\omega}_k^{(i)} \delta_{x_k^{(i)}}(x_k), \tag{2.67}$$

and

$$\tilde{\omega}_k^{(i)} = \omega_k^{(i)} / \textstyle\sum_{i=1}^{N} \omega_k^{(i)}, \tag{2.68}$$

$$x_k^{(i)} = q_k(\cdot|x_{k-1}^{(i)}, z_k), \tag{2.69}$$

$$\omega_k^{(i)} = \omega_{k-1}^{(i)} \frac{g_k(z_k|x_k^{(i)}) f_{k|k-1}(x_k^{(i)}|x_{k-1}^{(i)})}{q_k(x_k^{(i)}|x_{k-1}^{(i)}, z_k)}. \tag{2.70}$$

The SIS algorithm is subject to a problem known as particle depletion, where after several recursions, nearly all of the particles have negligible weights. This problem is mitigated with a resampling process, which is based on the premise that particles of negligible weights are replaced by replication of particles with higher weights [232]. As resampling is a non-parallelizable process, it is thus the main computational bottleneck in the particle filter. Recent works have investigated the use of graphic processing unit to improve execution time [245, 246]. Further development of the particle filter has been shown to improve tracking performance and computational tractability [232, 247–249].

Extension to Multiple Measurements

The aforementioned solutions to the Bayes filter are only for single object tracking with a single measurement. In actual tracking scenarios, the sensor may potentially pick up false measurements and be subjected to missed detections. In these cases, the aforementioned filters are not directly applicable. One simple way of tackling this issue is to use the nearest neighbor (NN) filter [250–253]. Both the NN filter and Kalman filter shares the same prediction step. In the update step, the closest measurement to the predicted state is used to compute the Kalman update. If no near measurements are available, no update is performed. While the NN filter is easy to implement, it performs poorly in dense clutter or/and low detection profiles due to the inability to pick up the true measurement.

The probabilistic data association (PDA) [250–253] filter is designed to be more robust than the NN filter. It shares a similar prediction step to the NN filter. In the update step, gating is used to select candidate measurements, which are used to compute the likelihoods given the predicted object state (the association probabilities). The Kalman update is subsequently conducted using an average of the weighted measurements. A variation is made to the PDA filter in [254], which propagates a Gaussian mixture filtering density compared to a single Gaussian in the standard PDA.

The multiple hypothesis tracking (MHT) filter [250, 253, 255] is a data-association-oriented method for MOT. The MHT filter seeks to search for all measurement-to-object associations across previous time steps that are likely to make up the object trajectories, in order to reduce the association uncertainty [255]. Since single-object tracking is a special case for multi-object tracking, the MHT filter can also be applied to single-object tracking.

2.4.3 Classical Approaches to Multi-Object Tracking

Standard Bayesian estimation techniques for single-object tracking are not directly applicable to MOT for the following reasons: the number of objects is unknown and subject to change over time; the measurements collected from various types of sensor are unlabeled and unordered, giving rise to the inherent data association (or the space-time permutation) problem, i.e., mappings of the received measurements to multiple objects are unknown and combinatorial; and the received measurements are subject to noise, missing measurements, and false measurements. Classical approaches to MOT have been constructed based on adaptations of the aforementioned single-object tracking solutions.

Global Nearest Neighbor filter

The global nearest neighbor (GNN) is designed for fixed and known number of multiple objects [250–253]. The object-to-measurement mappings of the GNN filter are based on the condition that one measurement can only be mapped onto one object at most. A mapping is obtained by minimizing a particular cost function so that the standard Kalman prediction and update can be performed individually on each object. The cost function can be the total summed distance or the association probability. Although easy to implement, the GNN filter performs poorly under dense clutter and low detection profiles.

Joint Probabilistic Data Association Filter

The joint probabilistic data association (JPDA) filter [251] is designed to handle a constant and known number of multiple objects and multiple unordered measurements. Similar to the GNN filter, the JPDA filter performs Kalman prediction for each object individually. The difference is in the update step where Kalman update is performed on a weighted average measurement, which is computed using neighboring measurements.

The drawback of the JPDA filter is that the complexity of the data association component increases with the number of measurements and objects exponentially. Suboptimal strategies have been proposed to reduce the complexity [256–258]. Subsequently, the author in [259] has proposed a variation of the JPDA called the joint integrated prob-

abilistic data association (JIPDA) filter to accommodate for multiple objects whereby the number of objects is time-varying and unknown.

Multiple Hypothesis Tracking

The multiple hypothesis tracking (MHT) filter [250, 253, 255] works by propagating multiple measurement-to-object associations from past to present, known as hypotheses. This formulation enables the MHT filter to defer difficult data association decisions, which usually happen when objects are close together or/and under a low detection profile. As it is infeasible to propagate all possible hypotheses, the MHT filter only selects those hypotheses with high weights calculated via the Kalman filer. Since a new set of hypotheses is generated with new measurements received, which could be assigned to an existing track, new track or clutter, the MHT is capable of handling multiple objects whereby the number of objects is time-varying and unknown.

The main limitation of the MHT filter is that the number of hypotheses grows exponentially after several recursions, rendering the algorithm intractable. To counter this issue, MHT uses the gating of measurements along with pruning/merging of hypotheses [250, 253, 255]. Improved deterministic techniques for selecting and propagating the best hypotheses have been proposed in [253, 260, 261]. A modified version of the MHT filter called the probabilistic multiple hypothesis tracking (PMHT) filter [262] is constructed based on the assumption that the data associations are independent over the objects tracks. By this assumption, the PMHT filter achieves a computational complexity lower than that of the MHT filter.

Rao-Blackwellized Particle Filter

In its most general form, the Rao-Blackwellized particle filter (RBPF) is designed to use the optimal estimator for linear-Gaussian subspaces to track time-varying and unknown number of objects [263]. The unknown births and deaths of objects are encapsulated by a stochastic process model. As oppose to computing the filtering equations with pure Monte Carlo sampling, the Rao–Blackwell theorem used in the formulation of this filter suggests that some of the filtering equations can be computed analytically, while the others are computed using Monte Carlo sampling [263]. RBPF performs the estimation of the filtering density by first resolving the data associations. Based on the computed data associations, the filter then applies single-object tracking. By conditioning on the data associations, RBPF calculates the filtering equations in closed-form rather than using particle sampling for all steps, which in principle, leads to better tracking results [263].

2.4.4 Random Finite Set and Multi-Object Filtering

Algorithms like JPDA and MHT hypothesize the associations between objects and measurements over time to achieve tracking. The common characteristic of these association-based techniques is that a hypothesis is treated as a state variable. A critique has been made as to the observability, consistency and correctness of modeling a hypothesis as a state variable that is to be estimated [237]. Hence, it is unclear whether or not these methods are Bayes-optimal and consistent with the Bayesian paradigm. To reach a rigorous formulation of multi-object Bayesian estimation, a set of principled mathematical tools is needed to underpin the derivation of a principled multi-object Bayesian filter and its approximations.

The RFS approach provides a mathematically consistent and association-free multi-object Bayesian formulation. The RFS approach has provided a systematic and principled framework for the multi-object Bayesian paradigm via the Finite Set Statistics (FISST) [122, 237]. Due to its rigorous construction and underpinnings, the framework is functional to many unconnected sub-disciplines of data fusion [236, 264, 265], and it has the potential to develop useful and tractable multi-object Bayes filter [21] and extensions, such as multi-sensor multi-object filtering [22] and multi-scan (batch) multi-object filtering [266].

By definition, an RFS is a random variable whereby the number of elements in the set as well as each element of the set are random. Note that the elements of an RFS are unordered and distinct from one another. Therefore, RFSs are natural representations of the multi-object measurements and states which are described later in this subsection. These representations along with appropriate models of the dynamics and measurements, lead to solutions that naturally incorporate track initiation (birth), termination (death), missing measurements (false negatives), and false measurements (false positives or clutter).

Probability Density and Finite Set Statistics

An RFS X on $\mathbb{X}$ is defined as a measurable mapping from a sample space to the space of finite subsets of $\mathbb{X}$ (i.e., $\mathcal{F}(\mathbb{X})$) [122, 267]. Similar to that of a random vector, the probability density of an RFS is essential as it a useful descriptor in Bayesian filtering and estimation. Contrary to a random vector, the space $\mathcal{F}(\mathbb{X})$ of an RFS does not inherit the usual Euclidean notion of a density. One of the key concepts of defining the notion of a density is the reference measure [268]. It has been shown in point process theory that the dimensionless *unnormalized distribution of a Poisson point process* is a conventional choice of reference measure for a probability density on $\mathcal{F}(\mathbb{X})$ [268]. With respect to this reference measure, the probability density of an RFS can be defined via Radon-Nikodym's derivative, which is dimensionless [268].

In the single-object case, derivative and integral transforms are fundamental to

Bayesian inferencing. These operations are also important in the multi-object statistics. The Finite Set-Statistics (FISST) is the first rigorous and principled treatment of multi-object calculus using RFS [237]. In FISST, three fundamental statistical descriptors of an RFS are the belief mass function, FISST density and probability generating functional (p.g.fl). These descriptors facilitate the derivations of many RFS-based MOT solutions in the literature. For more details on the importance of these descriptors, the reader may refer to [237] pg. 357. The set derivative and set integral are the basic FISST operations that describe the relationships between the statistical descriptors of an RFS.

For any subset $\mathcal{T} \subseteq \mathcal{F}(\mathbb{X})$, the set derivative of a belief mass function $\beta : \mathcal{T} \to [0, \infty)$ at a point $x \in \mathbb{X}$ is a mapping $(d\beta)_x : \mathcal{T} \to [0, \infty)$ defined as [268]

$$(d\beta)_x(\mathcal{T}) \equiv \lim_{\lambda_{\mathcal{K}}(\Delta_x)\to 0} \frac{\beta(\mathcal{T} \cup \Delta_x) - \beta(\mathcal{T})}{\lambda_{\mathcal{K}}(\Delta_x)}, \tag{2.71}$$

where $\lambda_{\mathcal{K}}(\Delta_x)$ is the volume (Lebesgue measure) of a neighborhood Δ_x of x in units of $\mathcal{K}$. The set derivative is given by the recursion [268]

$$(d\beta)_{\{x_1,\ldots,x_n\}}(\mathcal{T}) \equiv (d\,(d\beta)_{\{x_1,\ldots,x_{n-1}\}})_{x_n}(\mathcal{T}), \tag{2.72}$$

where $(d\beta)_{\emptyset} \equiv \beta$ by convention. Let π be a FISST density defined by $\pi(X) = (d\beta)_X(\emptyset)$, the set integral of π over $\mathcal{T} \subseteq \mathcal{F}(\mathbb{X})$ is defined as [268]

$$\int \pi(X)\delta X = \sum_{n=0}^{\infty} \frac{1}{n!} \int_{\mathcal{T}^n} \pi(\{x_1, \ldots, x_n\}) dx_1, \ldots, dx_n. \tag{2.73}$$

Since a FISST density is based on the measure $\lambda_{\mathcal{K}}$ that has a unit, all FISST densities are unit-dependent. It has been shown in [268] that a (measure-theoretic) probability density of an RFS with respect to the *unnormalized distribution of a Poisson point process* is a FISST density without its unit. Therefore, in this dissertation, FISST density and the probability density of an RFS are used interchangeably for convenience. Some common (unlabeled) RFSs, the standard multi-object transition and likelihood models that constitute the multi-object Bayes filter, and its approximate solutions are outlined below. In terms of the notations, this dissertation follows the convention in RFS tracking literature, where lower case letters (e.g., x, z) are used to denote single element and upper case letters (e.g., X, Z) are used to denote sets.

Common Unlabeled RFSs

Poisson RFS

A Poisson RFS X is completely described by an intensity function $v(\cdot)$. The cardinality of a Poisson RFS is Poisson distributed (on $\{0\} \cup \mathbb{N}$) with mean $N = \int v(x)dx$. The

elements of X for any finite cardinality are distributed independently and identically according to probability density (or a spatial distribution) $v(\cdot)/N$. The probability density of a Poisson RFS $\pi(\cdot)$ is given by [122, 267]:

$$\pi(\{x_1, ..., x_n\}) = e^{-N} \prod_{i=1}^{n} v(x_i), \tag{2.74}$$

with $\prod_{i=1}^{0} v(x_i) = 1$ by convention. Note that the intensity function is also known as the probability hypothesis density, which is the first-order moment of an RFS.

Independent and Identically Distributed Cluster RFS

An i.i.d cluster RFS X is a generalization of the Poisson RFS. The cardinality of this RFS is encapsulated by an arbitrary cardinality distribution $\rho(\cdot)$ on the condition that the mean of the cardinality distribution is $N = \int v(x)dx$. The i.i.d cluster RFS's probability density is given by [122, 267]:

$$\pi(\{x_1, ..., \pi_n\}) = n!\rho(n) \prod_{i=1}^{n} \frac{v(x_i)}{N}. \tag{2.75}$$

Bernoulli RFS

A Bernoulli RFS X in space $\mathcal{F}(\mathbb{X})$ is either a singleton with probability r, whereby the element is characterized by a probability density $p(\cdot)$ defined on $\mathbb{X}$, or an empty set with probability $1-r$. The probability density of a Bernoulli RFS is given by [122, 267]:

$$\pi(X) = \begin{cases} rp(x) & X = \{x\} \\ 1-r & X = \emptyset \\ 0 & |X| > 1 \end{cases}. \tag{2.76}$$

Multi-Bernoulli RFS

A union of finite and constant number of independent Bernoulli RFSs is a multi-Bernoulli RFS. This RFS can be characterized by a parameter set $\{(r^{(i)}, p^{(i)}(\cdot))\}_{i=1}^{M}$, where M is the number of Bernoulli RFSs and the pair $(r^{(i)}, p^{(i)}(\cdot))$ denotes the existence probability and the spatial probability density of the ith Bernoulli RFS. The probability density of such a multi-Bernoulli RFS is expressed as [122, 267]:

$$\pi(\{x_1, ..., x_n\}) = \prod_{j=1}^{M}(1 - r^{(j)}) \sum_{1 \le i_1 \ne ... \ne i_n \le M} \prod_{j=1}^{n} \frac{r^{(i_j)} p^{(i_j)}(x_j)}{1 - r^{(j)}}. \tag{2.77}$$

Standard Multi-Object Dynamic Model

At time k, let the collection of states be formally represented as a (finite) set or a multi-object state:

$$X_k = \{x_{k,1}, ..., x_{k,N_k}\}, \tag{2.78}$$

where $N_k = |X_k|$ is the number of objects and $|\cdot|$ is the cardinality of the set. In a standard multi-object system, new objects may appear (or be born) in the state space, existing objects may survive and transition to the next step with a new state, and some objects may disappear (death) or cease to exist.

Given X_{k-1}, an object with state $x_{k-1} \in X_{k-1}$ either survives with probability $P_S(x_{k-1})$ and transitions according to the transition density $f_{k|k-1}(\cdot|x_{k-1})$ to a new state, or not with probability $1 - P_S(x_{k-1})$. The RFS for the surviving objects is therefore modeled by a multi-Bernoulli RFS:

$$S_k(X_{k-1}) = \bigcup_{x_{k-1} \in X_{k-1}} F_k(x_{k-1}), \tag{2.79}$$

where $F_k(x_{k-1})$ is a Bernoulli RFS that is parameterized by $\big(P_S(x_{k-1}), f_{k|k-1}(\cdot|x_{k-1})\big)$.

At time k, the appearance of newly born objects is characterized by the RFS B_k, which can be modeled as either a multi-Bernoulli RFS, an i.i.d cluster RFS, or a Poisson RFS. Based on the surviving objects $S_k(X_{k-1})$ and newly born objects B_k, the multi-object state X_k at time k is the superposition of both the surviving and newly born objects:

$$X_k = S_k(X_{k-1}) \bigcup B_k. \tag{2.80}$$

It is important that the surviving objects $S_k(X_{k-1})$ and birth objects B_k are independent of one another.

Based on FISST convolution formula, the multi-object transition density $f_{k|k-1}(X_k|X_{k-1})$ for the multi-object set X_k given the multi-object set X_{k-1} is expressed as [122, 237]:

$$f_{k|k-1}(X_k|X_{k-1}) = \sum_{U \in X_k} f_S(U|X_{k-1}) f_B(X_k - U), \tag{2.81}$$

where $f_S(\cdot|X_{k-1})$ is the transition density of the surviving object set $S_k(X_{k-1})$ and $f_B(\cdot)$ is the probability density of the newly born object B_k, and $X_k - U$ is the set difference between X_k and U.

Standard Multi-Object Measurement Model

The measurements obtained at time k from any form sensor processing algorithm that produces point detections are represented as a finite set:

$$Z_k = \{z_{k,1}, ..., z_{k,|Z_k|}\}, \tag{2.82}$$

where $|Z_k|$ denotes the number of measurements. In the standard multi-object measurement model, each measurement in the measurement set Z_k may be a spurious measurement or generated by a certain object in the multi-object state X_k (i.e., a detected measurement).

If an object $x_k \in X_k$ either generates a noisy measurement z_k with probability $P_D(x_k)$ and likelihood $g_k(z_k|x_k)$, or missed detected with probability $1 - P_D(x_k)$, the RFS of all measurements generated the multi-object state X_k is modeled by a multi-Bernoulli RFS:

$$D_k(X_k) = \bigcup_{x_k \in X_k} J_k(x_k), \tag{2.83}$$

where $J_k(x_k)$ is a Bernoulli RFS with parameters $(P_D(x_k), g_k(\cdot|x_k))$ for the detection of object x_k.

A sensor may also generate incorrect or clutter measurements. Let K_k denote the RFS for clutter measurements at time k, it is conventional in the RFS tracking literature to model K_k as a Poisson RFS with intensity denoted by $\kappa_k(\cdot)$. Consequently, given the multi-object state X_k, the entire set of measurements Z_k is the superposition of the detected $D_k(X_k)$ and clutter measurements K_k:

$$Z_k(X_k) = D_k(X_k) \bigcup K_k. \tag{2.84}$$

It is important that the detected points $D_k(X_k)$ and clutter points K_k are both independent of one another.

Based on this RFS formulation, it follows from FISST convolution formula that the probability density for the multi-object observation set Z_k given the multi-object state X_k is derived as [122, 237]:

$$g_k(Z_k|X_k) = \sum_{U \subseteq Z_k} \pi_D(U|X_k)\pi_K(Z_k - U) \tag{2.85}$$

where $\pi_D(\cdot|X_k)$ is the probability density of the detected observations $D_k(X_k)$, $\pi_K(\cdot)$ is the probability density of the clutter measurements K_k, and $Z_k - U$ is the set difference between Z_k and U.

Multi-Object Bayes Recursion

Given the measurement history $Z_{1:k} \triangleq (Z_1, ..., Z_k)$, all information on the set of objects $X_{0:k} \triangleq (X_0, ..., X_k)$ is captured in the multi-object posterior density $\pi_{0:k}(\cdot|Z_{1:k})$, which is computed recursively for $k \geq 1$ according to:

$$\pi_{0:k}(X_{0:k}|Z_{0:k}) \propto g_k(Z_k|X_k) f_{k|k-1}(X_k|X_{k-1}) \pi_{0:k-1}(X_{0:k-1}|Z_{0:k-1}), \tag{2.86}$$

where g_k is the likelihood given in (2.85) and $f_{k|k-1}$ is transition density given in (2.81). Implementation of the (full) posterior Bayes recursion is computationally expensive. Several works in the tracking literature have demonstrated tractable approaches to propagate an approximate multi-object posterior [269–271]. In the context of labeled RFS, the recursion (2.86) has an analytic solution with a tractable implementation via Gibbs sampling [266].

A cheaper alternative is the propagation of the multi-object filtering density, which is the marginal of the multi-object posterior density at the current time. In Bayesian filtering, let $\pi_{k-1}(\cdot|Z_{1:k-1})$ be the filtering density for the multi-object state at time $k-1$, where $Z_{1:k-1}$ denotes all measurements received by the filter from time 1 up to time $k-1$. In the context of Bayesian filtering, this density $\pi_{k-1}(\cdot|Z_{1:k-1})$ at time $k-1$ is predicted forward to obtain the predicted density $\pi_{k|k-1}(\cdot|Z_{1:k-1})$ via the *Chapman-Kolmogorov* equation:

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X)\pi_{k-1}(X|Z_{1:k-1})\delta X. \tag{2.87}$$

Note that $f_{k|k-1}$ is the transition density expressed in (2.81), which is derived from the multi-object transition model that includes new object births, existing object survivals and deaths.

The predicted density $\pi_{k|k-1}(\cdot|Z_{1:k-1})$ for the multi-object state is updated with the measurement set Z_k received at time k using the Bayes rule to obtain the new filtering density $\pi_k(\cdot|Z_{1:k})$ at time k:

$$\pi_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\pi_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X)\pi_{k|k-1}(X|Z_{1:k-1})\delta X}. \tag{2.88}$$

Note that g_k is likelihood function given in (2.85), which is derived from the multi-object observation model that includes missed detections, true measurements and false alarms (clutter). Note that the integral with respect to δX is the set integral from FISST [237] as described in (2.73).

In summary, Eq. (2.87) and (2.88) constitute the recursive multi-object Bayes filter. In general, the complex computation of the set integrals and the combinatorial complexities in the multi-object densities render the implementation of the filter computationally expensive [272]. Implementations of the multi-object Bayes filter using SMC methods have been demonstrated in [268, 273–275]. Further, cheap and tractable solutions to the multi-object Bayes filter by means of moment and density approximations, for example, the PHD filter [267, 276], CPHD filter [277, 278], and the CBMeMBer filter [279] have been proposed. In the context of labeled RFS, the recursive multi-object Bayes filter has an analytic solution, which is discussed in Section 2.4.5.

Probability Hypothesis Density Filter

The PHD filter is derived by approximating the predicted and filtering multi-object RFSs as Poisson RFSs. According to FISST, the first moment of the multi-object filtering density is the PHD function, which, in this case, is the intensity function of a Poisson RFS [267]. The PHD filter operates by propagating the intensity function. Based on the standard multi-object dynamic and measurement models, the PHD filter operates only on the single-object state space and sidesteps the data association problem.

The complete derivation of the PHD filter is presented in [267], followed by an SMC implementation in [268] and a Gaussian mixture implementation in [276]. Convergence analyses for the PHD filter implementation in [268] are given in [280] and a convergence analysis for the implementation in [276] is given in [281]. Estimates for individual object states can be extracted by selecting the points in the single-object state space with the highest intensities. The PHD filter is computationally cheap yielding a algorithmic complexity that is linear in the number of filtered objects and measurements.

Cardinalized Probability Hypothesis Density Filter

Only a single parameter is used to describe the cardinality distribution of a PHD filter (i.e., the mean of the cardinality distribution, which is Poisson distributed) [278]. As a result, the estimation of the cardinality by a PHD filter has a high variance when the number of objects is high, because the cardinality is Poisson distributed and a Poisson distribution has equal variance and mean [278]. The CPHD filter generalizes the PHD filter by permitting an arbitrary cardinality distribution rather than confining it to be Poisson as the PHD filter does. Specifically, the CPHD filter is derived by assuming the filtering densities as i.i.d cluster RFSs.

The CPHD filter computes both the cardinality distribution and the intensity of the multi-object density recursively through time. While this means that CPHD filter is computationally more expensive than the PHD filter, the upshot is better accuracy in object state estimation and lower variance in the cardinality distribution. The groundwork for the CPHD filter was laid in [277] and analytic implementations ware given in [278]. Different implementations presented in [278] are based on the Kalman filter, EKF, UKF and the particle filter. It is also important to point out that based on the standard multi-object measurement and dynamic models, the CPHD filter sidesteps the requirement to perform data association, thereby contributing to a large saving in algorithmic complexity. It has been shown that the CPHD filter has a complexity that is cubic in the number of measurements and linear in the number of filtered objects.

Cardinality Balanced Multi-Bernoulli Filter

The CBMeMBer filter [279] approximates the multi-object filtering density by means of the parameters of a multi-Bernoulli RFS. The CBMeMBer filter can be implemented via a Gaussian mixture for linear Gaussian models, with EKF and UKF extensions for mildly nonlinear Gaussian scenarios [279]. In addition, an SMC implementation for the CBMeMBer filter is proposed to accommodate for nonlinear models [279]. The advantage of implementing a CBMeMBer filter is that the multi-object state extraction is more accurate and less computationally expensive compared to, for example, the SMC-PHD filters that require particle clustering to acquire the state estimates, which is computationally expensive [268]. Similar to the PHD filter, the CBMeMBer filter has a complexity that is linear in the number of received measurements and hypothesized objects. A convergence analysis of the SMC-CBMeMBer implementation is undertaken in [282].

2.4.5 Labeled Random Finite Set

The drawbacks of the aforementioned multi-object filters are two-fold. First, these filters are derived based on some form of functional approximations to the multi-object Bayes filter, resulting in certain limitations. For example, the PHD and CBMeMBer filters only work best in low clutter and high probability of detection scenarios [277, 279]. Second, these approximate multi-object filters are not formulated for estimating object trajectories, which is supposed to be the goal of MOT [19]. Thus, additional heuristic track management techniques (post-processing) are applied to the multi-object state estimates to obtain the object trajectories.

The notion of labeled RFSs addresses the trajectories of objects via the assignment of unique labels. When each object state is tagged by a unique label, the history of the multi-object states can then be interpreted as a set of object trajectories. Therefore, the estimation or filtering of multi-object labeled states amounts to multi-object trajectory estimation [266]. By definition, a labeled RFS is essentially an RFS on $\mathbb{X}\times\mathbb{L}$, where each element in the RFS is augmented with a unique label denoted by ℓ that takes values in a discrete label space $\mathbb{L}$ [19, 20]. As the labels are unique and distinct, the cardinality of the label set is equal to the cardinality of the labeled RFS itself. By convention, a labeled object state is represented by bold lower case (e.g., $\boldsymbol{x} = (x, \ell)$, where x denotes the object state and ℓ denotes the distinct label) and a labeled multi-object state is denoted by a bold uppercase (i.e., $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_{|\boldsymbol{X}|}\}$) [19, 20].

More labeled RFS notations and operators are specified in the following:

- The label extractor

$$\mathcal{L}(\boldsymbol{X}) = \{\mathcal{L}(\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{X}\} \text{ where } \mathcal{L}((x, \ell)) = \ell, \tag{2.89}$$

Note that a realization $\boldsymbol{X}$ of a labeled RFS always satisfies $|\mathcal{L}(\boldsymbol{X})| = |\boldsymbol{X}|$.

- Distinct label indicator

$$\Delta(\boldsymbol{X}) = \delta_{|X|}(|\mathcal{L}(\boldsymbol{X})|) \tag{2.90}$$

- Standard inner product

$$\langle f, g \rangle = \int f(x)g(x)dx \tag{2.91}$$

- Multi-object exponential

$$[h]^X = \prod_{x \in X} h(x) \tag{2.92}$$

- Inclusion function

$$1_Y(X) = \begin{cases} 1 & \text{if} \quad X \subseteq Y \\ 0 & \text{otherwise} \end{cases} \tag{2.93}$$

- Kronecker delta function

$$\delta_Y[X] = \begin{cases} 1 & \text{if} \quad X = Y \\ 0 & \text{otherwise} \end{cases} \tag{2.94}$$

A labeled RFS distributed according to $\boldsymbol{\pi}$ is related to its unlabeled version π according to [19]:

$$\pi(\{x_1, ..., x_n\}) = \sum_{(\ell_1,...,\ell_n) \in \mathbb{L}^n} \boldsymbol{\pi}(\{(x_1, \ell_1), ..., (x_n, \ell_n)\}). \tag{2.95}$$

This shows that marginalizing the labels of the labeled RFS density yields the density of its unlabeled version. The set integral for a function $h : \mathcal{F}(\mathbb{X} \times \mathbb{L}) \to \mathbb{R}$ is given as [19]:

$$\int h(\boldsymbol{X})\delta\boldsymbol{X} = \sum_{n=0}^{\infty} \frac{1}{n!} \sum_{(\ell_1,...,\ell_n) \in \mathbb{L}^n} \int_{\mathbb{X}^n} h(\{(x_1, \ell_1), ..., (x_n, \ell_n)\})d(x_1, ..., x_n). \tag{2.96}$$

The following subsections discuss how the integration of distinct labels into the RFS framework yields some simplified representations of the conventional RFSs and introduce a generalized density representation that is vital for the derivation of a closed-form multi-object Bayesian filter.

Labeled Poisson RFS

Augmenting the Poisson RFS X using labels from discrete label space $\mathbb{L}$, yields the labeled Poisson RFS $\boldsymbol{X}$ on $\mathbb{X} \times \mathbb{L}$ with intensity v. Note that the set of labeled states is

not distributed according to a Poisson RFS, rather its density is given by [19]:

$$\boldsymbol{\pi}(\{(x_1,\ell_1),...,(x_n,\ell_n)\}) = \delta_{\mathbb{L}(n)}[\{\ell_1,...,\ell_n\}]\text{Pois}_{\langle v,1\rangle}(n)\prod_{i=1}^{n}\frac{v(x_i)}{\langle v,1\rangle}, \tag{2.97}$$

where $\mathbb{L}(n) = \{\ell_i \in \mathbb{L}\}_{i=1}^{n}$ and $\text{Pois}_{\lambda_p}(n) = e^{-\lambda_p}\lambda_p^n/n!$ is the Poisson distribution with rate $\lambda_p = \langle v, 1\rangle$.

The procedure below demonstrates sampling from a labeled Poisson RFS [19]:

Drawing Samples from a Labeled Poisson RFS

Initialize $\boldsymbol{X} = \emptyset$
Draw $n \sim \text{Pois}_{\langle v,1\rangle}$
for $i = 1 : n$
  Draw $x \sim v(\cdot)/\langle v, 1\rangle$
  Assign $\boldsymbol{X} = \boldsymbol{X} \cup \{(x, \ell_i)\}$
end for

It can be shown that applying (2.95) to (2.97) and simplifying the sum of labels result in (2.74). Therefore, marginalizing the labels of a labeled Poisson RFS yields a Poisson RFS with intensity v.

Labeled Multi-Bernoulli RFS

Augmenting the successful Bernoulli components from a multi-Bernoulli RFS using labels from discrete label space $\mathbb{L}$, yields the labeled multi-Bernoulli (LMB) RFS $\boldsymbol{X}$ on $\mathbb{X}\times\mathbb{L}$ with a (finite) parameter set $\{(r^{(\zeta)}, p^{(\zeta)}) : \zeta \in \Psi\}$. Given a successful Bernoulli component $(r^{(\zeta)}, p^{(\zeta)})$, the label of a state is given by a 1-1 mapping $\varrho : \Psi \to \mathbb{L}$. The pseudocode below demonstrates how a sample from the LMB RFS is drawn [19]:

Drawing Samples from a LMB RFS

Initialize $\boldsymbol{X} = \emptyset$
for $\zeta \in \Psi$
  Draw $u \sim \text{Uniform}[0, 1]$
    If $u \leq r^{(\zeta)}$
      Draw $x \sim p^{(\zeta)}(\cdot)$
      Assign $\boldsymbol{X} = \boldsymbol{X} \cup \{(x, \varrho(\zeta))\}$
    end if
end for

The expression for the (multi-object) density of the LMB RFS on $\mathbb{X}\times\mathbb{L}$ with parameter set $\{(r^{(\zeta)}, p^{(\zeta)}) : \zeta \in \Psi\}$ is [19]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X})1_{\varrho(\Psi)}(\mathcal{L}(\boldsymbol{X}))[\Pi(\boldsymbol{X};\cdot)]^{\Psi}, \tag{2.98}$$

where

$$\Pi(\boldsymbol{X};\zeta) = \sum_{(x,\ell)\in\boldsymbol{X}} \delta_{\varrho(\zeta)}[\ell] r^{(\zeta)} p^{(\zeta)}(x) + (1 - 1_{\mathcal{L}(\boldsymbol{X})}(\varrho(\zeta)))(1 - r^{(\zeta)}). \tag{2.99}$$

It can be shown that applying (2.95) to (2.98) and simplifying the sum of labels result in (2.77). Therefore, marginalizing the labels of a LMB RFS yields a multi-Bernoulli RFS.

Generalized Labeled Multi-Bernoulli RFS

A generalized labeled multi-Bernoulli (GLMB) RFS on $\mathbb{X}\times\mathbb{L}$ is distributed according to [19]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{e\in\mathbb{E}} \omega^{(e)}(\mathcal{L}(\boldsymbol{X})) \left[p^{(e)}\right]^{\boldsymbol{X}} \tag{2.100}$$

where $\mathbb{E}$ is a discrete index set with $\omega^{(e)}(L)$ and $p^{(e)}$ satisfying

$$\sum_{L\in\mathbb{L}} \sum_{e\in\mathbb{E}} \omega^{(e)}(L) = 1, \tag{2.101}$$

$$\int_{x\in\mathbb{X}} p^{(e)}(x,\ell)dx = 1. \tag{2.102}$$

A GLMB has a mixture of components, where each component consists of a weight $\omega^{(e)}(\mathcal{L}(\boldsymbol{X}))$ and a multi-object exponential $[p^{(e)}]^{\boldsymbol{X}}$. Note that the GLMB weight $\omega^{(e)}(\mathcal{L}(\boldsymbol{X}))$ depends only on the set of labels, and $[p^{(e)}]^{\boldsymbol{X}}$ depends on the entire multi-object state. It is easy to verify that the GLMB density integrates to 1, as it should, by applying (2.96) to (2.100).

The previously described types of labeled RFSs are special cases of the GLMB RFS. For example, the labeled Poisson RFS is a special case of the GLMB RFS with

$$p^{(e)}(x,\ell) = v(x)/\langle v, 1\rangle, \tag{2.103}$$

$$\omega^{(e)}(L) = \delta_{\mathbb{L}(|L|)}[L]\text{Pois}_{\langle v,1\rangle}(|L|), \tag{2.104}$$

and the LMB RFS is a special case of the GLMB RFS with

$$p^{(e)}(x,\ell) = p^{(\ell)}(x), \tag{2.105}$$

$$\omega^{(e)}(L) = \prod_{i \in \mathbb{L}} \left(1 - r^{(i)}\right) \prod_{\ell \in L} \frac{1_{\mathbb{L}}(\ell) r^{(\ell)}}{1 - r^{(i)}}. \tag{2.106}$$

Note that the superscript (e) is not necessary because the index space has only one element.

2.4.6 The Generalized Labeled Multi-Bernoulli Filter

If a filtering density is modeled as a GLMB, all subsequent predicted and updated densities remain in the GLMB form based on the standard multi-object dynamic model and measurement model [19]. In other words, the GLMB density is a *conjugate prior* under the Bayes rule [19].

To specify the GLMB filter, we express the GLMB in a different but equivalent form [19]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{(I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} \omega^{(I,\xi)} \delta_I[\mathcal{L}(\boldsymbol{X})] \left[p^{(\xi)}\right]^{\boldsymbol{X}}, \tag{2.107}$$

where in this form (originally called the δ-GLMB), the GLMB is a set of weighted components generated over the space of $\mathcal{F}(\mathbb{L}) \times \Xi$. Each component is represented by (I,ξ), which is a pair of label set and association map history from space $\mathcal{F}(\mathbb{L}) \times \Xi$. In filtering, the GLMB density (2.107) is computed recursively through time. Hence, $I \in \mathcal{F}(\mathbb{L})$ is a set of labels that exists up to the current time while $\xi \triangleq (\theta_1, ..., \theta_k) \in \Xi$ is the history of association maps for the filtering density at time k, where θ_i is an association map at time i that maps object labels to their corresponding measurements.

Note that the history of association mappings only arises due to the standard multi-object measurement model where each measurement follows a positive 1-1 mapping to the object label. This is further illustrated later in this subsection. The pair (I,ξ) therefore represents a hypothesis that the label set I has the association map history ξ. The probability (also referred to as the weight) of the component (I,ξ) is represented by $\omega^{(I,\xi)}$ and $p^{(\xi)}(\cdot,\ell)$ represents the spatial probability density function for the label $\ell \in I$ and the association map history ξ. Also note that the $\Delta(\cdot)$ operator ensures that the probability of a multi-object state with repeated labels is zero. The GLMB filter operates by propagating recursively the predicted and filtering multi-object densities (in GLMB form) forward in time. Below, we show that this propagation is analytical (i.e., can be computed in closed form) and, therefore, an exact solution to the multi-object Bayes filter.

For the remainder of this chapter, we suppress the time index k and use '+' to indicate the next time step for sets, functions, parameters, and densities (e.g., $\boldsymbol{X}$ used in place of $\boldsymbol{X}_k$ and $\boldsymbol{X}_+$ used in place of $\boldsymbol{X}_{k+1}$).

The GLMB Prediction

The following is similar to Section 2.4.4, except that the multi-object dynamic model is now described by a labeled multi-object state and the labeled RFS. Given a labeled multi-object state $\boldsymbol{X}$ at the current time step, an object $\boldsymbol{x} = (x, \ell) \in \boldsymbol{X}$ either survives with probability of survival $P_S(\boldsymbol{x})$ and changes to a new state (x_+, ℓ_+) in the next time step according to the density $f_{S,+}(x_+|x, \ell)\delta_\ell[\ell_+]$, or not with probability $1 - P_S(\boldsymbol{x})$. With the inclusion of label into the state, the label remains unchanged during the transition and only the state of the object changes.

As discussed in Section 2.4.4, given the set of objects existing previously, the survivability of objects is modeled as a multi-Bernoulli RFS. As a corollary, let $\boldsymbol{S}_+$ be the set of surviving objects out of $\boldsymbol{X}$ in the next time step, it follows from (2.98) that $\boldsymbol{S}_+$ is a labeled multi-Bernoulli RFS expressed as [19]:

$$\boldsymbol{f}_S(\boldsymbol{S}_+|\boldsymbol{X}) = \Delta(\boldsymbol{S}_+)\Delta(\boldsymbol{X})1_{\mathcal{L}(\boldsymbol{X})}(\mathcal{L}(\boldsymbol{S}))[\boldsymbol{\Pi}(\boldsymbol{S}_+;\cdot)]^{\boldsymbol{X}} \tag{2.108}$$

where:

$$\boldsymbol{\Pi}(\boldsymbol{S}_+; x, \ell) = \sum_{(x_+,\ell_+)\in\boldsymbol{S}_+} \delta_\ell[\ell_+]P_S(x, \ell)f_{S,+}(x_+|x) + \left(1 + 1_{\mathcal{L}(\boldsymbol{S})}(\ell)\right)(1 - P_S(x, \ell)). \tag{2.109}$$

Aside from surviving objects, new objects may also be born in the next time step. Let $\mathbb{B}_+$ denote the label space for newly born objects. If $\mathbb{L}$ denotes the label space for objects at the current time step (which includes the labels of all objects born up to that time), then $\mathbb{L}_+$ is the label space for all objects at the next time step (i.e., $\mathbb{L}_+ = \mathbb{L} \cup \mathbb{B}_+$ and $\mathbb{L} \cap \mathbb{B}_+ = \emptyset$) [19]. This property reflects the uniqueness of the labels which is fundamentally the basis of labeled RFS.

Let $\boldsymbol{B}_+$ be the set of the new labeled objects. Using the labeled multi-Bernoulli birth model with parameter set $\left\{\left(r_{B_+}^{(\ell)}, f_{B_+}^{(\ell)}(\cdot)\right)\right\}_{\ell\in\mathbb{B}_+}$, the density of $\boldsymbol{B}_+$ is given as [19]:

$$\boldsymbol{f}_{\boldsymbol{B}}(\boldsymbol{B}_+) = \Delta(\boldsymbol{B}_+)\omega_B(\mathcal{L}(\boldsymbol{B}_+))[f_{B,+}(\cdot)]^{\boldsymbol{B}_+}, \tag{2.110}$$

where

$$\omega_B(\mathcal{L}(\boldsymbol{B}_+)) = \prod_{i\in\mathbb{B}_+}(1 - r_{B,+}^{(i)}) \prod_{\ell\in\mathcal{L}(\boldsymbol{B}_+)} \frac{1_{\mathbb{B}_+}(\ell)r_{B,+}^{(\ell)}}{1 - r_{B,+}^{(\ell)}}, \tag{2.111}$$

$$f_{B,+}(x, \ell) = f_{B,+}^{(\ell)}(x). \tag{2.112}$$

Consequently, the complete set of object states in the next time step is denoted by $\boldsymbol{X}_+ = \boldsymbol{X} \uplus \boldsymbol{B}_+$, which is the superposition of the sets of surviving and the newborn objects. Using (2.108) and (2.110), the multi-object transition kernel $\boldsymbol{f}_+(\cdot|\cdot)$ is given by [19]:

$$\boldsymbol{f}_+(\boldsymbol{X}_+|\boldsymbol{X}) = \boldsymbol{f}_S(\boldsymbol{X}_+ \cap (\mathbb{X}\times\mathbb{L})|\boldsymbol{X})\boldsymbol{f}_{\boldsymbol{B}}(\boldsymbol{X}_+ \cap (\mathbb{X}\times\mathbb{B}_+)). \tag{2.113}$$

Note that difference between (2.81) and (2.113) is that the combinatorial sum in (2.81) is nonexistent in the labeled version of multi-object transition kernel (2.113). This contributes to a huge computational saving.

Let a multi-object prior be a GLMB as expressed in (2.107), applying FISST and the *Chapman-Kolmogorov* equation yield a predicted multi-object density that remains in the same form, which is computed by [19]:

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+) = \Delta(\boldsymbol{X}_+) \sum_{(I_+,\xi_+)\in\mathcal{F}(\mathbb{L}_+)\times\Xi} \omega_+^{(I_+,\xi_+)} \delta_{I_+}[\mathcal{L}(\boldsymbol{X}_+)] \left[p_+^{(\xi)}\right]^{\boldsymbol{X}_+}, \tag{2.114}$$

where

$$\omega_+^{(I_+,\xi)} = \omega_B(I_+ \cap \mathbb{B}_+)\omega_S^{(\xi)}(I_+ \cap \mathbb{L}), \tag{2.115}$$

$$\omega_B(B) = \prod_{i\in\mathbb{B}_+} \left(1 - r_{B,+}^{(i)}\right) \prod_{\ell\in B} \frac{1_{\mathbb{B}_+}(\ell) r_{B,+}^{(\ell)}}{1 - r_{B,+}^{(i)}}, \tag{2.116}$$

$$\omega_S^{(\xi)}(L) = \left[\bar{P}_S^{(\xi)}\right]^L \sum_{I\supseteq L} \left[1 - \bar{P}_S^{(\xi)}\right]^{I-L} \omega^{(I_+,\xi)}, \tag{2.117}$$

$$p_+^{(\xi)}(x_+,\ell_+) = 1_{\mathbb{L}}(\ell_+) p_S^{(\xi)}(x_+,\ell_+) + 1_{\mathbb{B}_+}(\ell_+) f_{B,+}(x_+,\ell_+), \tag{2.118}$$

$$p_S^{(\xi)}(x_+,\ell_+) = \frac{\left\langle P_S(\cdot,\ell_+) f_{S,+}(x_+|\cdot,\ell_+), p^{(\xi)}(\cdot,\ell_+)\right\rangle}{\bar{P}_S^{(\xi)}(\ell_+)}, \tag{2.119}$$

$$\bar{P}_S^{(\xi)}(\ell_+) = \left\langle P_S(\cdot,\ell_+), p^{(\xi)}(\cdot,\ell_+)\right\rangle, \tag{2.120}$$

$$f_{B,+}(x_+,\ell_+) = f_{B,+}^{(\ell_+)}(x_+). \tag{2.121}$$

The GLMB Update

For a given $\boldsymbol{X}$, an object with labeled state $\boldsymbol{x} \in \boldsymbol{X}$ produces a measurements in one of two ways: there is no measurement with probability of missed detection $1 - P_D(\boldsymbol{x})$, or a measurement $z \in \mathbb{Z}$ is produced with detection probability $P_D(\boldsymbol{x})$ and likelihood $g(z|\boldsymbol{x})$. As discussed in Section 2.4.4, each labeled state $\boldsymbol{x} \in \boldsymbol{X}$ generates a Bernoulli RFS parameterized by $(P_D(\boldsymbol{x}), g(\cdot|\boldsymbol{x}))$. Given $\boldsymbol{X}$, all individual Bernoulli RFSs generated by the object states are conditionally independent of one another. Therefore, the set of detected measurements denoted by $W \subset \mathbb{Z}$ is modeled as a multi-Bernoulli RFS that is parameterized by $\{P_D(\boldsymbol{x}), g(\cdot|\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{X}\}$. Conditioned on $\boldsymbol{X}$, the detected set of measurements W has a probability density given by [19]:

$$\pi_D(W|\boldsymbol{X}) = \{P_D(\boldsymbol{x}), g(\cdot|\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{X}\}(W). \tag{2.122}$$

Apart from the detections, the multi-object measurement set Z may also contain

false or clutter measurements. Recall from Section 2.4.4 that the set $K \subset \mathbb{Z}$ denotes the set of false measurements, which by assumption is statistically independent of $W \subset \mathbb{Z}$. A Poisson RFS is the standard model for the collection of false measurements. Hence, the set of false measurements K has a probability density given by [19]:

$$\pi_K(K) = e^{-\langle \kappa, 1 \rangle} [\kappa]^K, \tag{2.123}$$

where $\kappa(\cdot)$ is the intensity of a Poisson RFS, and $\langle \kappa, 1 \rangle$ is the rate of the Poisson distribution

Consequently, the measurement set Z is the superposition of false measurements and detected measurements. Based on FISST convolution [122, 237], the multi-object likelihood of Z is given by [19]:

$$g(Z|\boldsymbol{X}) = \sum_{U \subseteq Z} \pi_D(U|\boldsymbol{X}) \pi_K(Z - U). \tag{2.124}$$

This expression is similar to that in Section 2.4.4 because only the object states are labeled while the measurements remain unlabeled as previously described. Consequently, the combinatorial sum remains in the likelihood of the GLMB filter.

The inclusion of labels in the object states enables the multi-object likelihood (2.124) to be expressed in a compact form [19]:

$$g(Z|\boldsymbol{X}) = e^{-\langle \kappa, 1 \rangle} \kappa^Z \sum_{\theta \in \Theta} \delta_{\theta^{-1}(\{0:|Z|\})} [\mathcal{L}(\boldsymbol{X})] [\psi_Z(\cdot; \theta)]^{\boldsymbol{X}}, \tag{2.125}$$

where:

$$\psi_Z(x, \ell; \theta) = \begin{cases} \frac{P_D(x,\ell) g(z_{\theta(\ell)}|x,\ell)}{\kappa(z_{\theta(\ell)})} & \theta(\ell) > 0 \\ 1 - P_D(x, \ell) & \theta(\ell) = 0 \end{cases}, \tag{2.126}$$

Θ is the space of positive 1-1 mappings $\theta : \mathbb{L} \to \{0 : |Z|\} \triangleq \{0, 1, ..., |Z|\}$ such that $\theta(i) = \theta(i') > 0$ implies $i = i'$. The mapping condition ensures that at any point in time, an object can only generate at most one measurement. The term $\delta_{\theta^{-1}(\{0:|Z|\})}(\mathcal{L}(\boldsymbol{X}))$ ensures that only mappings with domain $\mathcal{L}(\boldsymbol{X})$ are considered, which makes the summation of $[\psi_Z(\cdot; \theta)]^{\boldsymbol{X}}$ over all θ valid.

Let a multi-object prior be a GLMB as expressed in (2.107), then substituting the multi-object likelihood (2.125) into the Bayes formula and using FISST yield a filtering density that remains in the same form, which is computed by [19]:

$$\boldsymbol{\pi}(\boldsymbol{X}|Z) = \Delta(\boldsymbol{X}) \sum_{(I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} \sum_{\theta \in \Theta} \omega^{(I,\xi,\theta)}(Z) \delta_I[\mathcal{L}(\boldsymbol{X})] \left[p^{(\xi,\theta)}(\cdot|Z) \right]^{\boldsymbol{X}}, \tag{2.127}$$

where:

$$\omega^{(I,\xi,\theta)}(Z) = \frac{\delta_{\theta^{-1}(\{0:|Z|\})}(I)\omega^{(I,\xi)}\left[\bar{\psi}_Z^{(\xi,\theta)}\right]^I}{\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi}\sum_{\theta\in\Theta}\delta_{\theta^{-1}(\{0:|Z|\})}(I)\omega^{(I,\xi)}\left[\bar{\psi}_Z^{(\xi,\theta)}\right]^I}, \tag{2.128}$$

$$p^{(\xi,\theta)}(x,\ell|Z) = \frac{p^{(\xi)}(x,\ell)\psi_Z(x,\ell;\theta)}{\bar{\psi}_Z^{(\xi,\theta)}(\ell)}, \tag{2.129}$$

$$\bar{\psi}_Z^{(\xi,\theta)}(\ell) = \left\langle p^{(\xi)}(\cdot,\ell)\psi_Z(x,\ell;\theta)\right\rangle. \tag{2.130}$$

Implementation

A close examination of the predicted and filtering density of the GLMB filter reveals that it is not feasible to generate hypotheses on the entire spaces of $\mathcal{F}(\mathbb{L}_+)\times\Xi$ and $\mathcal{F}(\mathbb{L})\times\Xi\times\Theta$ respectively over time. To avoid an exhaustive computation of all GLMB hypotheses (or components), it is shown in [20] that keeping components with significant weights and discarding the insignificant ones minimizes the L_1-error from the actual density. In the original implementation, the predicted and filtering density of the GLMB filter are truncated via deterministic optimization algorithms (i.e., the k-shortest path [283] and *Murty's* ranked assignment [284]), which are implementable in polynomial time. Both algorithms seek to find the highest-weighted GLMB components without resorting to exhaustive enumeration. This implementation of the GLMB filter achieves an overall complexity that is, at best, cubic in the number of measurements [20].

Having two truncations of the predicted and filtering GLMB densities result in an issue where a large portion of predicted components would yield updated components with negligible weights. Consequently, the propagation is inefficient since most computations are wasted on predicted components that result in low-weighted updated components. An efficient implementation uses a joint prediction and update scheme [21], which requires exactly one joint truncation process per recursion. The joint propagation strategy avoids the wastage in the prediction while retaining the filtering performance. This efficient implementation of the GLMB filter truncates the filtering density based on Gibbs sampling which is a simulation-based approach that generates components according to their weights. Using the Gibbs sampler, the GLMB filter achieves a linear complexity in the number of measurements [21]. The computational savings of the Gibbs sampler comes from not being required to generate a solution that is ranked or ordered as opposed to the deterministic approaches. This is not an issue because such ordering of the components is not needed in the truncated GLMB densities.

Multi-Object State Estimation

Given a multi-target filtering density with a parameter set $\{(\omega^{(I,\xi)}, p^{(\xi)}) : (I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi\}$, the state estimates can be obtained via the Bayes-optimal joint multi-object estimator and marginal multi-object estimator [237]. However, these estimators are difficult to compute.

A suboptimal yet tractable implementation of the marginal multi-object estimator is achieved by first finding the *maximum a posteriori* (MAP) cardinality estimate from the cardinality distribution [19]:

$$\rho(n) = \sum_{(I,\xi)\in\mathcal{F}_n(\mathbb{L})\times\Xi} \omega^{(I,\xi)}, \tag{2.131}$$

$$\hat{n} = \underset{n}{\operatorname{argmax}} \quad \rho(n). \tag{2.132}$$

where $\mathcal{F}_n(\mathbb{L})$ denotes the finite subsets of $\mathbb{L}$ with exactly n elements. Then, compute the highest-weighted component $(\hat{I},\hat{\xi})$ based on the MAP cardinality estimate $\hat{n}$:

$$(\hat{I},\hat{\xi}) = \underset{(I,\xi)}{\operatorname{argmax}} \quad \omega^{(I,\xi)}\delta_{\hat{n}}(|I|). \tag{2.133}$$

Lastly, the estimate of the multi-object state $\hat{\boldsymbol{X}}$ is obtained by:

$$\hat{\boldsymbol{X}} = \{(\hat{x},\hat{\ell}) : \hat{\ell} \in \hat{I}\},$$

where the state estimate $\hat{x}$ of any track $\hat{\ell} \in \hat{I}$ is obtained from its probability density $p^{(\hat{\xi})}$ using, for example, its mean or mode. It is important to know that multi-object estimates produced by this estimator may result in track fragmentations, i.e., tracks that do not have consecutive labels over time. This is because the estimator is instantaneous and does not consider past observations.

Note that the GLMB filtering density is indeed consistent with respect to track continuity, as the density encapsulates the association histories of all tracks or labels according to the standard multi-object system model, which is consistent with the inherent constraints for objects and labels. Given an estimated association history $\hat{\xi} \triangleq (\hat{\theta}_1, ..., \hat{\theta}_k)$ for each estimated object with label $\hat{\ell}$, a posterior density $p^{(\hat{\xi})}_{1:k}(\cdot|z_{\hat{\theta}_1(\hat{\ell}):\hat{\theta}_k(\hat{\ell})})$ can be constructed via a recursive algorithm, e.g., the Rauch-Tung-Striebel or Kalman smoother [285, 286]. Extracting a state estimate $\hat{x}_{1:k}$ from the posterior density $p^{(\hat{\xi})}_{1:k}$ of the object guarantees a continuous track with no fragmentations. It is stressed that the presence of track fragmentations is caused by the choice of a naive estimator for the multi-object state estimates and is not due to any inconsistency in the propagation of the GLMB filtering density.

2.4.7 Developments of Labeled RFS

The establishment of the labeled RFS framework has prompted a myriad of efforts and developments in MOT. The following are some of the research topics related to the labeled RFS in the literature.

Centralized and Distributed Multi-Sensor Tracking

The GLMB filter can be extended to centralized multi-sensor networks with tractable implementations [22, 287]. To further alleviate computational burden, approximate solutions to multi-sensor tracking have been proposed [288, 289]. In scenarios where sensors are widely distributed, it is more prudent to implement a decentralized (or distributed) tracking system, which benefits from lower communication cost and fault occurrences. The consensus for distributed fusion for multi-sensor tracking systems are the generalized covariance intersection [290, 291], minimum information loss [292, 293], and distributed cross-entropy [294, 295].

Sensor Control

In advanced tracking systems, sensor control is implemented to improve the quality and information content of measurement data for better tracking performance. Sensor control typically involves orientation and movement of sensor platform, which affect the sensor's ability to observe objects in the scene. In the literature, single-sensor control method with GLMB filtering has been proposed in [296], while multi-sensor control methods have been proposed in [297–300]. Solutions that enable multiple sensors to cooperate with each other for information sharing are useful for application in autonomous vehicles and robotics [301–303]. A unified method that generalizes the GLMB filter for MOT and sensor management is proposed in [304].

Jump Markov System (Multiple Models)

In standard form, MOT solutions have assumed the same stochastic dynamic model for all objects. This assumption can be violated, for example, in traffic monitoring situations where some objects fit well in a NCV model, while faster maneuvering objects fit better in a NCT model. The GLMB filter has the versatility to be extended for multiple models as proposed in [305–307]. The tracking of maneuvering and interactive objects using LMB filtering is proposed in [308, 309].

Generic Measurement Models

In the standard measurement model, measurements are assumed to be preprocessed into point detections where each object only generates one detection at most. Based on this

model, the GLMB density is a conjugate prior for the standard multi-object measurement likelihood, which forms the basis for the GLMB filter [19–21]. In contrast, a generic measurement model has no simplifying assumptions on the multi-object likelihood, enabling it to encapsulate tracking with TBD, superpositional measurements, merged and extended measurements, and acoustic amplitude observations [310]. Since the GLMB density is not necessarily a conjugate prior for a generic measurement likelihood, several works have proposed methods to overcome this issue. For example, a labeled RFS filter with a generic measurement model [310–312], a labeled RFS filter for tracking objects with extended and merged measurements [313, 314], and a joint detection and tracking of multi-sensor image observations [315, 316].

Adaptive Models and Parameter Estimation

The GLMB filter typically requires *a priori* knowledge of the birth locations, process noise parameters, detection probability, and clutter rate. In practice, these parameters may not be known and may also be time-varying. To this end, several works have incorporated adaptive birth models for the GLMB filter [317, 318]. Moreover, an adaptive measurement gating technique has been proposed in [319]. The unknown detection profile and clutter rate can be estimated with an online multi-object tracker via the so-called multi-class GLMB filter [320]. Further robust GLMB filtering techniques have been proposed in [321–327].

Practical Applications

The GLMB filter has been applied to a variety of real-world applications. Aside from audio and visual tracking and separation, GLMB filtering has been used for tracking biological cells [328–332], space debris [333–337], sea ice [338], people [59, 227, 339], and vehicles with laser range finder [340, 341]. Further, several works have applied labeled RFS trackers in satellite management [342, 343], acoustic source tracking and separation [53, 344, 345], simultaneous localization and mapping [346–348], active sonar [349], autonomous drones [350], and radar tracking [351–359].

Chapter 3

Audio Multi-Source Tracking and Separation

THIS chapter proposes a novel solution for separating an unknown and time-varying number of moving acoustic sources in a blind setting using multiple microphone arrays. A standard steered-response power phase transform method is applied to extract source position measurements, which inevitably contain noise, false detections, missed detections, and are not labeled with the source identities. The imperfect measurements lead to the space-time permutation problem, as there is no information on how the measurements are associated to the sources in space, nor how the measurements are connected across time, if at all. To solve this problem, a labeled random finite set tracking framework is adopted to jointly estimate the source positions and their labels or identities. Based on these trajectory estimates, a corresponding set of time-varying generalized side-lobe cancellers is constructed to perform source separation. The overall algorithm operates in a block-wise or an online fashion and is scalable with the number of microphone arrays. The quality of the measurements, tracking, and separation, are evaluated respectively, with the OSPA metric, OSPA$^{(2)}$ metric, and ITU-T P.835 based listening tests, on both real-world and simulated data. The content of this chapter has been published in [53].

3.1 Introduction

In microphone array processing, blind source separation (BSS) is the estimation of source signals, using only the received mixture signals with no information about the original sources and the mixing process [24]. In a realistic auditory scene, one of the main challenges for separating a mixture of concurrent sources is not only that the sources are moving, but also that the number of sources is unknown and time-varying, i.e., new sources can appear and existing sources can disappear or undergo silence periods. For static sources, established solutions to BSS include independent component

analysis (ICA) [25], sparseness-based approaches [26, 80], and non-negative matrix factorization (NMF) [27]. These methods can be extended for moving sources by using a block-wise approach wherein moving sources are assumed to be static within a short time block [41, 42].

An alternative and a more recent block-wise approach is based on tracking of multiple moving sources, and followed by spatial filtering for extracting the signal-of-interest (SOI) from the estimated position/direction at each time [42–44, 137]. One of the main difficulties in tracking an unknown number of sources in a reverberant environment is that acoustic localization measurements are subject to noise and false positives or negative, i.e., spurious or missing measurements. Moreover, the more pertinent issue is the space-time permutation problem. As in space, it is not known which measurements are connected to which sources, and in time, it is not known how the measurements are connected across time frames with respect to the sources. Furthermore, the solution must cater for possible appearance of new sources, movement of active or inactive sources, and disappearance of existing sources.

Classical dynamic Bayesian estimation techniques such as the particle filter have been applied to single source tracking in [69, 71, 119]. For multiple sources, there is uncertainty not only in the source position, but also in the number of sources, and the latter is not accounted for within the classical Bayesian framework [8]. Recent solutions for addressing multiple sources have relied on adaptations of the Rao-Blackwellised Particle Filter (RBPF) [42, 120], the Probabilistic Multiple Hypothesis Tracker (PMHT) [44], and the Joint Probabilistic Data Association (JPDA) filter [121]. The newer RFS framework based on Finite Set Statistics (FISST) [122], offers a principled mechanism to cater for an unknown and time-varying number of sources in a Bayesian setting, and is directly applicable to acoustic tracking [8]. The first RFS based solution for multi-source acoustic tracking was proposed in [72]. Subsequent RFS-based solutions have been proposed for multi-source acoustic tracking with the Probability Hypothesis Density (PHD) filter [10, 43, 126, 360], the Cardinalized PHD filter [128], the Cardinality-Balanced Multi-Target Multi-Bernoulli filter [129], and the RFS Particle Filter [130].

However, these above methods do not directly estimate source tracks, which are source position estimates associated with a common label. Consequently, they require a post-processing step such as track management to resolve each track individually. These methods are thus suboptimal in the sense that they solve the space-time permutation problem separately. As the spatial filtering module relies on accurate label or identity estimates, the presence of labeling errors results in switching in the separated signal estimates. Solving the space-time permutation problem jointly has the potential to significantly improve tracking performance and hence separation performance. Furthermore, the above mentioned approaches do not scale linearly in the number of arrays used in the system, thereby making them impractical for online implementation when the number of arrays is large.

In this chapter, we propose a novel online solution for multi-array BSS with an unknown time-varying number of moving sources in a 3D auditory scene. Our solution follows the approach of first obtaining position measurements, then tracking of multiple sources, and finally separation using spatial filtering, all in an online or block-wise fashion. Source position measurements obtained through Steered-Response Power Phase Transform (SRP-PHAT) [70] exhibit the space-time permutation issue, where it is not known which measurement (if any) is connected to which source at the current time, nor which measurements are connected to the same source across time. This work is the first to formally address the space-time permutation problem, using a labeled random finite set (RFS) approach [19–21] to jointly estimate the number of sources, their positions and their labels. The solution invokes the Multi-Sensor Generalized Labeled Multi-Bernoulli (MS-GLMB) tracker [22], which is a tractable linear complexity recursive filter for estimating the source trajectories from raw measurements. The tracking estimates at each frame are used to construct a set of time-varying beamformers, known as the Generalized Side-lobe Canceller (GSC) [60], which are used for multi-source separation. The proposed method is evaluated using real recordings and under different reverberation times via simulation. We use the Optimal Sub-Pattern Assignment (OSPA) which is a metric for two sets of points [50], to evaluate the quality of the array measurements. The tracking performance is evaluated using a variant of the OSPA metric called the OSPA$^{(2)}$, which is a proper metric for two sets of tracks [51]. Finally, we evaluate the separation performance via subjective listening tests according to the ITU-T P.835 methodology [52].

3.2 Problem Formulation and Solution Overview

One of the main challenges in BSS for multiple moving sources is the inherent space-time permutation problem, since acoustic localization techniques are generally unable to identify and produce exactly one measurement for each source. It is then necessary to estimate the trajectory of each source from the measurements, which entails knowing when a source appears or enters the scene, disappears or exits the scene, and how its position changes over each time instance. This is effectively an online tracking problem where the objective is to estimate, at each time instance, the number of sources, their positions and unique labels. Knowledge of the correct source positions and their labels is crucial, as it resolves the inherent space-time permutation problem, thereby enabling the application of a set of time-varying spatial filters to achieve source separation. The underlying signal model and overview of the proposed solution are given below.

3.2.1 Signal Model

We consider a scenario consisting of $N(t)$ point sources where each source is indexed by $n \in \{1, ..., N(t)\}$ with 3D position denoted by $\alpha_n(t) \in \mathbb{R}^3$ at discrete time instance t. Each source signal is denoted by s_n, and all sources are assumed to be mutually uncorrelated, i.e., the cross power spectral density between two sources is zero. An array indexed by $q \in \{1, ..., Q\}$, comprises M_q microphone elements. The source signals impinge on each microphone element $m \in \{1, ..., M_q\}$ of array q, and are corrupted with non-directional diffuse noise $v^{(q,m)}$. The mixture signal at microphone (q,m) is represented by some mapping function ϱ of the source signals $s_1, ..., s_{N(t)}$, source positions $\alpha_1, ..., \alpha_{N(t)}$, and noise $v^{(q,m)}$, evaluated at time t:

$$y^{(q,m)}(t) = \varrho\left(s_1, ..., s_{N(t)}, \alpha_1, ..., \alpha_{N(t)}, v^{(q,m)}\right)(t). \tag{3.1}$$

For stationary sources in an invariant and homogeneous acoustic environment, the mixture signal can be modeled via the sum of the convolutions of source signals and the room impulse response (RIR), which encapsulates the direct path (time-delay) and multipath terms (reflections) between the sources and microphone element (q,m) [361, 362]. However when sources are moving, the effective RIR becomes time-varying. To circumvent this issue, we consider the source signal in blocks:

$$s_n(t) = \sum_{k=1}^{K} s_n(t)\varpi_T(t - (k-1)T) = \sum_{k=1}^{K} s_{k,n}(t), \tag{3.2}$$

where ϖ_T is a window function of length T, and k is the index of a time block/frame with length T. Specifically, we assume source stationarity at each frame k of length T, i.e., $\alpha_n(t) = \alpha_{k,n}$ and $N(t) = N_k$ for $t = (k-1)T, ..., kT$. Thus, the signal is filtered by a new RIR for each time frame:

$$y^{(q,m)}(t) \approx \sum_{k=1}^{K} \sum_{n=1}^{N_k} (s_{k,n} * \varphi_{k,\alpha_{k,n}}^{(q,m)})(t) + v^{(q,m)}(t), \tag{3.3}$$

where $*$ denotes convolution, and $\varphi_{k,\alpha_{k,n}}^{(q,m)}$ denotes the RIR between source n with position $\alpha_{k,n}$ and microphone element (q,m), at frame k. From this representation, each source signal is assumed to be a point source (in a fixed position) in frame k, which is filtered by a linear time-invariant system, where the time invariance is assumed over the frame at length T. For tractability reasons, we focus only on the direct path term and approximate the mixture signal as:

$$y^{(q,m)}(t) \approx \sum_{k=1}^{K} \sum_{n=1}^{N_k} \frac{s_{k,n}\left(t - \tau(\alpha_{k,n}, u^{(q,m)})\right)}{4\pi||\alpha_{k,n} - u^{(q,m)}||} + v^{(q,m)}(t), \tag{3.4}$$

where $||\cdot||$ is the Euclidean distance, $\tau(\alpha_{k,n}, u^{(q,m)}) \triangleq c_s^{-1}||\alpha_{k,n} - u^{(q,m)}||$ is the time delay between source n at position $\alpha_{k,n}$ and microphone (q,m) at position $u^{(q,m)} \in \mathbb{R}^3$ (c_s

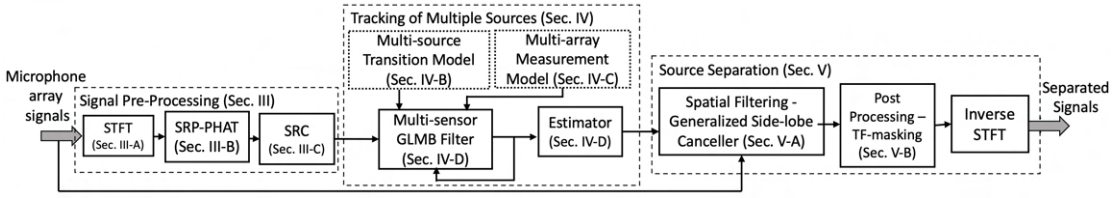


Figure 3.1: Processing Chain of the Proposed Method.

is the speed of sound). Based on this model, the objective is to estimate the individual source signals for every frame k (frame by frame) using only the mixture signals $y^{(1,1)}, ..., y^{(Q,M_Q)}$ and no prior knowledge on the sources.

3.2.2 Overview of the Proposed Method

The processing chain of the proposed method is depicted in Fig. 3.1. Raw microphone signals are segmented into frames and transformed into the frequency domain. Then, acoustic localization techniques that rely on source features such as direction-of-arrivals (DOAs), are used to acquire the source position candidates at each frame. The position candidates from each array are subjected to noise (disturbance), they may not reflect a source that is present (false negative), and some may not correspond to any source (false positive). Above all, there is a space-time permutation problem because the acquired position candidates from each array are unidentified (without labels) across time. As a result, there is no trajectory information on the sources, and spatial filtering cannot be applied for source separation. To remedy this, spatial distributions of the position candidates from all arrays are exploited to jointly estimate the number of sources, their positions and labels for each frame. The estimation of the source labels is important because it resolves the permutation ambiguity. Based on this information, a series of time-varying spatial filter can be constructed using the direct path model for source separation. The proposed method can be broken down into 3 stages: signal pre-processing, multi-source tracking and source separation.

In the first stage, raw microphone signals $y^{(1,1)}, ..., y^{(Q,M_Q)}$ from all arrays are pre-processed into frames of data in the frequency domain using the short-time Fourier transform (STFT). For each frame, we use the Steered-Response Power Phase Transform (SRP-PHAT) [70], and apply a region search algorithm known as Stochastic Region Contraction (SRC) proposed in [70], to obtain 3D position candidates from each array. Due to noise, false positives, false negatives, and the space-time permutation problem, the obtained source position candidates from all arrays are not fit for spatial filtering to achieve source separation.

In the second stage, we employ a Bayesian state estimation framework that processes the obtained position candidates from all arrays, herein referred to as the multi-array measurements, and produces estimates of the source positions and labels at each frame. The tracking filter works by recursively propagating a posterior density which

characterizes the uncertainty of a set of labeled states given all multi-array measurements up to the current time. This framework accounts for noise, false positives and false negatives in the multi-array measurements. Source labels, motions, appearances and disappearances are also incorporated into the formulation. The joint estimation of the source labels and positions resolves the space-time permutation problem.

In the third stage, source separation is achieved via constructing a type of spatial filter known as the Generalized Side-lobe Canceller (GSC) for each frame. The GSC aims to emphasize and separate the source of interest while actively cancelling interfering sources. In order to do this, it is necessary to have the estimated source positions and the labels at each frame, which is provided by the proposed tracking solution. In addition, we utilize the GSC signals to construct a time-frequency mask for enhancing the separated signals. Finally, the time-domain separated signals are recovered using the inverse STFT.

3.3 Signal Pre-processing

This section describes the segmentation of raw signals into frames of data using the short-time Fourier Transform (STFT), followed by the use of Steered-Response Power Phase Transform (SRP-PHAT) combined with Stochastic Region Contraction (SRC) to obtain the 3D source position candidates. The shortcomings of the obtained position candidates are outlined and discussed.

3.3.1 Short-Time Fourier Transform (STFT)

Each raw microphone signal $y^{(q,m)}$ is segmented into $y_1^{(q,m)}, ..., y_K^{(q,m)}$ via:

$$y_k^{(q,m)}(t) = y^{(q,m)}(t + (k-1)T)\varpi_T(t), \tag{3.5}$$

where ϖ_T is a selected window function of length T. The window function is chosen such that it captures enough information while reducing signal discontinuities at the edges, e.g., a Hann window $\varpi_T(t) = 0.5 - 0.5\cos(2\pi t/T),\ t = 0, \ldots, T-1$. We denote the discrete short-time Fourier transform of $y_k^{(q,m)}(t)$ by $Y_k^{(q,m)}(\lambda)$ where λ is the frequency bin index. To represent the segmented frequency-domain raw signals from all microphones at array q in a compact form, we stack them into a vector:

$$Y_k^{(q)}(\lambda) = \left[Y_k^{(q,i)}(\lambda)\right]_{i=1}^{M_q} \tag{3.6}$$

3.3.2 Steered-Response Power-Phase Transform

Steered-Response Power Phase Transform (SRP-PHAT) is an acoustic source localization solution well known for its robust performance in adverse acoustic environments

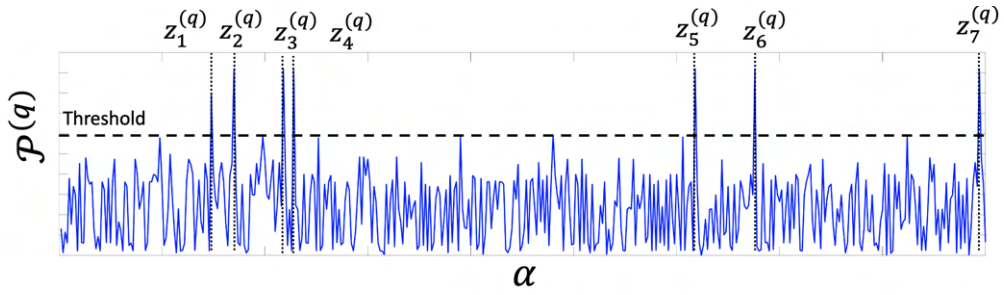


Figure 3.2: SRP-PHAT Measurements

[363]. The SRP is the output power of a delay-and-sum beamformer that is steered to a set of source positions which are defined under a specified spatial grid [70]. The Phase Transform (PHAT) is a weighting technique to avoid peak spreading in the SRP by emphasizing the phase information of the involved signals [70].

Given $Y_k^{(q)}$ received at array q, the spatial power that emanates from the direction of the source location $\alpha_k \in \mathbb{R}^3$ at each frame k, is computed using Steered-Response Power (SRP) with PHAT by [70]:

$$\mathcal{P}_k^{(q)}(\alpha) = \sum_{a=1}^{M_q-1} \sum_{b=a+1}^{M_q} \sum_{\lambda} \frac{Y_k^{(q,a)}(\lambda) Y_k^{*(q,b)}(\lambda)}{\left| Y_k^{(q,a)}(\lambda) Y_k^{*(q,b)}(\lambda) \right|} e^{j\omega_\lambda \left(\tau(\alpha, u^{(q,b)}) - \tau(\alpha, u^{(q,a)})\right)}, \tag{3.7}$$

where $\omega_\lambda = 2\pi(\lambda - 1)F_s/T$ (F_s is the sampling frequency), the PHAT weighting is the inverse magnitude of the frequency components of the involved signals, and the exponential term is responsible for time-aligning the microphone signals based on time-difference-of-arrival. Searching for multiple local maxima of (3.7) at any frame k corresponds to source position candidates that are present at that time frame. However, this process is computationally expensive as it involves a large search space.

3.3.3 Stochastic Region Contraction (SRC)

Using the computationally efficient SRC algorithm [70], the 3D source position candidates are obtained via peak-picking SRP-PHAT for every array with a certain threshold. For each array q, we denote the collection of the position candidates as a measurement set:

$$Z_k^{(q)} = \{z_{k,1}^{(q)}, ..., z_{k,|Z_k^{(q)}|}^{(q)}\}, \tag{3.8}$$

where $|Z_k^{(q)}|$ denotes the number of measurements (see Fig. 3.2). For multiple arrays, we define $Z_k \triangleq (Z_k^{(1)}, ..., Z_k^{(Q)})$ as the multi-array measurements. The multi-array measurements are utilized to deduce the optimal positions of the sources. However, due to nonlinearity, noise and reverberation (in real-world conditions), the multi-array measurements have the following issues:

- A measurement $z_k^{(q)}$ obtained from a single array (if it is generated by a source) is

noisy after undergoing a highly nonlinear transformation.

- The multi-array measurements contain false positives, which are measurements not generated by any active source; and false negatives, which are missing measurements even when sources are active.

- Furthermore, we are faced with the inherent space-time permutation problem as the multi-array measurements are unordered and have no identities/labels. Specifically, in space, it is not known which individual measurement in the sets $Z_k^{(1)},...,Z_k^{(Q)}$ is generated by which source. In time, it is not known how an individual measurement from the sets $Z_k^{(1)},...,Z_k^{(Q)}$ at the current frame, to the sets $Z_{k+1}^{(1)},...,Z_{k+1}^{(Q)}$ at the next frame, is connected with respect to an existing source. Also, the appearance of a new active source or the disappearance of an existing active source is unknown.

3.4 Tracking of Multiple Sources

This section presents a labeled RFS solution for estimating the source trajectories from the source measurements thereby addressing the space-time permutation problem. The solution entails the recursive multi-source Bayes filter, which requires specification of the multi-source transition and multi-array likelihood models. A tractable implementation is given in the form of the Multi-Sensor Generalized Labeled Multi-Bernoulli filter. These are summarized as follows.

3.4.1 Multi-Source Bayesian Tracking Filter

Given the multi-array measurements $Z_k \triangleq (Z_k^{(1)},...,Z_k^{(Q)})$, the objective is to estimate the number of the sources, their positions and labels at each frame k. In order to do so, it is necessary to have a stochastic model to characterize the time-varying nature of the number of sources and the individual source positions, which arises due to source appearance, disappearance and physical motion. Similarly, it is necessary to have a stochastic model to characterize the multi-array measurements as the number of measurements for each array is also time-varying, partly because the number of sources is time-varying, but also because the measurements are subjected to noise, false negatives and false positives.

A random finite set (RFS) is a natural representation for the collection of source positions (with labels), and for each of the array measurements, because an RFS is essentially a set-valued random variable, wherein the number of points as well as the values of individual points are random [19, 72, 122]. In order to develop an online solution for estimating the number of sources, their positions and labels based on RFS

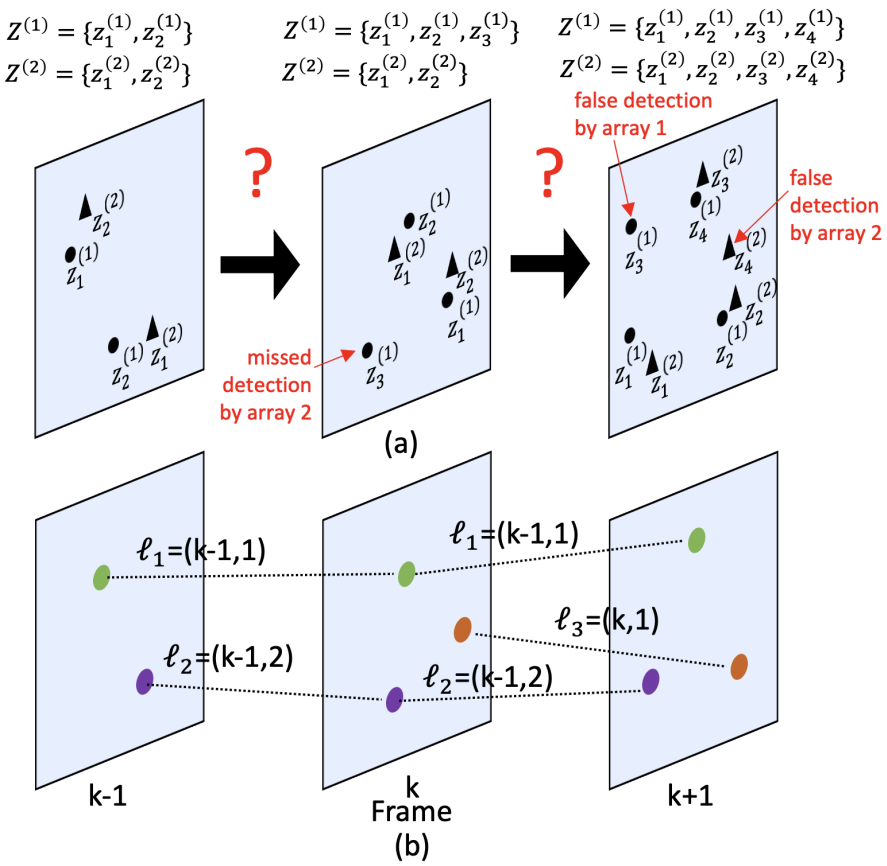


Figure 3.3: Two sources appear at frame $k-1$ and persist until frame $k+1$, while a third source appears at k and persists until $k+1$. (a) Illustration of measurements from two arrays, $Z^{(1)}$ and $Z^{(2)}$ (time subscript k is suppressed). (b) Illustration of desired tracking result to resolve the space-time permutation problem.

modeling for each frame, we cast the problem into a recursive Bayesian estimation framework.

In the context of this framework, source appearances and disappearances are referred to as source births and deaths respectively, while false negatives and false positives are referred to as missed detections and false detections respectively. Recall that the time permutation problem arises due to source motions, appearances and disappearances, while the space permutation problem arises due to the absence of labels in the array measurements, which are also subjected to noise, missed and false detections. The space-time permutation problem is referred to as the data association problem and can be addressed using the RFS tracking framework. Fig. 3.3 gives an illustration of the array measurements prior to tracking as well as the desired result after tracking is applied.

Each source at frame k has a state denoted by $\boldsymbol{x}_k \triangleq (x_k, \ell)$, where $x_k \triangleq (\alpha_k, \dot{\alpha}_k)$ is a vector capturing the 3D position and velocity of the source, and ℓ is a unique label from a discrete space $\mathbb{L}$. Note that the velocity component is an auxiliary variable needed for the state transition model in the Bayesian framework. The states of multiple sources at

each frame k, are represented as a finite set:

$$\boldsymbol{X}_k = \{\boldsymbol{x}_{k,1}, ..., \boldsymbol{x}_{k,N_k}\}, \tag{3.9}$$

herein referred to as a multi-source state. Note that the existence of unique labels in the multi-source state means that consecutive states with the same label across frames constitute the trajectory of the source movement (see Fig. 3.3 (b)).

The RFS representation of $\boldsymbol{X}_k$ naturally accounts for the movements of active sources, births of new sources and deaths of existing sources, while the RFS representation of the sets $Z_k^{(1)}, ..., Z_k^{(Q)}$ naturally accounts for, noise, missed detections, and false detections in the measurements across all arrays. In Bayesian RFS tracking, the aim is to estimate frame-by-frame (recursively) the multi-source state $\boldsymbol{X}_k$, given the multi-array measurements obtained from the beginning of time up to the current time frame k, i.e., $Z_{1:k} \triangleq (Z_1, ..., Z_k)$. The solution is the *multi-object Bayes filter*, which is a recursive mechanism that computes the probability density of $\boldsymbol{X}_k$ given $Z_{1:k}$ [19]. In the context of Bayesian filtering, this probability density is referred as the filtering density denoted by $\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})$. At any given frame k, all uncertainty in the multi-source state $\boldsymbol{X}_k$ given $Z_{1:k}$, is captured in $\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})$ [19].

The propagation of the filtering density is a recursive two-step procedure. The first step is the time update of the current filtering density $\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})$ via [19]:

$$\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k}) = \int \boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k)\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})\delta\boldsymbol{X}_k, \tag{3.10}$$

where the integral is not the usual Euclidean notion of integration, rather it is a set integral defined under Finite Set Statistics (FISST) for dealing with RFSs in a mathematically consistent manner [237], and $\boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k)$ is known as the *multi-source transition density* which gives the probability density that multi-source state $\boldsymbol{X}_k$ at frame k transitions to $\boldsymbol{X}_{k+1}$ at the next frame $k+1$. The *multi-source transition density* is formulated based on a stochastic model that encapsulates all possible source births, deaths and motions, i.e., the time permutation aspect. The details of this transition model are further discussed in Section 3.4.2. Consequently, the time-updated density (3.10) characterizes the transition of $\boldsymbol{X}_k$ to $\boldsymbol{X}_{k+1}$, given all multi-array measurements $Z_{1:k}$ up to the current time frame, and addresses the time permutation part of the data association problem. The second step is the data update of $\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k})$ with the multi-array measurements Z_{k+1} obtained at frame $k+1$ via [19]:

$$\boldsymbol{\pi}_{k+1|k+1}(\boldsymbol{X}_{k+1}|Z_{1:k+1}) = \frac{g(Z_{k+1}|\boldsymbol{X}_{k+1})\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k})}{\int g(Z_{k+1}|\boldsymbol{X}_{k+1})\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k})\delta\boldsymbol{X}_{k+1}}, \tag{3.11}$$

where $g(Z_{k+1}|\boldsymbol{X}_{k+1})$ is known as the *multi-array measurement likelihood* which gives the probability density of the multi-array measurements Z_{k+1}, given the multi-source state $\boldsymbol{X}_{k+1}$. The *multi-array measurement likelihood* is formulated based on a

Table 3.1: Parameters for the *Multi-Source Transition Density* (3.12)

Probability of survival	P_S
Single-source transition density	$f_S(\cdot\|\cdot)$
Probability of birth	$r_B(\cdot)$
Birth density	$f_B(\cdot)$

stochastic model that encapsulates noise, detections, missed detections, false detections and association uncertainty, i.e., the space permutation aspect, in the obtained multi-array measurements. The details of this multi-array measurement model are given in Section 3.4.3. The data-updated density (3.11) contains all information about the number of sources and their states (with labels) at the next time frame $k+1$, conditioned on the multi-array measurements up to that frame. This step consequently addresses the space permutation part of the data association problem.

In summary, the combination of both time-update and data-update steps in the propagation of the filtering density solves the space-time permutation problem. To obtain a multi-source state estimate at each frame, which contains the estimated number of sources, their positions and labels, a conventional Bayesian multi-source estimator is applied to the filtering density at each frame. The closed-form representation of the filtering density and the implementation of the filter, i.e., the tractable (recursive) propagation of the filtering density, are discussed in Section 3.4.4.

3.4.2 The Multi-Source Transition Model

The function $\boldsymbol{f}(\cdot|\cdot)$ is a probability density function characterizing all possible source births, deaths and motions that take place in the transition of a multi-source state from one frame to the next [19]. The function $\boldsymbol{f}(\cdot|\cdot)$ is parameterized as per Table 3.1, and explanations of these parameters are given as follows.

Given the multi-source state $\boldsymbol{X}_k$, each state $\boldsymbol{x}_k \triangleq (x_k, \ell_k) \in \boldsymbol{X}_k$ either survives with probability P_S and transition to a new state (x_{k+1}, ℓ_{k+1}) that inherits the same label whose uncertainty is captured by the transition density $f_S(x_{k+1}|x_k, \ell_k)\delta_{\ell_k}[\ell_{k+1}]$, or dies with probability $1-P_S$. At this next time, a set of new sources denoted by $\boldsymbol{B}_{k+1}$ with labels $\{\ell_{k+1} : (x_{k+1}, \ell_{k+1}) \in \boldsymbol{B}_{k+1}\}$ can be born or appear individually with probability $r_B(\ell_{k+1})$ and distributed according to the birth density $f_B(\cdot, \ell_{k+1})$. Recall that labels of a multi-source state are distinct/unique for all frames, hence a label is defined as $\ell_k = (\varsigma, \iota) \in \mathbb{L}_k$, where $\varsigma \in \{k\}$ denotes the time frame of birth and $\iota \in \mathbb{N}$ denotes the index of source born at the same time [19] (see Fig. 3.3 (b) for illustration). Consequently, the label space for all sources up to time k (including those born prior to k) is the disjoint union $\mathbb{L}_k = \biguplus_{i=0}^{k} \mathbb{B}_i$, where $\mathbb{B}_i$ denotes the label space for sources born at time i, (note that $\mathbb{L}_k = \mathbb{L}_{k-1} \uplus \mathbb{B}_k$).

The multi-source state $\boldsymbol{X}_{k+1}$ is the superposition of the surviving sources $\boldsymbol{W}_{k+1}$ and

Table 3.2: Parameters for the Multi-Array Measurement Likelihood (3.13)

Probability of detection	$P_D^{(1)}, \ldots, P_D^{(Q)}$
Single-source likelihood	$g^{(1)}(\cdot\|\cdot), \ldots, g^{(Q)}(\cdot\|\cdot)$
False detection intensity	$\kappa^{(1)}(\cdot), \ldots, \kappa^{(Q)}(\cdot)$

the new born sources $\boldsymbol{B}_{k+1}$, which are assumed to be statistically independent. Let $\boldsymbol{f}_S(\boldsymbol{W}_{k+1}|\boldsymbol{X}_k)$ and $\boldsymbol{f}_B(\boldsymbol{B}_{k+1})$ be the probability densities of the survivability of $\boldsymbol{X}_k$ to $\boldsymbol{W}_{k+1}$, and the new born sources $\boldsymbol{B}_{k+1}$ respectively, then the *multi-source transition density* is given by [19]:

$$\boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k) = \boldsymbol{f}_S(\boldsymbol{W}_{k+1}|\boldsymbol{X}_k)\boldsymbol{f}_B(\boldsymbol{B}_{k+1}). \tag{3.12}$$

The product in (3.12) presents a model for addressing the time permutation problem. In particular, source appearance, disappearance and motion are considered to be statistically independent. However, labels are kept the same for sources that move and continue to be active, and appearing active sources are assigned a new distinct label, while deactivated sources are removed. The derivation of (3.12) is beyond the scope of this dissertation, but interested readers are referred to [19].

3.4.3 The Multi-Array Measurement Likelihood Model

The function $g(\cdot|\cdot)$ is a probability density function characterizing noise, missed detections, false detections and association uncertainty in the multi-array measurements. The function $g(\cdot|\cdot)$ is parameterized as per Table 3.2, and explanations of these parameters are given as follows.

Given the multi-source state $\boldsymbol{X}_k$, each $\boldsymbol{x}_k = (x_k, \ell_k) \in \boldsymbol{X}_k$ is either detected at array q with probability $P_D^{(q)}$ and generates a detection $z_k^{(q)} \in Z_k^{(q)}$ with likelihood $g^{(q)}(z_k^{(q)}|x_k, \ell_k)$, or missed detected with probability $1 - P_D^{(q)}$. The detection process also generates false detections at array q, conventionally characterized by an intensity function $\kappa^{(q)}(\cdot) \triangleq \lambda_p^{(q)}\mathcal{U}(\cdot)$ on the measurement space [19, 122]. The number of false detections is modeled by a Poisson distribution with mean $\lambda_p^{(q)}$, and the false detections themselves are uniformly distributed in the measurement space according to $\mathcal{U}(\cdot)$. In standard multi-source tracking, it is standard to assume that the detections are statistically independent from the false detections [19].

A single-array association $\theta_k^{(q)} \in \Theta_k^{(q)}$ is defined as a mapping from the source labels to the measurement indexes, i.e., $\theta_k^{(q)} : \{\ell_k : (x_k, \ell_k) \in \boldsymbol{X}_k\} \to \{0 : |Z_k^{(q)}|\}$. Note that $\Theta_k^{(q)}$ is the space of all mappings, such that *no two distinct arguments are mapped to the same positive value* [19]. This property ensures each detection comes from at most one source. For example, $\theta_k^{(q)}(\ell_k) > 0$ corresponds to source ℓ_k generating detection $z^{(q)}_{k,\theta_k^{(q)}(\ell_k)}$ at array q, while $\theta_k^{(q)}(\ell_k) = 0$ means source ℓ_k is misdetected at array q. For

multiple arrays, a multi-array association is the vector $\theta_k \triangleq (\theta_k^{(1)},...,\theta_k^{(Q)}) \in \Theta_k$ of all single-array associations having the same aforementioned positive one-to-one property, where $\Theta_k \triangleq \Theta_k^{(1)} \times ... \times \Theta_k^{(Q)}$ is the space of all possible multi-array associations [22].

Under the assumption that the set $Z_k^{(q)}$ at array q is conditionally independent from those at other arrays, the *multi-array measurement likelihood* is given by [22]:

$$g(Z_k|\boldsymbol{X}_k) \propto \sum_{\theta_k^{(1)} \in \Theta_k^{(1)}} ,..., \sum_{\theta_k^{(Q)} \in \Theta_k^{(Q)}} \prod_{\substack{(x_k,\ell_k) \\ \in \boldsymbol{X}_k}} \prod_{q=1}^{Q} \psi_{Z_k^{(q)}}^{(q,\theta_k^{(q)}(\ell_k))}(x_k,\ell_k), \tag{3.13}$$

where

$$\psi_{Z_k^{(q)}}^{(q,j)}(x_k,\ell_k) = \begin{cases} \dfrac{P_D^{(q)} g^{(q)}\left(z_{k,j}^{(q)}|x_k,\ell_k\right)}{\kappa^{(q)}\left(z_{k,j}^{(q)}\right)}, & j>0 \\ 1-P_D^{(q)}, & j=0 \end{cases}. \tag{3.14}$$

It is important to note that the nested sum in (3.13) indicates the enumeration of all possible multi-array associations, thereby taking into account all possible combinations of missed detections, false detections and the source detections. In other words, the nested sum in (3.13) presents a model for addressing the space permutation problem by considering all possible mappings of position candidates to source labels. The derivation for (3.13) is beyond the scope of this dissertation, but interested readers are referred to [19, 22, 122].

3.4.4 The Multi-Sensor Generalized Labeled Multi-Bernoulli

Under the transition and measurement models as described above, the time-updated and data-updated (filtering) densities admit a closed-form solution via the Generalized Labeled Multi-Bernoulli (GLMB) density [19, 20, 22]:

$$\boldsymbol{\pi}(\boldsymbol{X}_k) = \Delta(\boldsymbol{X}_k) \sum_{\theta_{1:k} \in \Theta_{1:k}} \omega^{(\theta_{1:k})}(\mathcal{L}(\boldsymbol{X}_k)) \prod_{\boldsymbol{x}_k \in \boldsymbol{X}_k} p^{(\theta_{1:k})}(\boldsymbol{x}_k), \tag{3.15}$$

where $\mathcal{L}(\boldsymbol{X}_k) \triangleq \{\ell_k : (x_k,\ell_k) \in \boldsymbol{X}_k\}$, $\Delta(\cdot)$ is a distinct label indicator, i.e., $\Delta(\boldsymbol{X}_k) = 1$ if and only if the cardinality $|\mathcal{L}(\boldsymbol{X}_k)| = |\boldsymbol{X}_k|$, $\theta_{1:k} \in \Theta_{1:k}$ is the history of multi-array association mappings up to frame k, i.e., $\theta_{1:k} \triangleq (\theta_1,...,\theta_k)$. Each $\omega^{(\theta_{1:k})}(\mathcal{L}(\boldsymbol{X}_k))$ is a non-negative weight such that

$$\sum_{L \subseteq \mathbb{L}_{0:k}} \sum_{\theta_{1:k} \in \Theta_{1:k}} \omega^{(\theta_{1:k})}(L) = 1, \tag{3.16}$$

and can be interpreted as the probability of sources with label set $\mathcal{L}(\boldsymbol{X}_k)$ being active, as well as being associated with the detections given by the association history $\theta_{1:k}$. Each $p^{(\theta_{1:k})}(\cdot,\ell_k)$ is a probability density of the source state with label ℓ_k and association history $\theta_{1:k}$, where $p^{(\theta_{1:k})}(x_k,\ell_k)$ is the probability density of the source with label ℓ_k being located at state $x_k = (\alpha_k, \dot{\alpha}_k)$.

In plain terms, the GLMB (3.15) can be interpreted as a mixture model, i.e., a

weighted sum of the products of single-source probability densities, where each weight is a function of the labels in the multi-source state. In the context of Bayesian filtering, the GLMB density (3.15) at time k can be rewritten as [22]:

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{I,\xi} \omega^{(I,\xi)} \delta_I[\mathcal{L}(\boldsymbol{X})] \prod_{\boldsymbol{x}\in\boldsymbol{X}} p^{(\xi)}(\boldsymbol{x}), \tag{3.17}$$

where $I \in \mathcal{F}(\mathbb{L})$ and each $\xi \in \Xi$ represents a history of (multi-sensor) association maps, i.e., $\xi = (\theta_{1:k})$. For notational compactness, the time subscript k in the expression (3.17) has been suppressed. It follows from [22] that under the standard multi-source transition and measurement models as described in Section 3.4.2 and Section 3.4.3, respectively, a GLMB filtering density (3.17) at time k can be propagated forward to $k+1$ via a joint prediction and update given by:

$$\boldsymbol{\pi}_+(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{I,\xi,I_+,\theta_+} \omega^{(I,\xi)} \omega_{Z_+}^{(I,\xi,I_+,\theta_+)} \delta_{I_+}[\mathcal{L}(\boldsymbol{X})] \prod_{\boldsymbol{x}\in\boldsymbol{X}} p_{Z_+}^{(\xi,\theta_+)}(\boldsymbol{x}), \tag{3.18}$$

where the time subscript '+' is used to indicate the next time step, $I \in \mathcal{F}(\mathbb{L})$, $\xi \in \Xi$, $I_+ \in \mathcal{F}(\mathbb{L}_+)$, $\theta_+ \in \Theta_+(I_+)$, and

$$\begin{aligned}
\omega_{Z_+}^{(I,\xi,I_+,\theta_+)} &= 1_{\Theta_+(I_+)}(\theta_+) \left[1 - \bar{P}_S^{(\xi)}\right]^{I-I_+} \left[\bar{P}_S^{(\xi)}\right]^{I\cap I_+} \\
&\quad \times \left[1 - r_{B,+}\right]^{\mathbb{B}_+ - I_+} r_{B,+}^{\mathbb{B}_+\cap I_+} \left[\bar{\psi}_{Z_+}^{(\xi,\theta_+))}\right]^{I_+}, \qquad (3.19) \\
\bar{P}_S^{(\xi)}(\ell) &= \left\langle p^{(\xi)}(\cdot,\ell), P_S(\cdot,\ell) \right\rangle, \qquad (3.20) \\
\bar{\psi}_{Z_+}^{(\xi,\theta_+)}(\ell_+) &= \left\langle \bar{p}_+^{(\xi)}(\cdot,\ell_+), \psi_{Z_+}^{(\theta_+(\ell_+))}(\cdot,\ell_+) \right\rangle, \qquad (3.21) \\
\psi_{Z_+}^{(\theta_+(\ell_+))}(x_+,\ell_+) &= \prod_{q=1}^{Q} \psi_{Z_+^{(q)}}^{(q,\theta_+^{(q)}(\ell_+))}(x_+,\ell_+), \qquad (3.22) \\
\bar{p}_+^{(\xi)}(x_+,\ell_+) &= 1_{\mathbb{L}}(\ell_+) \frac{\left\langle P_S(\cdot,\ell_+) f_{S,+}(x_+|\cdot,\ell_+), p^{(\xi)}(\cdot,\ell_+) \right\rangle}{\bar{P}_S^{(\xi)}(\ell_+)} \\
&\quad + 1_{\mathbb{B}_+}(\ell_+) f_{B,+}(x_+,\ell_+), \qquad (3.23) \\
p_{Z_+}^{(\xi,\theta_+)}(x_+,\ell_+) &= \frac{\bar{p}_+^{(\xi)}(x_+,\ell_+) \psi_{Z_+}^{(\theta_+(\ell_+))}(x_+,\ell_+)}{\bar{\psi}_{Z_+}^{(\xi,\theta_+)}(\ell_+)}. \qquad (3.24)
\end{aligned}$$

From an implementation standpoint, the number of terms in the GLMB filtering density grows exponentially over time, partly due to the enumeration of all possible multi-array associations at each time frame. To maintain tractability, pruning of the terms with low weights is required, and has been shown to minimize the L_1 approximation error [22]. The Multi-Sensor GLMB (MS-GLMB) filter offers a polynomial time implementation mechanism that generates highly weighted components without exhaustive enumeration of the sum in (3.15), which has a linear complexity in the sum

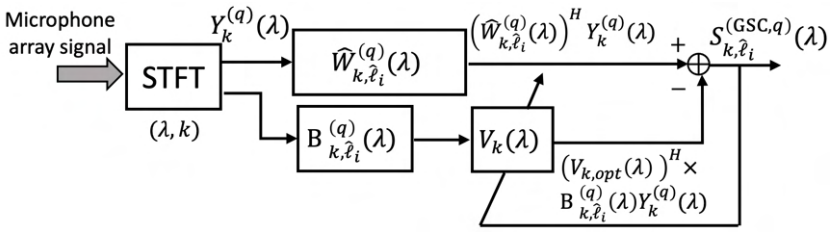


Figure 3.4: Spatial Filtering via Generalized Side-lobe Canceler (GSC)

of the total number of measurements across all arrays [22]. A multi-source state estimate can be obtained from the GLMB posterior density via a simple GLMB estimator [20, 22]. Since we only require the position component of the single-source state, the estimated multi-source state $\hat{\boldsymbol{X}}_k$ at frame k is:

$$\hat{\boldsymbol{X}}_k = \{(\hat{\alpha}_{k,1}, \hat{\ell}_1), \ldots, (\hat{\alpha}_{k,|\hat{\boldsymbol{X}}_k|}, \hat{\ell}_{|\hat{\boldsymbol{X}}_k|})\}, \tag{3.25}$$

where $\hat{N}_k = |\hat{\boldsymbol{X}}_k|$ is the estimated number of sources.

3.5 Source Separation

This section describes the use of the multi-source state estimate from the MS-GLMB filter to construct a Generalized Side-lobe Canceler (GSC) for source separation. For post-processing, we adopt a time-frequency masking step to further suppress interfering speech.

3.5.1 Spatial Filtering

At each frame k, the tracking filter provides the multi-source state estimate $\hat{\boldsymbol{X}}_k$, which contains the estimated source positions and labels from the available data. The combination of source positions and labels constitutes the estimated source tracks, thereby solving the space-time permutation problem that arises from the multi-array measurements as depicted in Fig. 3.3. With this information, we design a set of spatial filters that is changing at each frame depending on $\hat{\boldsymbol{X}}_k$, based on a free space near-field room model. We adopt a variant of the linearly constrained minimum variance beamformer called the Generalized Side-lobe Canceler (GSC). A GSC is a constrained beamformer that has been converted to a non-constrained design by means of a blocking matrix [60]. The GSC contains two parts: a beamformer that determines the response of the source of interest (SOI), and a mechanism that blocks the SOI from entering the canceler. Fig. 3.4 shows a block diagram of the GSC.

In the first part, we use a beamformer that emphasizes the direction of the SOI specified by label $\hat{\ell}_i$ with position $\hat{\alpha}_{k,i}$, while nulling other interfering sources specified by $\{(\hat{\alpha}_{k,j}, \hat{\ell}_j) \in \hat{\boldsymbol{X}}_k\}_{j=1}^{\hat{N}_k}$ for $i \neq j$. For each TF point (λ, k), the weight of the beamformer

$\hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)$ is given by:

$$\left(\mathrm{D}^{(q)}_{k,\hat{\boldsymbol{X}}_k}(\lambda)\right)^H \hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda) = l_{\hat{N}_k}(\hat{\ell}_i)$$
$$\hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda) = \left(\left(\mathrm{D}^{(q)}_{k,\hat{\boldsymbol{X}}_k}(\lambda)\right)^H\right)^{\dagger} l_{\hat{N}_k}(\hat{\ell}_i), \tag{3.26}$$

where the operator ^{H} is the Hermitian transpose, the dagger † denotes the Moore-Penrose pseudo-inverse, $l_{\hat{N}_k}$ is a selection vector whose dimension varies depending on the estimated number of sources $\hat{N}_k$, i.e., $l_{\hat{N}_k}(\hat{\ell}_i) = [\delta_{\hat{\ell}_1}[\hat{\ell}_i], \ldots, \delta_{\hat{\ell}_{\hat{N}_k}}[\hat{\ell}_i]]^T$ such that $\delta_i[j] = 1$ if $i = j$ and zero otherwise, and

$$\mathrm{D}^{(q)}_{k,\hat{\boldsymbol{X}}_k}(\lambda) = \begin{bmatrix} e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,1},u^{(q,1)})\right)} & \ldots & e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,\hat{N}_k},u^{(q,1)})\right)} \\ \vdots & \ddots & \vdots \\ e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,1},u^{(q,M_q)})\right)} & \ldots & e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,\hat{N}_k},u^{(q,M_q)})\right)} \end{bmatrix}, \tag{3.27}$$

is a matrix with columns representing the steering vectors for each estimated source. The number of columns depends on the estimated number of sources $\hat{N}_k$. Note that if $\hat{N}_k = 1$, (3.26) reduces to the classical delay-and-sum beamformer.

The second part involves a blocking matrix that is defined to be the orthogonal complement to $\left(\hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)\right)^H$ [60]:

$$\mathrm{B}^{(q)}_{k,\hat{\ell}_i}(\lambda) = \mathrm{I} - \hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)\left[\left(\hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)\right)^H \hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)\right]^{-1}\left(\hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda)\right)^H, \tag{3.28}$$

where I is an identity matrix. Subsequently, the weight vector of the GSC is defined by:

$$G^{(q)}_{k,\hat{\ell}_i}(\lambda) = \hat{W}^{(q)}_{k,\hat{\ell}_i}(\lambda) - \mathrm{B}^{(q)}_{k,\hat{\ell}_i}(\lambda) V_k(\lambda), \tag{3.29}$$

where

$$V_{k,opt}(\lambda) = \arg\min_{V} \sum_{\eta=1}^{k} \chi^{k-\eta} \left| \left(\hat{W}^{(q)}_{\eta,\hat{\ell}_i}(\lambda) - \mathrm{B}^{(q)}_{\eta,\hat{\ell}_i}(\lambda) V\right)^H Y^{(q)}_{\eta}(\lambda) \right|^2, \tag{3.30}$$

$\chi \in [0, 1]$ is a positive constant. Eq. (3.30) can be solved recursively using recursive least squares [364].

The output of the GSC for estimated source label $\hat{\ell}_i$ at each TF point (λ, k) and array q is given by:

$$S^{(\mathrm{GSC},q)}_{k,\hat{\ell}_i}(\lambda) = \left(G^{(q)}_{k,\hat{\ell}_i}(\lambda)\right)^H Y^{(q)}_{k}(\lambda). \tag{3.31}$$

3.5.2 Post-processing: Time-Frequency Masking

To improve the quality of the separated source signals, we exploit the spatial-spectral content of the GSC signals to construct a time-frequency (TF) mask following the approach in [365]. The construction of the TF mask relies on the assumption that the power spectrum of $S^{(\mathrm{GSC},q)}_{k,\hat{\ell}_i}(\lambda)$ is dominated by its corresponding source $\hat{\ell}_i$. For each source $\hat{\ell}_i$, a TF binary mask $\mathcal{M}^{(q)}_{k,\hat{\ell}_i}$ is constructed by comparing the relative power of the SOI to each of the interfering sources, with the intention of suppressing the interference. The estimated source is given by $\hat{S}^{(q)}_{k,\hat{\ell}_i}(\lambda) = \mathcal{M}_{k,\hat{\ell}_i}(\lambda) \cdot S^{(\mathrm{GSC},q)}_{k,\hat{\ell}_i}(\lambda)$, and the time-domain signal $\hat{s}^{(q)}_{\hat{\ell}_i}$ is given by the inverse STFT. In separating the source, we simply select the closest array to the estimated source position at each frame.

3.6 Experiments

In this section, we present the evaluations of the obtained multi-array measurements, the tracking filter performance, and the source separation performance on real data recorded in a physical room. Based on the same setting, we go further in evaluating the tracking and separation performance on simulated data with different reverberation times. The experimental setup is summarized in Section 3.6.1. The parameters used for the proposed method are explained in Section 3.6.2. Subsequently, we evaluate the quality of the SRP-PHAT multi-array measurements in Section 3.6.3, followed by the tracking performance of the multi-source Bayesian filter in Section 3.6.4, and the separation performance in Section 3.6.5.

3.6.1 Experimental Setup

The experiment is conducted in a $7.67\mathrm{m} \times 3.41\mathrm{m} \times 2.7\mathrm{m}$ enclosed room with reverberation measured at $T_{60} \approx 0.25\mathrm{s}$ using 4 linear arrays of 6 microphones (total of 24 mics), where all microphones are calibrated to the same gain/sensitivity. These microphones are connected into 3 *RME-OctaMic 8-channel* pre-amps. Each pre-amp is daisy-chained via MADI cables into the computer. All 4 microphone arrays are placed at the sides of the room as shown in Fig. 3.5.

As our proposed method is capable of handling an unknown number of moving sources, we design the experiment such that an active source (female speech) first appears in the scene and starts moving, followed later by another 2 active sources (male and female speech). It is also important to point out that the times at which these sources appear and disappear from the scene are unknown. The movement of each individual source is annotated by hand and the trajectories of the sources are illustrated in Fig. 3.5. In recording the source signals, we traverse each source according on the indicated path so that we can evaluate the tracking results. Note that the sources are continuously

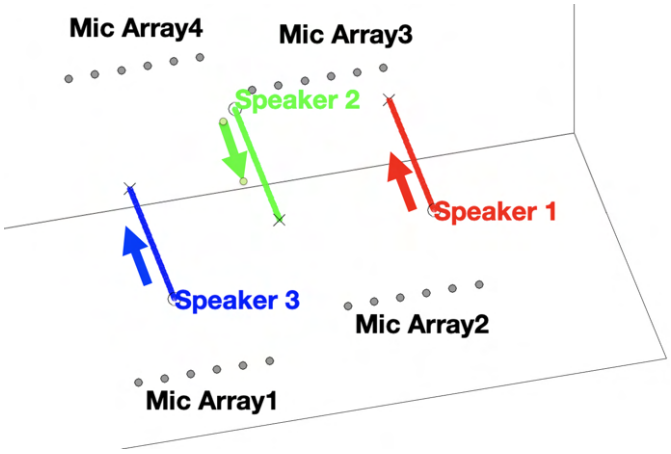


Figure 3.5: Experimental Room Setup

active with typical short pauses in speech.

To evaluate the performance of the proposed method with different reverberation times, i.e., $T_{60} = 0.05\text{s}, 0.25\text{s}, 0.55\text{s}$, we use the Image Source Model (ISM) [361, 362] to simulate the acoustic room response for these reverberation times. The movements of the sources are the same as the annotated (ground-truth) trajectories in Fig. 3.5, and the source signals are convolved with simulated room impulse responses using a 512-sample frame length.

3.6.2 Parameters Breakdown

Multi-Array Measurements

The microphone signals are sampled at F_s=16kHz and subjected to high-pass filtering with 1kHz cutoff to minimize the impact of reverberation on the multi-array measurements. The STFT of the raw signals is performed with a Hann window of frame length T=512, where each frame increment corresponds to a 32ms time frame. The multi-array position measurements are obtained via peak-picking with an empirically selected threshold.

Multi-Source Bayesian Tracking Filter

Recall that the parameters of the *multi-source transition density* are shown in Table 3.1 of Section 3.4.2. In audio speaker tracking where speech typically has short pauses, the Langevin model [69, 72, 234] is an appropriate choice for acoustic speaker tracking [71]. The motion model has the following state space equations [234]: $\alpha_{k+1} = \alpha_k + \phi\dot{\alpha}_k$, $\dot{\alpha}_{k+1} = e^{-\beta\phi}\dot{\alpha}_k + \nu\sqrt{1-e^{-2\beta\phi}}\upsilon_k$, where α_k and $\dot{\alpha}_k$ are the 3D position and velocity vectors respectively, β is the rate constant that controls the rate at which the velocity decays, ν is the steady-state root-mean-square velocity constant, ϕ is the discretization time step interval and υ_k is the process noise. The process noise υ_k models random disturbances in the state transition, and υ_k is a 3-dimensional Gaussian random vector

with zero mean and covariance $\sigma_\upsilon \sigma_\upsilon^T$, where σ_υ is a column vector of the component standard deviations. Note that each component of υ_k is a Gaussian random variable that is statistically independent of one another and across time.

Based on this motion model, we specify the single-source state transition density as $f_S(x_{k+1}|x_k) = \mathcal{N}(x_{k+1}; \mathrm{F}x_k, \mathrm{R}\mathrm{R}^T)$, where $x_k \triangleq (\alpha_k, \dot{\alpha}_k)$, $\mathcal{N}(\cdot\,; \mathrm{F}x_k, \mathrm{R}\mathrm{R}^T)$ is a Gaussian *pdf* with mean $\mathrm{F}x_k$ and covariance $\mathrm{R}\mathrm{R}^T$, $\mathrm{F} = \mathrm{F}_{\text{pseudo}} \otimes \mathrm{I}_3$, $\mathrm{R} = \mathrm{R}_{\text{pseudo}} \otimes \mathrm{I}_3$, I_3 an identity matrix of 3 dimensions, $\otimes$ is the Kronecker product, and

$$\mathrm{F}_{\text{pseudo}} = \begin{bmatrix} 1 & \phi \\ 0 & e^{-\beta\phi} \end{bmatrix} \quad \mathrm{R}_{\text{pseudo}} = \sigma_\upsilon \begin{bmatrix} 0 \\ \nu\sqrt{1 - e^{-2\beta\phi}} \end{bmatrix}.$$

In the experiment, the values of the Langevin model parameters are set to $\beta = 10\text{s}^{-1}$, $\nu = 1\text{ms}^{-1}$, and $\phi = 32\text{ms}$. The noise standard deviation is $\sigma_\upsilon = [4.7, 4.7, 0.7]^T\text{ms}^{-1}$, where the z-component standard deviation is lower than that of the other components because movements in the z-axis are small. A high probability of survival $P_S = 0.999$ is selected as existing sources are likely to be persist.

The birth parameters are given by $\{r_B(\ell_i), f_B(\cdot, \ell_i) \triangleq \mathcal{N}(\cdot; m_B^{(i)}, \mathrm{P}_B^{(i)})\}_{i=1}^3$, where $r_B(\ell_i)$ is the birth probability of a source with label ℓ_i and $f_B(\cdot, \ell_i)$ is the birth probability density which is a Gaussian with mean $m_B^{(i)}$ and covariance $\mathrm{P}_B^{(i)}$. The Gaussian mean is a vector containing the expected location of source birth while the associated covariance specifies its spatial uncertainty. In the experiment, the values of these parameters are: $r_B(\ell_1) = r_B(\ell_2) = r_B(\ell_3) = 0.005$, $m_B^{(1)} = [5.0\ 1.0\ 1.8\ 0\ 0\ 0]^T$, $m_B^{(2)} = [4.0\ 3.0\ 1.5\ 0\ 0\ 0]^T$, $m_B^{(3)} = [2.5\ 0.5\ 1.5\ 0\ 0\ 0]^T$, $\mathrm{P}_B^{(1)} = \mathrm{P}_B^{(2)} = \mathrm{P}_B^{(3)} = \text{diag}([0.15; 0.15; 0.15; 0.15; 0.15; 0.15]^T)^2$. Note that the Gaussian means have units of m for the 3D position components and ms^{-1} for the 3D velocity components.

Subsequently, recall that the parameters of the *multi-array measurement likelihood* are shown in Table 3.2 of Section 3.4.3. The obtained array measurements are noisy in nature. Hence, each measurement from each array $z_k^{(q)}$ is related to the source state x_k via the measurement equation: $z_k^{(q)} = \mathrm{H}x_k + w_k^{(q)}$ where $q = 1, \ldots, Q$, $\mathrm{H} = [\mathrm{I}_3, 0]$, and $w_k^{(q)}$ is an additive Gaussian random vector that is used to model noise in the measurement. Similar to the process noise, $w_k^{(q)}$ is a 3-dimensional Gaussian random vector with zero mean and covariance $\sigma_{w^{(q)}} \sigma_{w^{(q)}}^T$, where $\sigma_{w^{(q)}}$ is a column vector of the component standard deviations. Note that each component is a Gaussian random variable that is statistically independent of one another and across time. Based on this measurement model, the single-source likelihood for each array q is given as: $g^{(q)}(z_k^{(q)}|x_k) = \mathcal{N}(z_k^{(q)}; \mathrm{H}x_k, \sigma_{w^{(q)}} \sigma_{w^{(q)}}^T)$. In the experiment, the noise standard deviation vector is set to $\sigma_{w^{(q)}} = [0.1, 0.1, 0.1]^T\text{m}$ for $q = 1, \ldots, Q$. The probability of detection $P_D^{(q)} = 0.6$ for $q = 1, \ldots, Q$ is chosen to reflect the quality of the obtained measurements. The intensity function $\kappa^{(q)}(\cdot) = 10\mathcal{U}(\cdot)$ for $q = 1, \ldots, Q$ denotes an average of 10 false detections per frame where each individual false detection is uniformly distributed in its

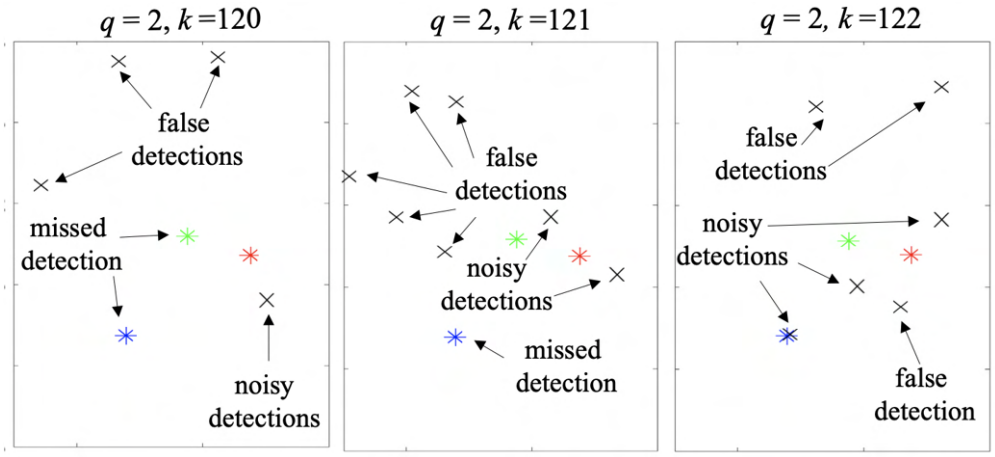


Figure 3.6: Observed measurements projected onto 2D ground plane as represented by black crosses at frames k=120, 121 and 122 for Array 2. The true positions for the active sources at the relevant times are denoted by colored asterisks.

space.

Source Separation

In the separation module, the STFT of the raw microphone signals is performed with a 1024-sample Hann window with 50% overlap to reduce the effect of windowing [366]. Since STFT from a 1024-sample with 50% overlapping window corresponds to the same number of frames as STFT from a 512-sample window with no overlapping, the frames are synchronized from the tracking module to the separation module, so that tracking estimates obtained at each frame are used for the separation accordingly.

3.6.3 Evaluation of SRP-PHAT Multi-Array Measurements

Due to space constraints, we only present the evaluation on real data. Fig. 3.6 (a) shows the real measurements obtained from an array compared with the ground-truth source positions. Notice that there is noise, missed detections (false negatives) and false detections (false positives) as expected across time frames. To evaluate the quality of the obtained multi-array measurements, we need a distance function between two sets of points, i.e., the set which contains the array measurements and the set which contains the ground-truth source positions. This distance function must be able to capture the accuracy of the individual points and the mismatch in number of points. Conceptually, the distance function must satisfy the three axioms of a metric: *identity*, *symmetry* and *triangle inequality*. While the first two axioms are often easily met, the triangle inequality is equally important. Conformity with the triangle inequality ensures the metric to be consistent with geometric interpretation, i.e., the shortest distance between two points is a straight line.

To this end we employ the Optimal Sub-Pattern Assignment (OSPA) distance which

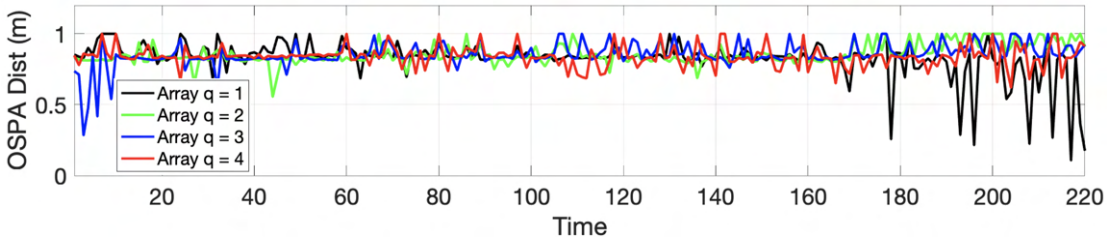


Figure 3.7: OSPA distance between the obtained source measurements and true source positions (lower is better) for each microphone array.

Table 3.3: Average OSPA distance on the obtained source measurements.

Array	Average OSPA Components (m)		
	Localization	Cardinality	OSPA
1	0.32	0.51	0.83
2	0.31	0.54	0.85
3	0.33	0.53	0.86
4	0.33	0.51	0.84

is an established mathematically consistent and physically meaningful metric between two finite sets of points [50]. The OSPA distance captures both localization and cardinality errors between two finite sets with a suitable base-distance between the points. The Euclidean distance (2-norm) is often used as the base-distance, and the resulting OSPA distance captures the perturbation error (localization) in the measurements caused by noise, and the error in the number of measurements (cardinality) caused by potential missed detections and false detections. Base-distances between two points that exceed the cutoff are capped at the cutoff value. The cutoff value is effectively the threshold at which a localization error is deemed as a cardinality error. A higher cutoff value brings more emphasis on the cardinality errors, and vice versa. The OSPA distance between the set containing the array measurements and the set containing true source positions is interpreted as a per-point error that ranges from zero to the cutoff value with units in meters. Interested readers can refer to [50] for full details.

For this evaluation, we compute the OSPA distance between the set of measurements obtained from each array and the set of source ground-truth trajectories with cutoff of 1m as shown in Fig. 3.7. It is observed that the OSPA distance for each array has a time average of about 0.8m. This is supported by Table 3.3, which shows the time average OSPA distance for each array along with its localization and cardinality components. The table indicates that the average localization error for each array is about 0.3m, while the average cardinality error for each array is about 0.5m. From these values, we observe that the OSPA distances have noticeable localization errors but are still dominated by cardinality errors. Consequently, when measurements corresponding to the direct path are obtained, they are somewhat noisy, while it it also clear that there is a high number of missed detections and false detections. Combined with the fact that the measurements have no identities or labels, and that the number of sources are

unknown and time-varying, it is clear that source separation via spatial filtering using the multi-array measurements is not viable.

3.6.4 Evaluation of Multi-Source Tracking Filter

The multi-array measurements are fed into the multi-source Bayesian tracking filter (MS-GLMB filter) at each frame, which outputs the filtering density. This output is fed back into the filter to process multi-array measurements at the next frame, and into the estimator to generate the multi-source state estimate which contains the estimated source tracks (positions and labels).

A track is defined when the source position estimates across frames are associated with a common label. Specifically, the mathematical definition of a track is a function whose domain is the set of time instants at which the source exists. In online tracking, a track can be fragmented or "broken" when the estimated source labels are not matching across time frames. Another common error is track switching which occurs when the label of a track switches to another. While the OSPA distance provides an indication of the acoustic measurement performance, it does not account for labeling errors between the estimated and true sets of tracks. As a result, it does not penalize track switching and fragmentation. In order to evaluate the estimated source tracks against the ground-truth source trajectories, we need a distance function to characterize the error between tracks over a time window.

To achieve this, we use the OSPA$^{(2)}$ metric which is defined for two sets of tracks, i.e., the set of estimated source tracks and the set of true source tracks. The construction of the OSPA$^{(2)}$ metric is based on the OSPA metric. In particular, OSPA$^{(2)}$ uses a time-averaged OSPA distance (over the common track times, with an appropriate cutoff) between a pair of tracks as the base-distance. The OSPA$^{(2)}$ distance treats the individual tracks as individual points in a larger space of tracks. The OSPA$^{(2)}$ distance is constructed as the OSPA distance between the two sets of tracks where the base distance is defined directly above [51]. Hence, the name OSPA$^{(2)}$ reflects the OSPA-on-OSPA nature in its construction. The OSPA$^{(2)}$ distance is capable of penalizing track switches (label changes) and fragmentations ("broken" tracks). The OSPA$^{(2)}$ is also parameterized by the cutoff value, which provides a sensitivity tradeoff between localization and cardinality errors between the tracks. The interpretation of the OSPA$^{(2)}$ distance evaluated over a fixed time window is consequently a time-averaged per-track error. The complete breakdown of the OSPA$^{(2)}$ metric can be found in [51].

For online tracking, it is desirable to have the tracking performance as a function of time. This can be achieved by computing the OSPA$^{(2)}$ distance over a sliding window instead of a fixed time window. This means that the OSPA$^{(2)}$ distance is plotted against time as the sliding window moves forward. Tracks whose domains lie outside the window are disregarded. This is useful for "forgetting" errors that were made further in the

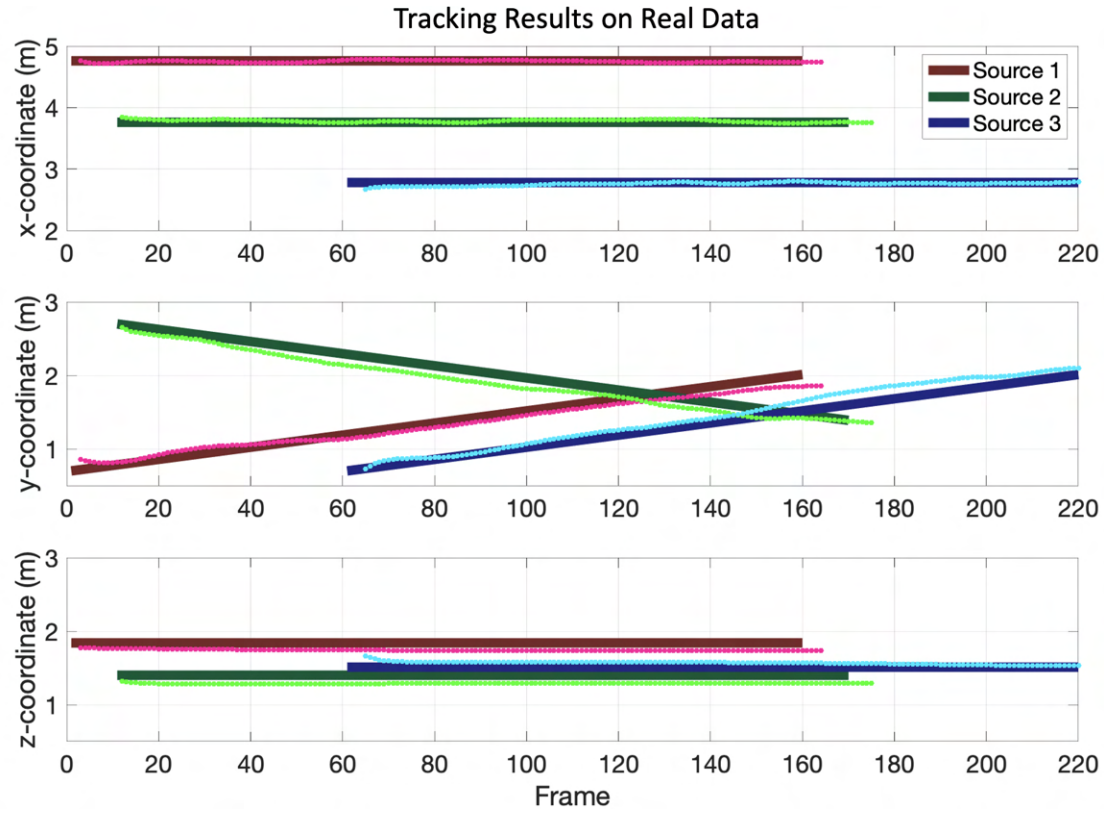


Figure 3.8: 3D estimated source tracks (colored dots) vs the true source trajectories (colored lines) plotted against time.

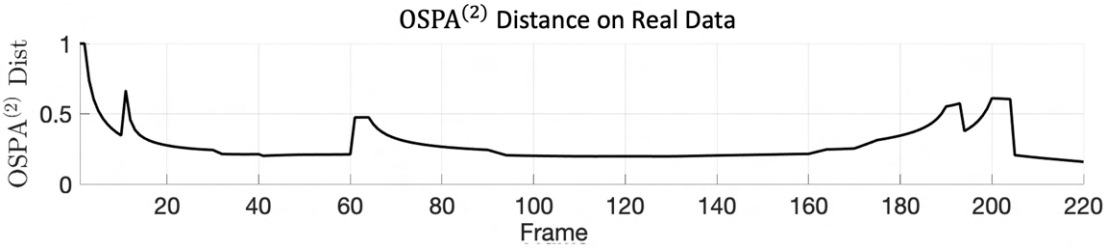


Figure 3.9: OSPA$^{(2)}$ distance between estimated and true source trajectories (lower is better).

past. For this evaluation, a cutoff of 1m and a window length of 30 frames are used.

Real Data

The 3D estimated tracks (colored dots) from the MS-GLMB tracking filter are compared with the source ground-truth trajectories (colored lines) in Fig. 3.8, where the color of a dot represents the label of a particular track. While the estimated tracks for Source 1 (red), 2 (green) and 3 (blue) at frame 1, 11 and 61 respectively have slight delays in the initiations, we observe that the tracking filter manages to initiate and maintain all 3 estimated tracks consistently across frames with respect to the ground-truth trajectories.

Fig. 3.9 shows the OSPA$^{(2)}$ distance between the estimated tracks and the ground-

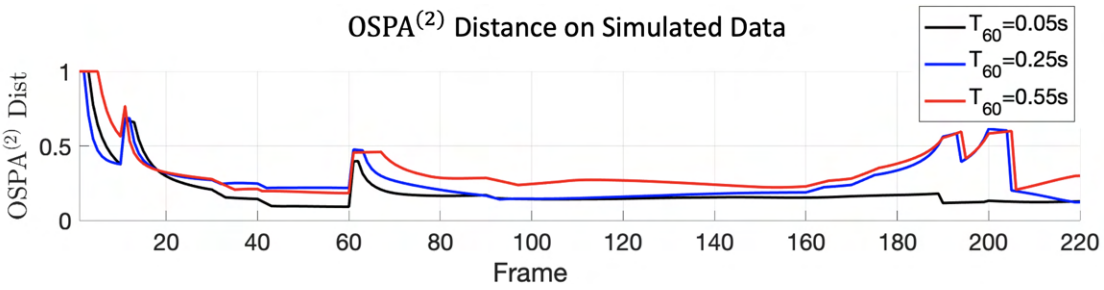


Figure 3.10: OSPA$^{(2)}$ distance between estimated and true source trajectories (lower is better).

truth trajectories plotted against time. Notice that the spikes of the curve correspond to the errors caused by the late track initiations and terminations of Source 1, 2 and 3 as depicted Fig. 3.8. Despite noise, false detections (false positives) and missed detections (false negatives) in the obtained multi-array measurements, the result validates the proposed tracking filter for solving the space-time permutation problem, and producing tracks for each source with reasonable accuracy as corroborated by both Fig. 3.8 and Fig. 3.9.

Simulated Data

Due to space constraints, we omit the 3D-track plots for simulated data and only present the OSPA$^{(2)}$ distances for the tracking estimates generated at reverberation times $T_{60} =$ 0.05s, 0.25s, and 0.55s in Fig. 3.10.

At $T_{60} = 0.05$s (in black), the MS-GLMB tracking filter achieves the lowest OSPA$^{(2)}$ distance compared to the other 2 curves, indicating that the tracking result is the best out of the other 2 examples. This is expected as the multi-array measurements capture the direct path.

At $T_{60} = 0.25$s (in blue), we see that the error curve is similar to the OSPA$^{(2)}$ error curve on real data, where the spikes are caused by the delays in track initiations and terminations. This indicates an agreement between the simulation and the real measurements.

At $T_{60} = 0.55$s (in red), we observe that the error curve is higher than that of the previous two curves, indicating a poorer tracking result. This increase in error is caused by late track initiations and terminations, and larger localization error due to higher reverberation.

3.6.5 Evaluation of Source Separation

For moving sources, the delay of the source signal with respect to any microphone array is changing over time. In our proposed method, the selection of the array for source separation depends on the source position at each frame. Therefore, perceptual measures such as PESQ [367], STOI [368] and PEASS [369] that rely on delay-compensation,

Table 3.4: Scales of SIG, BAK and OVRL in the Subjective Listening Test.

SIG	
Rating	Description
5	Very natural, no degradation
4	Fairly natural, little degradation
3	Somewhat natural, somewhat degraded
2	Fairly unnatural, fairly degraded
1	Very unnatural, very degraded

BAK	
Rating	Description
5	Not noticeable
4	Somewhat noticeable
3	Noticeable but not intrusive
2	Fairly conspicuous, somewhat intrusive
1	Very conspicuous, very intrusive

OVRL	
Rating	Description
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

are not directly applicable. One possibility for using these measures is to consider time frames where the sources are almost stationary. However, this is outside the scope of this dissertation as there may not be enough signal information in those frames, and a very complex study is needed with the development of suitable measures. Conventional BSS performance measures that are based on signal (energy) ratios, i.e., the BSSEval [370], require an exact time-alignment between the estimated and true signals to work [370]. As our experiment involves sources that are moving, and the exact times at which the sources appear in the scene are unknown, BSSEval is also not suitable for evaluating the source separation performance.

To evaluate the separation performance, we administered a subjective listening test on all scenarios based on the ITU-T P.835 methodology specifically designed to evaluate the distortions and overall quality of noise suppression algorithms [52]. In the test, each participant is instructed to listen to the clean speech signal (upper anchor reference), the separated speech signal (to be evaluated) and the mixture signal (lower anchor reference), then rate them on:

- The speech signal alone using a five-point scale of signal distortion (SIG);
- The background interfering noise alone using a five-point scale of background intrusiveness (BAK);

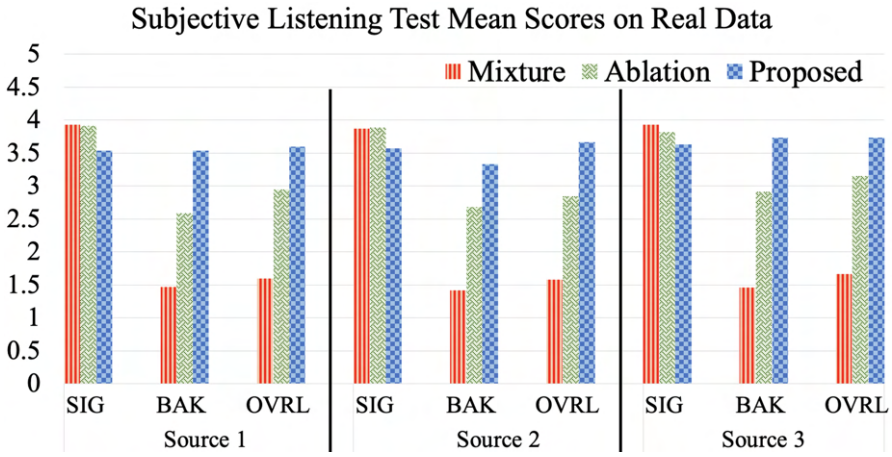


Figure 3.11: Mean scores for SIG, BAK, and OVRL for the estimated source signals, ablation study (estimation without post-processing), and original mixture signals evaluated on real data.

- The overall quality using the scale of mean opinion score (OVRL).

The scales of SIG, BAK and OVRL are described in Table 3.4. The listening tests are carried out on the separated signals both before and after the post-processing step. This form of ablation study is undertaken with the intention of understanding the trade-off between additional speech suppression and signal distortion due to the optional post-processing.

In this evaluation, 17 people (11 males, 6 females) of ages from 20 to 40 are recruited to partake in the listening test. To assess the overarching discrepancies between the test ratings on the separated speech signal and the unprocessed mixture signal, a statistical analysis of variance (ANOVA) is adopted to present the significant statistical difference between the quality of the separated speech signal and the unprocessed mixture based on a 0.05 significance level.

Real Data

For the subjective listening test, the mean scores over all 3 aspects, i.e., SIG, BAK and OVRL, of the separated/estimated source signals and the unprocessed mixture signals are presented in Fig. 3.11. We observe that the BAK and OVRL mean scores of all three estimated source signals from the proposed method (the blue bars) are relatively high as compared to the mean scores of the mixture signals, while the SIG mean scores of all estimated and mixture signals are relatively close. This indicates that the source signals are well separated with minimal signal distortions.

The p-values of the one-way ANOVA test between the estimated source signals and the unprocessed mixture signals are tabulated in Table 3.5. In terms of SIG, the table shows that all values of the proposed method are higher than 0.05, which means that there is no statistically significant difference in signal distortion between the estimated

Table 3.5: One-way ANOVA test between the estimated source signals and original mixture signals on real data, and corresponding ANOVA test for the ablation study (estimation without post-processing).

Source		p-value		
		SIG ↑	BAK ↓	OVRL ↓
1	Proposed	0.0791*	0.0001	0.0001
	Ablation	0.9247*	0.0058	0.0053
2	Proposed	0.1122*	0.0001	0.0001
	Ablation	0.9349*	0.0059	0.0051
3	Proposed	0.1494*	0.0001	0.0001
	Ablation	0.8694*	0.0054	0.0052

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

source signals and the mixture signals. In terms of BAK and OVRL, the table shows that all values of the proposed method are less than 0.05, indicating a statistically significant difference in speech intrusiveness and overall quality respectively.

From the results of the ablation study in Fig. 3.11 (the green bars) and Table 3.5, it can be seen that the BAK and OVRL means scores are slightly poorer than that of the proposed method, but the SIG mean scores are better than that of the proposed method across the board. Subsequently, the BAK and OVRL p-values indicate that there is a statistically significant difference in speech intrusiveness and overall quality, whereas the SIG p-values indicate that there is no statistically significant difference in signal distortion. These observations indicate that the proposed method minus post-processing achieves noticeable speech suppression with negligible signal distortion. The addition of the post-processing does indeed further enhance interference suppression, but at the cost of some signal distortion which manifests as musical noise in the estimated signals.

In summary, the proposed method achieves source separation with good noise (interfering speech) suppression, which is corroborated by both the mean scores and the ANOVA test in Fig. 3.11 and Table 3.5 respectively. The audio files for this experiment are available in https://github.com/researchwork888/BSMMS_via_Tracking.

Simulated Data

The mean scores for the subjective listening test on the estimated source signals and the unprocessed mixture signals, obtained under reverberation times $T_{60} = 0.05\text{s}, 0.25\text{s}$ and 0.55s, are shown in Fig. 3.12. Based on the relative differences for SIG, BAK and OVRL between all estimated and mixture signals at $T_{60} = 0.05\text{s}$ and 0.25s, we observe a similar pattern as for the real data, which shows that the proposed algorithm is capable of separating the sources reasonably well. However, at $T_{60} = 0.55\text{s}$, the separation performance degrades as the mean scores between all estimated and mixture signals are relatively close. Overall, we observe a downward trend in mean scores of the estimated

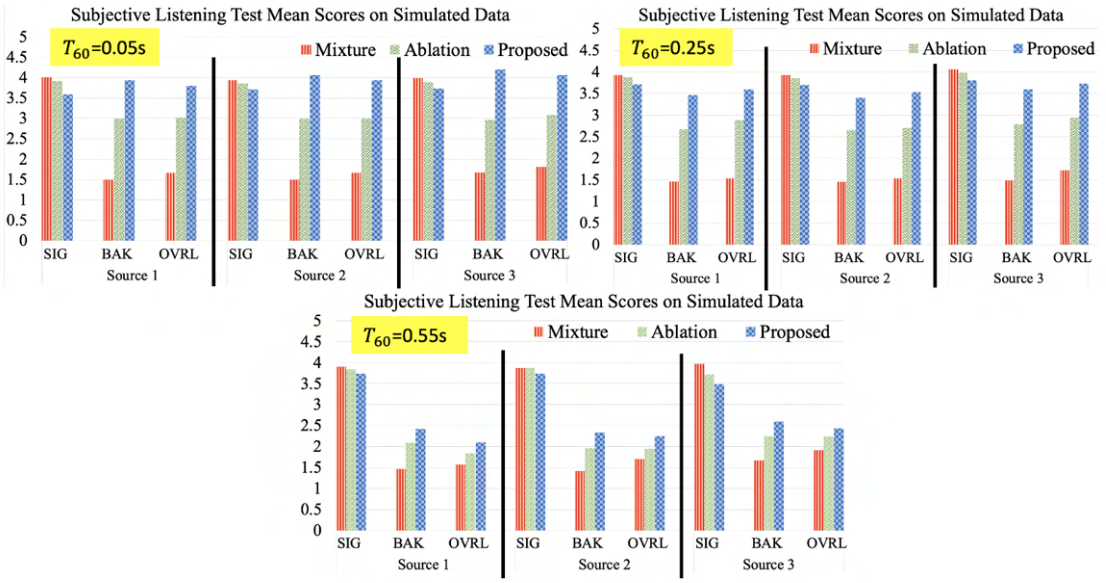


Figure 3.12: Mean scores for SIG, BAK, and OVRL for the estimated source signals, ablation study (estimation without post-processing), and original mixture signals evaluated on simulated data.

source signals from low T_{60} to high T_{60}. This degradation is expected because both tracking and separation performance degrades with increasing reverberation.

The p-values of the one-way ANOVA test between the estimated source signals and the unprocessed mixture signals are tabulated in Table 3.6. In terms of SIG, the p-values for all three sources at all reverberation times are above 0.05. This indicates that signal distortions between the estimated source signals and mixture signals are very similar. In terms of BAK and OVRL, the computed p-values for all three sources at $T_{60} = 0.05$s and 0.25s are below 0.05. This suggests that speech intrusiveness and the overall quality of the estimated source signals are statistically different from the mixture signals, thus indicating good separation.

At $T_{60} = 0.55$s however, we see that the p-values on OVRL for all three sources are higher than 0.05, suggesting that there is no statistically significant difference between the overall quality of the estimated source signals and the mixture signals. This, combined with the fact that the BAK values are fairly close to 0.05, suggests an overall poorer separation performance as interfering speech is not well suppressed. This decrease in performance is most likely due to two main reasons. The first being as reverberation time increases, the quality of tracking deteriorates, resulting in more localization errors. The second being the failure in the signal sparsity assumption, which results in leakage in the TF masking.

Examination of the ablation results in Fig. 3.12 and Table 3.6 reveals similar trends to those observed in the real-data experiments. For each of the three reverberation levels, the proposed method minus post-processing achieves noticeable suppression with negligible distortion, but the addition of the post-processing involves a trade-off between further suppression and audible distortion.

Table 3.6: One-way ANOVA test between the estimated source signals and original mixture signals on simulated data, and corresponding ANOVA test for the ablation study (estimation without post-processing).

T_{60}(s)	Source		p-value		
			SIG ↑	BAK ↓	OVRL ↓
0.05	1	Proposed	0.1903*	0.0001	0.0001
		Ablation	0.7393*	0.0031	0.0034
	2	Proposed	0.2279*	0.0001	0.0001
		Ablation	0.7618*	0.0031	0.0033
	3	Proposed	0.2169*	0.0001	0.0001
		Ablation	0.6511*	0.0031	0.0032
0.25	1	Proposed	0.1802*	0.0001	0.0001
		Ablation	0.8421*	0.0051	0.0048
	2	Proposed	0.1885*	0.0001	0.0001
		Ablation	0.7978*	0.0051	0.0047
	3	Proposed	0.1868*	0.0001	0.0001
		Ablation	0.7296*	0.0048	0.0049
0.55	1	Proposed	0.1629*	0.0355	0.1051*
		Ablation	0.8397*	0.0411	0.2481*
	2	Proposed	0.1741*	0.0396	0.3791*
		Ablation	0.9711*	0.0443	0.4521*
	3	Proposed	0.0791*	0.0343	0.2041*
		Ablation	0.8883*	0.0403	0.3451*

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

It can be seen that the separation performance on simulated data at $T_{60} = 0.25$s matches the results on real data, and that the separation performance generally degrades as reverberation time increases. This is corroborated by both the mean scores and the ANOVA test in Fig. 3.12 and Table 3.6. We also note the presence of more perceptible signal distortion in real data compared to simulated data. This is likely due to the mismatch between the real room environment and the simulated room model, which leads to additional spectral leakage in the time-frequency masking of the post-processing. The audio files for the experiments on simulated data are also provided in https://github.com/researchwork888/BSMMS_via_Tracking.

3.7 Conclusion

This chapter proposes a block-wise or online solution for blind source separation with multiple microphone arrays, which can accommodate an unknown time-varying number of acoustic moving sources in mild reverberation. The proposed solution is based on first obtaining source position measurements, then estimating the trajectories of the sources, and finally separating the mixed signal with corresponding spatial filtering. In real acoustic recordings measured at $T_{60} \approx 0.25$s, it is observed that the SRP-PHAT

source measurements are relatively noisy, and contain significant false and missed detections. In addition, the measurements are unlabeled, and coupled with the unknown appearance, disappearance and movement of sources, it is not known which source generated which measurement at the current time, nor which measurements are connected to the same source across time. These observations verify the extent of the inherent space-time permutation problem, which is then addressed with the application of a labeled RFS based MS-GLMB tracking filter. Results indicate that the tracking filter is able to recover the source trajectories (i.e., the positions and identities) from the imperfect source measurements with some delay in initiation and termination. Separation is carried out via a corresponding set of time-varying generalized side-lobe cancellers. Evaluations with subjective listening tests confirm acceptable performance in mild reverberation. Additional experiments via acoustic room simulations with the ISM method indicate clear separation performance at lower reverberation $T_{60} = 0.05$s, matching performance in mild reverberation $T_{60} = 0.25$s, and noticeable deterioration at higher reverberation $T_{60} = 0.55$s.

Chapter 4

Visual Multi-Object Tracking with Occlusion Handling

THIS chapter proposes an online multi-camera multi-object tracker that only requires monocular detector training, independent of the multi-camera configurations, allowing seamless extension/deletion of cameras without retraining effort. The proposed algorithm has a linear complexity in the total number of detections across the cameras, and hence scales gracefully with the number of cameras. It operates in the 3D world frame, and provides 3D trajectory estimates of the objects. The key innovation is a high fidelity yet tractable 3D occlusion model, amenable to optimal Bayesian multi-view multi-object filtering, which seamlessly integrates, into a single Bayesian recursion, the sub-tasks of track management, state estimation, clutter rejection, and occlusion/misdetection handling. The proposed algorithm is evaluated on the latest WILDTRACKS dataset, and demonstrated to work in very crowded scenes on a new dataset. The content of this chapter has been published in [59].

4.1 Introduction

The interest of visual tracking is to jointly estimate an unknown time-varying number of object trajectories from a stream of images [209]. The challenges of visual tracking are the random appearance/disappearance of the objects, false positives/negatives, and data association uncertainty [194]. Multiple object tracking (MOT) algorithms can operate online to produce current estimates as data arrives, or in batch which delay the estimation until further data is available [197, 198]. In principle, batch algorithms are more accurate than online as they allow better data integration into the estimates [193–196]. Online algorithms, however, tend to be faster and hence better suited for time-critical applications [198–200, 202, 210].

The common sub-tasks, traditionally performed by separate modules in a MOT system are track management, state estimation, clutter rejection, and occlusion/misdetection

handling. Track management involves the initiation, termination and identification of trajectories of individual objects, while state estimation is concerned with determining the state vectors of the trajectories. Problems such as track loss, track fragmentation and identity switching are caused by false negatives that can arise from occlusions when objects of interest are visually blocked from a sensor, or from misdetections when the sensor/detector fails to register objects of interest. On the other hand, false positives can lead to false tracks and identity switching. Hence, occlusion/misdetection handling and clutter rejection are critical for improving tracking performance.

While occlusion handling is just as challenging compared with the other sub-tasks, theoretical developments are far and few [175]. This is due mainly to the complex object-to-object and object-to-background relationships, as well as computational tractability because, theoretically, all possible partitions of the set of objects need to be considered [198]. In a single-view setting, useful *a priori* information about the objects of interest are exploited to resolve occlusions [193, 194, 202, 371]. However, there are fundamental limitations on what can be achieved with single-view data. In contrast, a multi-view setting naturally allows exploiting complementary information from the data to resolve occlusions since an object occluded in one view may not be occluded in another [54]. Furthermore, from an information theoretic standpoint, data from diverse views will reduce the uncertainty on the set of objects of interest, thereby improving overall tracking performance. Given the proliferation of cameras in today's world, it is imperative to develop effective means for making the best of the information-rich multi-view data sources, not only for occlusion handling, but ultimately to achieve better visual tracking.

The perennial challenge in multi-view visual MOT is the high-dimensional data association problem between the detections and objects, across different views/cameras [174, 175]. Two common architectures for multi-view MOT are shown in Fig. 4.1. So far the best solutions are batch algorithms with the architecture in Fig. 4.1 (a). These solutions are based on: generative modeling and dynamic programming [174]; convolutional neural network (CNN) multi-camera detection (MCD), trained on multi-view datasets [55], followed by track management [57]; and MCD via multi-view CNN training combined with Conditional Random Fields (CRF) models to exploit multi-camera geometry (followed by track management) [56]. These MCD based MOT solutions, which produce trajectories on the ground plane, have been shown to outperform previous works [55], and demonstrated remarkable performance in crowded scenarios [56]. Note that such data-centric MCDs require retraining when the multi-camera system is extended/reconfigured, and that training/learning is expensive as the input space is very high-dimensional due to the large number of possible combinations across the cameras [58].

In practice, it is desirable to have an online multi-view MOT system whose complexity scale linearly with the number of cameras, and do not require multi-view train-

ing so that reconfiguration (including addition and deletion) of cameras can be performed without interruption to the operation. Moreover, in a multi-view context, it is more prudent to have trajectories in the 3D world frame for applications such as sports analytics, age care, school environment monitoring, etc. While there are solutions to online 3D multi-view MOT with monocular data such as [372, 373], they do not scale gracefully with the number of cameras. Similar to the mentioned batch-processing methods, these solutions are more data-centric as they rely, respectively, on deep training for object depth information, and motion learning.

At the other end of the spectrum are the model-centric approaches that rely largely on physical models of the dynamics of the objects, the geometry and characteristics of the sensors/cameras. Such model-based solutions to 3D online MOT with monocular data, using 2D object detections, 3D object proposals, and 3D point cloud techniques were developed, respectively, in [5, 216, 222]. From a state-space modeling perspective, a natural choice for online MOT is the multi-object Bayes filter [267]. Since the inception of the Random Finite Sets (RFS) framework for multi-object state-space models, a number of multi-objects Bayesian filters have been developed [122, 237] and applied to visual MOT problems [198, 210, 374].

In addition to algorithms, datasets for performance evaluation are also an important aspect of 3D multi-view MOT research. Existing multi-view datasets include DukeMTMC [375], PETS 2009 S2.L1 [376], EPFL - Laboratory, Terrace and Passageway [174], SALSA [377], Campus [197] and EPFL-RLC [55]. However, in [57] the authors discussed a number of their shortcomings and introduced a seven-camera high-definition (HD) unscripted pedestrian dataset known as WILDTRACKS to provide a high quality, highly crowded and cluttered evaluation scenario. It comes with accurate joint (extrinsic and intrinsic) calibration, and 7 series of 400 annotated frames for detection at a rate of 2 frames per second (fps). The annotations of the tracks are given both as locations on the ground plane and 2D bounding boxes projected onto each view.

While WILDTRACKS is more extensive than earlier datasets, it is still not sufficient for comprehensive 3D MOT performance evaluation. Specifically, for actual 3D MOT applications where objects may also move vertically (e.g., sport analytics, age care, etc.), ground plane annotations are simply not adequate for evaluating tracking performance in full 3D, i.e., changes in all 3 x, y, z-coordinates. To this end, the Curtin Multi-Camera (CMC) dataset is proposed to enrich the datasets and to enable performance evaluation in full 3D. This new dataset comprises four calibrated cameras, on scenarios of varying difficulties in crowd density and occlusion, as well as scenarios with people jumping and falling, all with 3D centroid-with-extent annotations, along with camera locations and parameters. Note that in addition to extrinsic and intrinsic parameters, we also provide the absolute camera locations needed for testing and evaluation of model-centric solutions that exploit multi-camera geometry.

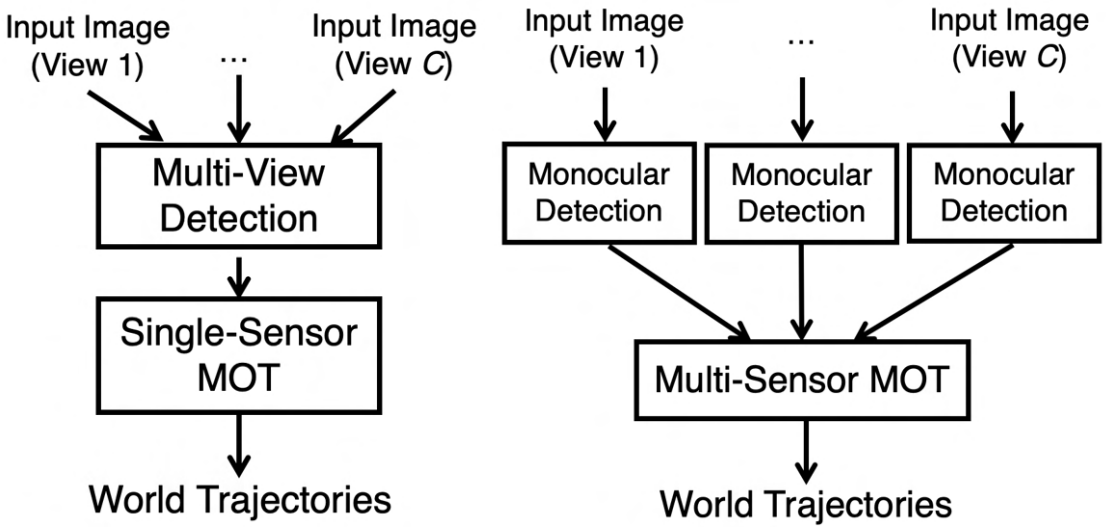


Figure 4.1: Multi-view Architectures: (a) Multi-view Detection + Single-sensor Multi-object Tracking; (b) Monocular Detection + Multi-sensor Multi-object Tracking.

This chapter proposes a model-centric, online multi-view visual MOT solution that only requires one-off monocular detector training (or off-the-shelf monocular detectors), independent of the multi-camera configurations, via the architecture of Fig. 4.1 (b). Hence, no retraining of the detectors is needed when the multi-camera system is extended/reconfigured. More importantly, our algorithm has a linear complexity in the total number of detections, thereby scales gracefully with the number of cameras. The algorithm intrinsically operates in the 3D world frame by exploiting multi-camera geometry, allowing it to track people jumping and falling, suitable for applications such as sports analytics, age care, school environment monitoring, etc. The proposed method is validated on the latest WILDTRACKS dataset on ground plane and show comparable results with Deep-Occlusion+KSP+ptrack [57]. To evaluate tracking performance in the 3D world frame, the new CMC dataset is used, which has varying degrees of difficulties on scenarios with very closely spaced people, with addition/deletion of cameras during operation, and with people jumping and falling.

The key innovation is a high fidelity yet tractable 3D occlusion model, amenable to Bayesian multi-sensor multi-object filtering [22], which seamlessly integrates, into a single Bayesian recursion, the sub-tasks of track management, state estimation, clutter rejection, and occlusion/misdetection handling. In the Bayesian paradigm, the multi-object filtering density captures all information on the set of trajectories in 3D, encapsulated in the observations, as well as dynamic and observation models. The novel occlusion model, incorporated in the multi-object measurement likelihood function, enables the MOT Bayesian filter to correctly maintain occluded tracks that would have otherwise been incorrectly terminated. The schematic in Fig. 4.2 shows the integration of the novel occlusion model into a near-optimal multi-sensor multi-object Bayes filter known

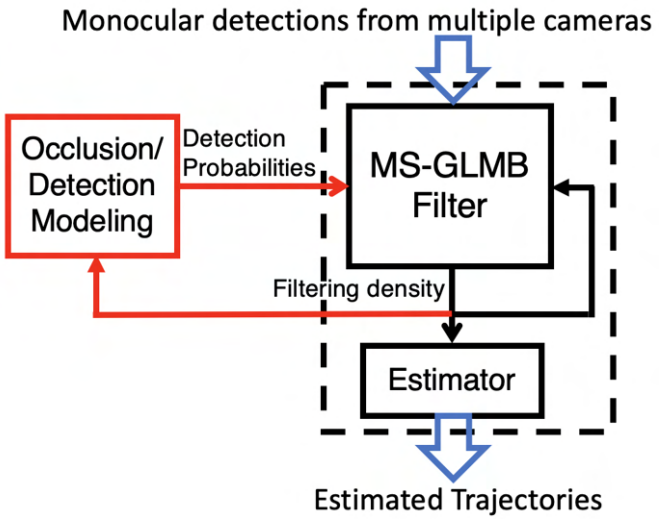


Figure 4.2: MV-GLMB-OC filter Processing Chain. Monocular detections from multiple cameras are fed into the filter, which outputs the filtering density. This output is fed into: the estimator to generate track estimates; and back into the filter to process detections at the next time. The Occlusion Model (red) is an add-on that takes the filter output and compute the detection probabilities for the filter on-the-fly.

as the Multi-Sensor Generalized Labeled Multi-Bernoulli (MS-GLMB) filter [22]. This configuration enables the proposed algorithm, herein referred to as Multi-View GLMB with OCclusion modeling (MV-GLMB-OC), to address occlusions, and inherits the numerical efficiency of the MS-GLMB filter. In short, the main technical contributions are:

- A tractable and realistic detection model that accommodates 3D occlusion by taking into account the Lines of Sights (LoSs) of all objects in the scene with respect to the cameras. In contrast, conventional detection models either neglect the LoSs of the objects or are computationally intractable, leading to poor tracking performance in the presence of occlusions. Our new detection model can be regarded as a generalization of tractable conventional detection models;
- The first Bayesian multi-view MOT filter for such detection model, which resolves occlusion online and is scalable with the number of sensors. Experiments show better performance than the latest multi-camera tracking algorithm;
- A new dataset with full 3D annotations (not restricted to the ground plane), in terms of position and extent in all 3 x, y, z-coordinates, including sequences that involve changes in the z-coordinate due to people jumping and falling. Instead of reporting performance for the entire scenario duration (as done traditionally), we also introduce live or online tracking performance evaluation over time, using the OSPA$^{(2)}$ metric [51], to characterize the behavior of the algorithm and demonstrate uninterrupted operation when the multi-camera system is extended/reconfigured.

4.2 Bayesian Formulation

This section formulates the multi-view MOT problem (Sections 4.2.1-4.2.4), including the proposed occlusion/detection model (Section 4.2.5), and the new tractable filter with occlusion handling capability (Section 4.2.6).

4.2.1 Bayes Filter

We first recall the classical Bayesian filter where the state x of the object, in some finite dimensional state space $\mathbb{X}$, is modeled as a random vector. The dynamic of the state is described by a Markov chain with transition density $f_+(x_+|x)$, i.e., the probability density of a transition to the state x_+ at the next time given the current state x. Note that for simplicity we omit the subscript for current time and use the subscript '+' denotes the next time step. Additionally, the current state x generates an observation z described by the likelihood function $g(z|x)$, i.e., the probability density of receiving the observation z given x. All information on the current the state is encapsulated in the filtering density[1] p, which can be propagated to the next time as p_+, via the celebrated Bayes recursion [232]

$$p_+(x_+) \propto g(z_+|x_+) \int f_+(x_+|x)\, p(x)\, dx. \tag{4.1}$$

The multi-view MOT Bayes filter used in this work is conceptually identical to the classical Bayes filter above by replacing: x and x_+ with the sets $\boldsymbol{X}$ and $\boldsymbol{X}_+$; p and p_+ with the multi-object filtering densities $\boldsymbol{\pi}$ and $\boldsymbol{\pi}_+$; f_+ and g with the multi-object transition density $\boldsymbol{f}_+$ and multi-object observation likelihood $\boldsymbol{g}$; z_+ with the observation set Z_+; and the integral with the set integral [122], i.e.,

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+) \propto \boldsymbol{g}(Z_+|\boldsymbol{X}_+) \int \boldsymbol{f}_+(\boldsymbol{X}_+|\boldsymbol{X})\, \boldsymbol{\pi}(\boldsymbol{X})\, \delta\boldsymbol{X}. \tag{4.2}$$

The sets $\boldsymbol{X}$ (and $\boldsymbol{X}_+$) containing the object states at the current (and next) time, is called the current (and next) multi-object state. Each element of the multi-object state $\boldsymbol{X}$ is an ordered pair $\boldsymbol{x} = (x, \ell)$, where $x \in \mathbb{X}$ is a state vector, and $\ell \triangleq (\varsigma, \iota)$ is a unique label consisting of the object's time of birth ς, and an index ι to distinguish those born at the same time [19]. The cardinality (number of elements) of $\boldsymbol{X}$ and $\boldsymbol{X}_+$ may differ due to the appearance and disappearance of objects from one frame to the next.

Under the Bayesian paradigm, the multi-object state is modeled as a random finite set, i.e., a finite-set-valued random variable, characterized by Mahler's multi-object density [122, 237] (equivalent to a probability density [268]). The multi-object transition density $\boldsymbol{f}_+$ captures the motions as well as births and deaths of objects. The multi-object observation likelihood $\boldsymbol{g}$ captures the detections, false alarms, occlusions,

[1] The filtering densities are conditioned on the observations, which have been omitted for notational compactness.

and misdetections.

4.2.2 Motion and Birth/Death Models

An object at time k, represented by a state $\boldsymbol{x} = (x, \ell)$, either survives with probability $P_S(\boldsymbol{x})$ and evolves to state $\boldsymbol{x}_+ = (x_+, \ell_+)$ at the next time with transition density

$$\boldsymbol{f}_{S,+}(\boldsymbol{x}_+|\boldsymbol{x}) = f_{S,+}(x_+|x, \ell)\delta_\ell[\ell_+], \tag{4.3}$$

or dies with probability $1 - P_S(\boldsymbol{x})$ [19]. At this next time, an object with label ℓ is born with probability $r_{B,+}(\ell)$, and with feature-vector x distributed according to a probability density $f_{B,+}(\cdot, \ell)$. Note that the label of an object remains the same over time, and hence the *trajectory* of an object is a sequence of consecutive states with a common label [19].

Let $\mathbb{B}_k$ denote the finite set of all possible labels for objects born at time k, then the label space for all objects up to time k is the disjoint union $\mathbb{L}_k = \biguplus_{t=0}^{k} \mathbb{B}_t$. For simplicity we omit the time subscript k, and let $\mathcal{L}(\boldsymbol{x})$ denote the label of an $\boldsymbol{x} \in \mathbb{X} \times \mathbb{L}$. For any finite $\boldsymbol{X} \subset \mathbb{X} \times \mathbb{L}$, we define $\mathcal{L}(\boldsymbol{X}) \triangleq \{\mathcal{L}(\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{X}\}$, and the *distinct label indicator* $\Delta(\boldsymbol{X}) \triangleq \delta_{|\boldsymbol{X}|}[|\mathcal{L}(\boldsymbol{X})|]$. At any time, the set $\boldsymbol{X}$ of (states of) objects in the scene must have distinct labels, i.e., $\Delta(\boldsymbol{X}) = 1$. Conditional on the current set of objects, it is standard practice to assume that objects are born or displaced at the next time, independently of one another. The expression for the multi-object transition density $\boldsymbol{f}_+$ is not needed in this work, interested readers are referred to [19].

4.2.3 Multi-Sensor Observation Model

Suppose that at time k, there are C cameras (sensors), and a set $\boldsymbol{X}$ of current objects. Each $\boldsymbol{x} \in \boldsymbol{X}$ is either: detected by camera $c \in \{1{:}C\}$, with probability $P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X} - \{\boldsymbol{x}\})$ and generates an observation $z^{(c)}$ in the measurement space $\mathbb{Z}^{(c)}$ with likelihood $g^{(c)}(z^{(c)}|\boldsymbol{x})$; or missed with probability $1 - P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X} - \{\boldsymbol{x}\})$. Note that to account for occlusions (and uncertainty in the detection process), the probability of detecting an object $\boldsymbol{x}$ also depends on the states of other current objects $\boldsymbol{X} - \{\boldsymbol{x}\}$. However, most MOT algorithms neglect this dependence for computational tractability.

The detection process also generates false positives at camera c, usually characterized by an intensity function $\kappa^{(c)}$ on $\mathbb{Z}^{(c)}$. The standard model is a Poisson distribution, with mean $\langle \kappa^{(c)}, 1 \rangle$, for the number of false positives, and the false positives themselves are i.i.d. according to the probability density $\kappa^{(c)}/\langle \kappa^{(c)}, 1 \rangle$ [237, 251, 253]. Moreover, conditional on the set $\boldsymbol{X}$ of objects, detections are assumed to be independent from false positives, and that the set $Z^{(c)}$ of detections and false positives at sensor c, are independent from those at other sensors.

An association hypothesis (at time k) associating labels with detections from camera c is a mapping $\gamma^{(c)}{:}\mathbb{L} \to \{-1{:}|Z^{(c)}|\}$, such that *no two distinct arguments are mapped to*

the same positive value [19]. This property ensures each detection comes from at most one object. Given an association hypothesis $\gamma^{(c)}$: $\gamma^{(c)}(\ell) = -1$ means object ℓ does not exist; $\gamma^{(c)}(\ell) = 0$ means object ℓ is not detected by camera c; $\gamma^{(c)}(\ell) > 0$ means object ℓ generates detection $z_{\gamma^{(c)}(\ell)}$ at camera c; and the set $\mathcal{L}(\gamma^{(c)}) \triangleq \{\ell \in \mathbb{L} : \gamma^{(c)}(\ell) \geq 0\}$ are the *live labels* of $\gamma^{(c)}$. Under standard assumptions, the (multi-object) likelihood for camera c is given by the following sum over the space $\Gamma^{(c)}$ of association hypotheses with domain $\mathbb{L}$ and range $\{-1{:}|Z^{(c)}|\}$ [19]:

$$\boldsymbol{g}^{(c)}(Z^{(c)}|\boldsymbol{X}) \propto \sum_{\gamma^{(c)} \in \Gamma^{(c)}} \delta_{\mathcal{L}(\gamma^{(c)})}[\mathcal{L}(\boldsymbol{X})] \left[\psi_{\boldsymbol{X}-\{\cdot\}}^{(c,\gamma^{(c)})}(\cdot)\right]^{\boldsymbol{X}}, \tag{4.4}$$

where $Z^{(c)} = \{z_{1:|Z^{(c)}|}^{(c)}\}$, and

$$\psi_{\boldsymbol{X}-\{\boldsymbol{x}\}}^{(c,\gamma^{(c)})}(\boldsymbol{x}) = \begin{cases} 1 - P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X}-\{\boldsymbol{x}\}), & \gamma^{(c)}(\mathcal{L}(\boldsymbol{x})) = 0 \\ \frac{P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X}-\{\boldsymbol{x}\}) g^{(c)}(z_j^{(c)}|\boldsymbol{x})}{\kappa^{(c)}(z_j^{(c)})}, & \gamma^{(c)}(\mathcal{L}(\boldsymbol{x})) = j > 0 \end{cases}, \tag{4.5}$$

Note that $\psi_{\boldsymbol{X}-\{\boldsymbol{x}\}}^{(c,\gamma^{(c)})}(\boldsymbol{x})$ also depends on $Z^{(c)}$, but we omitted it for clarity. Interested readers are referred to the texts [122, 237] for the derivation/discussion.

A multi-sensor (association) hypothesis is an array $\gamma \triangleq (\gamma^{(1)}, ..., \gamma^{(C)})$ of association hypotheses with the same set of live labels, denoted as $\mathcal{L}(\gamma)$. The likelihood that $\boldsymbol{X}$ generates the multi-sensor observation $Z \triangleq (Z^{(1:C)})$ is the product $\prod_{c=1}^{C} \boldsymbol{g}^{(c)}(Z^{(c)}|\boldsymbol{X})$, which can be rewritten as [22]

$$\boldsymbol{g}(Z|\boldsymbol{X}) \propto \sum_{\gamma \in \Gamma} \delta_{\mathcal{L}(\gamma)}[\mathcal{L}(\boldsymbol{X})] \left[\psi_{\boldsymbol{X}-\{\cdot\}}^{(\gamma)}(\cdot)\right]^{\boldsymbol{X}}, \tag{4.6}$$

where Γ is the set of all multi-sensor hypotheses,

$$\delta_{\mathcal{L}(\gamma)}[J] \triangleq \prod_{c=1}^{C} \delta_{\mathcal{L}(\gamma^{(c)})}[J], \tag{4.7}$$

$$\psi_{\boldsymbol{X}-\{\boldsymbol{x}\}}^{(\gamma)}(\boldsymbol{x}) \triangleq \prod_{c=1}^{C} \psi_{\boldsymbol{X}-\{\boldsymbol{x}\}}^{(c,\gamma^{(c)})}(\boldsymbol{x}). \tag{4.8}$$

Remark: The sets of objects, observations, and possibly the number of sensors and their parameters, may vary with time. However, for clarity we suppressed the time index.

4.2.4 Multi-Sensor GLMB Filter

Most of the literature on tracking assumes the probability of detection $P_D^{(c)}(\boldsymbol{x}; \boldsymbol{X}-\{\boldsymbol{x}\}) = P_D^{(c)}(\boldsymbol{x})$, i.e., independent of $\boldsymbol{X}-\{\boldsymbol{x}\}$. In this case, the Bayes recursion (4.2) admits an

analytical solution based on Generalized Labeled Multi-Bernoulli (GLMB) models.

A GLMB is a multi-object density of the form [19]

$$\boldsymbol{\pi}(\boldsymbol{X}) = \Delta(\boldsymbol{X}) \sum_{I,\xi} \omega^{(I,\xi)} \delta_I[\mathcal{L}(\boldsymbol{X})] \left[p^{(\xi)}\right]^{\boldsymbol{X}}, \tag{4.9}$$

where: $I \in \mathcal{F}(\mathbb{L})$ the space of all finite subsets of $\mathbb{L}$; $\xi \in \Xi$ the space of all (multi-sensor) association hypotheses histories up to the current time, i.e., $\xi \triangleq \gamma_{1:k}$; each $\omega^{(I,\xi)}$ is a non-negative weight such that $\sum_{I,\xi} \omega^{(I,\xi)} = 1$; and each $p^{(\xi)}(\cdot,\ell)$ is a probability density on $\mathbb{X}$. For convenience, we represent a GLMB by its parameter-set

$$\boldsymbol{\pi} \triangleq \left\{ \left(\omega^{(I,\xi)}, p^{(\xi)} \right) : (I,\xi) \in \mathcal{F}(\mathbb{L}) \times \Xi \right\}. \tag{4.10}$$

Each GLMB *component* (I,ξ) can be interpreted as a hypothesis with probability $\omega^{(I,\xi)}$, and each individual object $\ell \in I$ of this hypothesis has probability density $p^{(\xi)}(\cdot,\ell)$.

A simple multi-object state estimate can be obtained from a GLMB by first determining: the most probable cardinality n^* from the cardinality distribution [19]

$$\text{Prob}(|\boldsymbol{X}| = n) = \sum_{I,\xi} \delta_n[|I|] \omega^{(I,\xi)}; \tag{4.11}$$

and then the hypothesis (I^*,ξ^*) with highest weight such that $|I^*| = n^*$. The current state estimate for each object $\ell \in I^*$ can be computed from $p^{(\xi^*)}(\cdot,\ell)$, e.g., the mode or mean. Alternatively, the entire trajectory of object $\ell \in I^*$ can be estimated using the forward-backward algorithm, starting from its current filtering density $p^{(\xi^*)}(\cdot,\ell)$ and propagating backward to its time of birth [22, 286].

Under the Bayes recursion (4.2), and the standard multi-object model (i.e., with no occlusions, $P_D^{(c)}(\boldsymbol{x};\boldsymbol{X}-\{\boldsymbol{x}\}) = P_D^{(c)}(\boldsymbol{x})$ hence $\psi^{(\gamma)}_{\boldsymbol{X}-\{\boldsymbol{x}\}}(\boldsymbol{x}) = \psi^{(\gamma)}(\boldsymbol{x})$), the multi-object filtering density at any time is a GLMB [19]. Moreover, if (4.10) is the current GLMB filtering density, then the next GLMB filtering density

$$\boldsymbol{\pi}_+ = \left\{ \left(\omega_+^{(I_+,\xi_+)}, p_+^{(\xi_+)} \right) : (I_+,\xi_+) \in \mathcal{F}(\mathbb{L}_+) \times \Xi_+ \right\}, \tag{4.12}$$

can be computed via the *MS-GLMB recursion* Ω [22]

$$\boldsymbol{\pi}_+ = \Omega\left(\boldsymbol{\pi}; P_{D,+}\right), \tag{4.13}$$

where $P_{D,+} \triangleq (P_{D,+}^{(1)}, ..., P_{D,+}^{(C)})$ and the mathematical expressions for the MS-GLMB recursion operator $\Omega : \boldsymbol{\pi} \mapsto \boldsymbol{\pi}_+$ are:

$$I_+ = \mathbb{B}_+ \uplus I, \; \xi_+ = (\xi, \gamma_+) \tag{4.14}$$

$$\omega_+^{(I_+,\xi_+)} = 1_{\mathcal{F}(\mathbb{B}_+ \uplus I)}(\mathcal{L}(\gamma_+)) \omega^{(I,\xi)} \left[\bar{\omega}^{(\xi,\gamma_+)}\right]^{\mathbb{B}_+ \uplus I} \tag{4.15}$$

$$p_{+}^{(\xi_{+})}(x_{+},\ell) \propto= \begin{cases} \langle \Lambda_{S}^{(\gamma_{+})}(x_{+}|\cdot,\ell), p^{(\xi)}(\cdot,\ell)\rangle, & \ell \in \mathcal{L}(\gamma_{+}) - \mathbb{B}_{+} \\ \Lambda_{B}^{(\gamma_{+})}(x_{+},\ell), & \ell \in \mathcal{L}(\gamma_{+}) \cap \mathbb{B}_{+} \end{cases} \tag{4.16}$$

$$\bar{\omega}^{(\xi,\gamma_{+})}(\ell) = \begin{cases} 1 - \bar{P}_{S}^{(\xi)}(\ell), & \ell \in \overline{\mathcal{L}(\gamma_{+})} - \mathbb{B}_{+} \\ \overline{\Lambda}_{S}^{(\xi,\gamma_{+})}(\ell), & \ell \in \mathcal{L}(\gamma_{+}) - \mathbb{B}_{+} \\ 1 - r_{B,+}(\ell), & \ell \in \overline{\mathcal{L}(\gamma_{+})} \cap \mathbb{B}_{+} \\ \overline{\Lambda}_{B}^{(\gamma_{+})}(\ell), & \ell \in \mathcal{L}(\gamma_{+}) \cap \mathbb{B}_{+} \end{cases}, \tag{4.17}$$

$$\bar{P}_{S}^{(\xi)}(\ell) = \langle P_{S}(\cdot,\ell), p^{(\xi)}(\cdot,\ell)\rangle, \tag{4.18}$$

$$\Lambda_{B}^{(\gamma_{+})}(x_{+},\ell) = \psi^{(\gamma_{+})}(x_{+},\ell)\, f_{B,+}(x_{+},\ell) r_{B,+}(\ell), \tag{4.19}$$

$$\Lambda_{S}^{(\gamma_{+})}(x_{+}|y,\ell) = \psi^{(\gamma_{+})}(x_{+},\ell)\, f_{S,+}(x_{+}|y,\ell) P_{S}(y,\ell), \tag{4.20}$$

$$\overline{\Lambda}_{B}^{(\gamma_{+})}(\ell) = \int \Lambda_{B}^{(\gamma_{+})}(x,\ell)dx, \tag{4.21}$$

$$\overline{\Lambda}_{S}^{(\xi,\gamma_{+})}(\ell) = \int \langle \Lambda_{S}^{(\gamma_{+})}(x|\cdot,\ell), p^{(\xi)}(\cdot,\ell)\rangle dx. \tag{4.22}$$

Note that the recursion operator Ω also depends on the measurement Z_{+}, and model parameters for birth $(r_{B,+}, f_{B,+})$, death/survival P_S, motion $\boldsymbol{f}_{S,+}$, false alarms $\kappa_{+} \triangleq (\kappa_{+}^{(1)}, \ldots, \kappa_{+}^{(C)})$, and detection $g_{+} \triangleq (g_{+}^{(1)}, \ldots, g_{+}^{(C)})$ (described in Section 4.2.3). However, for our purpose it suffices to show the dependence on detection probabilities.

While the MS-GLMB filter can applied directly to multi-view MOT, a detection probability (of an object $\boldsymbol{x}$) that does not depend on other objects, i.e., $\boldsymbol{X} - \{\boldsymbol{x}\}$, is unable to capture the effect of occlusions. On the other hand, accounting for occlusions with $P_{D}^{(c)}(\boldsymbol{x}; \boldsymbol{X} - \{\boldsymbol{x}\})$ that actually depends on $\boldsymbol{X} - \{\boldsymbol{x}\}$, results in filtering densities that are not GLMBs. One example is the merged-measurement model [314], which involves summing over all partitions of the set $\boldsymbol{X}$, making it intractable [314]. Although the resulting filtering density can be approximated by a GLMB, this solution is still computationally demanding and not suitable for large number of objects [314]. In what follows, we propose a new detection model that addresses occlusions and permits efficient multi-view MOT implementations.

4.2.5 Detection Model with Occlusion

For tracking in 3D, we consider the state $\boldsymbol{x} = (x, \ell)$, where:

$$x = (x^{(p)}, \dot{x}^{(p)}, x^{(s)}); \tag{4.23}$$

$x^{(p)}$ is the object's position (centroid) in 3D Cartesian coordinates; $\dot{x}^{(p)}$ is its velocity; and $x^{(s)}$ is its shape parameter. The region in $\mathbb{R}^3$ occupied by an object with labeled state $\boldsymbol{x}$ is denoted by $R(\boldsymbol{x})$.

Consider camera c and the set $\boldsymbol{X}$ of current objects. In this work, an object $(x,\ell)\in \boldsymbol{X}$ is regarded as occluded from camera c when its position $x^{(p)}$ is not in the line of sight (LoS) of the camera, i.e., $x^{(p)}$ is in the *shadow regions* of the other objects in $\boldsymbol{X}$. Assuming straight LoSs, the shadow region of an object with labeled state $\boldsymbol{x}'$, relative to camera c (see Fig. 4.3), is given by

$$\mathcal{S}^{(c)}(\boldsymbol{x}') = \left\{\alpha \in \mathbb{R}^3 : \overline{(u^{(c)},\alpha)} \cap R(\boldsymbol{x}') \neq \emptyset\right\}, \tag{4.24}$$

where $\overline{(u^{(c)},\alpha)} \triangleq \{\chi\alpha + (1-\chi)u^{(c)} : \chi \in [0,1]\}$ is the line segment joining the position $u^{(c)}$ of camera c and α. Note that for an ellipsoidal region $R(\boldsymbol{x}')$, the indicator function $1_{\mathcal{S}^{(c)}(\boldsymbol{x}')}(\cdot)$ of its shadow region can be computed in closed form (see Section 4.3.1).

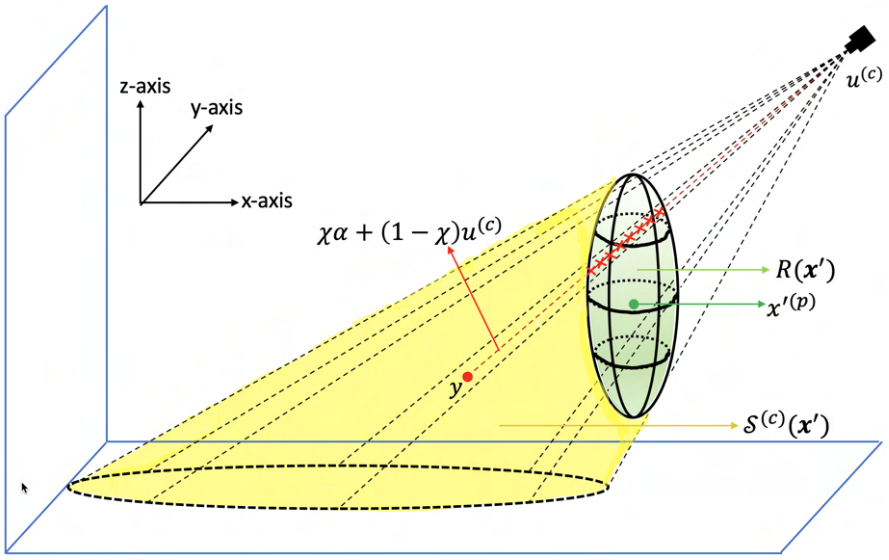


Figure 4.3: The shadow region (in yellow) of object with labeled state $\boldsymbol{x}'$, relative to camera c.

To incorporate the effect of occlusions into the detection model, the probability that $\boldsymbol{x} \in \boldsymbol{X}$ be detected by camera c should be close to zero when it is occluded from camera c. This can be accomplished by extending the standard detection probability so that: when $\boldsymbol{x}$ is in the LoS of camera c, its detection probability is $P_D^{(c)}(\boldsymbol{x})$; and when occluded by the other objects its detection probability scales down to $\mu P_D^{(c)}(\boldsymbol{x})$, where μ is a small positive number. More explicitly,

$$P_D^{(c)}(\boldsymbol{x};\boldsymbol{X}-\{\boldsymbol{x}\}) = P_D^{(c)}(\boldsymbol{x})\Big(\mathcal{V}(\boldsymbol{x};\boldsymbol{X}-\{\boldsymbol{x}\}) + \mu\big(1-\mathcal{V}(\boldsymbol{x};\boldsymbol{X}-\{\boldsymbol{x}\})\big)\Big), \tag{4.25}$$

where

$$\mathcal{V}(\boldsymbol{x};\boldsymbol{X}-\{\boldsymbol{x}\}) = \prod_{\boldsymbol{x}'\in\boldsymbol{X}-\{\boldsymbol{x}\}} \left(1 - 1_{\mathcal{S}^{(c)}(\boldsymbol{x}')}(\boldsymbol{x})\right) \tag{4.26}$$

Conditional on detection, $\boldsymbol{x}$ is observed at camera c as a bounding box $z^{(c)} \triangleq (z_p^{(c)}, z_e^{(c)})$,

where $z_p^{(c)}$ is the center, and $z_e^{(c)}$ is the extent, parameterized by the logarithms of the width (x-axis) and height (y-axis), in image coordinates. The observed $z^{(c)}$ is a noisy version of the box $\Upsilon^{(c)}(\boldsymbol{x})$ bounding the image of $R(\boldsymbol{x})$ in the camera's image plane, under the projection of the camera matrix $\mathrm{P}_{3\times4}^{(c)}$. This matrix projects homogeneous points in the world coordinate frame to homogeneous points in the image plane of camera c, and can be obtained by standard calibration techniques (see [378] for details). Note that for an ellipsoidal region $R(\boldsymbol{x})$, the axis-aligned $\Upsilon^{(c)}(\boldsymbol{x})$ on the image plane can be computed analytically (see Section 4.3.1). This observation process can be modeled by the likelihood

$$g(z^{(c)}|\boldsymbol{x}) = \mathcal{N}\left(z^{(c)}; \Upsilon^{(c)}(\boldsymbol{x}) + \begin{bmatrix} 0_{2\times1} \\ -w_e^{(c)}/2 \end{bmatrix}, \mathrm{diag}\left(\begin{bmatrix} w_p^{(c)} \\ w_e^{(c)} \end{bmatrix}\right)\right), \tag{4.27}$$

where $w_p^{(c)}$ and $w_e^{(c)}$ are respectively the vector of noise variances for the center and the extent (in logarithm) of the box. This Gaussian model of the logarithms of the width and height is equivalent to modeling the actual width and height as log-normals, which ensures that they are non-negative. Note that these log-normals have mean 1, and variances $e^{w_{e,1}^{(c)}} - 1$ and $e^{w_{e,2}^{(c)}} - 1$, where $w_{e,1}^{(c)}$ and $w_{e,2}^{(c)}$ are the two components of $w_e^{(c)}$. This means the observed width and height are randomly scaled versions of their nominal values, with an expected scaling factor of 1.

4.2.6 Multi-View GLMB Filtering with Occlusions

This subsection presents a tractable GLMB approximation to the multi-view Bayes filter to address occlusions. The proposed filter (with the new detection model to account for occlusion) is referred to as Multi-View GLMB with occlusion modeling (MV-GLMB-OC).

Given the current GLMB filtering density (4.10), the predicted density $\int \boldsymbol{f}_+(\boldsymbol{X}_+|\boldsymbol{X})\boldsymbol{\pi}(\boldsymbol{X})\delta\boldsymbol{X}$ in the Bayes recursion (4.2) is also a GLMB [19], which we denote by

$$\widehat{\boldsymbol{\pi}}_+(\boldsymbol{X}_+) = \Delta(\boldsymbol{X}_+)\sum_{I_+,\xi} w_+^{(I_+,\xi)}\delta_{I_+}[\mathcal{L}(\boldsymbol{X}_+)]\left[p_+^{(\xi)}\right]^{\boldsymbol{X}_+}, \tag{4.28}$$

where $I_+ \in \mathcal{F}(\mathbb{L}_+)$. Multiplying (4.28) by the likelihood (4.8) yields the next (unnormalized) multi-object density

$$\boldsymbol{\pi}_+(\boldsymbol{X}_+) \propto \Delta(\boldsymbol{X}_+)\sum_{I_+,\xi,\gamma_+} \delta_{\mathcal{L}(\gamma_+)}[\mathcal{L}(\boldsymbol{X}_+)]w_+^{(I_+,\xi)}\delta_{I_+}[\mathcal{L}(\boldsymbol{X}_+)]\left[p_{\boldsymbol{X}_+-\{\cdot\}}^{(\xi,\gamma_+)}(\cdot)\right]^{\boldsymbol{X}_+}, \tag{4.29}$$

where

$$p_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}^{(\xi,\gamma_+)}(\boldsymbol{x}_+) = p_+^{(\xi)}(\boldsymbol{x}_+)\psi_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}^{(\gamma_+)}(\boldsymbol{x}_+). \tag{4.30}$$

As previously alluded to, the multi-object density (4.29) is not a GLMB because $p^{(\xi,\gamma_+)}_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}$ depends on $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$. Nonetheless, a good GLMB approximation of (4.29) can be obtained by approximating $p^{(\xi,\gamma_+)}_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}$ with a density that is independent of $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$.

Note that $\psi^{(\gamma_+)}_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}$ is the only factor of $p^{(\xi,\gamma_+)}_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}$, which depends on $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$ (see (4.30)). Further inspection of (4.5) and (4.8) reveals that the detection probability functions $P^{(c)}_{D,+}(\cdot;\boldsymbol{X}_+-\{\boldsymbol{x}_+\})$, $c\in\{1{:}C\}$ are the only constituent terms that depend on $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$. Moreover, it follows from (4.25) that $P^{(c)}_{D,+}(\boldsymbol{x}_+;\boldsymbol{X}_+-\{\boldsymbol{x}_+\})$ only takes on two values, depending on whether $\boldsymbol{x}_+$ falls in the shadow region of $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$ w.r.t. camera c. Assuming the positions of the elements of $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$ are concentrated around their predicted values according to the prediction densities $p^{(\xi)}_+(\cdot,\ell)$, $\ell\in\mathcal{L}(\boldsymbol{X}_+-\{\boldsymbol{x}_+\})$, we can approximate $P^{(c)}_{D,+}(\cdot;\boldsymbol{X}_+-\{\boldsymbol{x}_+\})$ by replacing the set $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$ with its predicted value. Noting that the term $\delta_{I_+}[\mathcal{L}(\boldsymbol{X}_+)]$ in (4.29) implies $\mathcal{L}(\boldsymbol{X}_+)=I_+$, the prediction of $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$ is

$$\boldsymbol{X}^{(\xi,I_+)}_+=\{(x^{(\xi,\ell)}_+,\ell):\ell\in I_+-\mathcal{L}(\boldsymbol{x}_+)\},\tag{4.31}$$

where $x^{(\xi,\ell)}_+$ denotes an estimate (e.g., mean, mode) from the density $p^{(\xi)}_+(\cdot,\ell)$, which is either the birth density $f_{B,+}(\cdot,\ell)$ if $\ell\in\mathbb{B}_+$ or $\int f_{S,+}(\cdot|x,\ell)p^{(\xi)}(x,\ell)\,dx$ if $\ell\notin\mathbb{B}_+$ [19].

The above approximation translates to

$$p^{(\xi,\gamma_+)}_{\boldsymbol{X}_+-\{\boldsymbol{x}_+\}}\approx p^{(\xi,\gamma_+)}_{\boldsymbol{X}^{(\xi,I_+)}_+},\tag{4.32}$$

which is independent of $\boldsymbol{X}_+-\{\boldsymbol{x}_+\}$, thereby turning (4.29) into a GLMB. Moreover, the computation of this GLMB approximation to (4.29) only differs from the MS-GLMB recursion (4.13) in the detection probabilities

$$P^{(\xi,I_+)}_{D,+}(\ell)\triangleq\left(P^{(1)}_{D,+}((\hat{x}_+,\ell);\boldsymbol{X}^{(\xi,I_+)}_+),\ldots,P^{(C)}_{D,+}((\hat{x}_+,\ell);\boldsymbol{X}^{(\xi,I_+)}_+)\right),\tag{4.33}$$

where $\ell=\mathcal{L}(\boldsymbol{x}_+)$, and $\hat{x}_+$ denotes an estimate (e.g., mean, mode) from the density $p^{(\xi)}_+(\cdot,\ell)$. Specifically, the GLMB approximation of the multi-object filtering density can be propagated by the MS-GLMB recursion

$$\boldsymbol{\pi}_+=\Omega\left(\boldsymbol{\pi};\{P^{(\xi,I_+)}_{D,+}(\ell):\ell\in I_+,(\xi,I_+)\in\Xi\times\mathcal{F}(\mathbb{L}_+)\}\right).\tag{4.34}$$

The integration of the proposed occlusion model (via the detection probabilities) into the MS-GLMB filter is shown in Fig. 4.2. The implementation of this so-called MV-GLMB-OC filter is discussed in the next section.

4.3 Implementation

This section describes the implementation of the proposed filter for ellipsoidal objects. Section 4.3.1 provides mathematical representations for the objects and the multi-object

model parameters. Propagation of the MV-GLMB-OC filtering density is then described in Section 4.3.2.

4.3.1 Object Representation and Model Parameters

Each object is represented by an axis-aligned ellipsoid. For an object with labeled state $\boldsymbol{x} = (x, \ell)$, the position $x^{(p)}$ is the centroid, and the shape parameter $x^{(s)}$ is a vector containing the logarithms of the half-lengths of the ellipsoid's principal axes. Further, the time-evolution of the state vector x is modeled by a linear Gaussian transition density:

$$f_{S,+}(x_+|x,\ell) = \mathcal{N}\left(x_+; \mathrm{F}x + \begin{bmatrix} 0_{6\times 1} \\ -\upsilon^{(s)}/2 \end{bmatrix}, \mathrm{Q}\right), \tag{4.35}$$

where

$$\mathrm{F} = \left[\begin{array}{c:c} \mathrm{I}_3 \otimes \begin{bmatrix} 1 & \phi \\ 0 & 1 \end{bmatrix} & 0_{6\times 3} \\ \hdashline 0_{3\times 6} & \mathrm{I}_3 \end{array}\right], \tag{4.36}$$

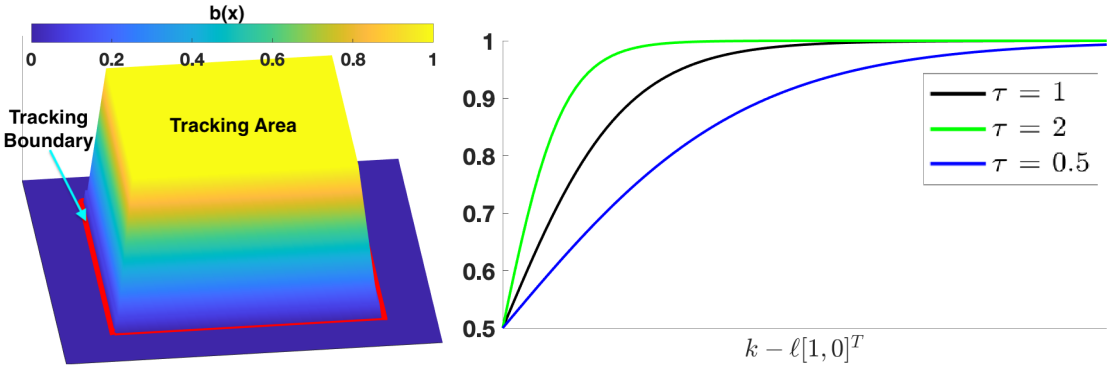
$$Q = \left[\begin{array}{c:c} \mathrm{diag}(\upsilon^{(p)}) \otimes \begin{bmatrix} \frac{\phi^2}{2} \\ \phi \end{bmatrix} \begin{bmatrix} \frac{\phi^2}{2} & \phi \end{bmatrix} & 0_{6\times 3} \\ \hdashline 0_{3\times 6} & \mathrm{diag}(\upsilon^{(s)}) \end{array}\right], \tag{4.37}$$

ϕ is the sampling period, $\upsilon^{(p)}$ and $\upsilon^{(s)}$ are, respectively, 3D vectors of noise variances for the components of the centroid and shape parameter (in logarithm) of the ellipsoid. This transition density describes a nearly constant velocity model for the centroid and a Gaussian random-walk for the shape parameter. Gaussianity of the logarithms of the half-lengths is equivalent to modeling the half-lengths as log-normals, which ensure that they are non-negative. Note that these log-normals have mean 1, and variances $e^{\upsilon_i^{(s)}} - 1$, $i = 1,2,3$, where $\upsilon_i^{(s)}$ is the i^{th} components of $\upsilon^{(s)}$. Hence, the observed half-lengths are randomly scaled versions of their nominal values, with an expected scaling factor of 1.

Empirically, objects that are in the scene for a long time, are more likely to remain in the scene, unless they are close to the borders (exit regions). This can be modeled via the following object survival probability [198]:

$$P_S(x,\ell) = \frac{b(x)}{1 + \exp(-\tau(k - \ell[1,0]^T))}, \tag{4.38}$$

where $b(x)$ is the the scene mask (chosen to be close to one in the middle of the scene, and close to zero in the designated exit regions and beyond) as depicted in Fig. 4.4 (a), and τ is the control parameter of the sigmoid function that is dependent on the duration



(a) (b)

Figure 4.4: Illustration of the survival probability model: (a) The scene mask $b(x)$; (b) The control parameter τ of the sigmoid function.

(age) of the track $k - \ell[1,0]^T$ as depicted in Fig. 4.4 (b).

The detection probability (4.25)-(4.26) can be computed in closed form when the objects extents are ellipsoids. As alluded to in Section 4.2.5, the shadow region indicator function $1_{\mathcal{S}^{(c)}(\boldsymbol{x}')}(\cdot)$ used for checking whether an object is in the shadow region of the object $\boldsymbol{x}'$, can be determined analytically. Suppose that $R(\boldsymbol{x}')$ in (4.24) is a quadric, then it intersects the line $\overline{(u^{(c)}, x^{(p)})}$ (between $u^{(c)}$ and $x^{(p)}$) if the roots of a certain quadratic equation are real [379]. Consequently, for an axis-aligned ellipsoidal object representation, the shadow region indicator function is given by

$$1_{\mathcal{S}^{(c)}(\boldsymbol{x}')}(\boldsymbol{x}) = \begin{cases} 1, & \left(\mathcal{B}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'}\right)^2 - 4\mathcal{A}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} C^{(c)}_{\boldsymbol{x}'} \geq 0 \\ 0, & \text{otherwise} \end{cases}, \tag{4.39}$$

where

$$\mathcal{A}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} = (x^{(p)} - u^{(c)})^T \left(\text{diag}(x^{(s)'})\right)^{-2} (x^{(p)} - u^{(c)}), \tag{4.40}$$

$$\mathcal{B}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} = (x^{(p)} - u^{(c)})^T \left[2\left(\text{diag}(x^{(s)'})\right)^{-2} u^{(c)} + \hbar_{\boldsymbol{x}'}\right], \tag{4.41}$$

$$C^{(c)}_{\boldsymbol{x}'} = (u^{(c)})^T \left[\left(\text{diag}(x^{(s)'})\right)^{-2} u^{(c)} + \hbar_{\boldsymbol{x}'}\right] + \mathcal{E}_{\boldsymbol{x}'}, \tag{4.42}$$

$$\hbar_{\boldsymbol{x}'} = -2\frac{x^{(p)'}}{(x^{(s)'} \cdot x^{(s)'})}, \; \mathcal{E}_{\boldsymbol{x}'} = \left\| x^{(p)'} / x^{(s)'} \right\|_2^2 - 1, \tag{4.43}$$

and $u^{(c)}$ is the position of camera c, with multiplication/division of two vectors of the same dimension to be understood as point-wise multiplication/division. The derivation of the shadow region indicator function is given in Appendix A.

In addition, using quadric projection [380, pp. 201], the relationship between the estimated bounding box $\Upsilon^{(c)}(\boldsymbol{x})$ and measured bounding box $z^{(c)}$ captured in the mea-

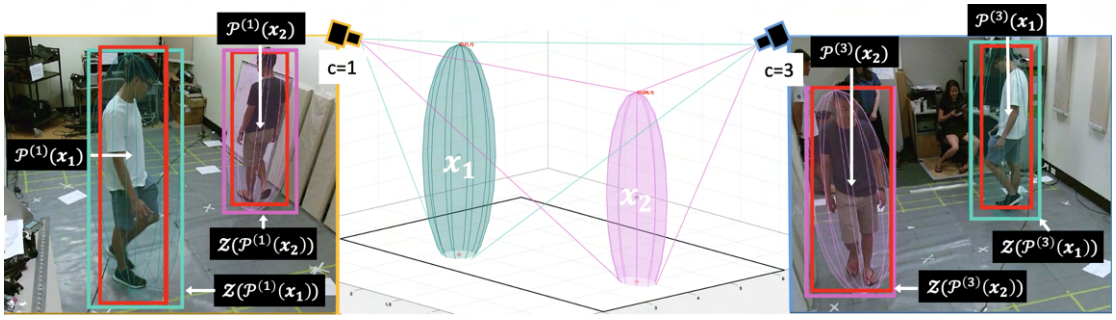


Figure 4.5: The projections $\mathcal{P}^{(c)}$ of two quadrics (in cyan and pink) onto two image views ($c = 1,3$) result in 2D conics. The transformation $\mathcal{Z}$ yields the corresponding estimated bounding boxes (in cyan and pink). The estimated bounding box and the measured bounding box (in red) from monocular detector formulate the measurement likelihood (4.27).

surement likelihood (4.27), has the following closed form

$$\Upsilon^{(c)}(\boldsymbol{x}) \triangleq \mathcal{Z}(\mathcal{P}^{(c)}(\boldsymbol{x})), \tag{4.44}$$

where

$$\mathcal{P}^{(c)}(\boldsymbol{x}) = \left(\mathrm{P}^{(c)}_{3\times4}\begin{bmatrix}(\mathrm{diag}(x^{(s)}))^{-2} & \hbar_{\boldsymbol{x}}/2 \\ \hbar_{\boldsymbol{x}}^T/2 & \mathcal{E}_{\boldsymbol{x}}\end{bmatrix}^{-1}(\mathrm{P}^{(c)}_{3\times4})^T\right)^{-1}, \tag{4.45}$$

$$\mathcal{Z}\left(\begin{bmatrix}\mathrm{A} & r \\ r^T & e\end{bmatrix}\right) = \begin{bmatrix}-Q\mathrm{D}^{-1}Q^T r \\ 2\nu\left\|[1,0]Q\mathrm{D}^{-0.5}\right\|_2 \\ 2\nu\left\|[0,1]Q\mathrm{D}^{-0.5}\right\|_2\end{bmatrix}, \tag{4.46}$$

$$\nu = (r^T Q\mathrm{D}^{-1}Q^T r - e)^{0.5}, \tag{4.47}$$

Q is a matrix containing the eigenvectors of A, and D is a diagonal matrix of the eigenvalues of A. Given the camera matrices $\mathrm{P}^{(1)}_{3\times4}, ..., P^{(C)}_{3\times4}$, $\mathcal{P}^{(c)}(\cdot)$ is a matrix-to-matrix projection that transforms the quadric into a conic on each image of camera c [380, pp. 201]. $\mathcal{Z}(\cdot)$ is a matrix-to-vector transformation that transforms the conic into a 4D bounding box (in the same format as $z^{(c)}$). The illustration of the overall transformation (4.44) is depicted in Fig. 4.5, and the derivation is given in Appendix B.

The Poisson false alarms intensity for camera c is $\kappa^{(c)} \triangleq \lambda_p \mathcal{U}(\cdot)$, where λ_p is the false-positive (clutter) rate, and $\mathcal{U}(\cdot)$ is a uniform distribution on the measurement space $\mathbb{Z}^{(c)}$. In many visual tracking cases, this value can either be estimated offline or manually tuned. The false alarm intensity can be estimated by the Cardinalized Probability Hypothesis Density (CPHD) clutter estimator [381]. In this work, we bootstrap the CPHD clutter intensity estimator output to the tracker [359].

4.3.2 MV-GLMB-OC Filter Implementation

The number of components of the GLMB filtering density grows super-exponentially over time. To maintain tractability in GLMB filter implementations, truncating insignificant components has been proven to minimize the L_1 approximation error [22]. This truncation strategy can be formulated as an NP-hard multi-dimensional assignment problem [22]. Nonetheless, it can be solved by exploiting certain structural properties, and suitable adaptation of 2D assignment solutions such as Murty's or Auction [22].

The MV-GLMB-OC recursion described in Section 4.2.6, can be directly implemented with separate prediction and update, i.e., by computing a truncated version of the prediction (4.28) and the corresponding detection probabilities $\{P_{D,+}^{(\xi,I_+)}(\ell) : \ell \in I_+, (\xi, I_+) \in \Xi \times \mathcal{F}(\mathbb{L}_+)\}$, then using these to compute a truncated version of the update (4.34). This strategy requires keeping a significant portion of the predicted components that would end up as updated components with negligible weights, thereby wasting computations in solving a large number of 2D assignment problems. Thus, this approach is inefficient and becomes infeasible for systems with many sensors [22].

In this work, we exploit an efficient GLMB truncation strategy that has a linear complexity in the sum of the measurements across all sensors [22]. This approach bypasses the prediction truncation, and returns the significant components of the next GLMB filtering density (4.34) by sampling from a discrete probability distribution proportional to the weights of the components [22]. This means GLMB components with higher weights are more likely to be selected than those with lower weights. For the MV-GLMB-OC recursion, this discrete probability distribution $\vartheta(\cdot; P_{D,+})$ of the GLMB components, is determined by the detection probabilities $P_{D,+} \triangleq \{P_{D,+}^{(\xi,I_+)}(\ell) : \ell \in I_+, (\xi, I_+) \in \Xi \times \mathcal{F}(\mathbb{L}_+)\}$ (and other multi-object system parameters, which are suppressed for clarity) [22]. However, since truncation of the prediction (4.28) has been bypassed, the predicted components $\{(\xi, I_+) \in \Xi \times \mathcal{F}(\mathbb{L}_+)\}$ and their corresponding detection probabilities are not available. Nonetheless, importance sampling can be used to generate weighted samples of $\vartheta(\cdot; P_{D,+})$ by sampling from $\vartheta(\cdot; \widehat{P}_{D,+})$, where $\widehat{P}_{D,+} \triangleq \{P_{D,+}^{(\xi,I \uplus \mathbb{B}_+)}(\ell) : \ell \in I \uplus \mathbb{B}_+, (\xi, I) \in \Xi \times \mathcal{F}(\mathbb{L})\}$, and then re-weight the resulting samples accordingly [232]. Note that the detection probabilities $\widehat{P}_{D,+}$ can be readily computed from the components of the (truncated) current GLMB filtering density $\{(\omega^{(I,\xi)}, p^{(\xi)}) : (I, \xi) \in \mathcal{F}(\mathbb{L}) \times \Xi\}$. Moreover, $P_{D,+}^{(\xi,I \uplus \mathbb{B}_+)} \preceq P_{D,+}^{(\xi,I_+)}$, for any $I_+ \subseteq I \uplus \mathbb{B}_+$, it follows from [21] that $\vartheta(\cdot; \widehat{P}_{D,+})$ is more diffused than $\vartheta(\cdot; P_{D,+})$, i.e., the support of $\vartheta(\cdot; \widehat{P}_{D,+})$ contains the support of $\vartheta(\cdot; P_{D,+})$.

The MS-GLMB and MV-GLMB-OC recursions are presented in Algorithm 4.1 and 4.2 respectively. Observe that the main difference is the additional computation of the detection probabilities prior to and re-weighting after the Gibbs sampling step in the MV-GLMB-OC filter.

In this work, the object's birth density $f_{B,+}(\cdot, \ell)$, single-object transition (4.35) and likelihood (4.27) are all Gaussians. Standard Kalman prediction and Unscented Kalman update are used to evaluate the single-object filtering density $p_+^{(\xi_+)}$, which results in a

Gaussian.

Algorithm 4.1 MS-GLMB Filter [22]

Global Input: $\{(r_{B,+}(\ell), f_{B,+}(\cdot,\ell))\}_{\ell\in\mathbb{B}_+}, \boldsymbol{f}_{S,+}(\cdot|\cdot), P_S(\cdot)$
Global Input: κ, P_D, g
Input: $\boldsymbol{\pi} \triangleq \left\{\left(\omega^{(I,\xi)}, p^{(\xi)}\right) : (I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi\right\}$
Output: $\boldsymbol{\pi}_+ \triangleq \left\{\left(\omega_+^{(I_+,\xi_+)}, p_+^{(\xi_+)}\right) : (I_+,\xi_+)\in\mathcal{F}(\mathbb{L}_+)\times\Xi_+\right\}$

for $(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi$
  Construct stationary distribution from inputs
  Run Gibbs sampler to obtain samples γ_+ [22, Algorithm 3]
  Use samples γ_+ to compute $\boldsymbol{\pi}_+$
end for
Extract labeled state estimates

Algorithm 4.2 MV-GLMB-OC Filter

Global Input: $\{(r_{B,+}(\ell), f_{B,+}(\cdot,\ell))\}_{\ell\in\mathbb{B}_+}, \boldsymbol{f}_{S,+}(\cdot|\cdot), P_S(\cdot)$
Global Input: κ, P_D, g
Input: $\boldsymbol{\pi} \triangleq \left\{\left(\omega^{(I,\xi)}, p^{(\xi)}\right) : (I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi\right\}$
Output: $\boldsymbol{\pi}_+ \triangleq \left\{\left(\omega_+^{(I_+,\xi_+)}, p_+^{(\xi_+)}\right) : (I_+,\xi_+)\in\mathcal{F}(\mathbb{L}_+)\times\Xi_+\right\}$

for $(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi$
  Compute occlusion-based probability of detection
    $\{P_{D,+}^{(\xi,I\uplus\mathbb{B}_+)}(\ell) : \ell\in I\uplus\mathbb{B}_+\}$ via (4.33)
  Construct stationary distribution from inputs and
    $\{P_{D,+}^{(\xi,I\uplus\mathbb{B}_+)}(\ell) : \ell\in I\uplus\mathbb{B}_+\}$
  Run Gibbs sampler to obtain samples γ_+ [22, Algorithm 3]
  Update occlusion-based probability of detection
    $\{P_{D,+}^{(\xi,\mathcal{L}(\gamma_+))}(\ell) : \ell\in\mathcal{L}(\gamma_+)\}$, via (4.33)
  Use samples γ_+, $\{P_{D,+}^{(\xi,\mathcal{L}(\gamma_+))}(\ell) : \ell\in\mathcal{L}(\gamma_+)\}$ to compute $\boldsymbol{\pi}_+$
end for
Extract labeled state estimates

4.4 Experiments

This section demonstrates the three main advantages of the proposed MV-GLMB-OC approach. The first is the capability to produce 3D object trajectories using independent monocular detections from multiple views, where each object is represented as a 3D ellipsoid of unknown location and extent (Section 4.4.2). The second is the amenability

for uninterrupted/seamless operation in the event that cameras are added, removed or repositioned on the fly (Section 4.4.3). The third is the flexibility of not confining objects to the ground plane, which is demonstrated by tracking people jumping and falling (Section 4.4.4). The effectiveness of the proposed occlusion model is also studied, by comparing the tracking performance of the MV-GLMB-OC against that of the standard MS-GLMB filter.

We first focus our demonstrations on the latest WILDTRACKS dataset[2], which involves seven-cameras at 1920×1080 resolution with overlapping views. The WILDTRACKS dataset is also supplied with calibrated intrinsic and extrinsic camera parameters, along with 3D ground plane annotations although these are restricted to the ground plane. WILDTRACKS was initially introduced to address various perceived shortcomings in older multi-view datasets, the arguments for which were originally presented in [57] and are summarized as follows. The DukeMTMC dataset [375] is essentially non-overlapping in views and is now no longer available. The PETS 2009 S2.L1 dataset [376] has supposed inconsistencies when projecting 3D points across the views. The EPFL, SALSA and Campus datasets [174, 197, 377] involve a relatively small number of people, and are relatively sparse in terms of person density, but do not provide 3D annotations. In addition, the EPFL-RLC dataset [55] only provides annotations for a small subset of the last 300 of 8000 frames. For the same reasons that the authors of WILDTRACKS were motivated to introduce their new dataset, the older multi-view datasets superseded by WILDTRACKS are not suitable for evaluating the MV-GLMB-OC filter in the 3D world frame.

In the context of demonstrating the MV-GLMB-OC approach however, the WILDTRACKS dataset is not suitable for evaluating tracking performance in full 3D, i.e., changes in all 3 x, y, z-coordinates. While WILDTRACKS provides 3D annotations, these are restricted to the ground plane. Moreover the annotations are for centroids only, and do not capture the extent (in terms of length, width and height) of objects in the world coordinates. In our performance comparisons, the outputs of the proposed MV-GLMB-OC filter on WILDTRACKS are limited to the estimated centroids projected onto the ground plane. To demonstrate the full capabilities of MV-GLMB-OC, it is critical to have annotations of the 3D centroids and their 3D extent, along with the ground truths for each of the camera locations. Consequently we introduce a new Curtin Multi-Camera (CMC) dataset which meets these requirements.

The new CMC dataset is a four-camera 1920x1024 resolution dataset recorded at 4fps in a room with dimensions 7.67m x 3.41m x 2.7m. The CMC dataset has 5 different sequences with varying levels of person density and occlusion: CMC1 has a maximum of 3 people and virtually no occlusion; CMC2 has a maximum of 10 people with some occlusion; CMC3 has a maximum of 15 people with significant occlusion; while CMC4 and CMC5 involve people jumping and falling with a maximum of 3 and 7

[2] https://www.epfl.ch/labs/cvlab/data/data-wildtrack/

people respectively. CMC1 and CMC4 have low person density and are intended for basic testing, while CMC2, CMC3 and CMC5 have higher person density and significant visual occlusions across multiple overlapping cameras, and are intended to highlight performance differences. The convention for the world coordinate frame is illustrated in Fig. 4.6. The origin is at the lower corner and the ground plane corresponds to the x-y plane i.e., $z = 0$. In every sequence, each person enters the tracking area at $(2.03\text{m}, 0.71\text{m})$ with an average height of 1.7m. The dataset is also supplied with camera locations and parameters, along with annotations for 3D centroid and extent. The 2D monocular annotation for bounding boxes is carried out with the MATLAB Image Labeler Tool, and the world coordinates are obtained by averaging the homographic projection of the feet coordinates from each view. The actual height and width of each person is used for the annotation.

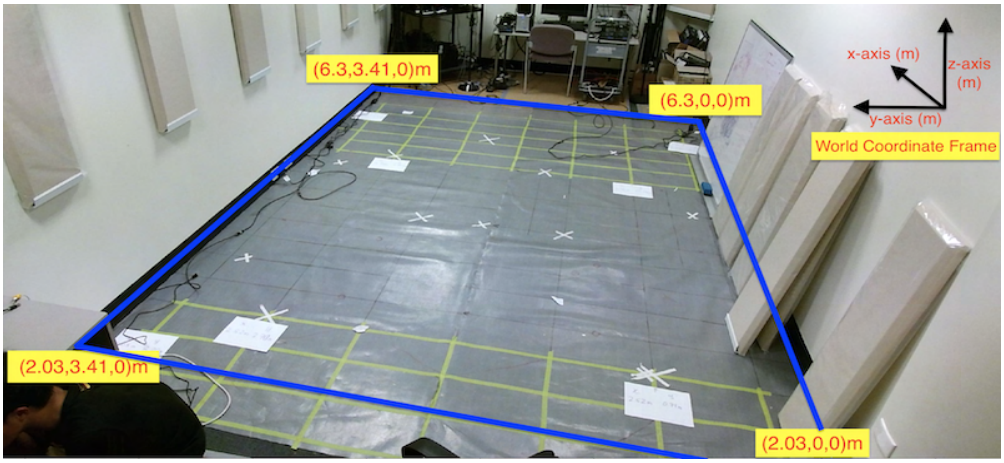


Figure 4.6: Layout for CMC dataset: The blue line denotes the boundary of the tracking area. The yellow boxes denote the coordinates of the boundary in (x,y,z) axes. The 4 cameras are positioned (in sequence) at the top 4 corners of the room.

A common setting for object survival and detection model parameters is used in both evaluations on the WILDTRACKS and CMC datasets. Specifically: the survival probability $P_S(\boldsymbol{x})$ given by (4.38), is parameterized by the control parameter $\tau = 0.5$ and the scene mask $b(\cdot)$ with a margin of 0.3m inside the border of the tracking area; the detection probability, given in Section 4.2.5 is parameterized by $P_D^{(c)}(\boldsymbol{x}) = 0.9$ and $\mu = 0.1$. For all cameras, the observed bounding box model is described in (4.27), with position noise parameterized by $w_p^{(c)} = [400, 400]^T$, and the extent noise parameterized by $w_e^{(c)} = [0.01, 0.0025]^T$ (on the logarithms of the half-lengths of the principal axes).

4.4.1 Performance Evaluation Criteria

Standard Evaluation on 3D Position Estimates

The performance of various combinations of detectors and trackers are evaluated using the CLEAR MOT devkit provided in [382]. For computing CLEAR MOT, we adhere to

the convention of using the Euclidean distance (L_2-norm) on the estimated 3D centroid with a threshold of 1m.

For MOT, the following performance indicators are reported: Multiple Object Tracking Accuracy (MOTA) which penalizes normalized false negatives (FNs), false positives (FPs) and identity switches (IDs) between consecutive frames; Multiple Object Tracking Precision (MOTP) which accounts for the overall dissimilarity between all true positives and the corresponding ground truth objects [383]; Mostly Tracked (MT), Partially Tracked (PT), Mostly Lost (MT) which indicate how much of the trajectory is retained or lost by the tracker; Fragmentations (FM) which account for interrupted tracks based on ground truth trajectories; Identity Precision (IDP), Identity Recall (IDR) and F_1 score (IDF1) which are analogous to the standard *precision*, standard *recall* and F_1 score with identifications (tracks) [375]. For reference, we also provide performance indicators on the bounding box detections, where we set the threshold at 0.5 and report: Multiple Object Detection Accuracy (MODA) which accounts for misdetections and false alarms; Multiple Object Detection Precision (MODP) which accounts for the spatial overlap information between the bounding boxes; *precision* which is the measure of exactness; and *recall* which is the measure of quality.

We note that CLEAR MOT is traditionally calculated over the entire scenario window, and thus the tracking performance is reported after the entire data stream has been processed. To evaluate the live or online tracking performance over time, we employ the Optimal Sub-Pattern Assignment (OSPA$^{(2)}$) distance between two sets of tracks [51]. This distance is based on the OSPA metric that captures both localization and cardinality errors between two finite sets of a metric space with a suitable base-distance between objects (e.g., the Euclidean distance) [50]. The OSPA$^{(2)}$ metric is defined as the OSPA distance between two sets of tracks over a time window. Details for OSPA and OSPA$^{(2)}$ metrics are given in Appendix C. By design, OSPA$^{(2)}$ captures both localization and cardinality errors between the set of true and estimated tracks, and penalizes switched tracks or label changes [51]. The resultant metric carries the interpretation of a time-averaged per-track error. In our evaluation of the position estimate in real world coordinates, we use a 3D Euclidean base-distance for OSPA$^{(2)}$ with order parameter 1 and cutoff parameter 1m. Performance evaluation for live or online tracking is given by plotting the error over a sliding window of length $L_w = 10$ frames, while overall performance is captured in a single number by calculating the error over the entire scenario window.

GIoU Based Evaluation on 3D Position with Extent

As the proposed MV-GLMB-OC filter outputs 3D estimates of the object centroid and extent, we extend the performance evaluations to capture the joint error in the centroid and extent. This is achieved by employing an alternative base-distance between two objects, in this case a 3D generalized intersection over union (GIoU), which extends

the commonly used IoU to non-overlapping bounding boxes [384]. The details for the IoU and GIoU metrics are given in Appendix D. It is important to note that if there is no overlap between the ground truth and estimated shape, the IoU distance is zero regardless of their separation, whereas the GIoU distance captures the extent of the error while retaining the metric property [384]. We present evaluations of the estimated centroid with extent for CLEAR MOT (using a GIoU base-distance with a threshold of 0.5) and OSPA$^{(2)}$ metric with GIoU base-distance (and with unit order and cut-off parameters). We refer the reader to [385] for the rationale and discussions on the use of OSPA$^{(2)}$-GIoU for performance evaluation.

4.4.2 WILDTRACKS Dataset

We test MV-GLMB-OC against the latest multi-camera detector (Deep-Occlusion) [56] coupled with the k-shortest-path (KSP) algorithm [196] and *ptrack* as shown in [57] (Deep-Occlusion+KSP+ptrack). KSP is an optimization algorithm that finds the most likely sequence of ground plane occupancies (trajectories) given by the multi-camera detector, and *ptrack* described in [386] improves and smooths over tracks by learning motion patterns. As a baseline comparison, we employ the Deep-Occlusion multi-camera detector combined with single-view GLMB (Deep-Occlusion+GLMB). Since WILDTRACKS provides annotations in real-world coordinates but restricted to the ground plane, tracking is performed in real-world coordinates but also restricted to the ground plane. To further explore the performance of MV-GLMB-OC, we also run experiments using monocular detections from each of the cameras. For the detectors, we use the monocular backbone of the Deep-Occlusion detector i.e., VGG16-net trained using Faster-RCNN [160], and separately with the newer YOLOv3 [169], to produce separate monocular detections for input to MV-GLMB-OC. Since WILDTRACKS does not supply the camera positions required for our proposed occlusion model, we reconstruct the camera positions from the given camera parameters. We note that KSP and/or ptrack is an offline or batch method, while GLMB is online or recursive, and provides estimates on the fly.

Model Parameters

The birth density is adaptive/measurement-driven (see Section F in [387]) with $r_{B,+}(\ell) = 0.001$ and $f_{B,+}(x,\ell) = \mathcal{N}(x; m^{(\ell)}_{B,+}, 0.1^2\mathrm{I}_9)$ where $m^{(\ell)}_{B,+}$ is obtained via clustering (e.g., k-means). The single-object transition is as described in (4.35) with position noise and extent (in logarithm) noise parameterized by:

$$\upsilon^{(p)} = [0.0016, 0.0016, 0.0016]^T,$$
$$\upsilon^{(s)} = [0.0036, 0.0036, 0.0004]^T.$$

Discussion

Table 4.1 shows the CLEAR MOT and OSPA$^{(2)}$ benchmarks for MV-GLMB-OC (with occlusion modeling) and MS-GLMB (without occlusion modeling) with two different detectors YOLOv3 and Faster-RCNN(VGG16). Results for Deep-Occlusion+KSP+ptrack being the reference, are reproduced directly from the original paper [57]. The results indicate that the two trackers based on multi-camera detections, i.e., Deep-Occlusion+KSP+ptrack and Deep-Occlusion+GLMB, have very similar tracking performance in terms of MOTA/MOTP and OSPA$^{(2)}$. Importantly, closer examination of the tracking results based on multiple monocular detections indicates that performance is significantly improved with the addition of the occlusion model. This can be seen from the relative changes in the MOTA/MOTP and OSPA$^{(2)}$. Several observations can also be drawn from comparing the multi-camera detector with batch processing method (Deep-Occlusion+ KSP+ptrack), and the related monocular detector with online processing (Faster-RCNN(VGG16)+MV-GLMB-OC). While the MOTP improves due to the use of multiple monocular detectors, the MOTA degrades due to the use of an online method which is unable to correct past estimates. This is corroborated by the overall OSPA$^{(2)}$ value which improves slightly from Deep-Occlusion+KSP+ptrack to Faster-RCNN(VGG16)+MV-GLMB-OC. Surprisingly, the results based on YOLOv3 are better across the board than that for Faster-RCNN(VGG16), even though YOLO v3 is more efficient than Faster-RCNN(VGG16). For reference, the CLEAR evaluations for the detectors used in the experiment are presented in Appendix E, from which it is noted that the monocular detections are generally much poorer than the multi-camera detections due to severe occlusions.

Table 4.1: WILDTRACKS Performance Benchmarks for 3D Position Estimates (restricted to the ground plane)

Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	74.3%	**85.0%**	**75.9%**	**136**	111	37	**424**	1333	104	86	69.7%	**73.2%**	**0.69m**
YOLOv3+MS-GLMB	74.2%	79.0%	69.9%	116	85	83	841	1951	139	105	61.9%	68.3%	0.81m
Faster-RCNN(VGG16)+MV-GLMB-OC	76.5%	84.5%	70.0%	119	118	47	545	1621	104	81	65.3%	71.9%	0.72m
Faster-RCNN(VGG16)+MS-GLMB	75.5%	76.8%	74.3%	98	104	82	1114	1716	179	116	61.5%	65.8%	0.88m
Deep-Occlusion+GLMB	72.5%	82.7%	72.2%	160	86	39	960	**990**	107	**64**	70.1%	63.1%	0.73m
Deep-Occlusion+KSP+ptrack	**78.4%**	84.4%	73.1%	72	**74**	**25**	2007	5830	**103**	95	**72.2%**	60.3%	0.75m

CLEAR MOT scores and OSPA$^{(2)}$ distance are calculated on standard position estimates (↑ means higher is better while ↓ means lower is better). Three different detectors are considered -Deep-Occlusion (multiocular), Faster-RCNN(VGG16) *(monocular) and YOLOv3 (monocular). Three types of trackers are considered -KSP+ptrack or GLMB (single-sensor), MV-GLMB-OC (multi-view with occlusion model) and MS-GLMB (multi-sensor without occlusion model).*

4.4.3 Curtin Multi-Camera Dataset 1, 2 and 3

This subsection focuses on scenarios with people walking in order of increasing difficulty, i.e., CMC1-CMC3. Similar to the WILDTRACKS evaluation, we evaluate our method based on 2 monocular detectors, namely Faster-RCNN(VGG16) and YOLOv3. For each sequence, the effect of the occlusion model is studied by comparing the proposed MV-GLMB-OC with the standard MS-GLMB filter.

Model Parameters

Unlike WILDTRACKS where objects enter the scene from anywhere at the boundary, in CMC we know the location of objects entering the scene. Hence, we specify the birth parameters as $r_{B,+}(\ell) = 0.001$ and $f_{B,+}(x,\ell) = \mathcal{N}(x; m_{B,+}, 0.1^2\mathrm{I}_9)$ where

$$m_{B,+} = [2.03\ 0\ 0.71\ 0\ 0.825\ 0\ -1.2\ -1.2\ -0.18]^T.$$

We use the single-object transition density (4.35) with position noise and extent (in logarithm) noise parameterized by:

$$\upsilon^{(p)} = [0.0012, 0.0012, 0.0012]^T,$$
$$\upsilon^{(s)} = [0.0036, 0.0036, 0.0004]^T.$$

Effectiveness of Occlusion Model

Table 4.2 shows the CLEAR MOT and OSPA$^{(2)}$ benchmarks with a Euclidean base-distance, for the estimated 3D centroids only. Table 4.3 shows the CLEAR MOT and OSPA$^{(2)}$ benchmarks with a 3D GIoU base-distance, for the estimated 3D centroids and extent. Both tables compare the tracking performance with and without and occlusion model, i.e., MV-GLMB-OC and MS-GLMB respectively. The asterisked entry denotes the multi-camera reconfiguration case which is discussed later on. All results are presented for two different detectors YOLOv3 and Faster-RCNN(VGG16).

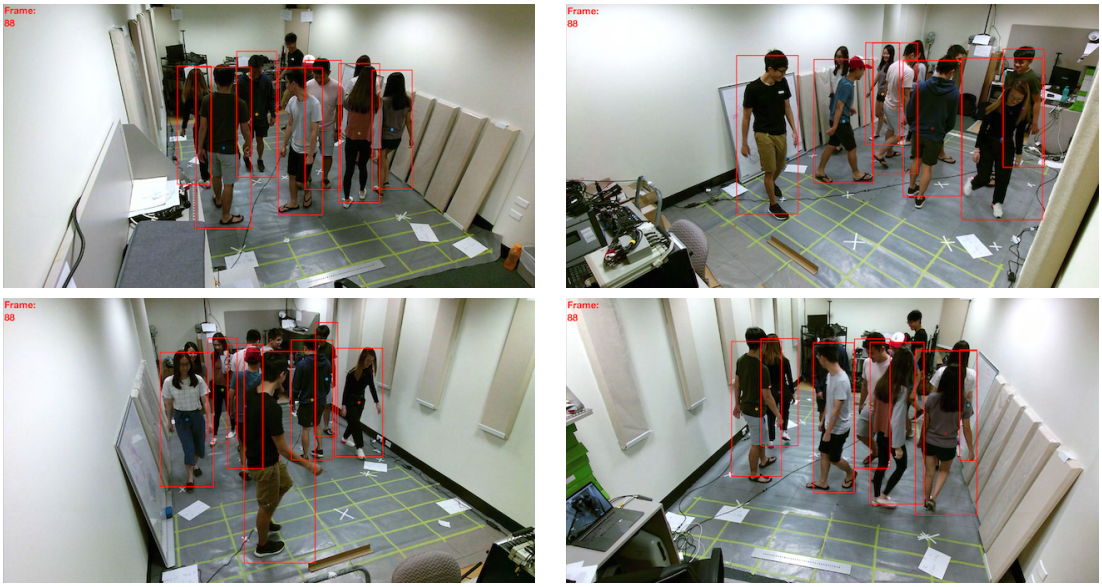
We focus our initial examination on the non-asterisked entries in Tables 4.2 and 4.3. This corresponds to the case where all cameras are operational. For the sparse scenario CMC1, both MV-GLMB-OC and MS-GLMB on either detectors achieved a close to perfect CLEAR MOT scores in MOTA and MOTP. Some of the flagged FPs are caused by track initiation/termination mismatches with the ground truths (annotations). The OSPA$^{(2)}$ values are relatively low due to the sparsity of the scenario.

For the medium scenario CMC2, Fig. 4.7 shows a screenshot of the detections and the MV-GLMB-OC estimates. In this case, MV-GLMB-OC on both detectors managed to maintain consistent tracks and accurate estimates overall. The CLEAR MOT benchmarks for CMC2 show high MOTA and MOTP but with some FNs and FPs. We observe an improvement in performance for MV-GLMB-OC over MS-GLMB, and on both de-

tectors due to the inclusion of occlusion modeling. The improvement in performance due to occlusion modeling is also reflected in the OSPA$^{(2)}$.

For the dense scenario CMC3, MV-GLMB-OC on both detectors managed to achieve acceptable MOTA/MOTP scores, but is penalized with high FPs, FNs, IDs and FMs. This outcome occurs even with the proposed occlusion model, as the algorithm fails when a person is totally occluded in all views. An example of this occurrence is illustrated in Fig. 4.8, where the red bounding boxes denote detections, while the yellow bounding boxes indicate people who are undetected in all views. Such an event could cause track termination/switching and is reflected in the performance evaluation. It is evident from Tables 4.2 and 4.3 that the tracking performance improves considerably with the occlusion model. Examination of the OSPA$^{(2)}$ error leads to a similar conclusion.

Overall, YOLOv3+MV-GLMB-OC performs slightly better than Faster-RCNN (VGG16)+MV-GLMB-OC due to better detections. The tracking performance of the proposed MV-GLMB-OC filter generally degrades as the number of people in the scene increases, since the visual occlusions become more frequent and more difficult to resolve. The results of this study on the proposed occlusion model suggest that without proper modeling of the probability of detection, the algorithm fails to maintain tracks, resulting in poorer tracking results. The CLEAR evaluation for the monocular detectors used are given in Appendix E.



(a)

(b)

Figure 4.7: CMC2 Camera 1 to 4 (top left to bottom right): (a) YOLOv3 detections and (b) MV-GLMB-OC estimates.

Figure 4.8: CMC3 Camera 1 to 4 (top left to bottom right): YOLOv3 detections (red bounding boxes) and people that are occluded in all four cameras (yellow bounding boxes).

Table 4.2: CMC1,2,3 Performance Benchmarks for 3D Position Estimates

CMC1 (Maximum/Average 3 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**99.7%**	**99.4%**	**100%**	**3**	**0**	**0**	**4**	**0**	**0**	**0**	**99.4%**	**91.8%**	**0.13m**
YOLOv3+MV-GLMB-OC*	98.9%	97.9%	99.8%	**3**	**0**	**0**	14	1	**0**	**0**	97.7%	90.5%	0.16m
YOLOv3+MS-GLMB	95.9%	92.3%	99.8%	**3**	**0**	**0**	55	1	1	**0**	91.3%	91.4%	0.34m
Faster-RCNN(VGG16)+MV-GLMB-OC	99.5%	99.1%	**100%**	**3**	**0**	**0**	6	**0**	**0**	**0**	99.1%	**91.8%**	**0.13m**
Faster-RCNN(VGG16)+MV-GLMB-OC*	95.5%	91.4%	**100%**	**3**	**0**	**0**	62	**0**	1	**0**	90.4%	90.5%	0.14m
Faster-RCNN(VGG16)+MS-GLMB	99.6%	99.2%	**100%**	**3**	**0**	**0**	5	**0**	**0**	**0**	99.2%	91.4%	0.36m
CMC2 (Maximum/Average 10 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**91.0%**	**91.1%**	**91.3%**	**10**	**0**	**0**	16	**11**	**9**	**2**	**98.3%**	81.7%	**0.30m**
YOLOv3+MV-GLMB-OC*	90.1%	90.2%	90.0%	**10**	**0**	**0**	38	29	11	7	96.2%	78.9%	0.34m
YOLOv3+MS-GLMB	67.7%	79.9%	58.9%	4	6	**0**	8	550	34	30	71.5%	74.4%	0.70m
Faster-RCNN(VGG16)+MV-GLMB-OC	90.6%	90.5%	90.9%	**10**	**0**	**0**	50	37	**9**	5	95.4%	**83.7%**	0.35m
Faster-RCNN(VGG16)+MV-GLMB-OC*	86.2%	85.5%	87.5%	**10**	**0**	**0**	120	60	25	13	90.1%	79.8%	0.48m
Faster-RCNN(VGG16)+MS-GLMB	75.3%	81.9%	69.7%	7	3	**0**	**7**	316	23	19	83.3%	80.4%	0.58m
CMC3 (Maximum/Average 15 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**77.9%**	**79.7%**	**76.1%**	**13**	**2**	**0**	63	**191**	**44**	33	**89.5%**	**76.4%**	**0.51m**
YOLOv3+MV-GLMB-OC*	72.1%	77.9%	67.2%	11	4	**0**	47	437	51	37	81.1%	72.3%	0.61m
YOLOv3+MS-GLMB	50.5%	69.9%	39.5%	0	15	**0**	5	1234	54	51	54.2%	67.8%	0.83m
Faster-RCNN(VGG16)+MV-GLMB-OC	71.7%	74.9%	68.8%	12	3	**0**	71	303	**44**	**32**	85.2%	73.5%	0.61m
Faster-RCNN(VGG16)+MV-GLMB-OC*	67.7%	72.1%	63.8%	10	5	**0**	92	419	59	44	79.8%	68.0%	0.70m
Faster-RCNN(VGG16)+MS-GLMB	54.3%	73.2%	43.1%	0	15	**0**	**3**	1165	53	55	56.8%	65.9%	0.81m

CLEAR MOT scores and OSPA$^{(2)}$ distance are calculated on standard position estimates (↑ means higher is better while ↓ means lower is better). Two different detectors are considered - Faster-RCNN(VGG16) (monocular) and YOLOv3 (monocular). Two types of trackers are considered - MV-GLMB-OC (multi-view with occlusion model) and MS-GLMB (multi-sensor without occlusion model). The asterisk (*) indicates the multi-camera reconfiguration experiment.

Table 4.3: CMC1,2,3 Performance Benchmarks for 3D Centroid with Extent Estimates

CMC1 (Maximum/Average 3 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**99.7%**	**99.4%**	**100%**	**3**	**0**	**0**	**4**	**0**	**0**	**0**	**99.4%**	**67.8%**	**0.20**
YOLOv3+MV-GLMB-OC*	98.9%	97.9%	99.8%	**3**	**0**	**0**	14	1	**0**	**0**	97.7%	66.7%	**0.20**
YOLOv3+MS-GLMB	95.9%	92.3%	99.8%	**3**	**0**	**0**	55	1	1	**0**	91.3%	67.5%	0.40
Faster-RCNN(VGG16)+MV-GLMB-OC	99.5%	99.1%	**100%**	**3**	**0**	**0**	6	**0**	**0**	**0**	99.1%	67.5%	**0.20**
Faster-RCNN(VGG16)+MV-GLMB-OC*	95.5%	91.4%	**100%**	**3**	**0**	**0**	62	**0**	1	**0**	90.4%	67.2%	**0.20**
Faster-RCNN(VGG16)+MS-GLMB	99.6%	99.2%	**100%**	**3**	**0**	**0**	5	**0**	**0**	**0**	99.2%	66.9%	0.40
CMC2 (Maximum/Average 10 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**87.9%**	**87.5%**	**87.7%**	**10**	**0**	**0**	**19**	**14**	**8**	**2**	**98.0%**	**62.3%**	**0.32**
YOLOv3+MV-GLMB-OC*	87.3%	87.1%	87.5%	**10**	**0**	**0**	53	44	14	12	94.7%	57.0%	0.38
YOLOv3+MS-GLMB	59.4%	69.9%	51.7%	4	6	**0**	21	563	30	31	70.4%	55.7%	0.62
Faster-RCNN(VGG16)+MV-GLMB-OC	86.7%	86.5%	87.0%	**10**	**0**	**0**	68	55	10	8	93.6%	60.9%	0.34
Faster-RCNN(VGG16)+MV-GLMB-OC*	81.3%	80.2%	82.5%	**10**	**0**	**0**	127	67	33	15	89.1%	55.0%	0.45
Faster-RCNN(VGG16)+MS-GLMB	68.6%	74.6%	63.5%	7	3	**0**	23	332	23	21	81.8%	57.1%	0.52
CMC3 (Maximum/Average 15 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**70.7%**	**72.3%**	**69.1%**	**14**	**1**	**0**	94	**222**	**45**	**37**	**87.2%**	**52.8%**	**0.53**
YOLOv3+MV-GLMB-OC*	60.8%	65.7%	56.6%	9	6	**0**	91	481	66	56	77.4%	46.4%	0.60
YOLOv3+MS-GLMB	41.4%	57.3%	32.4%	0	15	**0**	**10**	1239	64	60	53.5%	46.7%	0.76
Faster-RCNN(VGG16)+MV-GLMB-OC	63.7%	66.6%	61.1%	12	3	**0**	97	329	63	41	82.7%	**52.8%**	0.58
Faster-RCNN(VGG16)+MV-GLMB-OC*	57.3%	61.0%	54.0%	10	5	**0**	133	460	78	60	76.3%	47.9%	0.66
Faster-RCNN(VGG16)+MS-GLMB	45.7%	61.7%	36.3%	0	15	**0**	13	1175	61	67	55.8%	46.6%	0.75

CLEAR MOT scores and OSPA$^{(2)}$ distance are calculated with a 3D GIoU base-distance for estimates of 3D centroid with extent (↑ means higher is better while ↓ means lower is better). Two different detectors are considered - Faster-RCNN(VGG16) (monocular) and YOLOv3 (monocular). Two types of trackers are considered - MV-GLMB-OC (multi-view with occlusion model) and MS-GLMB (multi-sensor without occlusion model). The asterisk (*) indicates the multi-camera reconfiguration experiment.

Multi-Camera Reconfiguration

The MV-GLMB-OC approach requires only a one-off training on each monocular detector, and hence can operate without retraining and without interruption, in the event that cameras are added, removed or repositioned on the fly. To demonstrate this capability, we design a multi-camera reconfiguration experiment. At the start of the sequence, all four cameras are operational. Later, one camera is taken offline to mimic a camera failure. Subsequently, two cameras are taken offline to mimic a more severe camera failure. After this, the two previously offline cameras are made operational, while the previously operational cameras are taken offline, which mimics the event that the two operational cameras are moved to different locations. We benchmark the multi-camera reconfiguration experiment against the ideal case when all cameras are operational.

Results for the experiments on multi-camera reconfiguration are denoted with an asterisk in Tables 4.2 and 4.3. The reported CLEAR MOT scores and OSPA$^{(2)}$ errors show similar trends in respect of inclusion of the occlusion model, increasing scenario density, and relative performance on the two detectors. The tracking performance in the multi-camera reconfiguration case is generally worse than the case when all cameras are active. This relative observation is in line with expectations, as there is less sensor data to resolve occlusions and perform estimation.

To facilitate an examination of the relative performance in further detail, Fig. 4.9 plots the OSPA$^{(2)}$ error with 3D GIoU base-distance over a sliding window with time. As a reference point for the performance comparison, the YOLOv3+MV-GLMB-OC with all cameras operational case is also shown. The spikes in the error curve at the beginning and the end of the scenario are due to mismatches in track initiation and termination with the ground truths. For CMC1, we observe that the error curves are relatively close to the reference case. This would be expected for a sparse scenario as there are virtually no occlusions even when some cameras are offline. For CMC2 and CMC3, the error curves for both YOLOv3+MV-GLMB-OC* and Faster-RCNN(VGG16)+MV-GLMB-OC* begin to deviate midway into sequence from the all cameras operational reference. The errors become more pronounced entering the 2-camera only segment, as the more crowded scenarios exacerbate the effect of occlusions and misdetections. Nonetheless, the results show that the MV-GLMB-OC filter is able to accommodate on-the-fly changes to the camera configurations.

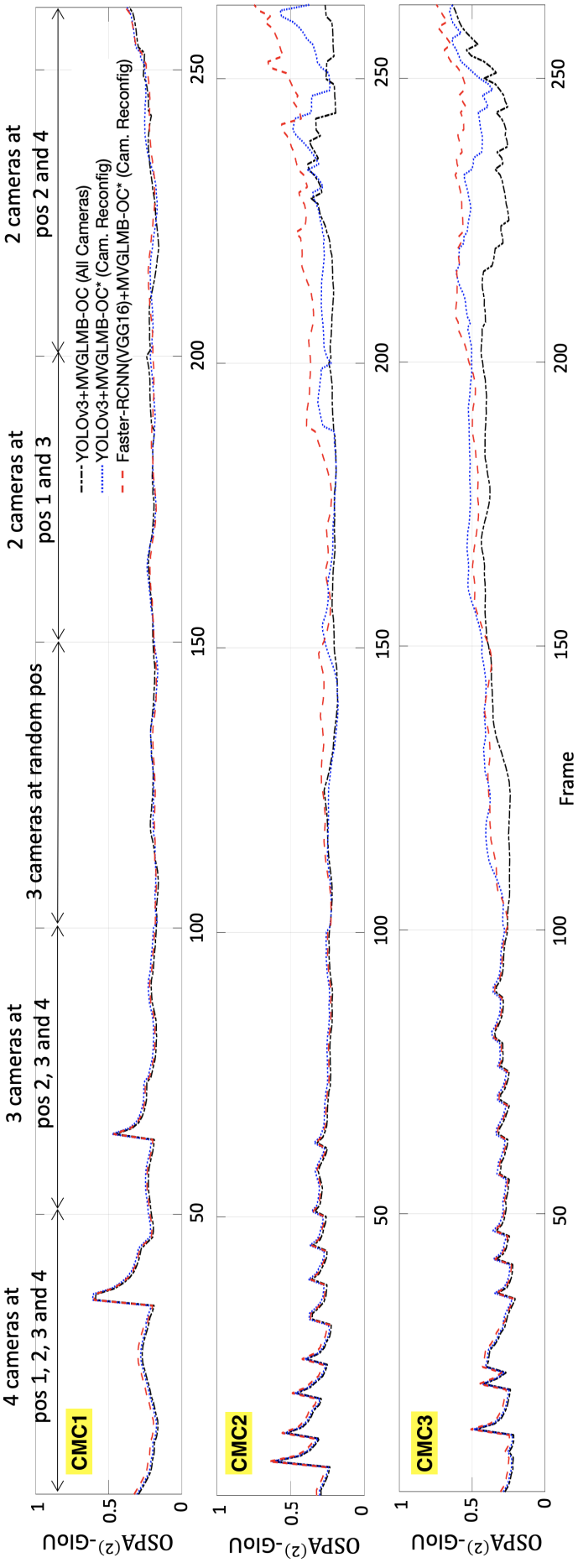


Figure 4.9: Multi-Camera Reconfiguration Experiment: OSPA$^{(2)}$ plots with 3D GIoU base-distance for estimates of 3D centroid with extent. Three trackers are considered: YOLOv3+MV-GLMB-OC* (multi-camera reconfiguration) and Faster-RCNN+MV-GLMB-OC* (multi-camera reconfiguration) and with YOLOv3+MV-GLMB-OC (all cameras operational).

4.4.4 Curtin Multi-Camera Dataset 4 and 5

Here we present the first multi-camera dataset with people jumping and falling, which is more challenging for MOT than scenarios with only normal walking. We demonstrate the versatility of the proposed MOT framework by using a Jump Markov System (JMS), to cater for potential switching between upright and fallen modes [320].

Model Parameters

Each state is augmented $\boldsymbol{x}$ with a discrete mode or class $o \in \{0,1\}$, where $o=0$ corresponds to a standing state and $o=1$ corresponds to a fallen state. We consider the single-object state as $(\boldsymbol{x}, o)$, with single-object density $p^{(\xi)}(\boldsymbol{x}, o) = p^{(\xi)}(\boldsymbol{x}|o)\eta^{(\xi)}(o)$. The following single-object transition density and observation likelihood are used

$$\boldsymbol{f}_{S,+}(\boldsymbol{x}_+ o_+|\boldsymbol{x}, o) = f_{S,+}^{(o_+)}(x_+|x, \ell, o)\delta_{\ell}[\ell_+]\eta_+(o_+|o),$$

$$g^{(c)}(z^{(c)}|\boldsymbol{x}, o) \propto g_e^{(c)}(z_e^{(c)}|o)\mathcal{N}\left(z^{(c)}; \Upsilon^{(c)}(\boldsymbol{x}) + \begin{bmatrix} 0_{2\times 1} \\ -w_e^{(c,o)}/2 \end{bmatrix}, \operatorname{diag}\left(\begin{bmatrix} w_p^{(c)} \\ w_e^{(c,o)} \end{bmatrix}\right)\right).$$

The mode transition probabilities are $\eta_+(0|0) = 0.6$, $\eta_+(1|0) = 0.4$, $\eta_+(0|1) = 0.6$ and $\eta_+(1|1) = 0.4$.

For a standing object, i.e., $o=0$, we have $w_e^{(c,0)} = w_e^{(c)} = [0.01, 0.0025]^T$ in the above observation likelihood. Further, standing objects typically have a bounding box size ratio (y-axis/x-axis) greater than one, thus the mode dependent likelihood component is chosen as $g_e^{(c)}(z_e^{(c)}|0) = e^{\rho\left(([0,1]z_e^{(c)}/[1,0]z_e^{(c)})-1\right)}$ for all cameras, where $\rho = 2$ is a control parameter. The transition density to another standing state $f_{S,+}^{(0)}(x_+|x, \ell, 0)$, is the same as per the previous subsection.

For a fallen object, i.e., $o=1$, we have $w_e^{(c,1)} = [0.0025, 0.01]^T$ in the above observation likelihood, and the mode dependent likelihood component is chosen as $g_e^{(c)}(z_e^{(c)}|1) = e^{-\rho\left(([0,1]z_e^{(c)}/[1,0]z_e^{(c)})-1\right)}$ for all cameras because fallen objects typically have a bounding box size ratio (y-axis/x-axis) less than one. The transition density to another fallen state $f_{S,+}^{(1)}(x_+|x, \ell, 1)$ is the same as that for standing-to-standing except for the large variance $\upsilon^{(s)} = [0.15, 0.15, 0.04]^T$ to capture all possible orientations during the fall.

For a state transition involving a mode switch i.e., standing-to-fallen or fallen-to-standing, the transition density $f_+^{(1)}(x_+|x, \ell, 0)$ or $f_+^{(0)}(x_+|x, \ell, 1)$ takes the form (4.35), with position noise and extent (in logarithm) noise parameterized by:

$$\begin{aligned} \upsilon^{(p)} &= [0.0049, 0.0049, 0.0049]^T, \\ \upsilon^{(s)} &= [0.01, 0.01, 0.01]^T. \end{aligned}$$

Notice that the position noise is increased in the case of a mode switch compared to the case of no switching, in order to capture the abrupt change in the size of the object

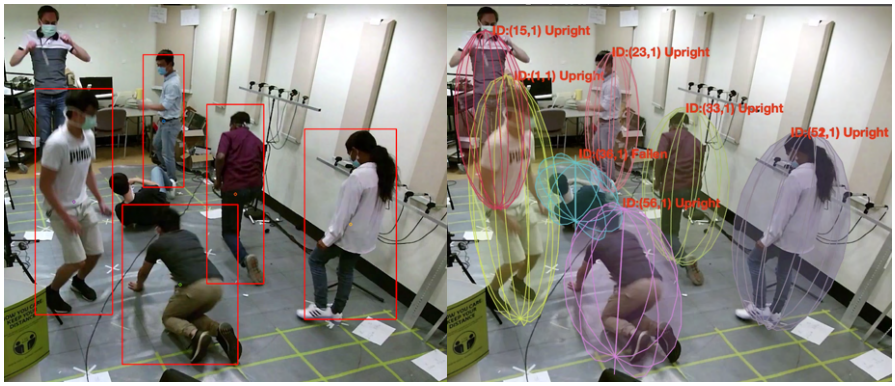


Figure 4.10: CMC5 Camera 1: YOLOv3 detections (left) and MV-GLMB-OC estimates (right).

during mode switching.

The birth density is an LMB with parameters $r_{B,+}(\ell) = 0.001$ and

$$
\begin{aligned}
f_{B,+}(x,\ell,0) =& 0.9\mathcal{N}(x;m_{B,+,0},\mathrm{P}_{B,+,0}),\\
f^{(\ell)}_{B,+}(x,\ell,1) =& 0.1\mathcal{N}(x;m_{B,+,1},\mathrm{P}_{B,+,1}),\\
m_{B,+,0} =& [2.03\ 0\ 0.71\ 0\ 0.825\ 0\ -1.2\ -1.2\ -0.18]^T,\\
m_{B,+,1} =& [2.03\ 0\ 0.71\ 0\ 0.413\ 0\ -0.18\ -0.18\ -1.2]^T,\\
\mathrm{P}_{B,+,0} =& \mathrm{P}_{B,+,1} = 0.1^2\mathrm{I}_9.
\end{aligned}
$$

Effectiveness of Occlusion Model

Tables 4.4 and 4.5 show the CLEAR MOT and OSPA$^{(2)}$ benchmarks for MV-GLMB-OC and MS-GLMB on both detectors YOLOv3 and Faster-RCNN(VGG16). The CLEAR evaluations for the monocular detections are given in Appendix E.

For CMC4 which has a maximum of 3 people, both MV-GLMB-OC and MS-GLMB on either detectors achieved high CLEAR MOT scores in MOTA/MOTP, and low OSPA$^{(2)}$ errors. The incidence of FPs and FNs is caused by track initiation/termination mismatches with the ground truths. Nonetheless, we observe that on MOTA/MOTP and OSPA$^{(2)}$, MV-GLMB-OC outperforms MS-GLMB.

For CMC5 which has a maximum of 7 people, both MV-GLMB-OC and MS-GLMB on either detectors were still capable of producing reasonable MOTA/MOTP scores and OSPA$^{(2)}$ errors. Fig. 4.10 shows a snapshot of detections and estimates on a single view. However, due to poor detections and more occlusions in CMC5, we observe many IDs and FNs. Again on MOTA/MOTP and OSPA$^{(2)}$, MV-GLMB-OC outperforms MS-GLMB.

Multi-Camera Reconfiguration

The multi-camera reconfiguration experiment described in Section 4.4.3 is repeated for the multi-modal datasets CMC4 and CMC5. The results for the multi-camera reconfiguration are denoted with asterisks in Tables 4.4 and 4.5. The plot for $\text{OSPA}^{(2)}$ with 3D GIoU base-distance over a sliding window with time is given in Fig. 4.11. While similar observations can be made from the experiments without jumping and falling (CMC1-CMC3), the results for CMC4-CMC5 exhibit different behavior for people in the fallen state. The estimated extent is warped out of its ordinary shape when the person is on the ground, and more data is required to infer the corresponding state of the fallen person. In CMC4-CMC5, the effect of occlusions or misdetections is exacerbated by having fewer cameras when the person is on the ground, which would likely lead to track termination or switching. Nonetheless, the results confirm that the JMS variant of the MV-GLMB-OC algorithm can automatically accommodate multi-camera reconfiguration.

Table 4.4: CMC4,5 Performance Benchmarks for 3D Position Estimates

CMC4 (Jumping and Falling, Maximum/Average 3 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**99.3%**	**99.0%**	**99.5%**	**3**	**0**	**0**	4	2	**0**	**0**	**98.5%**	**89.5%**	**0.16m**
YOLOv3+MV-GLMB-OC*	95.0%	93.5%	96.5%	**3**	**0**	**0**	17	4	5	1	93.6%	87.7%	0.18m
YOLOv3+MS-GLMB	95.9%	94.0%	97.8%	**3**	**0**	**0**	21	5	4	1	92.6%	86.4%	0.21m
Faster-RCNN(VGG16)+MV-GLMB-OC	98.0%	98.5%	97.5%	**3**	**0**	**0**	**3**	7	2	1	97.0%	87.1%	0.18m
Faster-RCNN(VGG16)+MV-GLMB-OC*	85.1%	82.2%	88.1%	**3**	**0**	**0**	29	**0**	6	**0**	91.3%	86.6%	0.19m
Faster-RCNN(VGG16)+MS-GLMB	89.3%	85.8%	93.1%	**3**	**0**	**0**	34	**0**	12	**0**	88.6%	87.0%	0.22m
CMC5 (Jumping and Falling, Maximum/Average 7 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**60.5%**	**63.5%**	**61.3%**	**3**	4	**0**	**388**	**933**	**55**	**47**	**61.1%**	**69.3%**	**0.63m**
YOLOv3+MV-GLMB-OC*	59.3%	58.1%	60.1%	**3**	3	1	418	1172	69	60	56.7%	63.9%	0.69m
YOLOv3+MS-GLMB	50.9%	51.1%	47.6%	**3**	**2**	2	735	1699	85	69	50.7%	59.5%	0.79m
Faster-RCNN(VGG16)+MV-GLMB-OC	60.1%	62.5%	60.1%	**3**	4	**0**	410	1185	61	49	60.3%	64.1%	0.66m
Faster-RCNN(VGG16)+MV-GLMB-OC*	56.2%	55.6%	59.2%	**3**	3	1	534	1493	63	61	55.7%	63.6%	0.70m
Faster-RCNN(VGG16)+MS-GLMB	49.1%	49.7%	46.1%	**3**	3	1	781	1337	92	69	49.6%	61.6%	0.80m

CLEAR MOT scores and OSPA$^{(2)}$ distance are calculated on standard position estimates (↑ means higher is better while ↓ means lower is better). Two different detectors are considered - Faster-RCNN(VGG16) (monocular) and YOLOv3 (monocular). Two types of trackers are considered - MV-GLMB-OC (multi-view with occlusion model) and MS-GLMB (multi-sensor without occlusion model). The asterisk (*) indicates the multi-camera reconfiguration experiment.

Table 4.5: CMC4,5 Performance Benchmarks for 3D Centroid with Extent Estimates

CMC4 (Jumping and Falling, Maximum/Average 3 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**99.3%**	**99.0%**	**99.5%**	**3**	**0**	**0**	4	2	**0**	**0**	**98.5%**	**60.1%**	**0.18**
YOLOv3+MV-GLMB-OC*	95.0%	93.5%	96.5%	**3**	**0**	**0**	17	4	5	1	93.6%	58.9%	0.20
YOLOv3+MS-GLMB	95.9%	94.0%	97.8%	**3**	**0**	**0**	21	5	4	1	92.6%	57.0%	0.26
Faster-RCNN(VGG16)+MV-GLMB-OC	98.0%	98.5%	97.5%	**3**	**0**	**0**	**3**	7	2	1	97.0%	59.3%	0.20
Faster-RCNN(VGG16)+MV-GLMB-OC*	85.1%	82.2%	88.1%	**3**	**0**	**0**	29	**0**	6	**0**	91.3%	56.2%	0.24
Faster-RCNN(VGG16)+MS-GLMB	89.3%	85.8%	93.1%	**3**	**0**	**0**	34	**0**	12	**0**	88.6%	55.3%	0.28
CMC5 (Jumping and Falling, Maximum/Average 7 people)													
Detector and Tracker	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	PT ↓	ML ↓	FP ↓	FN ↓	IDs ↓	FM ↓	MOTA ↑	MOTP ↑	OSPA$^{(2)}$↓
YOLOv3+MV-GLMB-OC	**59.8%**	**61.0%**	**60.8%**	**3**	4	**0**	**404**	**951**	**67**	**54**	**60.6%**	**45.0%**	**0.65**
YOLOv3+MV-GLMB-OC*	55.9%	54.9%	57.1%	**3**	3	1	689	1125	80	85	55.3%	43.4%	0.71
YOLOv3+MS-GLMB	49.5%	50.1%	45.0%	**3**	**2**	2	715	1750	94	91	49.3%	42.6%	0.78
Faster-RCNN(VGG16)+MV-GLMB-OC	58.1%	60.8%	59.4%	**3**	4	**0**	451	1008	72	57	59.9%	43.1%	0.66
Faster-RCNN(VGG16)+MV-GLMB-OC*	55.9%	53.6%	51.6%	**3**	3	1	569	1519	81	88	51.4%	42.7%	0.75
Faster-RCNN(VGG16)+MV-GLMB-OC	48.8%	45.3%	41.7%	**3**	3	1	734	1493	96	98	43.3%	43.9%	0.81

CLEAR MOT scores and OSPA$^{(2)}$ distance are calculated with a 3D GIoU base-distance for estimates of 3D centroid with extent (↑ means higher is better while ↓ means lower is better). Two different detectors are considered - Faster-RCNN(VGG16) (monocular) and YOLOv3 (monocular). Two types of trackers are considered - MV-GLMB-OC (multi-view with occlusion model) and MS-GLMB (multi-sensor without occlusion model). The asterisk (*) indicates the multi-camera reconfiguration experiment.

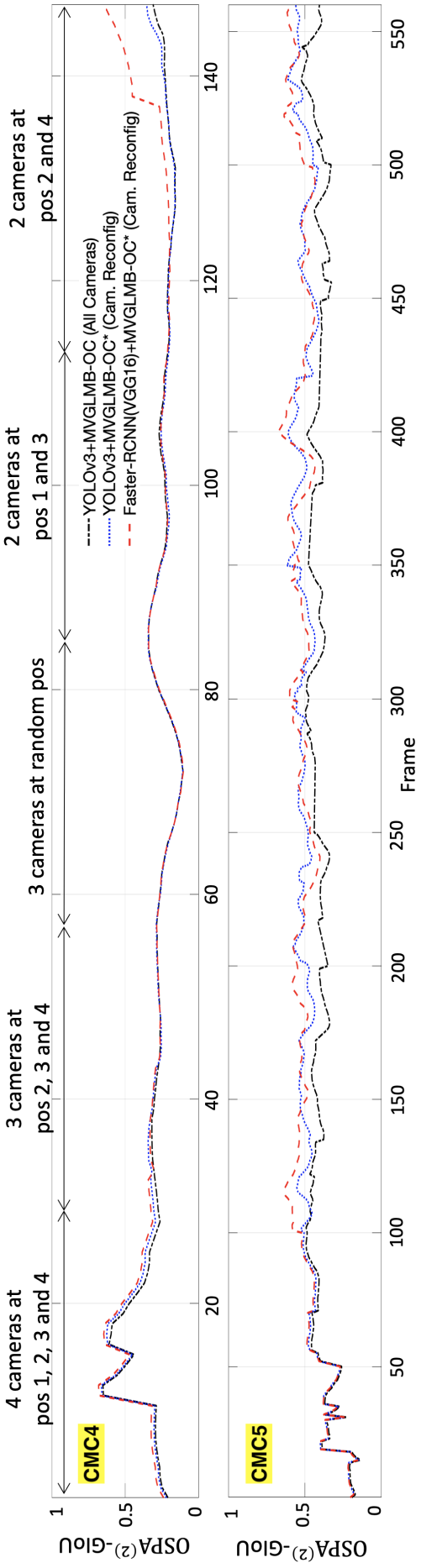


Figure 4.11: Multi-Camera Reconfiguration Experiment: OSPA$^{(2)}$ plots with 3D GIoU base-distance for estimates of 3D centroid with extent. Three trackers are considered: YOLOv3+MV-GLMB-OC* (multi-camera reconfiguration) and Faster-RCNN+MV-GLMB-OC* (multi-camera reconfiguration) and with YOLOv3+MV-GLMB-OC (all cameras operational).

Table 4.6: MV-GLMB-OC Runtime on WILDTRACKS and CMC

Dataset (Cams)	Frames	No. Obj (avg)	Exec. Time (s/frame)
W.T. (7)	400	20	18.0
CMC1(4)	261	3	0.1
CMC2 (4)	263	10	3.2
CMC3 (4)	263	15	7.9
CMC4 (4)	147	3	0.4
CMC5 (4)	560	7	5.5

4.4.5 Runtimes

The runtimes for the MV-GLMB-OC filter on the WILDTRACKS and CMC datasets are summarized in Table 4.6. The current implementation is via unoptimized MATLAB code. The reported runtimes appear to be consistent with the computational complexity of the MV-GLMB-OC algorithm: quadratic in the number of objects and linear in the sum of the number of detections across all cameras.

4.5 Conclusion

By developing a tractable 3D occlusion model, we have derived an online Bayesian multi-view multi-object filtering algorithm that only requires monocular detector training, independent of the multi-camera configurations. This enables the multi-camera system to operate uninterrupted in the event of extension/reconfiguration (including camera failures), obviating the need for multi-view retraining. Moreover, it addresses the multi-camera data association problem in a way that is scalable in the total number of detections. Experiments on existing 3D multi-camera datasets have demonstrated similar performance to the state-of-the-art batch method. We also demonstrated the ability of the proposed algorithm to track in densely populated scenarios with high occlusions, and with people jumping/falling in the 3D world frame.

Chapter 5

Audio-Visual Multi-Source Tracking and Separation

MEETING or conference assistance is a popular application that typically requires compact configurations of co-located audio and visual sensors. This chapter proposes a novel solution for online separation of an unknown and time-varying number of moving sources using only a single microphone array co-located with a single visual device. The approach exploits the complementary nature of simultaneous audio and visual measurements, accomplished by a model-centric 3-stage process of detection, tracking, and (spatial) filtering, which performs separation in a block-wise or recursive fashion. Fusing the measurements requires solving the multi-modal space-time permutation problem, since the audio and visual measurements reside in different observation spaces, but also are unidentified or unlabeled (with respect to the unknown and time-varying number of sources), and are subject to noise, extraneous measurements and missing measurements. A labeled random finite set tracking filter is applied to resolve the permutation problem and recursively estimate the source identities and trajectories. A time-varying set of generalized side-lobe cancellers is constructed based on the tracking estimates to perform online separation. Evaluations are undertaken with live human speakers. The content of this chapter has been published in [61][1].

5.1 Introduction

Source separation refers to the estimation of individual source signals from an unknown mixture signal recorded by one or more microphones. A common challenge in source separation is the permutation ambiguity problem [24]. Traditional approaches to blind source separation (BSS) such as independent component analysis (ICA) [25], sparseness-based solutions [26, 80] and non-negative matrix factorization (NMF) [27]

[1] © 2022 IEEE. Reprinted, with permission, from J. Ong, B. T. Vo, S. Nordholm, B. -N. Vo, D. Moratuwage and C. Shim, "Audio-Visual Based Online Multi-Source Separation," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30, pp. 1219-1234, 2022.

have demonstrated strong interference suppression with minimal signal distortion on a mixture of static speech sources. These approaches typically assume a fixed and known number of stationary sources and exploit their individual statistics in order to achieve separation. More recent deep neural network (DNN) based approaches such as uPIT [29], DPCL [28], DANet [39], and TasNet [40] have also shown promising separation performance for pre-trained speaker models. Similarly, these approaches rely on the assumption that the number of speakers and their characteristics are fixed and known during training and testing [31].

Source separation for an unknown and time-varying number of moving speakers is even more challenging since the room impulse response for each source varies in both time and position [8]. As a result, standard BSS techniques which rely on stationarity assumptions may not be directly applicable [41, 42]. In addition, it is not clear if DNN based approaches can be extended to accommodate the unknown and spontaneous appearance and disappearance of active sources. An alternative to BSS and DNN approaches is a model-centric approach based on a 3-step process of detection, tracking, and filtering (DTF), which has the salient feature of being able to accommodate an unknown and time-varying number of moving sources without pre-training [42, 44, 137]. In our previous work [53], the DTF approach was further demonstrated for online or recursive operation using the latest generation of random finite set (RFS) tracking techniques [122, 237], where separation of multiple speech sources was achieved through initially taking audio measurements from multiple microphone arrays, then tracking the sources in space and time, and finally carrying out beamforming in the direction of the estimated source. While each of the abovementioned approaches has relative advantages and disadvantages in different applications, the common element is that they exclusively rely on audio content to perform separation.

In noisy or loud settings, humans can employ both audio and visual cues to hone in on the speaker of interest, and are thought to incorporate the audio-visual correspondence between lip movements and speech utterances [30]. Motivated by traditional BSS approaches, an unsupervised audio-visual solution is proposed in [228], which employs low-rank matrices to model the background audio-visual information, while sparsity is used to extract sources through correlations between the audio and visual modalities. The DNN-based solution proposed in [388] uses an off-the-shelf face detector in combination with a face recognition model to extract face embeddings and estimate the associations of speech signals to their respective speakers. Subsequent works in [31, 389] incorporate a DNN-module that extracts lip embeddings and facial appearance directly from video streams, exploiting joint audio-visual features in matching lip movements and voice fluctuations to the correct speaker. The work in [390] further analyzes the close connection between facial motion and emitted speech, proposing that the consistency between voice elements and facial appearance can facilitate the isolation of speech from overlapping sounds.

DNN-based solutions for audio-visual source separation have also been specialized to exploit the naturally occurring features in the case of musical sources. Live musical sounds typically emanate from a person playing an instrument with a unique action, and it is possible to exploit the distinctive correspondence between the audio and visual cues of music generation to achieve separation. To date, numerous DNN-based solutions have shown promising audio-visual based separation performance. The work in [391] demonstrates that a mix of different musical instruments playing on video can be separated by locating the cluster of pixels corresponding to the sound from a particular instrument. This method exploits the natural synchronization of audio and visual modalities to enable joint audio-visual learning without supervision [391], and was extended to train a self-supervised network for vehicle tracking with stereo sound [392].

When multiple similar instruments are playing, relying solely on audio and visual semantics is typically insufficient. The more recent solution in [393] additionally incorporates temporal motion information from the video to improve source differentiation and hence sound separation. An alternative approach in [394] considers the correspondence between body dynamics and finger movements to create a context-aware network which enables more robust audio-visual separation of both heterogeneous and homogeneous musical sources. Network training can further be improved with a so-called sounding object visual grounding technique [395], which distinguishes between active and silent sources to avoid learning noise from the latter. Noting that simultaneous musical instruments are usually interactive in their timing, the approach in [396] improves on one-time separation solutions by recursively minimizing the residual information in the spectrogram. DNN-based audio-visual solutions have also found applications in robot navigation [397, 398], automatic speech recognition [399–401], and person recognition [402–405].

The abovementioned approaches to audio-visual based separation are broadly classified as being data-centric, in the sense that they require some form of training to capture the correspondence between the two complementary modes. Data-centric approaches generally rely on large training sets to work desirably [388, 390] which can be computationally intensive during the learning stage. Moreover, the abovementioned data-centric approaches are generally regarded as offline or batch methods, as the output decompositions are produced only after processing the entire input history, as opposed to online methods where the output and input are synchronized up to a fixed delay. In addition, it is not immediately clear if such approaches are amenable for the separation of an unknown and time-varying number of moving sources.

In contrast to data-centric, model-centric DTF approaches to audio-visual based separation are virtually unexplored. The use of co-located audio and visual sensors is intuitively appealing since the two complementary modalities are used to observe the same scene. This approach is also naturally suited to online conferencing or meeting

analysis type applications, where both modes are readily available and are likely to be more effective than using audio data alone. One of the main difficulties lies in fusing the two measurement modes since the 3D audio measurements and 2D video measurements reside in different observation spaces even though they observe the same physical space or state space. Furthermore, the audio and visual measurements are subject to noise, spurious or missing measurements, and are unlabeled or unidentified. In addition, active sources can move, while new sources can appear and existing sources can disappear. Collectively, these issues give rise to the *multi-modal space-time permutation problem*, since it is not known which measurements are connected to which sources (if any at all) in both measurement modes and across space and time.

Multi-source separation becomes far more challenging in the popular commercial application of meeting or conference assistance. Such applications require a compact configuration with a small number of co-located audio and visual sensors for spatial efficiency and portability as well as facilitating synchronization and calibration [406]. The ensuing technological question is whether multi-source separation can be achieved with this minimal configuration. Apart from the low observability, the absence of widely spaced sensors reduces the available spatial information, thereby causing more noise in the measurements [407]. A co-located sensor configuration therefore relies on the complementarity of both modalities to yield accurate tracking results and improve source separation. Intuitively, visual observations are used to reduce the uncertainty in 3D localization and assist the audio measurement [408, 409], which facilitates better directionality and suppression in spatial filtering.

This work proposes a novel model-centric DTF based algorithm for online source separation, using only a single microphone array co-located with a single visual device. The proposed approach caters for an unknown and time-varying number of moving sources, without pre-training, by exploiting the complementary nature of simultaneous audio and visual measurements. An RFS framework [122, 237] is adopted to address the fusion of the multi-modal measurements and to facilitate the tracking of multiple moving sources. The RFS approach entails the development of stochastic models which capture the physical relationship between the measurements and the sources, including the abovementioned uncertainties. An RFS tracking filter known as the Multi-Sensor Generalized Labeled Multi-Bernoulli (MS-GLMB) filter [19–22] is applied to recursively estimate the number of sources as well as their identities and trajectories, thereby addressing the multi-modal space-time permutation problem. The tracking estimates inform the construction of a time-varying set of spatial filters, known as Generalized Side-lobe Cancellers (GSCs) [60] for achieving source separation. Near-field and far-field evaluations are undertaken with live human speakers.

In summary, our main contribution is a novel audio-visual source separation algorithm, which is the first to demonstrate

- Model-based solution via detection, tracking and filtering,

- Operation in an online fashion or as the data arrives,
- An unknown time-varying number of moving sources,
- Separation without pre-training of the audio signals.

5.2 Problem Formulation and Solution Overview

5.2.1 Signal Model

Consider a scenario where the number of sources is time-varying, and let $N(t)$ denote the number of sources in the scene at discrete time instance t. Each source indexed by $n \in \{1,...,N(t)\}$ is located at position vector $\alpha_n(t) \in \mathbb{R}^3$ at the time instance t. The signal emitted by source n is denoted by s_n, and all signals are assumed to be mutually uncorrelated. The source signals propagate and are received by a single array of M microphones, where each microphone element indexed by $m \in \{1,...,M\}$ is corrupted with non-directional diffuse noise $v^{(m)}$. In this work, we assume source stationarity at each frame k of length T, i.e., $\alpha_n(t) = \alpha_{k,n}$ and $N(t) = N_k$ for $t = (k-1)T,...,kT$. In this case, the source signal s_n can be represented in blocks of frames:

$$s_n(t) = \sum_{k=1}^{K} s_n(t)\varpi_T(t-(k-1)T) = \sum_{k=1}^{K} s_{k,n}(t), \tag{5.1}$$

where ϖ_T is a window function of length T, and k is the index of a time block/frame with length T. Based on the direct path term only, the mixture received by microphone element m is approximated by:

$$y^{(m)}(t) \approx \sum_{k=1}^{K}\sum_{n=1}^{N_k} \frac{s_{k,n}\left(t-\tau(\alpha_{k,n},u^{(m)})\right)}{4\pi||\alpha_{k,n}-u^{(m)}||} + v^{(m)}(t), \tag{5.2}$$

where $||\cdot||$ is the Euclidean distance, $\tau(\alpha_{k,n},u^{(m)}) \triangleq c_s^{-1}||\alpha_{k,n}-u^{(m)}||$ is the time delay between source n at position $\alpha_{k,n}$ and microphone m at position $u^{(m)} \in \mathbb{R}^3$, and c_s is the speed of sound propagation. The objective is to estimate the individual source signals frame by frame using the mixture signals $y^{(1)},...,y^{(M)}$ with no prior knowledge on the number of sources, their positions and identities/labels.

5.2.2 Visual Assistance

To estimate the individual source signals, knowledge of the source positions and their labels is crucial, as they are needed to direct a set of time-varying spatial filters to perform source separation. In our previous work [53], this is achieved by tracking multiple sources in 3D space using audio-only data obtained from four microphone arrays that

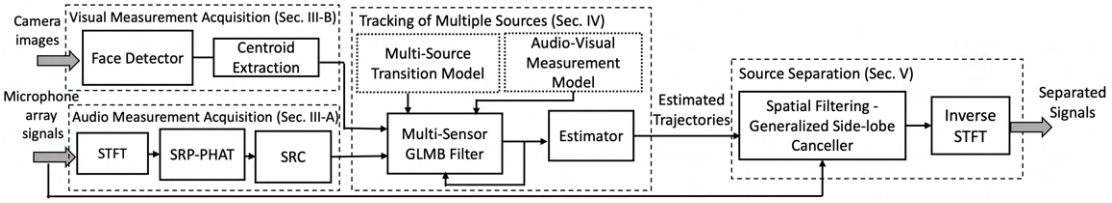


Figure 5.1: System Diagram.

are spaced around the room. The use of multiple microphone arrays is necessary because the audio measurements obtained from a single array alone typically have insufficient observability to allow accurate 3D tracking. An alternative to multiple microphone arrays is to use complementary audio-visual data to observe multiple human speakers in a common physical space. According to recent surveys [15, 140], visual detections or measurements via standard object detectors, e.g., body [169], face [410], and pose [411], have become highly robust and accurate over the years. Thus the use of a single visual device in combination with a single microphone array is likely to facilitate accurate tracking performance. Due to the complementary nature of the audio and visual measurements, which are conditionally independent measurements of the same active sources in a common physical space, it is natural to exploit both modalities simultaneously. To incorporate 2D visual measurements with 3D audio measurements, it is necessary to specify the physical relationship $\mathcal{P}_V^{(c)}$, which maps the 3D source position α to the 2D camera projection $\alpha_V^{(c)}$. Details of this relationship are given in the Section 5.3.2.

5.2.3 Overview of the Proposed Method

The processing chain of the proposed method is shown in Fig. 5.1. Audio and visual measurements of the same (multiple) sources in a common (physical) space are synchronized and segmented into frames indexed by discrete time $k = 1, ..., K$. At each frame, raw microphone signals are fed into an acoustic localization technique to acquire the 3D source position candidates. In parallel, images from multiple cameras are fed into a monocular face detection algorithm to acquire 2D centroid measurements of the same sources present. Measurements acquired from both modalities are subjected to noise (disturbance), they may not reflect a source that is present (false negative), and some may not correspond to any source (false positive). Furthermore, the audio and visual measurements undergo different transformations and hence reside in different observations spaces. Consequently, the audio and visual measurements have an inherent *multi-modal space-time permutation* issue, since the measurements are unlabeled or unidentified with respect to the time-varying and unknown number of sources. The space permutation aspect refers to the fact that in a given frame, it is not known which measurements (if any) correspond to which sources, while the time permutation aspect refers to the fact that across time, it is not known which measurements (if any) correspond to the same source. A labeled RFS approach [19–22] can be used to model the

stochastic relationship between the multi-modal measurements and source states, and jointly estimate the number of sources, their positions and labels. Based on the tracking estimates, a set of time-varying spatial filters can be constructed based on the direct path signal model to perform source separation. The proposed method can be described in three stages: audio-visual measurement acquisition, multi-source tracking, and source separation.

Audio-Visual Measurement Acquisition

In the first stage, audio measurements are obtained by first performing the Short-Time Fourier Transform (STFT) on the raw microphone signals. For each frame, the Steered-Response Power Phase Transform (SRP-PHAT) and a region search algorithm known as Stochastic Region Contraction (SRC) [70], are used to obtain 3D position candidates from the microphone array. In parallel, visual measurements are obtained by passing images into the Dual-Shot Face Detector (DSFD) [410] to acquire visual detections in the form of bounding boxes, and then picking the centroids as 2D position candidates of the human lips.

Multi-Modal Multi-Source Tracking

In the second stage, we adopt a labeled RFS framework [19–22] to fuse the multi-modal (audio-visual) measurements, and produce estimates of the 3D source positions and labels at each frame, in a statistically consistent manner. In this framework, the relationship between the multi-modal measurements and multi-source states is established by the multi-sensor audio-visual measurement model. The motion, appearance, and disappearance of sources are encapsulated by the multi-source transition model. Specifically, a tracking filter known as the Multi-Sensor Generalized Labeled Multi-Bernoulli (MS-GLMB) filter [22] is employed. The recursive filter propagates a so called filtering density, which provides a stochastic description of the set of labeled source states at the current time frame, given all audio-visual measurements up to the current time frame. An estimator is applied to the filtering density to output the source positions and labels at each frame.

Source Separation via Spatial Filtering

In the third stage, source separation is achieved via constructing a type of spatial filter known as the Generalized Side-lobe Canceller (GSC) [60]. A set of GSCs is constructed, one for each source present, using the estimated source positions and the labels at each frame. Each GSC is employed to emphasize each source of interest while simultaneously suppressing other interfering sources. Finally, the time-domain separated signals are recovered using the inverse STFT.

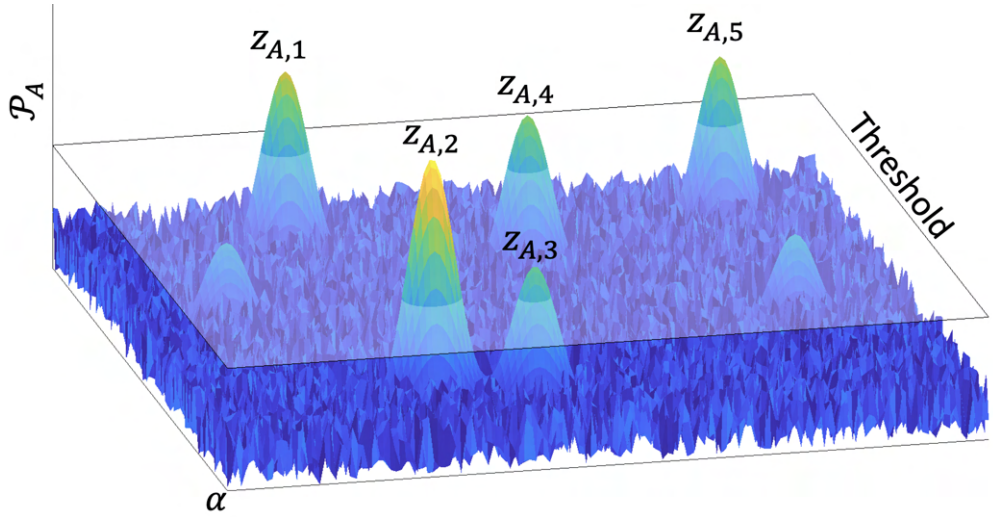


Figure 5.2: SRP-PHAT Measurements.

5.3 Audio-Visual Data Pre-Processing

5.3.1 Audio Measurement Acquisition

Each raw microphone signal $y^{(m)}$ is segmented into $y_1^{(m)}, \ldots, y_K^{(m)}$ via:

$$y_k^{(m)}(t) = y^{(m)}(t + (k-1)T)\varpi_T(t), \tag{5.3}$$

where ϖ_T is a selected window function of length T. The window function is chosen to capture enough signal information while reducing signal discontinuities at the edges, e.g., a Hann window $\varpi_T(t) = 0.5 - 0.5\cos(2\pi t/T), \ \ t = 0, \ldots, T-1$.

We denote the discrete STFT of $y_k^{(m)}$ by $Y_k^{(m)}$. To represent the segmented frequency-domain raw signals from all microphones in a compact form, we stack them into a vector (where λ is the frequency bin index):

$$Y_k(\lambda) = \left[Y_k^{(i)}(\lambda) \right]_{i=1}^{M}. \tag{5.4}$$

Given Y_k received at the array, the spatial power that emanates from the direction of the source location $\alpha_k \in \mathbb{R}^3$, is computed using SRP-PHAT by [70]:

$$\mathcal{P}_{A,k}(\alpha) = \sum_{a=1}^{M-1} \sum_{b=a+1}^{M} \sum_{\lambda} \frac{Y_k^{(a)}(\lambda) Y_k^{*(b)}(\lambda)}{\left| Y_k^{(a)}(\lambda) Y_k^{*(b)}(\lambda) \right|} e^{j\omega_\lambda \left(\tau(\alpha, u^{(b)}) - \tau(\alpha, u^{(a)}) \right)}, \tag{5.5}$$

where $\omega_\lambda = 2\pi(\lambda - 1)F_s/T$, F_s is the sampling frequency, the PHAT weighting is frequency-dependent, and the exponential term time-aligns the microphone signals based on the propagation delays. Using the computationally efficient SRC algorithm [70], the 3D source position candidates are obtained via peak-picking on SRP-PHAT with a certain threshold (see Fig. 5.2). We denote the collection of the 3D position

candidates as a measurement set:

$$Z_{A,k} = \{z_{A,k,1}, ..., z_{A,k,|Z_k|}\}, \tag{5.6}$$

where $|Z_{A,k}|$ denotes the number of elements of $Z_{A,k}$.

5.3.2 Visual Measurement Acquisition

Objects in the 3D world coordinate frame are observed by multiple cameras indexed by $c \in \{1, ..., C\}$, wherein each camera produces object detections as 2D points in the camera image coordinate frame. Each camera is treated as a projective device that converts 3D world points onto the 2D image plane [380]. The perspective projection of a point in the 3D coordinate frame (world) to a point in a 2D coordinate frame (plane) is a nonlinear transformation because it can be interpreted as a many-to-one morphism $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ (except for an orthographic projection). Alternatively, this projection can be realized as a linear transformation in the homogeneous coordinates of the projective space $\mathbb{P}$, which is an extension of Euclidean space by adding an extra dimension [380].

Let $\mathcal{P}_V^{(c)}$ be the projective transformation of camera c that takes an arbitrary point α in 3D to a point $\alpha_V^{(c)}$ in 2D (see Fig. 5.3). Based on the pinhole camera model [380], the transformation $\mathcal{P}_V^{(c)}$ first converts the vector $\alpha = (\alpha_1, \alpha_2, \alpha_3)^T$ into its homogeneous form $\tilde{\alpha} = (\alpha_1, \alpha_2, \alpha_3, 1)^T$ (where the subscript indexes refer to the respective coordinate values), and then performs a linear transformation via the camera matrix $\mathrm{P}_{3\times4}^{(c)}$ to obtain the projected homogeneous point $\tilde{\alpha}_V^{(c)}$ on camera c, i.e.,

$$\tilde{\alpha}_V^{(c)} = \mathrm{P}_{3\times4}^{(c)} \tilde{\alpha}. \tag{5.7}$$

The actual 2D point on the image plane $\alpha_V^{(c)}$ is recovered via dividing the first two coordinate values of $\tilde{\alpha}_V^{(c)} = (\tilde{\alpha}_{V,1}^{(c)}, \tilde{\alpha}_{V,2}^{(c)}, \tilde{\alpha}_{V,3}^{(c)})^T$ by the value of its last coordinate, i.e.,

$$\alpha_V^{(c)} = (\tilde{\alpha}_{V,1}^{(c)} / \tilde{\alpha}_{V,3}^{(c)}, \tilde{\alpha}_{V,2}^{(c)} / \tilde{\alpha}_{V,3}^{(c)})^T. \tag{5.8}$$

The camera matrix $\mathrm{P}_{3\times4}^{(c)}$ of camera c captures the intrinsic parameters (the focal length, skew coefficient and projection center), and the extrinsic parameters (the rotation and translation of the camera), which are obtainable via standard camera calibration techniques [378].

Denote the image obtained from camera c at time frame k by $\mathcal{I}_k^{(c)}$. The image is fed into a Dual-Shot Face-Detector [410] which is represented as detection operator $\mathcal{D}^{(c)}$ and produces a set of 2D visual detections at frame k:

$$Z_{V,k}^{(c)} = \mathcal{D}^{(c)}(\mathcal{I}_k^{(c)}) = \{z_{V,k,1}^{(c)}, ..., z_{V,k,|Z_{V,k}^{(c)}|}^{(c)}\}, \tag{5.9}$$

where $z_{V,k}^{(c)} = (\alpha_{V,k,1}^{(c)}, \alpha_{V,k,2}^{(c)})^T$ is a point specified in 2D image coordinates, $|Z_{V,k}^{(c)}|$ denotes

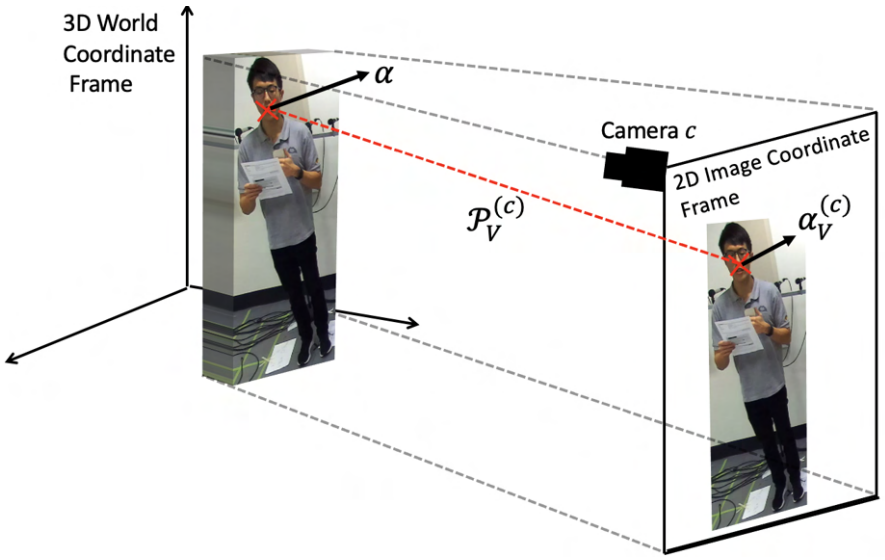


Figure 5.3: Projective transformation $\mathcal{P}_V^{(c)}$ of a point α in 3D to a point $\alpha_V^{(c)}$ in 2D for camera c.

the number of visual measurements at camera c. Note that the projective transformation $\mathcal{P}_V^{(c)}$ between a 3D point in world coordinates and the observed point in 2D image coordinates establishes the relationship between the 3D source positions and the 2D visual measurements.

5.3.3 Audio-Visual Measurements

The multi-modal measurements Z_k at frame k comprise all the constituent measurement sets from the audio and visual sensors, i.e.,

$$Z_k = (Z_{A,k}, Z_{V,k}), \tag{5.10}$$

where $Z_{V,k} \triangleq (Z_{V,k}^{(1)}, ..., Z_{V,k}^{(C)})$. The multi-modal measurements are the basis for estimating the states and labels of the sources. However, the following difficulties arise in the estimation:

- While the audio and visual sensors observe the same scene and same sources, the individual audio measurements $z_{A,k} \in Z_{A,k}$ and individual visual measurements $z_{V,k}^{(c)} \in Z_{V,k}^{(c)}$ are in different observation spaces.
- Due to undergoing different and highly non-linear transformations, individual measurements are noisy, and each measurement set may contain false positives (measurements not generated by any source) and false negatives (missing measurements or missed detections).
- These factors give rise to the inherent *multi-modal space-time permutation problem*, since in space it is not known how the audio measurements from $Z_{A,k}$ and

the visual measurements from $Z_{V,k}^{(1)}, ..., Z_{V,k}^{(C)}$ are associated, or generated by which source if any; and in time, it is not known how the individual audio and visual measurements from $Z_{A,k}$ and $Z_{V,k}^{(1)}, ..., Z_{V,k}^{(C)}$ at the current frame are connected to those from $Z_{A,k+1}$ and $Z_{V,k+1}^{(1)}, ..., Z_{V,k+1}^{(C)}$ at the next frame.

In the next section, we show how the multi-modal space-time permutation problem can be solved using a dynamic Bayesian estimation framework. A labeled RFS model [19–22] facilitates a statistically consistent specification of the *multi-source transition model* and the *multi-modal measurement model*. The transition model is given by a transition density that captures the appearance, disappearance and motion of the sources over time, and captures the uncertainties due to the time permutation issue. The measurement model is given by a likelihood which is based on the assumption that the audio and visual measurements are conditionally independent given the source states, since the audio and video sensors produce complementary measurements of the same sources in a common scene. Consequently, the audio-visual measurement likelihood is separable and can be written as a product of the audio likelihood and visual likelihood. The audio likelihood function describes the relationship between the SRP-PHAT measurements and the source positions, including the uncertainties due to the space permutation issue. The visual likelihood function describes the relationship between the DSFD measurements and the source positions, based on the pinhole camera model, including the uncertainties due to the space permutation issue. Based on these stochastic transition and measurement models, a Bayesian RFS filter recursively estimates the source trajectories and labels.

5.4 Tracking of Multiple Sources

5.4.1 Multi-Source Bayes Tracking Filter

The Bayesian RFS framework [122, 237, 268] facilitates the stochastic modeling of the time-varying nature of the number of sources and the individual source positions, as well as the stochastic modeling of the time-varying nature of the number of measurements which are subjected to noise, false measurements (false positives) and missing measurements (false negatives). In tracking terminology, false negatives and false positives are termed missed detections and false detections respectively, while source appearance and disappearance are termed birth and death respectively. The multi-modal space-time permutation problem is referred to as the data association problem and can be addressed using a labeled RFS tracking filter [19–22]. A visual illustration of the nature of the multi-modal measurements along with the desired tracking result is shown in Fig. 5.4.

Each source at frame k has a state denoted by $\boldsymbol{x}_k \triangleq (x_k, \ell_k)$, where $x_k \triangleq (\alpha_k, \dot{\alpha}_k)$ is a vector capturing the 3D position and velocity of the source, and ℓ_k is a unique label

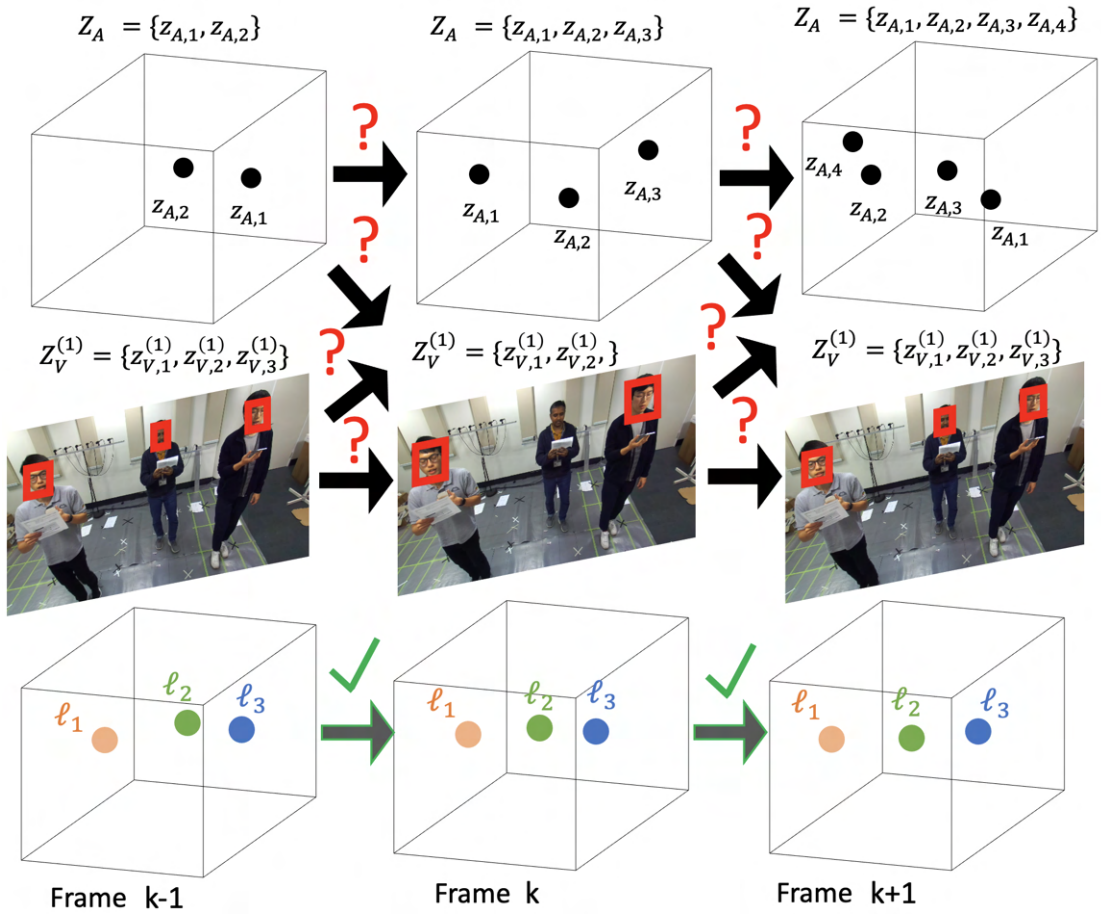


Figure 5.4: Three sources existing from frame $k-1$ to $k+1$. The top row shows an illustration of the audio measurements (3D position candidates). The middle row shows an illustration of the visual measurements (2D point detections). The bottom row shows an illustration of the tracking result addressing the multi-modal space-time permutation problem.

from a discrete space $\mathbb{L}_{0:k}$. The inclusion of the velocity component is necessary as an auxiliary variable for the specification of the state transition model. At each frame k, the collection of states for multiple sources is represented as a finite set:

$$\boldsymbol{X}_k = \{\boldsymbol{x}_{k,1}, ..., \boldsymbol{x}_{k,N_k}\}, \tag{5.11}$$

herein referred to as a multi-source state, where N_k is the number of sources. A key feature of labeled RFS modeling is the assumption of unique labels in the multi-source state, which treats the trajectory of an individual source as a sequence of states with a common label (see Fig. 5.4).

In Bayesian RFS filtering, the aim is to estimate frame-by-frame (recursively) the multi-source state $\boldsymbol{X}_k$, given the multi-modal measurements obtained from the beginning of time up to the current time frame k, i.e., $Z_{1:k} \triangleq (Z_1, ..., Z_k)$. The *multi-source Bayes filter* is a recursive mechanism for computing the probability density of $\boldsymbol{X}_k$ given $Z_{1:k}$. In the Bayesian paradigm, such a probability density is referred to as the filtering density denoted by $\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})$, which captures all uncertainty in the multi-source state given $Z_{1:k}$.

The propagation of the filtering density is a recursive procedure consisting of a time-update followed by a data-update. The first step is given by:

$$\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k}) = \int \boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k)\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k|Z_{1:k})\delta\boldsymbol{X}_k, \tag{5.12}$$

where the above set integral is derived from Finite Set Statistics (FISST) for dealing with probability densities of RFSs in a mathematically consistent manner [122, 237], and the probability density $\boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k)$ is the *multi-source transition density* or the probability density that multi-source state $\boldsymbol{X}_k$ at frame k transitions to $\boldsymbol{X}_{k+1}$ at the next frame $k+1$. The *multi-source transition density* is derived from a stochastic model that captures all possible source births, deaths and motions, i.e., the previously discussed time permutation aspect. The parameters for the transition model are given in Section 5.4.2. The time-updated density (or predicted density) (5.12) describes the uncertainty in $\boldsymbol{X}_{k+1}$, given all multi-modal measurements $Z_{1:k}$ up to the current time frame, and addresses the time permutation part of the data association problem.

The second step is given by:

$$\boldsymbol{\pi}_{k+1|k+1}(\boldsymbol{X}_{k+1}|Z_{1:k+1}) = \frac{g(Z_{k+1}|\boldsymbol{X}_{k+1})\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k})}{\int g(Z_{k+1}|\boldsymbol{X}_{k+1})\boldsymbol{\pi}_{k+1|k}(\boldsymbol{X}_{k+1}|Z_{1:k})\delta\boldsymbol{X}_{k+1}}, \tag{5.13}$$

where the probability density $g(Z_{k+1}|\boldsymbol{X}_{k+1})$ is the *multi-modal (audio-visual) measurement likelihood* or the probability density of the multi-modal measurements Z_{k+1} given the multi-source state $\boldsymbol{X}_{k+1}$. The *multi-modal measurement likelihood* is derived from a stochastic model that encapsulates noise, detections, missed detections, false detections

and multi-modal association uncertainty, i.e., the previously discussed audio-visual space permutation aspect. The parameters for the multi-modal measurement model are given in Section 5.4.3. The data-updated density (or new filtering density) (5.13) contains all information about $\boldsymbol{X}_{k+1}$, conditioned on the multi-modal measurements $Z_{1:k+1}$ up to the new time frame, and addresses the space permutation part of the data association problem.

5.4.2 The Standard Multi-Source Transition Model

Given the multi-source state $\boldsymbol{X}_k$, each state $\boldsymbol{x}_k \triangleq (x_k, \ell_k) \in \boldsymbol{X}_k$ either persists and survives with probability P_S and transition to a new state (x_{k+1}, ℓ_{k+1}) that inherits the same label with transition density $f_S(x_{k+1}|x_k, \ell_k)\delta_{\ell_k}[\ell_{k+1}]$, or dies with probability $1-P_S$. The single-source transition density $f_S(x_{k+1}|x_k, \ell_k)$ gives the probability density of source label ℓ_k moving from state x_k to state x_{k+1}. For tracking live human speakers, a popular choice for the transition density is the Langevin model [69, 72, 234], which takes on the form:

$$f_S(x_{k+1}|x_k, \ell_k) = \mathcal{N}(x_{k+1}; \mathrm{F}x_k, \mathrm{R}\mathrm{R}^T), \tag{5.14}$$

where $\mathcal{N}(\cdot\,; \mathrm{F}x_k, \mathrm{R}\mathrm{R}^T)$ is a Gaussian probability density function with mean $\mathrm{F}x_k$ and covariance $\mathrm{R}\mathrm{R}^T$, $\mathrm{F} = \mathrm{F}_{\text{pseudo}} \otimes \mathrm{I}_3$, $\mathrm{R} = \mathrm{R}_{\text{pseudo}} \otimes \mathrm{I}_3$, I_3 an identity matrix of 3 dimensions, $\otimes$ is the Kronecker product,

$$\mathrm{F}_{\text{pseudo}} = \begin{bmatrix} 1 & \phi \\ 0 & e^{-\beta\phi} \end{bmatrix} \quad \mathrm{R}_{\text{pseudo}} = \sigma_\upsilon \begin{bmatrix} 0 \\ \nu\sqrt{1-e^{-2\beta\phi}} \end{bmatrix}. \tag{5.15}$$

Here, β is the rate constant that controls the rate at which the velocity decays, ν is the steady-state root-mean-square velocity constant, ϕ is the discretization time step interval, and σ_υ is a 3D column vector of the component standard deviations of the process noise.

At this next time, a set of new sources denoted by $\boldsymbol{B}_{k+1}$ with labels $\{\ell_{k+1} : (x_{k+1}, \ell_{k+1}) \in \boldsymbol{B}_{k+1}\}$ can appear individually with probability $r_B(\ell_{k+1})$ and distributed according to the birth density $f_B(\cdot, \ell_{k+1})$. A label follows the convention $\ell_k = (\varsigma, \iota) \in \mathbb{L}_k$, where $\varsigma \in \{k\}$ denotes the time frame of birth and $\iota \in \mathbb{N}$ denotes the index of source born at the same time [19]. Consequently, the labels of a multi-source state are distinct/unique for all frames, and the label space for sources at frame k is constructed recursively by $\mathbb{L}_{0:k} = \mathbb{L}_{0:k-1} \cup \mathbb{L}_k$.

The RFS $\boldsymbol{X}_{k+1}$ is the union of the survivals $\boldsymbol{W}_{k+1}$ and births $\boldsymbol{B}_{k+1}$ which are assumed to be statistically independent. Denote by $\boldsymbol{f}_S(\boldsymbol{W}_{k+1}|\boldsymbol{X}_k)$ and $\boldsymbol{f}_B(\boldsymbol{B}_{k+1})$, the probability densities of the surviving sources $\boldsymbol{W}_{k+1}$ from $\boldsymbol{X}_k$, and the births of new sources $\boldsymbol{B}_{k+1}$ respectively. The *multi-source transition density* is given by [19]:

$$\boldsymbol{f}(\boldsymbol{X}_{k+1}|\boldsymbol{X}_k) = \boldsymbol{f}_S(\boldsymbol{W}_{k+1}|\boldsymbol{X}_k)\boldsymbol{f}_B(\boldsymbol{B}_{k+1}). \tag{5.16}$$

The above product is a stochastic model for addressing the time permutation problem. Under this model, source appearance, disappearance and motion are statistically independent. Importantly, distinct/unique labels are propagated for existing sources that continue to be active. The appearance of new sources is catered for with new distinct labels, while deactivated sources are removed without reusing their labels. The derivation and full expression for (5.16) is not required for this dissertation, however readers are referred to the original work [19] for details. The transition density (5.16) captures all possible source births, deaths and motions in the transition of a multi-source state from one frame to the next, and is parameterized by: the probability of survival P_S, single-source transition density f_S, probability of birth r_B, and the birth density f_B. Specific values for these parameters are provided in the experimental section.

5.4.3 The Standard Multi-Sensor Measurement Model

Microphone Array Measurements

Given a multi-source state $\boldsymbol{X}_k$, each $\boldsymbol{x}_k = (x_k, \ell_k) \in \boldsymbol{X}_k$ is either detected by the microphone array with probability $P_{A,D}$ and generates a detection $z_{A,k} \in Z_{A,k}$ with a likelihood $g_A(z_{A,k}|x_k, \ell_k)$, or is missed with probability $1 - P_{A,D}$. The audio single-source likelihood $g_A(z_{A,k}|x_k, \ell_k)$ gives the probability density of the audio measurement $z_{A,k}$ given the source state (x_k, ℓ_k). For SRP-PHAT measurements, the likelihood has the form:

$$g_A(z_{A,k}|x_k, \ell_k) = \mathcal{N}(z_{A,k}; \mathrm{H}x_k, \sigma_A \sigma_A^T), \tag{5.17}$$

where $\mathrm{H} = [\mathrm{I}_3, 0]$, and σ_A is a 3D column vector of the component standard deviations describing the uncertainty in the audio measurement ($\sigma_A \sigma_A^T$ is the 3-by-3 noise covariance matrix).

The detection process also generates false detections, conventionally characterized by an intensity function $\kappa_A(z_{A,k}) \triangleq \lambda_{A,p} \mathcal{U}_A(z_{A,k})$ on the measurement space [122, 237]. The number of false detections is modeled by a Poisson distribution with mean $\lambda_{A,p}$, and the false detections themselves are uniformly distributed in the audio measurement space according to $\mathcal{U}_A$. It is standard to assume that the audio detections are statistically independent from the false detections [122, 237].

Let $\mathcal{L}(\boldsymbol{X}_k)$ be a set of all distinct source labels present in $\boldsymbol{X}_k$, i.e., $\mathcal{L}(\boldsymbol{X}_k) \triangleq \{\ell : (x_k, \ell) \in \boldsymbol{X}_k\}$. A single-array association $\theta_{A,k} \in \Theta_{A,k}$ is defined as a mapping from the source labels to the audio measurement indices, i.e., $\theta_{A,k} : \{\ell_k : \ell_k \in \mathcal{L}(\boldsymbol{X}_k)\} \to \{0 : |Z_{A,k}|\}$, such that *no two distinct arguments are mapped to the same positive value* [19]. This property ensures each audio measurement comes from at most one source. For example, $\theta_{A,k}(\ell_k) > 0$ corresponds to source ℓ_k generating detection $z_{A,k,\theta_{A,k}(\ell_k)}$, while $\theta_{A,k}(\ell_k) = 0$ means a missed detection for source ℓ_k.

The multi-source audio measurement likelihood is given by:

$$g_A(Z_{A,k}|\boldsymbol{X}_k) \propto \sum_{\theta_{A,k}\in\Theta_{A,k}} \prod_{\substack{(x_k,\ell_k)\\ \in \boldsymbol{X}_k}} \psi_{A,Z_{A,k}}^{(\theta_{A,k}(\ell_k))}(x_k,\ell_k), \tag{5.18}$$

where

$$\psi_{A,Z_{A,k}}^{(j)}(x_k,\ell_k) = \begin{cases} \frac{P_{A,D}\, g_A(z_{A,k,j}|x_k,\ell_k)}{\kappa_A(z_{A,k,j})}, & j>0 \\ 1-P_{A,D}, & j=0 \end{cases}. \tag{5.19}$$

The mixture form of the audio measurement likelihood (5.18) takes in account all possible combinations of missed detections, false detections and the source detections that can occur in the audio measurements.

Camera Measurements

Given a multi-source state $\boldsymbol{X}_k$, each $\boldsymbol{x}_k = (x_k,\ell_k) \in \boldsymbol{X}_k$ is either detected by the camera c with probability $P_{V,D}^{(c)}$ and generates a detection $z_{V,k}^{(c)} \in Z_{V,k}^{(c)}$ with a likelihood $g_V^{(c)}(z_{V,k}^{(c)}|x_k,\ell_k)$, or is missed by camera c with probability $1-P_{V,D}^{(c)}$. The visual single-source likelihood $g_V^{(c)}(z_{V,k}^{(c)}|x_k,\ell_k)$ for camera c gives the probability density of the visual measurement $z_{V,k}^{(c)}$ given the source state (x_k,ℓ_k). For 2D camera detections, the likelihood for camera c takes on the form:

$$g_V^{(c)}(z_{V,k}^{(c)}|x_k,\ell_k) = \mathcal{N}(z_{V,k}^{(c)}; \mathcal{P}_V^{(c)}(\mathrm{H}x_k), \sigma_V^{(c)}\sigma_V^{(c)T}), \tag{5.20}$$

where $\mathcal{P}_V^{(c)}$ is the transformation described in Section 5.3.2, and $\sigma_V^{(c)}$ is a 2D column vector of the component standard deviations describing the uncertainty in the visual measurement ($\sigma_V^{(c)}\sigma_V^{(c)T}$ is the 2-by-2 noise covariance matrix).

The detection process also generates false measurements or detections, conventionally characterized by an intensity function $\kappa_V^{(c)}(z_{V,k}^{(c)}) \triangleq \lambda_{V,p}^{(c)} \mathcal{U}_V(z_{V,k}^{(c)})$ on the measurement space for camera c [122, 237]. The number of false detections is modeled by a Poisson distribution with mean $\lambda_{V,p}^{(c)}$, and the false detections themselves are uniformly distributed in the visual measurement space according to $\mathcal{U}_V$. It is standard to assume that the visual detections are statistically independent from the false detections [122, 237].

A single-camera association $\theta_{V,k}^{(c)} \in \Theta_{V,k}^{(c)}$ is defined as a mapping from the source labels to the visual measurement indices, i.e., $\theta_{V,k}^{(c)} : \{\ell_k : \ell_k \in \mathcal{L}(\boldsymbol{X}_k)\} \to \{0 : |Z_{V,k}^{(c)}|\}$, such that *no two distinct arguments are mapped to the same positive value* [19]. This property ensures each visual measurement comes from at most one source. For multiple cameras, a multi-camera association is the vector $\theta_{V,k} \triangleq (\theta_{V,k}^{(1)},\ldots,\theta_{V,k}^{(C)}) \in \Theta_{V,k}$ of all single-camera associations having the same aforementioned positive one-to-one prop-

erty, where $\Theta_{V,k} \triangleq \Theta_{V,k}^{(1)} \times ... \times \Theta_{V,k}^{(C)}$ is the space of all possible multi-camera associations [22].

The multi-source visual measurement likelihood is given by:

$$g_V(Z_{V,k}|\boldsymbol{X}_k) \propto \sum_{\theta_{V,k}^{(1)}} ... \sum_{\theta_{V,k}^{(C)}} \prod_{(x_k,\ell_k) \in \boldsymbol{X}_k} \prod_{c=1}^{C} \psi_{V,Z_{V,k}^{(c)}}^{(c,\theta_{V,k}^{(c)}(\ell_k))}(x_k,\ell_k), \tag{5.21}$$

where $\theta_{V,k}^{(1)} \in \Theta_{V,k}^{(1)}, ..., \theta_{V,k}^{(C)} \in \Theta_{V,k}^{(C)}$, and

$$\psi_{V,Z_{V,k}^{(c)}}^{(c,j)}(x_k,\ell_k) = \begin{cases} \dfrac{P_{V,D}^{(c)} g_V^{(c)}\left(z_{V,k,j}^{(c)}|x_k,\ell_k\right)}{\kappa_V^{(c)}\left(z_{V,k,j}^{(c)}\right)}, & j>0 \\ 1-P_{V,D}^{(c)}, & j=0 \end{cases}, \tag{5.22}$$

The mixture form of the visual measurement likelihood (5.21) takes in account all possible combinations of missed detections, false detections and the source detections that can occur in the visual measurements.

Audio-Visual Measurement Likelihood

While the audio and visual sensors produce different measurements in different observation spaces, they are nonetheless observing the same human speakers in a common physical space. Consequently, the measurement sets from each (audio or visual) sensor can be treated as conditionally independent given the multi-source state, and the multi-modal measurement likelihood at time k can be written as:

$$g(Z_k|\boldsymbol{X}_k) = g_A(Z_{A,k}|\boldsymbol{X}_k) \cdot g_V(Z_{V,k}|\boldsymbol{X}_k). \tag{5.23}$$

Each constituent likelihood function in (5.23), i.e., g_A or g_V, contains a nested sum that enumerates all possible associations in that measurement domain, thereby taking into account all possible combinations of missed detections, false detections and the source detections. The product of g_A and g_V in (5.23) therefore contains all combinations of cross-domain associations, thereby presenting a model for addressing the multi-modal space-time permutation problem.

In summary, the *multi-modal measurement likelihood* describes the statistical connection between the audio measurements $Z_{A,k}$ and the visual measurements $Z_{V,k}$ which are complementary observations of the same state $\boldsymbol{X}_k$. The *multi-modal measurement likelihood* is parameterized by: the audio sensor's probability of detection $P_{A,D}$, single-source likelihood g_A, false detection intensity, κ_A; and the visual sensors' probabilities of detection $P_{V,D}^{(1)}, ..., P_{V,D}^{(C)}$, single-source likelihoods $g_V^{(1)}, ..., g_V^{(C)}$, false detection intensities, $\kappa_V^{(1)}, ..., \kappa_V^{(C)}$.

5.4.4 Implementation and State Estimation

The MS-GLMB filter [22] is the analytic solution to the multi-source Bayes recursion (i.e., (5.12) and (5.13)) under the standard multi-source transition and multi-sensor measurement models. The filter propagates the time-updated and data-updated filtering densities in a GLMB form:

$$\boldsymbol{\pi}_{k|k}(\boldsymbol{X}_k) = \Delta(\boldsymbol{X}_k) \sum_{\theta_{1:k} \in \Theta_{1:k}} \omega_{k|k}^{(\theta_{1:k})}(\mathcal{L}(\boldsymbol{X}_k)) \prod_{\boldsymbol{x}_k \in \boldsymbol{X}_k} p_{k|k}^{(\theta_{1:k})}(\boldsymbol{x}_k), \tag{5.24}$$

where $\Delta(\cdot)$ is a distinct label indicator, i.e., $\Delta(\boldsymbol{X}_k) = 1$ if the cardinality $|\mathcal{L}(\boldsymbol{X}_k)| = |\boldsymbol{X}_k|$, $\theta_{1:k} \in \Theta_{1:k}$ is the history of multi-sensor association mappings up to frame k, i.e., $\theta_{1:k} \triangleq (\theta_1, ..., \theta_k)$ where $\theta_k \triangleq (\theta_{A,k}, \theta_{V,k})$ and $\Theta_k \triangleq \Theta_{A,k} \times \Theta_{V,k}$. Each $\omega_{k|k}^{(\theta_{1:k})}(\cdot)$ is a non-negative weight such that

$$\sum_{L \subseteq \mathbb{L}_{0:k}} \sum_{\theta_{1:k} \in \Theta_{1:k}} \omega_{k|k}^{(\theta_{1:k})}(L) = 1, \tag{5.25}$$

and can be interpreted as the probability of sources with label set L being active, as well as being associated with the audio and visual measurements given by the association history $\theta_{1:k}$. Each $p_{k|k}^{(\theta_{1:k})}(\cdot, \ell)$ is the probability density of the source state with label ℓ and association history $\theta_{1:k}$.

The MS-GLMB filter offers a polynomial time implementation mechanism, which has a linear complexity in the sum of the total number of measurements across all sensors [22]. At each frame k, the MS-GLMB filter outputs a multi-source state estimate

$$\hat{\boldsymbol{X}}_k = \{(\hat{\alpha}_{k,1}, \hat{\ell}_1), ..., (\hat{\alpha}_{k,|\hat{\boldsymbol{X}}_k|}, \hat{\ell}_{|\hat{\boldsymbol{X}}_k|})\}, \tag{5.26}$$

via a standard GLMB estimator applied to the GLMB filtering density (5.24) [22]. The source positions and labels over time constitute the estimated source tracks, thereby resolving the space-time permutation problem that arises from the multi-modal measurements as depicted in Fig. 5.4.

5.5 Source Separation

5.5.1 Spatial Filtering

The estimate $\hat{\boldsymbol{X}}_k$ acquired at each frame from the tracking filter informs the construction of a set of time-varying beamformers based on a free space direct-path model. We use the GSC [60], which contains two parts: a beamformer that determines the response of the source of interest (SOI), and a blocking mechanism to prevent the SOI from entering the canceler.

To estimate the SOI specified by label $\hat{\ell}_i$, the corresponding beamformer is constructed to achieve two objectives: select the direction of the source specified by the

estimated position $\hat{\alpha}_{k,i}$, and suppress other interfering sources specified by $\{(\hat{\alpha}_{k,j}, \hat{\ell}_j) \in \hat{\boldsymbol{X}}_k\}_{j=1}^{\hat{N}_k}$ for $i \neq j$, where $\hat{N}_k = |\hat{\boldsymbol{X}}_k|$ is the estimated number of sources. For each time-frequency (TF) point (λ, k), the weight of the beamformer $\hat{W}_{k,\hat{\ell}_i}(\lambda)$ is given by [53]:

$$\hat{W}_{k,\hat{\ell}_i}(\lambda) = \left(\left(\mathrm{D}_{k,\hat{\boldsymbol{X}}_k}(\lambda)\right)^H\right)^{\dagger} l_{\hat{N}_k}(\hat{\ell}_i), \tag{5.27}$$

where ^{H} is the Hermitian transpose, $\dagger$ denotes the Moore-Penrose pseudo-inverse, $l_{\hat{N}_k}$ is a selection vector whose dimension varies depending on the estimated number of sources $\hat{N}_k$, i.e., $l_{\hat{N}_k}(\hat{\ell}_i) = [\delta_{\hat{\ell}_1}[\hat{\ell}_i], \ldots, \delta_{\hat{\ell}_{\hat{N}_k}}[\hat{\ell}_i]]^T$ such that $\delta_i[j] = 1$ if $i = j$ and zero otherwise, and

$$\mathrm{D}_{k,\hat{\boldsymbol{X}}_k}(\lambda) = \begin{bmatrix} e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,1},u^{(1)})\right)} & \ldots & e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,\hat{N}_k},u^{(1)})\right)} \\ \vdots & \ddots & \vdots \\ e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,1},u^{(M)})\right)} & \ldots & e^{j\omega_\lambda\left(\tau(\hat{\alpha}_{k,\hat{N}_k},u^{(M)})\right)} \end{bmatrix}, \tag{5.28}$$

is a matrix with columns representing the steering vectors for each estimated source. The number of columns depends on the estimated number of sources $\hat{N}_k$. Note that if $\hat{N}_k = 1$, (5.27) reduces to the classical delay-and-sum beamformer.

The blocking matrix is defined to be the orthogonal complement to $\left(\hat{W}_{k,\hat{\ell}_i}(\lambda)\right)^H$ [53, 60]:

$$\mathrm{B}_{k,\hat{\ell}_i}(\lambda) = \mathrm{I} - \hat{W}_{k,\hat{\ell}_i}(\lambda)\left[\left(\hat{W}_{k,\hat{\ell}_i}(\lambda)\right)^H \hat{W}_{k,\hat{\ell}_i}(\lambda)\right]^{-1}\left(\hat{W}_{k,\hat{\ell}_i}(\lambda)\right)^H, \tag{5.29}$$

where I is an identity matrix. Subsequently, the GSC weight vector is defined by:

$$G_{k,\hat{\ell}_i}(\lambda) = \hat{W}_{k,\hat{\ell}_i}(\lambda) - \mathrm{B}_{k,\hat{\ell}_i}(\lambda) V_k(\lambda), \tag{5.30}$$

where

$$V_{k,opt}(\lambda) = \arg\min_{V} \sum_{\eta=1}^{k} \chi^{k-\eta} \left| \left(\hat{W}_{\eta,\hat{\ell}_i}(\lambda) - \mathrm{B}_{\eta,\hat{\ell}_i}(\lambda) V\right)^H Y_\eta(\lambda) \right|^2, \tag{5.31}$$

$\chi \in [0, 1]$ is a positive constant. Eq. (5.31) can be solved recursively using Recursive Least Squares (RLS) [364].

The output of the GSC beamformer for the estimated source label $\hat{\ell}_i$ at each TF point (λ, k) is given by:

$$S_{k,\hat{\ell}_i}(\lambda) = \left(G_{k,\hat{\ell}_i}(\lambda)\right)^H Y_k(\lambda). \tag{5.32}$$

Finally, the estimated time-domain signal $\hat{s}_{\hat{\ell}_i}$ of source label $\hat{\ell}_i$ is given by the inverse STFT.

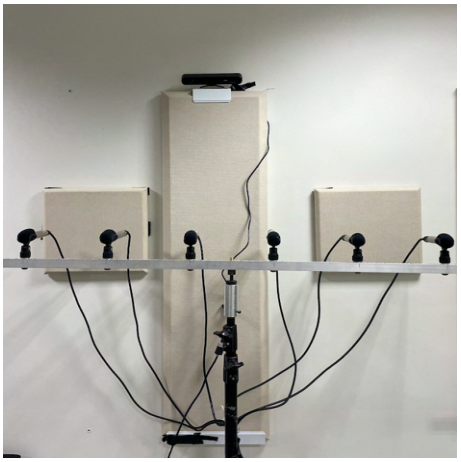


Figure 5.5: Audio-Visual Sensor Setup.

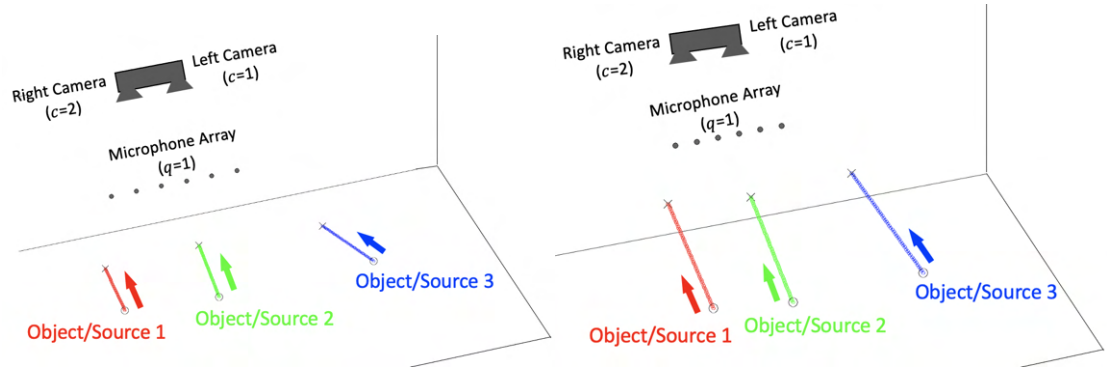


Figure 5.6: Scenario 1A (left) and Scenario 2 (right).

5.6 Experiments

In this section, we present the evaluations for the proposed audio-visual based separation method for live human speakers in an acoustic room. The algorithm is tested in scenarios where human speakers are talking and walking at the same time. We initially consider a detailed analysis of the proposed algorithm in near-field vs far-field. In Scenario 1A, the human speakers are situated closer to the audio-visual sensors, while in Scenario 2, human speakers are situated farther away from the audio-visual sensors. In addition, we present an ablation study for each scenario whereby the measurements, tracking and separation are performed using the audio data only. This is undertaken to demonstrate the improvement in performance due to the combination of audio and visual data. The experimental setup is summarized in Section 5.6.1, and the parameters used for the proposed algorithm are explained in Section 5.6.2. The evaluation of the accuracy of the SRP-PHAT measurements is given in Section 5.6.3, followed by the tracking performance of the MS-GLMB filter in Section 5.6.4, and the separation performance in Section 5.6.5. Subsequently in Section 5.6.6, we consider two additional near-field experiments. Scenario 1B has up to three moving sources appearing at different times, and Scenario 1C has at most one source but with two distinct modes of background interference.

5.6.1 Experimental Setup

The experiment is conducted in a 7.67m × 3.41m × 2.7m enclosed room with reverberation measured at $T_{60} \approx 0.25$s, using a single linear array of 6 microphones, which are calibrated to the same gain/sensitivity. These microphones are connected into 3 *RME-OctaMic 8-channel* pre-amps. Each pre-amp is daisy-chained via MADI cables into the computer. For the visual sensor, a ZED 2 stereo camera from *StereoLabs* is used to record at 1080p. The linear microphone array and ZED 2 stereo camera are co-located and placed close to the wall of the room as shown in Fig. 5.5.

To demonstrate the multi-source tracking and source separation performance of the proposed method, Scenario 1A considers three people talking and walking towards the sensors as shown in Fig. 5.6 (left). The participants stop talking and turn their faces away from the cameras at different times to simulate an exit. A more challenging Scenario 2 employs a similar setup but with the speakers further away from the sensors as shown in Fig. 5.6 (right). To acquire the original speech signals for evaluation, the participants self-recorded their speech while performing the experiments.

5.6.2 Algorithm Parameters

Table 5.1: Parameters for microphone array measurements

F_s	16kHz
High-pass filtering	1kHz
Window function	Hann
T	2048
Detector	SRP-PHAT [70]

Table 5.2: Parameters for visual device measurements

c	1 (left camera) and 2 (right camera)			
FPS	8			
$\mathrm{P}^{(1)}_{3\mathrm{x}4}$	−1021.7	−827.2	−575.8	7071.5
	31.8	142.0	−1184.0	2012.8
	0.04	−0.83	−0.56	3.81
$\mathrm{P}^{(2)}_{3\mathrm{x}4}$	−1021.9	−822.9	−579.0	6940.6
	28.3	131.3	−1192.6	2030.0
	0.03	−0.82	−0.57	3.81
Detector	Dual-Shot Face Detector [410]			

Table 5.3: Parameters for MS-GLMB transition

Multi-source transition density	
β	10s^{-1}
ν	1ms^{-1}
ϕ	0.128s
σ_υ	$[1.2, 1.2, 0.2]^T\text{ms}^{-1}$
P_S	0.999
$\{r_B(\ell_i)\}_{i=1}^3$	$r_B(\ell_i)=0.005$ for all i
$\{f_B(\cdot,\ell_i) \triangleq \mathcal{N}(\cdot;m_B^{(i)},\mathrm{P}_B^{(i)})\}_{i=1}^3$	$m_B^{(1)}=[2.0\ 0.7\ 1.7\ 0\ 0\ 0]^T$, $m_B^{(2)}=[3.0\ 0.5\ 1.7\ 0\ 0\ 0]^T$, $m_B^{(3)}=[4.0\ 0.6\ 1.7\ 0\ 0\ 0]^T$, $\mathrm{P}_B^{(i)}=0.2^2\mathrm{I}_9$ for all i

Table 5.4: Parameters for MS-GLMB likelihood

Audio likelihood	
σ_A	$[0.1, 0.1, 0.1]^T\text{m}$
$P_{A,D}$	0.6
κ_A	$10\mathcal{U}_A$
Visual likelihood	
$\sigma_V^{(c)}$	$[20, 20]^T$ for $c=1,2$
$P_{V,D}^{(c)}$	0.99 for $c=1,2$
$\kappa_V^{(c)}$	$1\mathcal{U}_V$ for $c=1,2$

Table 5.5: Parameters for source separation via spatial filtering

Beamformer	Generalized Side-lobe Canceller
Solver	Recursive Least Squares
Window function	Hann
T	2048
Overlap	50%

5.6.3 Evaluation of SRP-PHAT Measurements

The audio measurements generated from the single microphone array via SRP-PHAT are in the form of 3D position candidates for active sources. The measurements are not only noisy, but are also subjected to false measurements and missing measurements. To evaluate the accuracy of the audio measurements at each frame, the Optimal Sub-Pattern Assignment (OSPA) metric [50] is applied to quantify the error between the set of audio measurements and the set of true source positions. The OSPA metric typically

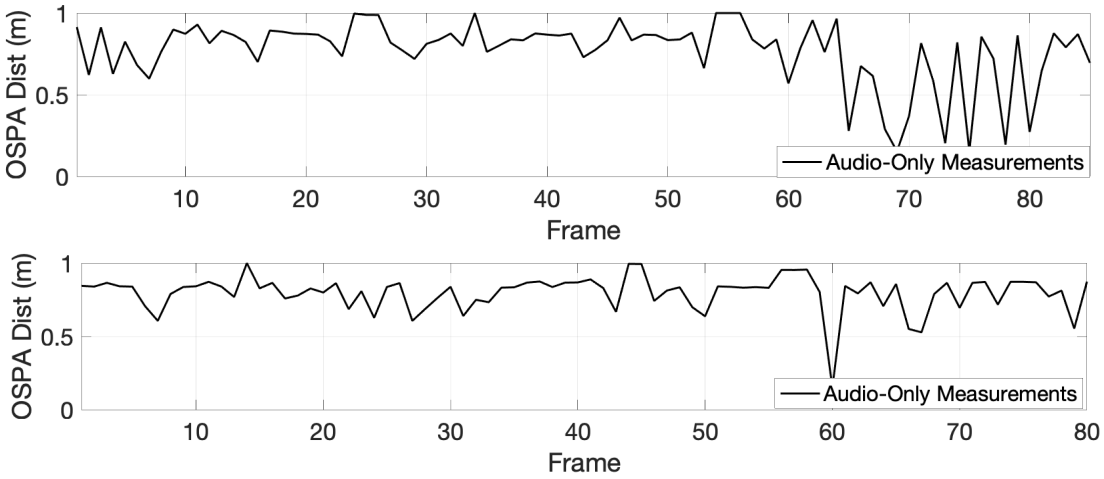


Figure 5.7: Scenario 1A (top) and Scenario 2 (bottom): OSPA distance on the SRP-PHAT measurements (lower is better).

Table 5.6: Average OSPA distance on the obtained SRP-PHAT measurements.

Scenario	Average OSPA Components (m)		
	Localization	Cardinality	OSPA
1A	0.253	0.561	0.814
2	0.291	0.595	0.886

uses a standard Euclidean distance as a base distance, and a cut-off value beyond which a localization error is deemed to be cardinality error. Consequently, the OSPA metric captures both localization and cardinality errors between the set of measurements and set of truths. The numerical value of the OSPA metric lies between zero and the chosen cut-off, which can be interpreted as a per-point error with units of meters. Further details on the OSPA metric can be found in [50].

The OSPA metric with a cut-off at 1m is shown versus time in Fig. 5.7 for Scenarios 1A and 2. It can be seen that the error values are consistently high in both scenarios and occasionally saturate at the cut-off value. The time averaged OSPA errors are shown in Table 5.6, along with the localization and cardinality components. The high average value indicates that the audio-based measurements alone are inaccurate. Furthermore, the large localization component indicates significant positional errors, and the relatively high proportion of the cardinality component indicates significant false and missing measurements. The overall higher errors in Scenario 2 compared to Scenario 1A are due to the sources being farther away from the array. Consequently, the OSPA results in both scenarios suggest that the audio measurements alone are insufficient for accurate tracking of the sources, due to the lack of observability with only a single microphone array.

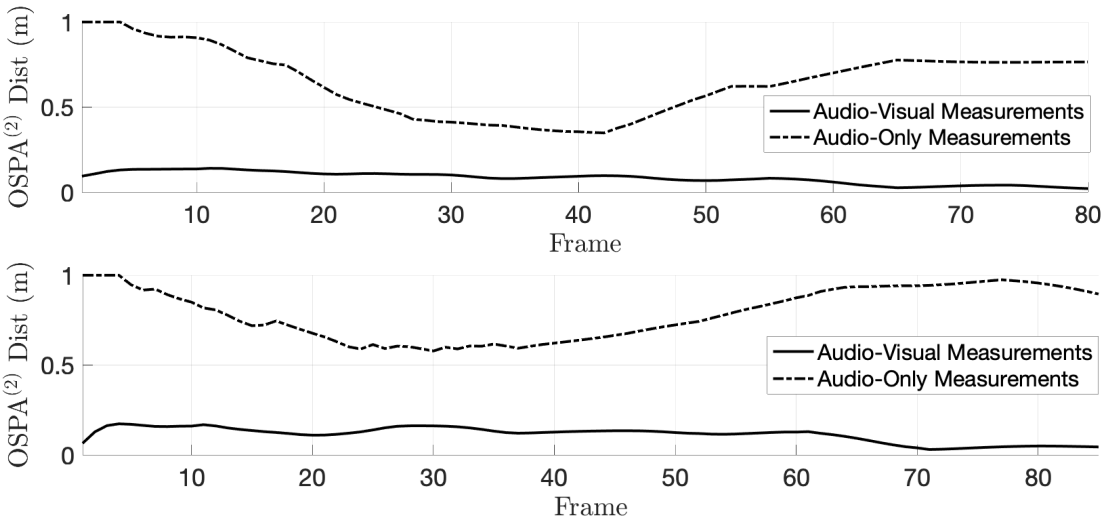


Figure 5.8: Scenario 1A (top) and Scenario 2 (bottom): OSPA$^{(2)}$ distance between estimated and true source trajectories (lower is better).

5.6.4 Evaluation of Multi-Source Tracking Filter

The multi-modal audio and visual measurements are modeled in the RFS framework and processed into trajectory estimates with the MS-GLMB filter. The output of the MS-GLMB tracking filter is a set of unique source labels and corresponding position estimates over time which together constitute a set of tracks or trajectories. Due to the imperfect nature of the multi-modal measurements, it is possible that the estimated trajectories will be noisy, in addition to potentially having incorrect labels and/or misaligned starting and finishing times, and extraneous or missing trajectories. To evaluate the accuracy of the audio-visual source tracking, the OSPA$^{(2)}$ metric [51, 53] can be used, which quantifies the error between the two sets of estimated and true source trajectories. The OSPA$^{(2)}$ metric uses a time averaged OSPA distance as a base distance between two individual tracks, and has a separate cut-off value beyond which a tracking error is deemed to be a labeling error. Consequently, the OSPA$^{(2)}$ metric captures both tracking and labeling errors, and the numerical value is interpreted as time-averaged per-track error with units of meters. The metric is typically calculated over a moving window and plotted versus time. Further details on the OSPA$^{(2)}$ metric can be found in [51].

For this evaluation, a cut-off of 1m is used, with a 10-scan moving window. The OSPA$^{(2)}$ evaluation for combined audio-visual tracking is shown in Fig. 5.8 for Scenarios 1A and 2, which for comparison also shows the OSPA$^{(2)}$ evaluation for audio-only tracking with the single microphone array. It can be seen that in Scenario 1A, combined audio-visual tracking is consistently accurate with low errors below 0.1m. Similarly for Scenario 2, the combination of audio and visual measurements produces consistently accurate tracking estimates with low errors below 0.2m, although the average errors are higher than in Scenario 1A, due to increased distance of the sources from the sensors. Furthermore, the tracking results with only audio measurements from a single micro-

Table 5.7: Scales of SIG, BAK and OVRL in the Subjective Listening Test.

SIG	
Rating	Description
5	Very natural, no degradation
4	Fairly natural, little degradation
3	Somewhat natural, somewhat degraded
2	Fairly unnatural, fairly degraded
1	Very unnatural, very degraded

BAK	
Rating	Description
5	Not noticeable
4	Somewhat noticeable
3	Noticeable but not intrusive
2	Fairly conspicuous, somewhat intrusive
1	Very conspicuous, very intrusive

OVRL	
Rating	Description
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

phone array are consistently poor with very high errors in both scenarios. The cause of the relatively high errors for tracking with only audio measurements are not only due to the high positional errors, but also due to label switching errors, and some incidence of extraneous and missing source trajectories. These observations suggest that the multi-modal combination of audio and video measurements enables accurate multi-source tracking, and further highlight the limitations on the observability of the source trajectories with only a single microphone array.

5.6.5 Evaluation of Source Separation

The set of position and identity estimates from the MS-GLMB tracking filter are used to perform spatial filtering or source separation via a set of GSCs. As the sources are moving within the room, the delays of each source signal, with respect to the microphone array, are changing over time. Therefore, perceptual measures such as PESQ [367], STOI [368] and PEASS [369], that rely on delay-compensation, are not directly applicable for performance evaluations. While it may be possible to apply these measures on time blocks during which sources are almost stationary, there may be insufficient signal information within each short block to allow a meaningful evaluation [53].

Instead, we administer subjective listening tests based on the ITU-T P.835 methodology which evaluates the extent of signal distortion and the overall quality of noise

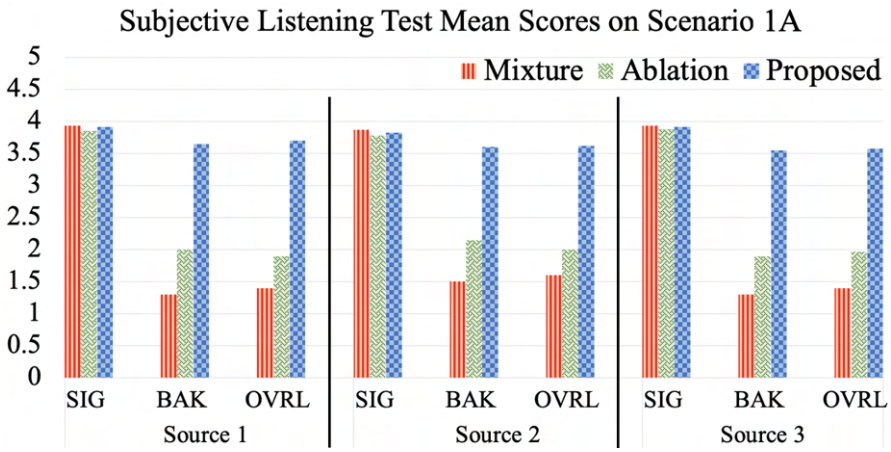


Figure 5.9: Mean scores for SIG, BAK, and OVRL for the estimated source signals and original mixture signals evaluated on Scenario 1A.

suppression [52]. In the test, each participant is instructed to listen to the clean speech signal (upper anchor reference), the separated speech signal (to be evaluated) and the mixture signal (lower anchor reference), and then rate them on: The speech signal alone using a five-point scale of signal distortion (SIG); The background interfering noise alone using a five-point scale of background intrusiveness (BAK); The overall quality using a five-point scale of mean opinion score (OVRL). The scales for SIG, BAK and OVRL are described in Table 5.7.

The evaluation considers the separation performance based on a single microphone array combined with visual tracking assistance from a single camera device (proposed method), and for comparison considers the separation performance using audio-only data without visual tracking assistance (ablation study). In the evaluation, 20 people (12 males, 8 females) of ages from 20 to 30 are recruited to participate in the listening test. A statistical analysis of variance (ANOVA) test at a 0.05 significance level is used to determine if there is a statistically significant difference between the quality of the separated speech signal and the mixture. All video/audio files for both scenarios are available via GitHub: https://github.com/researchwork888/AVseparation.

Scenario 1A

Examination of the audio-visual outputs suggests that there is some degree of interference suppression, though the overall performance is naturally constrained by the use of a single microphone array. The mean scores of all 3 criteria, i.e., SIG, BAK and OVRL, are shown in Fig. 5.9. Some difference is observed in the BAK and OVRL mean scores for all 3 estimated source signals (blue bars) and the mixture signals (orange bars), while the SIG mean scores are relatively similar across the board, which confirms the observed suppression with minimal distortion.

The corresponding p-values for the ANOVA test are given in Table 5.8. The BAK and OVRL p-values for all three sources are below the 0.05 significance value, which

Table 5.8: One-way ANOVA test between the estimated source signals and original mixture signals on Scenario 1A.

Source		*p*-value		
		SIG ↑	BAK ↓	OVRL ↓
1	Proposed	0.871*	0.0052	0.0058
	Ablation	0.831*	0.0641*	0.0931*
2	Proposed	0.913*	0.0069	0.0072
	Ablation	0.893*	0.0591*	0.1213*
3	Proposed	0.844*	0.0044	0.0051
	Ablation	0.884*	0.0626*	0.0824*

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

suggests a statistically significant difference between the separated and mixture signals in terms of background interference level and overall speech quality. The SIG *p*-values are well above the 0.05 significance level, which suggests that there is no statistically significant difference in terms of signal distortion between the estimated and the mixture signals.

The BAK and OVRL mean scores for the audio-only ablation method (green bars) are much lower than for the proposed audio-visual method, while the SIG mean scores are on par across the board. Furthermore, the BAK and OVRL *p*-values for the ablation are above 0.05 for all sources, which suggests that the audio-only approach produces poor separation performance. In particular, the separated signals produced by the audio-only approach not only have poor interference suppression and overall quality, but are truncated at the start and end of the signals due to late tracking initiation and termination.

Consequently, a co-located audio-visual configuration is capable of performing separation, but is naturally constrained by the limited spatial coverage of the single microphone array. Nonetheless, the use of visual assistance to complement the audio data is still significantly better than an audio-only approach, which is due to vastly improved tracking performance as observed in the previous subsection.

Scenario 2

The mean scores of all 3 criteria, i.e., SIG, BAK and OVRL, are shown in Fig. 5.10, and the results for ANOVA test are given in Table 5.9. A similar trend is observed to Scenario 1A, although now with lower SIG and BAK scores in Scenario 2. As expected, the proposed audio-visual based approach still achieves a small degree of separation but clearly deteriorates as the sources are placed farther away from the microphones.

The results for the audio-only ablation indicate more pronounced failures. The BAK and OVRL means scores for all estimated source signals are low and almost match the scores of the mixture signals. The results of the ANOVA tests also confirm poor separation performance. These failures in the audio-only ablation are expected since the

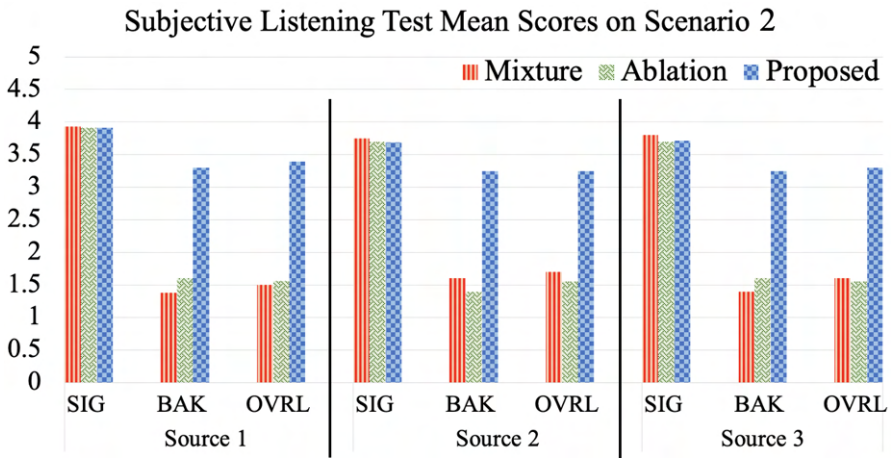


Figure 5.10: Mean scores for SIG, BAK, and OVRL for the estimated source signals and original mixture signals evaluated on Scenario 2.

Table 5.9: One-way ANOVA test between the estimated source signals and original mixture signals on Scenario 2.

Source		*p*-value		
		SIG ↑	BAK ↓	OVRL ↓
1	Proposed	0.811*	0.0077	0.0081
	Ablation	0.781*	0.2542*	0.3415*
2	Proposed	0.753*	0.0091	0.0089
	Ablation	0.803*	0.3218*	0.4035*
3	Proposed	0.714*	0.0072	0.0074
	Ablation	0.694*	0.2966*	0.3211*

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

effectiveness of the GSC beamformer is highly dependent on the accuracy of the tracking estimates, which in this case have large localization errors, in addition to extraneous and missing tracks, as well as late initiations and terminations.

In short, while the proposed audio-visual tracking maintains accuracy when sources are farther away, the separation performance degrades with increasing distance between the sources and the single microphone array. However, compared to using audio-only where the separation fails due to erroneous tracking information, the audio-visual approach still maintains consistency in the output.

5.6.6 Additional Near-field Experiments

In the previous subsections, it was observed that near-field performance (Scenario 1A) was markedly better than far-field performance (Scenario 2), in all aspects of measurements, tracking, and separation. It was also observed via the ablation studies that audio-visual based separation is much more effective than audio-only separation. We now further explore the audio-visual near-field case with two additional scenarios as described below.

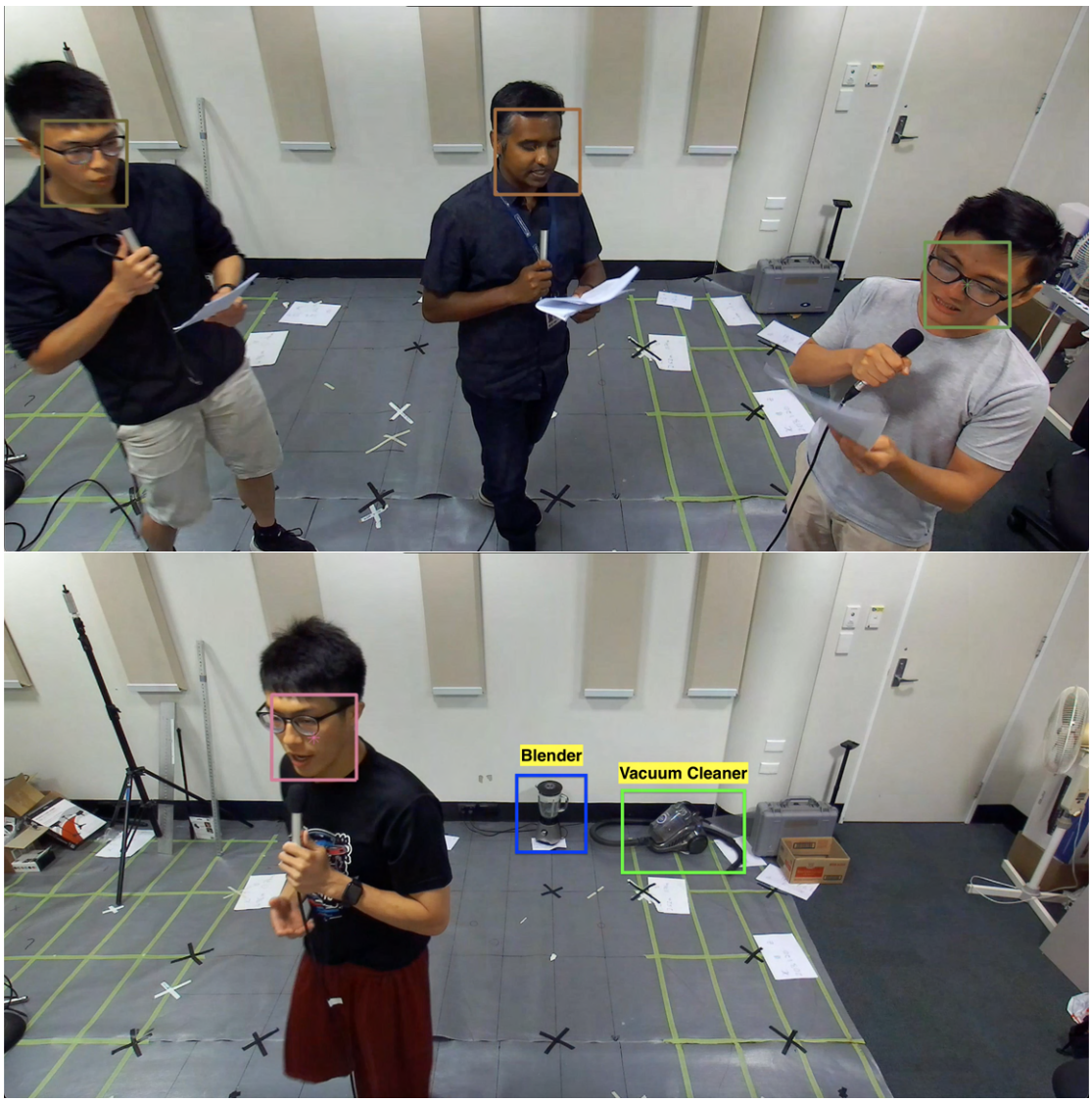


Figure 5.11: Screenshots of Scenario 1B (top) and Scenario 1C (bottom).

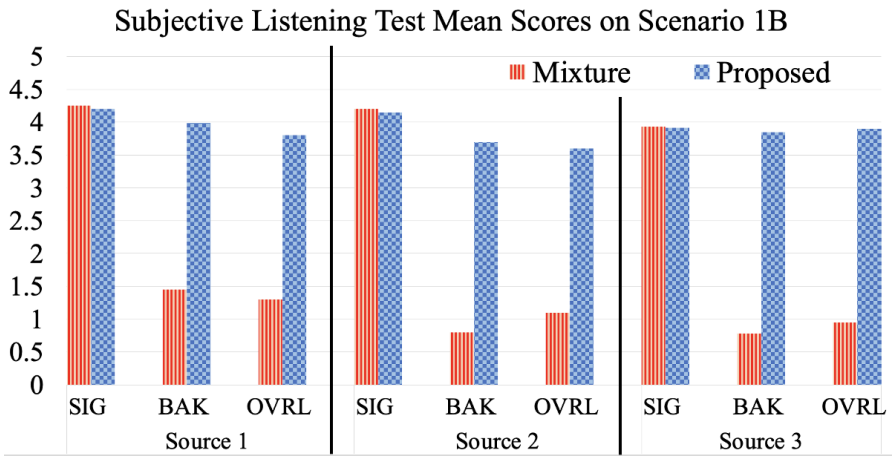


Figure 5.12: Mean scores for SIG, BAK, and OVRL for the estimated source signals and original mixture signals evaluated on Scenario 1B.

Table 5.10: One-way ANOVA test between the estimated source signals and original mixture signals on Scenario 1B.

Source		*p*-value		
		SIG ↑	BAK ↓	OVRL ↓
1	Proposed	0.891*	0.0057	0.0061
2	Proposed	0.853*	0.0041	0.0044
3	Proposed	0.824*	0.0039	0.0051

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

In Scenario 1B, three distinct sources enter the scene at different times, and all are moving while they are speaking. In Scenario 1C, the source enters mid-scenario but its audio is obscured by background noise from a blender and a vacuum cleaner in the room. In both cases the algorithm has no knowledge of the number of sources or the times of their entry. The objective is to separate the mixture of an unknown and time varying number of moving sources.

The screenshots in Fig. 5.11 illustrate the setup of the two additional scenarios. Due to space constraints we omit the evaluation of the measurements and tracking, as well as the ablation study with audio-only measurements. We only present the evaluation of the separation in a similar manner to Section 5.6.5. All video/audio files for the additional scenarios are available via GitHub: https://github.com/researchwork888/AVseparation.

Scenario 1B (Time-varying Number of Speakers)

The mean scores of all 3 criteria, i.e., SIG, BAK and OVRL, are shown in Fig. 5.12, and the results for ANOVA test are given in Table 5.10. The mean scores and *p*-values of the OVRL and BAK criteria suggest that all three estimated sources achieve good overall speech quality with moderate interference suppression, and similarly the mean scores and *p*-values of the SIG component indicate there is minimal signal degradation

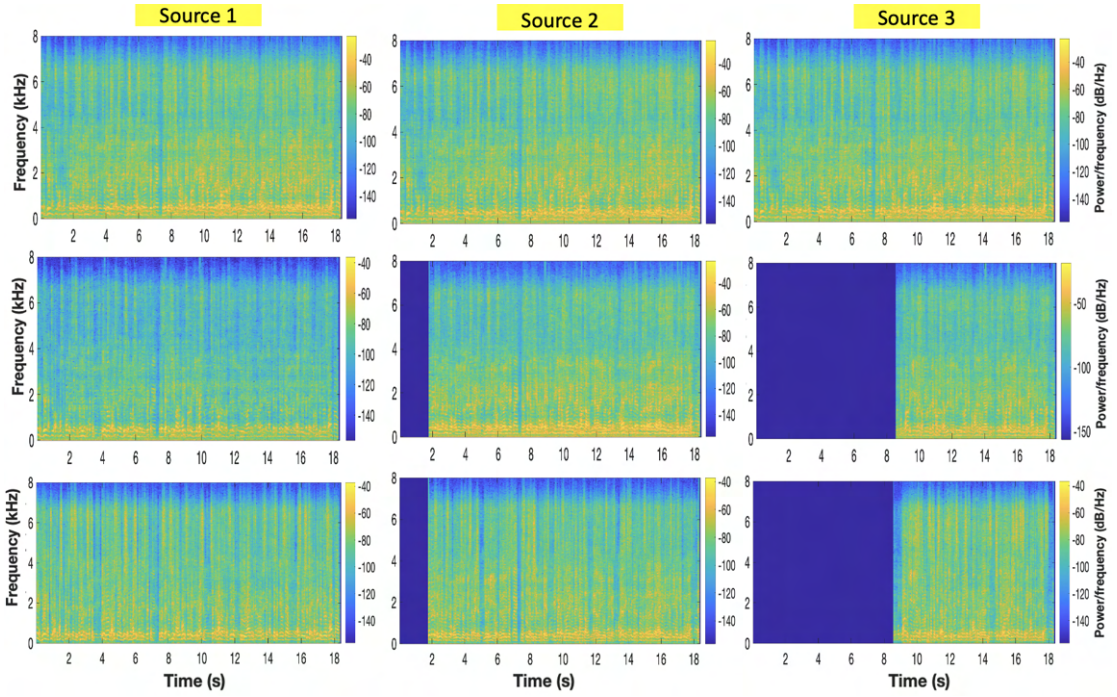


Figure 5.13: Spectrograms for signals from Scenario 1B. Top row: mixtures; middle row: estimated signals; bottom row: ground-truth signals.

or distortion. Additionally, the spectrograms for each of estimated signals are presented in Fig. 5.13. In this scenario, Source 2 enters the scene a few seconds after Source 1, and Source 3 first appears a few seconds after Source 2. Examination of the spectrograms confirms that the proposed method is able to detect and track all three sources from the point they each enter the scene. As a result, the individual signals for each of the three sources is reconstructed correctly. It is also important to point out that there are no identity switches in the estimation of the trajectories of the sources, which is necessary for the correct reconstruction of the three uninterrupted waveforms. Overall, the results of this scenario demonstrate that the proposed method can handle an unknown and time-varying number of moving sources.

Scenario 1C (Loud Background Noise)

The mean scores of all 3 criteria, i.e., SIG, BAK and OVRL, are shown in Fig. 5.15, and the results for ANOVA test are given in Table 5.11. Additionally, the spectrograms of the obtained signals are presented in Fig. 5.14. The results indicate that the proposed method is able to detect and track Source 1 quite accurately, and as a consequence, is able to achieve moderate noise suppression with close to no signal distortion. The onset of the source at the two second mark is also correctly initiated with negligible delay, even in the presence of background noise. This is largely due to the exploitation of the complementary audio and visual modes. The results indicate that the proposed method is able to identify the presence, and enhance the speech signal of the moving speaker, with both a blender and vacuum cleaner running simultaneously in the background.

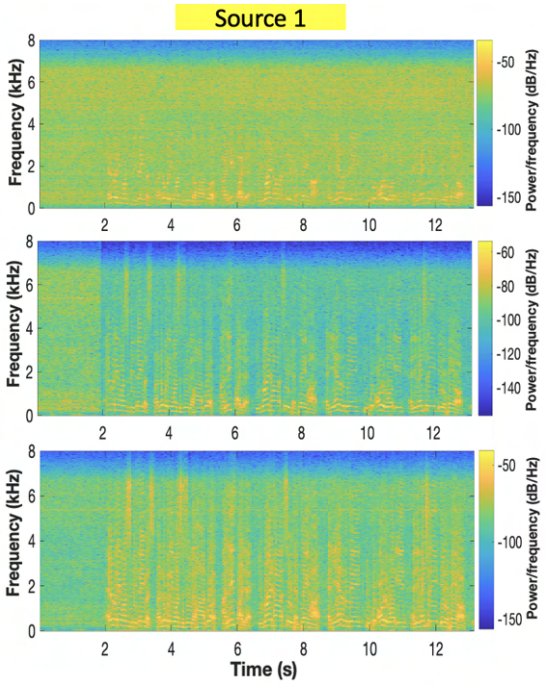


Figure 5.14: Spectrograms for signals from Scenario 1C. Top row: mixtures; middle row: estimated signals; bottom row: ground-truth signals.

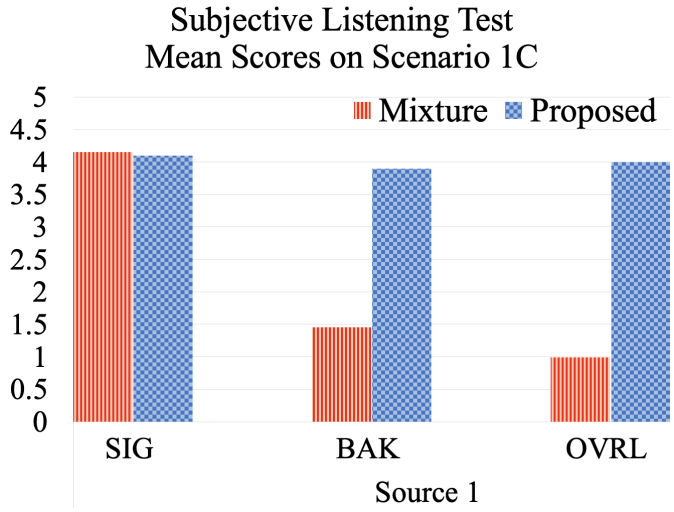


Figure 5.15: Mean scores for SIG, BAK, and OVRL for the estimated source signals and original mixture signals evaluated on Scenario 1C.

Table 5.11: One-way ANOVA test between the estimated source signals and original mixture signals on Scenario 1C.

Source		*p*-value		
		SIG ↑	BAK ↓	OVRL ↓
1	Proposed	0.841*	0.0097	0.0088

The asterisk (*) denotes values that are above the selected significance level, i.e., 0.05. (↑ means higher is better while ↓ means lower is better.)

5.7 Conclusion

This chapter proposes a solution for online separation of an unknown and time-varying number of moving sources, based on a model-centric approach involving sequential stages of detection, tracking, and spatial filtering. The solution exploits simultaneous audio and video measurements, taken from a single microphone array co-located with a single visual device, to produce complementary measurements of an active scene. A labeled random finite set model describes the underlying statistical relationship between the audio-visual measurements and the multi-source states, including the inherent multi-modal space-time permutation uncertainty. A Multi-Sensor GLMB filter is applied to resolve the permutation problem and recursively estimate the source trajectories and labels. A corresponding time-varying set of generalized side-lobe cancellers then performs online source separation.

The proposed solution is evaluated in a real experimental setting with up to 3 live and moving human speakers. An ablation study on audio-only data without the visual mode confirms audibly poor performance due to limited observability with a single microphone array. With the addition of a co-located visual sensor, in near-field experiments, we demonstrate that multi-source separation is possible, despite the limited spatial coverage of the single microphone array. For far-field experiments, the performance is considerably reduced, but still maintains consistency in the output. In both near-field and far-field experiments, the audio-visual approach demonstrably outperforms the audio-only approach. The proposed combination of audio-visual modes is easily extended to the case of multiple visual devices with multiple microphone arrays, which should significantly improve separation performance.

Chapter 6

Conclusion and Future Works

In this dissertation, online audio-visual separation for multiple sources where the number of sources is time-varying and unknown, is achieved using the three-step approach of detection, tracking, and (spatial) filtering (DTF). In a dynamic multi-source scenario, the construction of a time-varying set of spatial filters to separate each source signal and suppress interfering sources requires knowledge of every source position and its unique label at each time frame. During each frame, audio measurements obtained from peaks of a cross-correlation or power-response function are either in the time or space domain respectively, while visual measurements obtained from object detectors lie in the 2D image domain. It is clear that both types of measurements are different entities of multiple observation spaces that are not common to the source state space. Since these measurements are typically unlabeled, it is not known how measurements of different modes are associated together with respect to an observed source across space and time. Moreover, the measurements are subject to missing measurements, false measurements, and noise, in addition to the sources themselves being subject to unknown movement, appearance, and disappearance.

These factors give rise to the multi-modal space-time permutation ambiguity problem (data association problem), which must be resolved in order to inform the construction of spatial filters for separation (in a blind multi-source condition). To address the problem, a joint audio-visual stochastic model that captures the relationship between the audio-visual measurements and the source states and a Bayesian mechanism to solve the inherent multi-modal space-time permutation problem are required. The labeled RFS framework provides a principled mechanism for combining multiple modalities in a statistically consistently manner. Therefore, the framework facilitates the specification of the joint audio-visual stochastic model that encapsulates the uncertainties in the audio and visual measurements, as well as their respective physical relationships to the sources. In support of building the model, this dissertation has proposed a labeled RFS-based audio model for multi-source tracking and separation and a labeled RFS-based visual model for MOT with occlusion handling. The joint audio-visual model is then developed in a principled manner.

In short, the proposed solutions revolve around the multi-sensor generalized labeled multi-Bernoulli (MS-GLMB) tracking filter which facilitates an online estimation of the positions and labels of the sources jointly. The MS-GLMB filter consists of the multi-source dynamic model (which characterizes the dynamics of the source movements, appearance and disappearance in a statistically consistent manner) and the multi-source measurement model (which characterizes the nature of the measurements and is amenable to multiple modalities given that a physical relationship between the source states and the measurements for each mode can be established). The MS-GLMB filter has a complexity that scales linearly with the total number of audio-visual measurements and sensors, which is well-suited for online audio-visual applications. Based on the online estimates, a corresponding spatial filter can then be constructed to perform source separation all in an online fashion. The following subsections present the conclusions for the audio DTF approach, visual MOT, and audio-visual DTF approach.

6.1 Audio Multi-Source Tracking and Separation

In Chapter 3, based on a direct-path signal model, multiple microphone arrays are used for online separation involving multiple sources where the number of sources is time-varying and unknown. The SRP-PHAT localization method is applied on real acoustic recordings from a mild reverberation room to obtain the position candidates of the sources, which are referred to as audio measurements. These measurements are unlabeled, noisy, and containing many missing and false measurements as indicated via the OSPA metric. Moreover, the sources are subject to unknown movement, disappearance, and appearance. Consequently, these factors give rise to the space-time permutation problem, because the associations between sources and the measurements from multiple arrays across space and time are unknown.

The MS-GLMB tracking filter is adopted to address the space-time permutation issue. The tracking filter is derived based on the labeled RFS framework. The source evolutions and kinematics and the nature of the audio measurements are modeled in a statistically consistent manner via the RFS-based multi-source transition and measurement models. Despite the imperfect source measurements from real-world data, it has been demonstrated that the MS-GLMB filter is able to estimate the source trajectories with some delay in initiation and termination, as indicated via the $\text{OSPA}^{(2)}$ metric. Finally, a time-varying set of GSCs (one for each source present) is used to perform source separation and suppression. Further interference suppression is achieved via a post-processing step (i.e., time-frequency masking). The estimated signals acquired before post-processing and after post-processing are evaluated using the ITU-T P.835 based listening tests.

Results from the listening tests show that the proposed method can produce well-separated source signals with minimal signal distortions. The post-processing step fur-

ther enhances interference suppression but introduces some signal distortion (musical noise) in the estimated signals. Additional experiments using the image source model (ISM) for room simulation indicate strong separation performance in low reverberation, matching performance with the real-world setup in mild reverberation, and poor performance in higher reverberation. Such results are in line with the use of the direct-path signal model.

6.2 Visual Multi-Object Tracking with Occlusion Handling

In Chapter 4, multiple cameras were used to perform online MOT in full 3D. State-of-the-art multi-view tracking methods have relied on a data-centric multi-camera detector that requires expensive training and retraining for any multi-camera system extension or reconfiguration. In contrast, the proposed online MOT solution is based on monocular detector training, thereby avoiding any training process when there is an extension or a reconfiguration in the multi-camera system. Further, the solution is able to operate uninterrupted during a camera failure.

The unknown time-varying number of moving (human) objects and the nature of the image-domain visual measurements from multiple cameras are encapsulated by the RFS-based multi-object transition and multi-camera measurement models, respectively. The physical relationship between the objects in 3D and visual measurements in 2D is established via the camera matrix, which can be obtained using standard camera calibration techniques. In addition, a tractable 3D detection model that serves to inform the detection probabilities is incorporated into the measurement likelihood function, enabling the modified MS-GLMB filter to retain occluded tracks correctly, all while maintaining scalability in the number of visual detections and cameras.

The proposed online method was evaluated on the latest multi-camera dataset called WILDTRACKS [57], which only provides the ground truths on the 2D ground plane. Results indicate comparable results with the best-performing batch (offline) method of WILDTRACKS. The proposed method was further evaluated on the new full 3D CMC dataset, which has varying degrees of crowded scenarios. Results indicate that in an extremely crowded scenario when some people are not visible by the cameras, the performance of the proposed tracker drops. This is expected since the number of severe occlusion increases, which become more difficult to resolve. The ablation study on the proposed 3D detection model indicates that the algorithm fails to retain tracks even with mild occlusion when the probability of detection for each object is not correctly assigned. Moreover, evaluations on this dataset have shown that the proposed filter can accommodate for reconfigurations in the multi-camera system. Lastly, results indicate that the proposed algorithm can be extended via a Jump Markov System (JMS) for

tracking people falling and jumping.

6.3 Audio-Visual Multi-Source Tracking and Separation

In Chapter 5, the task of separating moving live human sources (where the number of sources is time-varying and unknown) using both audio and visual data is performed using a single microphone array co-located with a single visual device. The proposed approach is online and model-centric involving the sequential stages of audio and visual detection, tracking and spatial filtering. Standard detection algorithms are applied to both audio and visual data to produce complementary measurements of the scene of interest. These measurements are unlabeled and do not reside in the same observation space. Additionally, they are susceptible to noise, false measurements, and missing measurements. As a result, these issues give rise to the multi-modal space-time permutation problem, since the associations between measurements of both modes and sources across space and time are unknown.

The labeled RFS framework provides a principled and systematic mechanism for fusing both audio and visual measurements in a statistically consistent manner. The labeled RFS framework is employed to specify the multi-source dynamic model that captures the nature of multiple sources where the number of sources is time-varying and unknown. The framework also specifies the multi-modal measurement model that captures the stochastic uncertainties and imperfections of the audio and visual measurements, as well as the physical relationship between the visual measurements and sources through a 3D-to-2D camera (transformation) model. Collectively, the MS-GLMB filter resolves the multi-modal space-time permutation problem and recursively estimates the labels and positions of the sources. Finally, a time-varying set of GSCs is utilized to perform separation, all in an online fashion.

The proposed approach was evaluated on live and moving human sources in two scenarios, a near-field condition and a far-field condition. An ablation study was carried out for each scenario whereby the measurements, tracking and separation were performed using the audio data only. The OSPA evaluations for both scenarios reveal that the audio measurements are highly erroneous, mainly due to the lack of observability with only a single microphone array. The $\text{OSPA}^{(2)}$ evaluations for both scenarios highlight the limitation on the observability of the source trajectories with a single microphone array but suggest that the multi-modal combination of audio and video measurements enables accurate multi-source tracking. Lastly, the listening test results confirm that in near-field experiments, the proposed method exhibits modest separation performance, while for far-field experiments, the performance is limited. Nonetheless, the ablation study confirms poor separation performance for both scenarios.

6.4 Future Works

This dissertation has developed a solution to the inherent multi-modal space-time permutation problem in audio-visual separation of a time-varying and unknown number of moving sources. This problem has been explored in the three core chapters of this dissertation, i.e., by using only audio measurements (Chapter 3), then visual measurements (Chapter 4), and finally audio-visual measurements (Chapter 5).

The following delineates future directions of the proposed solutions that are beyond the current scope of this dissertation. These are given in the interest of improving the algorithm as well as pointing out the shortcomings in the proposed solutions for further development.

6.4.1 Audio Multi-Source Tracking and Separation

The multichannel localization model assumed in the current algorithm is based on a generic time-delay model that indirectly estimates the direction-of-arrival (DOA) for each microphone element via the time difference of arrival. This effectively makes the localization algorithm a far-field model where directionality is the only spatial component about the source. In scenarios where sources are in the near-field region of a microphone array, it is advantageous to apply a near-field source localization approach to improve tracking performance. A near-field localization model assumes that wavefronts impinging on a sensor array are curved as opposed to planar (far-field), requiring both ranges and arrival angles of the sources to be estimated. The addition of range component decreases ambiguity in audio measurements, which should lead to better track estimates and consequently separation results via the proposed filters. A 3D near-field signal model for localization has been established by various works [412–414]. However, localizing multiple unknown sources entails decomposing a spatial covariance matrix into orthogonal subspaces which ultimately requires a multi-dimensional search and is often not feasible without knowing the number of sources [415, 416]. Therefore, future work may research on near-field solutions that can blindly estimate the 3D ranges and arrival angles of the sources. To robustly localize multiple sources under high reverberation, various works have exploited a fundamental principle whereby all signal onsets are dominated by the direct path [417–419]. Given a pair of microphone signals, the method in [418, 419] has introduced a so-called direct-path relative transfer function (DP-RTF) feature in the TF domain, and proposed a consistency test to either retain the DP-RTF feature in a given TF bin that is associated to one of the active sources or disregard it. A complex Gaussian mixture model is used to cluster the selected TF bins and localization is achieved by selecting Gaussian components with large weights. In a blind scenario with reverberation up to 500ms, their results have shown that the method outperformed SRP-PHAT in all aspects of localization error, missed detection rate and false alarm rate [418]. Given that the approach operates on a frame-by-frame basis and

is able to blindly localize multiple sources via peak selection, it would be interesting to adapt this algorithm for an array of microphones and use the obtained measurements to improve tracking performance under high reverberation.

6.4.2 Visual Multi-Object Tracking with Occlusion Handling

Chapter 4 has proposed a 3D multi-view multi-object tracking algorithm with a novel detection model capable of maintaining smooth track estimates in spite of occluded and closely-spaced human objects. In a realistic people tracking scenario, a person may exit the scene and re-enters after a period of time. In this circumstance, the algorithm may erroneously assign a new label/identity to the returning object as the proposed algorithm only processes spatial measurements without any appearance information of the object. Hence, it would be interesting to investigate integrating an appearance model into the current labeled RFS tracking framework, which would enable a formulation of a likelihood for the state labels given some form of statistical representation of visual features of the objects. A survey of appearance models in visual object tracking can be found in [420]. Appearance features such as patterns, object contours, and colors, are key to defining the uniqueness of an object. A recent work relevant to this proposition is the Siamese neural network for online object tracking [421, 422]. One of the salient features of a Siamese network in object tracking is the ability to use the available data online to adjust the weights of a pre-trained network [421]. A learning strategy proposed in [423] has enabled a Siamese network to effectively learn background suppression and target appearance variation from previous frames. Since each hypothesis for a track in the GLMB has a stored online memory of all its past (measured) visual features, it is conceptually feasible to leverage this information to build an appearance model that updates its discriminative power over time [424].

6.4.3 Audio-Visual Multi-Source Tracking and Separation

Chapter 5 has proposed a audio-visual multi-source separation algorithm that processes audio and video measurements, taken from a single microphone array co-located with a single visual device, to track and separate multiple moving sources. Results have demonstrated better tracking and separation performance in a near-field setting as compared to a far-field setting. To improve both the tracking and separation performance for a larger field, a future direction may include distributing multiple visual devices and multiple microphone arrays in the tracking field. This is feasible because the MS-GLMB filter has a complexity that scales linearly with the total number of measurements across all sensors. The extension to multiple distributed cameras and microphones makes the work interesting, as more cameras and microphones observing the scenario from different vantage points provide more significant information for both tracking and separation. This also enables the integration of the proposed occlusion

model in Chapter 4 and the flexibility of selecting the closest microphone array to a source for beamforming in order to achieve better separation quality. Moreover, future work may investigate and incorporate the abovementioned future works on audio tracking under reverberation and visual tracking with source reappearance. This could potentially culminate in the development of an audio-visual multi-source separation solution that is robust against reverberation and source reappearance.

Appendix A

Derivation of The Shadow Region Indicator

Let an object of labeled state $\boldsymbol{x} = (x, \ell)$ be an axis-aligned ellipsoid, where $x \triangleq (x^{(p)}, x^{(s)})$, $x^{(p)}$ is a vector representing the centroid, and $x^{(s)}$ is a vector containing the half-lengths of the ellipsoid's principal axes. From Section 4.3.1 of Chapter 4, the shadow region indicator function is given by

$$1_{S^{(c)}(\boldsymbol{x}')}(\boldsymbol{x}) = \begin{cases} 1, & \left(\mathcal{B}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'}\right)^2 - 4\mathcal{A}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} C^{(c)}_{\boldsymbol{x}'} \geq 0 \\ 0, & \text{otherwise} \end{cases}, \tag{A.1}$$

where

$$\mathcal{A}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} = (x^{(p)} - u^{(c)})^T \left(\text{diag}(x^{(s)'})\right)^{-2} (x^{(p)} - u^{(c)}), \tag{A.2}$$

$$\mathcal{B}^{(c)}_{\boldsymbol{x},\boldsymbol{x}'} = (x^{(p)} - u^{(c)})^T \left[2\left(\text{diag}(x^{(s)'})\right)^{-2} u^{(c)} + \hbar_{\boldsymbol{x}'}\right], \tag{A.3}$$

$$C^{(c)}_{\boldsymbol{x}'} = (u^{(c)})^T \left[\left(\text{diag}(x^{(s)'})\right)^{-2} u^{(c)} + \hbar_{\boldsymbol{x}'}\right] + \mathcal{E}_{\boldsymbol{x}'}, \tag{A.4}$$

$$\hbar_{\boldsymbol{x}'} = -2\frac{x^{(p)'}}{(x^{(s)'} \cdot x^{(s)'})}, \; \mathcal{E}_{\boldsymbol{x}'} = \left\| x^{(p)'} / x^{(s)'} \right\|_2^2 - 1, \tag{A.5}$$

and $u^{(c)}$ is the position of camera c, with multiplication/division of two vectors of the same dimension to be understood as point-wise multiplication/division. The derivation of (A.1) is given in the following.

Consider a set of objects $\boldsymbol{X}$ and a camera of index c, the region occupied by a labeled state $\boldsymbol{x}' \in \boldsymbol{X}$ is given by [380]:

$$R(\boldsymbol{x}') = \{\alpha \in \mathbb{R}^3 : \alpha^T \Lambda_{x^{(s)'}} \alpha + \hbar_{x'}{}^T \alpha + \mathcal{E}_{x'} \leq 0\}, \tag{A.6}$$

where

$$\Lambda_{x^{(s)\prime}} = \left(\text{diag}(x^{(s)\prime})\right)^{-2}, \tag{A.7}$$

$$\hbar_{x'} = -2\frac{x^{(p)\prime}}{(x^{(s)\prime} \cdot x^{(s)\prime})}, \tag{A.8}$$

$$\mathcal{E}_{x'} = \left\| x^{(p)\prime} / x^{(s)\prime} \right\|_2^2 - 1. \tag{A.9}$$

The shadow region of an object with labeled state $\boldsymbol{x'} \in \boldsymbol{X}$, relative to camera c, is given by (as described in Section 4.2.5 of Chapter 4):

$$\mathcal{S}^{(c)}(\boldsymbol{x'}) = \left\{ \alpha \in \mathbb{R}^3 : \overline{(u^{(c)}, \alpha)} \cap R(\boldsymbol{x'}) \neq \emptyset \right\}, \tag{A.10}$$

where $\overline{(u^{(c)}, \alpha)} \triangleq \{\chi\alpha + (1-\chi)u^{(c)} : \chi \in [0,1]\}$ is the line segment joining the position $u^{(c)}$ of camera c and α.

The indication of $\boldsymbol{x} \in \boldsymbol{X}$ falling in the shadow region of $\boldsymbol{x'}$, i.e. $1_{\mathcal{S}^{(c)}(\boldsymbol{x'})}(\boldsymbol{x}) = 1$, is determined by whether the line segment between $u^{(c)}$ and $x^{(p)}$ of $\boldsymbol{x}$, crosses the boundary of the ellipsoid $\boldsymbol{x'}$ twice. Therefore, substituting the line $\overline{(u^{(c)}, x^{(p)})}$ into the ellipsoid boundary equation of $\boldsymbol{x'}$ yields [425]:

$$\begin{aligned} &\left((x^{(p)} - u^{(c)})^T \Lambda_{x^{(s)\prime}} (x^{(p)} - u^{(c)})\right) \chi^2 + \\ &\qquad \left((x^{(p)} - u^{(c)})^T \left[2\Lambda_{x^{(s)\prime}} u^{(c)} + \hbar_{x'}\right]\right) \chi + \\ &\qquad\qquad \left((u^{(c)})^T \left[\Lambda_{x^{(s)\prime}} u^{(c)} + \hbar_{x'}\right] + \mathcal{E}_{x'}\right) = 0, \end{aligned} \tag{A.11}$$

where $\Lambda_{x^{(s)\prime}}$, $\hbar_{x'}$, and $\mathcal{E}_{x'}$ are specified in (A.7), (A.8), and (A.9) respectively.

Consequently, the indicator function is specified by determining whether the roots of (A.11) are real, or equivalently, checking whether the discriminant of the (A.11) is a positive number, i.e.

$$1_{\mathcal{S}^{(c)}(\boldsymbol{x'})}(\boldsymbol{x}) = \begin{cases} 1 & \text{if the discriminant of (A.11) is a positive number} \\ 0 & \text{otherwise} \end{cases}. \tag{A.12}$$

Appendix B

Derivation of The Object-to-Detection Transformation

Let an object of labeled state $\boldsymbol{x} = (x, \ell)$ be an axis-aligned ellipsoid, where $x \triangleq (x^{(p)}, x^{(s)})$, $x^{(p)}$ is a vector representing the centroid, and $x^{(s)}$ is a vector containing the half-lengths of the ellipsoid's principal axes. The transformation $\Upsilon^{(c)}(\boldsymbol{x})$ in the measurement likelihood from Section 4.2.5 of Chapter 4 has the following closed form

$$\Upsilon^{(c)}(\boldsymbol{x}) \triangleq \mathcal{Z}(\mathcal{P}^{(c)}(\boldsymbol{x})), \tag{B.1}$$

where

$$\mathcal{P}^{(c)}(\boldsymbol{x}) = \left(\mathrm{P}^{(c)}_{3\times4} \left[\begin{array}{c:c} (\mathrm{diag}(x^{(s)}))^{-2} & \hbar_{\boldsymbol{x}}/2 \\ \hdashline \hbar_{\boldsymbol{x}}^{T}/2 & \mathcal{E}_{\boldsymbol{x}} \end{array} \right]^{-1} (\mathrm{P}^{(c)}_{3\times4})^{T} \right)^{-1}, \tag{B.2}$$

$$\hbar_{\boldsymbol{x}} = -2 \frac{x^{(p)}}{(x^{(s)} \cdot x^{(s)})}, \tag{B.3}$$

$$\mathcal{E}_{\boldsymbol{x}} = \left\| x^{(p)} / x^{(s)} \right\|_2^2 - 1. \tag{B.4}$$

$$\mathcal{Z}\left(\left[\begin{array}{c:c} \mathrm{A} & r \\ \hdashline r^{T} & e \end{array} \right] \right) = \left[\begin{array}{c} -Q\mathrm{D}^{-1}Q^{T}r \\ 2\nu \left\| [1,0]Q\mathrm{D}^{-0.5} \right\|_2 \\ 2\nu \left\| [0,1]Q\mathrm{D}^{-0.5} \right\|_2 \end{array} \right], \tag{B.5}$$

$$\nu = (r^{T}Q\mathrm{D}^{-1}Q^{T}r - e)^{0.5}, \tag{B.6}$$

Q is a matrix containing the eigenvectors of A, and D is a diagonal matrix of the eigenvalues of A. The derivations for $\mathcal{P}^{(c)}(\cdot)$ and $\mathcal{Z}(\cdot)$ are given in the following.

The function $\mathcal{P}^{(c)}(\cdot)$ is a matrix-to-matrix projection that transforms the quadric $\Psi_{\boldsymbol{x}}$ into a conic $C_x^{(c)}$ on each image of camera c. In the homogeneous coordinate system,

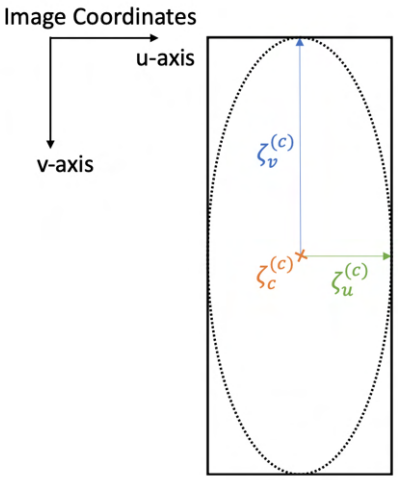


Figure B.1: Illustration of a conic to bounding box transformation.

an axis-aligned ellipsoid $\boldsymbol{x}$ is a quadric represented by [380]:

$$\Psi_{\boldsymbol{x}} = \left[\begin{array}{c:c} (\mathrm{diag}(x^{(s)}))^{-2} & \hbar_{\boldsymbol{x}}/2 \\ \hdashline \hbar_{\boldsymbol{x}}^T/2 & \mathcal{E}_{\boldsymbol{x}} \end{array}\right], \tag{B.7}$$

The perspective projection of a quadric (B.7) under the *camera matrix* $\mathrm{P}_{3\times4}^{(c)}$ of camera c, results in a conic represented by a symmetric 3×3 matrix $C_x^{(c)}$ [380, pp. 201]:

$$C_x^{(c)} = \left(\mathrm{P}_{3\times4}^{(c)} \Psi_{\boldsymbol{x}}^{-1} (\mathrm{P}_{3\times4}^{(c)})^T\right)^{-1}, \tag{B.8}$$

The function $\mathcal{Z}(\cdot)$ is a matrix-to-vector transformation that transforms the conic $C_x^{(c)}$ into a 4D bounding box specified by its center and extent (width and height):

$$\mathcal{Z}(C_x^{(c)}) = \begin{bmatrix} \zeta_{x,c_1}^{(c)} \\ \zeta_{x,c_2}^{(c)} \\ 2\left\|(\zeta_{x,u_1}, \zeta_{x,v_1})\right\|_2 \\ 2\left\|(\zeta_{x,u_2}, \zeta_{x,v_2})\right\|_2 \end{bmatrix}, \tag{B.9}$$

where $\zeta_{x,c}^{(c)} = (\zeta_{x,c_1}^{(c)}, \zeta_{x,c_2}^{(c)})^T$ is the centroid of the conic, and $\zeta_{x,u}^{(c)} = (\zeta_{x,u_1}^{(c)}, \zeta_{x,u_2}^{(c)})^T$, $\zeta_{x,v}^{(c)} = (\zeta_{x,v_1}^{(c)}, \zeta_{x,v_2}^{(c)})^T$ are the orthogonal half-length vectors of the conic respectively (see Fig. B.1). To compute $\zeta_{x,c}^{(c)}, \zeta_{x,u}^{(c)}, \zeta_{x,v}^{(c)} \in \mathbb{R}^2$ from $C_x^{(c)}$, the conic

$$C_x^{(c)} = \left[\begin{array}{c:c} \mathrm{A} & r \\ \hdashline r^T & e \end{array}\right], \tag{B.10}$$

is expressed in its polynomial form:

$$\alpha^T \mathrm{A}\alpha + (2r)^T\alpha + e = 0, \tag{B.11}$$

where $\alpha \in \mathbb{R}^2$ dummy/free variable, A is a 2×2 positive symmetric definite matrix, r is a vector and e is a number. Subsequently, we conduct a change of basis to obtain the standard form of the ellipse, so that the ellipse centroid and lengths of the half-axes can be identified.

Based on eigendecomposition, the positive symmetric definite matrix A can be decomposed into, $\mathrm{A} = Q\mathrm{D}Q^{-1}$, where Q is an orthogonal matrix containing the eigenvectors of A, and D is a diagonal matrix of the eigenvalues of A. Substituting this into (B.11), yields

$$\left(\alpha^T Q\right)\mathrm{D}\left(Q^{-1}\alpha\right) + (2r)^T\alpha + e = 0. \tag{B.12}$$

Let $\alpha' = \left(Q^{-1}\alpha\right)$ be the dummy variable in this new coordinate system, then the equation (B.12) becomes

$$(\alpha')^T\mathrm{D}(\alpha') + \left((2r)^T Q\right)\alpha' + e = 0. \tag{B.13}$$

By completing the square of the above equation, and arranging the terms, we get the standard form of an ellipse (note: an inverse of an orthogonal matrix is equal to its transpose):

$$\frac{\left(\alpha'_1 - \left(-\mathrm{D}^{-1}Q^T r\right)_1\right)^2}{\left(\left(r^T Q\mathrm{D}^{-1}Q^T r - e\right)^{-0.5}\left\|[1,0]\mathrm{D}^{-0.5}\right\|_1\right)^2} + \frac{\left(\alpha'_2 - \left(-\mathrm{D}^{-1}Q^T r\right)_2\right)^2}{\left(\left(r^T Q\mathrm{D}^{-1}Q^T r - e\right)^{-0.5}\left\|[0,1]\mathrm{D}^{-0.5}\right\|_1\right)^2} = 1, \tag{B.14}$$

where the subscripts 1 and 2 denote the x-axis and y-axis of the vector respectively, and $\|\cdot\|_n$ is the n-norm. With the standard form of an ellipse, it is straightforward to identify the ellipse's centroid and lengths of the half-axes.

To retrieve the centroid vector $\zeta_{x,c}^{(c)}$ in the original coordinate system, we undo the transformation by multiplying with the matrix Q, i.e.

$$\zeta_{x,c}^{(c)} = -Q\mathrm{D}^{-1}Q^T r. \tag{B.15}$$

Subsequently, the two orthogonal half-length vectors of the conic $\zeta_{x,u}^{(c)}, \zeta_{x,v}^{(c)}$ are obtained by scaling the eigenvectors by the length of the half-axes:

$$\left[\zeta_{x,u}^{(c)}, \zeta_{x,v}^{(c)}\right] = \left(r^T Q\mathrm{D}^{-1}Q^T r - e\right)^{-0.5} Q\mathrm{D}^{-0.5}. \tag{B.16}$$

Consequently, substituting (B.15) and (B.16) into (B.9) yields:

$$\mathcal{Z}(C_x^{(c)}) = \begin{bmatrix} -Q\mathrm{D}^{-1}Q^T r \\ 2\left(r^T Q\mathrm{D}^{-1}Q^T r - e\right)^{-0.5} \left\| [1,0]Q\mathrm{D}^{-0.5} \right\|_2 \\ 2\left(r^T Q\mathrm{D}^{-1}Q^T r - e\right)^{-0.5} \left\| [0,1]Q\mathrm{D}^{-0.5} \right\|_2 \end{bmatrix}. \tag{B.17}$$

Appendix C

OSPA/OSPA$^{(2)}$ Metrics

Consider a space $\mathbb{W}$ with $\underline{d} : \mathbb{W}\times\mathbb{W} \to [0;\infty)$ as the *base-distance* between the elements of $\mathbb{W}$. Let $\underline{d}^{(c)}(x,y) = \min\left(c, \underline{d}(x,y)\right)$, and Π_n be the set of permutations of $\{1,2,...,n\}$. The Optimal Sub-Pattern Assignment (OSPA) distance of order $p \geq 1$, and cut-off $c > 0$, between two finite sets of points $X = \{x_1,...,x_m\}$ and $Y = \{y_1,...,y_n\}$ of $\mathbb{W}$ is defined by [50]

$$d_0^{(p,c)}(X,Y) = \left(\frac{1}{n}\left(\min_{\pi\in\Pi_n}\sum_{i=1}^{m}\underline{d}^{(c)}\left(x_i, y_{\pi(i)}\right)^p + c^p(n-m)\right)\right)^{\frac{1}{p}}, \tag{C.1}$$

if $n \geq m > 0$, and $d_0^{(p,c)}(X,Y) = d_0^{(p,c)}(Y,X)$ if $m > n > 0$. In addition, $d_0^{(p,c)}(X,Y) = c$ if one of the set is empty, and $d_0^{(p,c)}(\emptyset,\emptyset) = 0$. The integer p plays the same role as the order of the ℓ_p-distance for vectors. The cut-off parameter c provides a weighting between cardinality and location errors. A large c emphasizes cardinality error while a small c emphasizes location error. However, a small c also decreases the sensitivity to the separation between the points due to the saturation of $\underline{d}^{(c)}$ at c.

The OSPA$^{(2)}$ distance between two sets of tracks is the OSPA distance with the following base-distance between two tracks f and g [51]:

$$\underline{d}^{(c)}(f,g) = \sum_{t\in\mathcal{D}_f\cup\mathcal{D}_g}\frac{d_0^{(c)}(\{f(t)\},\{g(t)\})}{\left|\mathcal{D}_f\cup\mathcal{D}_g\right|}, \tag{C.2}$$

if $\mathcal{D}_f \cup \mathcal{D}_g \neq \emptyset$, and $\underline{d}^{(c)}(f,g) = 0$ if $\mathcal{D}_f \cup \mathcal{D}_g = \emptyset$, where $\mathcal{D}_f \cup \mathcal{D}_g$ denotes the set of instants when at least one of the tracks exists, and $d_0^{(c)}(\{f(t)\},\{g(t)\})$ denotes the OSPA distance between the two sets containing the states of the two tracks at time t (the set $\{f(t)\}$ (or $\{g(t)\}$) would be empty if the track f (or g) does not exist at time t). Note that the order parameter p of the OSPA distance in (C.2) is redundant because only sets of at most one element are considered.

Appendix D

Intersection-over-Union (IoU) and Generalized IoU (GIoU) Metrics

For bounding boxes x, y, the IoU similarity index is given by

$$IoU(x, y) = |x \cap y|/|x \cup y| \in [0; 1], \tag{D.1}$$

where $|\cdot|$ denotes hyper-volume, while the Generalized IoU index is given by

$$GIoU(x, y) = IoU(x, y) - |C(x \cup y) \setminus (x \cup y)|/|C(x \cup y)|, \tag{D.2}$$

where $C(x \cup y)$ is the convex hull of $x \cup y$ [65]. The metric forms of IoU and GIoU, respectively are $\underline{d}_{IoU}(x, y) = 1 - IoU(x, y)$ and $\underline{d}_{GIoU}(x, y) = \frac{1-GIoU(x,y)}{2}$, both of which are bounded by one [384].

Appendix E

Monocular Detector Results

Table E.2 shows the CLEAR evaluation for detections on the CMC dataset, which is referenced from Section 4.4.3 and Section 4.4.4 of Chapter 4. Table E.1 shows the CLEAR evaluation for detections on WILDTRACKS dataset, which is referenced from Section 4.4.2 of Chapter 4.

Table E.1: CLEAR Evaluation for Detection Results on WILDTRACKS Dataset

	Detector	MODA ↑	MODP ↑	Precision ↑	Recall ↑
C1	YOLOv3	12.2%	70.1%	0.55	0.62
	F-RCNN(VGG16)	-17.1%	69.6%	0.44	0.62
C2	YOLOv3	31.7%	68.5%	0.68	0.58
	F-RCNN(VGG16)	28.4%	68.3%	0.67	0.57
C3	YOLOv3	-24.4%	69.2%	0.42	0.68
	F-RCNN(VGG16)	-34.6%	69.0%	0.40	0.69
C4	YOLOv3	-272.4%	71.1%	0.14	0.57
	F-RCNN(VGG16)	-300.0%	70.1%	0.14	0.57
C5	YOLOv3	-94.4%	70.0%	0.29	0.69
	F-RCNN(VGG16)	-113.0%	67.8%	0.27	0.71
C6	YOLOv3	-12.6%	63.4%	0.44	0.50
	F-RCNN(VGG16)	-30.5%	65.4%	0.39	0.53
C7	YOLOv3	-79.2%	70.1%	0.33	0.77
	F-RCNN(VGG16)	-100.3%	69.3%	0.31	0.77
All	Deep-Occlusion	74.1%	53.8%	0.95	0.80

Table E.2: CLEAR Evaluation for Detection Results on CMC1 to CMC5

CMC1	Detector	MODA ↑	MODP ↑	Prcn ↑	Rcll ↑
Cam1	YOLOv3	20.6%	80.2%	0.56	0.97
	F-RCNN(VGG16)	12.0%	80.3%	0.53	0.97
Cam2	YOLOv3	20.5%	78.8%	0.56	0.97
	F-RCNN(VGG16)	12.0%	80.1%	0.53	0.98
Cam3	YOLOv3	13.2%	79.7%	0.53	0.97
	F-RCNN(VGG16)	10.1%	80.8%	0.51	0.97
Cam4	YOLOv3	12.1%	79.7%	0.51	0.96
	F-RCNN(VGG16)	11.1%	80.3%	0.41	0.96
CMC2	Detector	MODA ↑	MODP ↑	Prcn ↑	Rcll ↑
Cam1	YOLOv3	51.2%	76.2%	0.77	0.72
	F-RCNN(VGG16)	37.5%	76.5%	0.67	0.73
Cam2	YOLOv3	45.3%	76.5%	0.72	0.72
	F-RCNN(VGG16)	35.5%	76.6%	0.66	0.73
Cam3	YOLOv3	43.4%	77.2%	0.71	0.72
	F-RCNN(VGG16)	34.4%	77.2%	0.66	0.72
Cam4	YOLOv3	47.3%	77.7%	0.74	0.71
	F-RCNN(VGG16)	37.4%	78.0%	0.67	0.72
CMC3	Detector	MODA ↑	MODP ↑	Prcn ↑	Rcll ↑
Cam1	YOLOv3	44.9%	76.4%	0.79	0.60
	F-RCNN(VGG16)	33.1%	76.0%	0.67	0.61
Cam2	YOLOv3	39.8%	75.3%	0.73	0.62
	F-RCNN(VGG16)	30.9%	75.4%	0.66	0.63
Cam3	YOLOv3	36.1%	74.4%	0.72	0.58
	F-RCNN(VGG16)	29.6%	74.0%	0.66	0.61
Cam4	YOLOv3	37.0%	74.9%	0.72	0.59
	F-RCNN(VGG16)	27.6%	74.6%	0.65	0.60
CMC4	Detector	MODA ↑	MODP ↑	Prcn ↑	Rcll ↑
Cam1	YOLOv3	86.8%	82.0%	0.93	0.93
	F-RCNN(VGG16)	76.7%	82.6%	0.84	0.94
Cam2	YOLOv3	75.2%	79.1%	0.87	0.88
	F-RCNN(VGG16)	68.3%	80.3%	0.82	0.88
Cam3	YOLOv3	86.7%	84.6%	0.93	0.93
	F-RCNN(VGG16)	77.3%	87.0%	0.84	0.95
Cam4	YOLOv3	81.5%	82.7%	0.94	0.87
	F-RCNN(VGG16)	75.9%	82.2%	0.82	0.97

CMC5	Detector	MODA ↑	MODP ↑	Prcn ↑	Rcll ↑
Cam1	YOLOv3	48.7%	75.1%	0.77	0.68
	F-RCNN(VGG16)	50.3%	74.8%	0.71	0.69
Cam2	YOLOv3	49.8%	75.6%	0.66	0.65
	F-RCNN(VGG16)	45.3%	76.4%	0.67	0.61
Cam3	YOLOv3	50.7%	73.1%	0.65	0.66
	F-RCNN(VGG16)	44.7%	74.3%	0.65	0.65
Cam4	YOLOv3	49.8%	76.2%	0.65	0.68
	F-RCNN(VGG16)	46.7%	74.1%	0.61	0.69

Appendix F

Statements of Contribution

Publication 1

To whom it may concern, I, Jonah Ong Soon Xuan, contributed to the theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper titled:

J. Ong, B. T. Vo and S. Nordholm, "Blind Separation for Multiple Moving Sources With Labeled Random Finite Sets," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 2137-2151, 2021, doi: 10.1109/TASLP.2021.3087003.

_____

(Jonah Ong Soon Xuan)

The co-authors listed below contributed towards theoretical developments, drafting, and editing the paper, suggesting the design of the experiments, surveying for suitable existing techniques for comparisons, and providing insights on the evaluation of the source separation results.

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

_____

(Ba Tuong Vo)

_____

(Sven Nordholm)

Publication 2

To whom it may concern, I, Jonah Ong Soon Xuan, contributed to the theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper titled:

J. Ong, B. -T. Vo, B. -N. Vo, D. Y. Kim and S. Nordholm, "A Bayesian Filter for Multi-view 3D Multi-object Tracking with Occlusion Handling," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, Issue: 5, pp. 2246-2263, 2022, doi: 10.1109/TPAMI.2020.3034435.

_____

(Jonah Ong Soon Xuan)

The co-authors listed below contributed by way of making theoretical developments, drafting and editing the paper, documenting the novel occlusion model and tracking filter, providing insights into state-of-the-art visual detection and tracking algorithms and surveying related datasets for experimental comparisons.

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

_____

(Ba Tuong Vo)

_____

(Ba-Ngu Vo)

_____

(Du Yong Kim)

_____

(Sven Nordholm)

Publication 3

To whom it may concern, I, Jonah Ong Soon Xuan, contributed to the theoretical development of the algorithm, implementation (MATLAB), evaluation and writing of the paper titled:

J. Ong, B. T. Vo, S. Nordholm, B. -N. Vo, D. Moratuwage and C. Shim, "Audio-Visual Based Online Multi-Source Separation," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30, pp. 1219-1234, 2022, doi: 10.1109/TASLP.2022.3156758.

_____

(Jonah Ong Soon Xuan)

B.T. Vo, S. Nordholm and B.N. Vo contributed towards theoretical developments, drafting and editing the paper. D. Moratuwage and C. Shim contributed by way of recording the dataset, providing insights into carrying out the experimental evaluations and editing the paper.

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

_____ _____

(Ba Tuong Vo) (Changbeom Shim)

_____

(Sven Nordholm)

_____

(Ba-Ngu Vo)

_____

(Diluka Moratuwage)

Publication 4

To whom it may concern, I, Jonah Ong Soon Xuan, contributed to the theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper titled:

J. Ong, D. Y. Kim and S. Nordholm, "Multi-sensor Multi-target Tracking Using Labelled Random Finite Sets with Homography Data," 2019 International Conference on Control, Automation and Information Sciences (ICCAIS), 2019, pp. 1-7, doi: 10.1109/ICCAIS46528.2019.9074716.

_____

(Jonah Ong Soon Xuan)

The co-authors listed below contributed by way of editing the paper and proposing ideas for the experiments.

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

_____

(Du Yong Kim)

_____

(Sven Nordholm)

Publication 5

To whom it may concern, I, Jonah Ong Soon Xuan, contributed to the theoretical development of the algorithm, implementation (MATLAB), evaluation and drafting of the paper titled:

J. Ong, D. Y. Kim and C.T. Do, "A Tractable Multi-Target Detection Model for Line-of-Sight Measurements," 2021 International Conference on Control, Automation and Information Sciences (ICCAIS), 2021, doi: 10.1109/ICCAIS52680.2021.9624664.

_____

(Jonah Ong Soon Xuan)

The co-authors contributed by way of editing the paper and proposing ideas for the experiments.

I, as a co-author, endorse that this level of contribution by the candidate indicated above is appropriate.

_____

(Du Yong Kim)

_____

(Cong-Thanh Do)

References

[1] S. T. Shivappa, M. M. Trivedi, and B. D. Rao, "Audiovisual information fusion in human-computer interfaces and intelligent environments: A survey," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1692–1715, 2010.

[2] S. Choi, A. Cichocki, H.-M. Park, and S.-Y. Lee, "Blind source separation and independent component analysis: A review," *Neural Information Processing-Letters and Reviews*, vol. 6, no. 1, pp. 1–57, 2005.

[3] R. Gribonval and S. Lesage, "A survey of sparse component analysis for blind source separation: principles, perspectives, and new challenges," in *ESANN'06 proceedings-14th European Symposium on Artificial Neural Networks*, 2006, pp. 323–330.

[4] Z. Luo, C. Li, and L. Zhu, "A comprehensive survey on blind source separation for wireless adaptive processing: Principles, perspectives, challenges and new research directions," *IEEE Access*, vol. 6, pp. 66 685–66 708, 2018.

[5] X. Weng and K. Kitani, "A baseline for 3D multi-object tracking," *arXiv preprint arXiv:1907.03961*, 2019.

[6] G. Ciaparrone, F. L. 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, 2020.

[7] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim, "Multiple object tracking: A literature review," *Artificial Intelligence*, p. 103448, 2020.

[8] 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 Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 1620–1643, 2020.

[9] K. Imoto and N. Ono, "Spatial cepstrum as a spatial feature using a distributed microphone array for acoustic scene analysis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1335–1343, 2017.

[10] C. Evers and P. A. Naylor, "Acoustic SLAM," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1484–1498, 2018.

[11] K. Reindl, S. Meier, H. Barfuss, and W. Kellermann, "Minimum mutual information-based linearly constrained broadband signal extraction," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 22, no. 6, pp. 1096–1108, 2014.

[12] S. Markovich-Golan, S. Gannot, and W. Kellermann, "Combined LCMV-TRINICON beamforming for separating multiple speech sources in noisy and reverberant environments," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 2, pp. 320–332, 2016.

[13] S. Harding, J. Barker, and G. J. Brown, "Mask estimation for missing data speech recognition based on statistics of binaural interaction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 58–67, 2005.

[14] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[15] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[16] L. Leal-Taixé, A. Milan, K. Schindler, D. Cremers, I. Reid, and S. Roth, "Tracking the trackers: an analysis of the state of the art in multiple object tracking," *arXiv preprint arXiv:1704.02781*, 2017.

[17] H. Zhu, M.-D. Luo, R. Wang, A.-H. Zheng, and R. He, "Deep audio-visual learning: A survey," *International Journal of Automation and Computing*, pp. 1–26, 2021.

[18] D. Michelsanti, Z.-H. Tan, S.-X. Zhang, Y. Xu, M. Yu, D. Yu, and J. Jensen, "An overview of deep-learning-based audio-visual speech enhancement and separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[19] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, 2013.

[20] B.-N. Vo, B.-T. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6554–6567, 2014.

[21] B.-N. Vo, B.-T. Vo, and H. G. Hoang, "An efficient implementation of the generalized labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1975–1987, 2017.

[22] B.-N. Vo, B.-T. Vo, and M. Beard, "Multi-sensor multi-object tracking with the generalized labeled multi-bernoulli filter," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5952–5967, 2019.

[23] S. Makino, T.-W. Lee, and H. Sawada, *Blind speech separation*. Springer, 2007, vol. 615.

[24] X. Yu, D. Hu, and J. Xu, *Blind source separation: theory and applications*. John Wiley & Sons, 2013.

[25] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[26] O. Yilmaz and S. Rickard, "Blind separation of speech mixtures via time-frequency masking," *IEEE Transactions on signal processing*, vol. 52, no. 7, pp. 1830–1847, 2004.

[27] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 3, pp. 1066–1074, 2007.

[28] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 31–35.

[29] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901–1913, 2017.

[30] R. Lu, Z. Duan, and C. Zhang, "Listen and look: Audio–visual matching assisted speech source separation," *IEEE Signal Processing Letters*, vol. 25, no. 9, pp. 1315–1319, 2018.

[31] J. Wu, Y. Xu, S.-X. Zhang, L.-W. Chen, M. Yu, L. Xie, and D. Yu, "Time domain audio visual speech separation," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 667–673.

[32] M. T. Akhtar, T.-P. Jung, S. Makeig, and G. Cauwenberghs, "Recursive independent component analysis for online blind source separation," in *2012 IEEE*

International Symposium on Circuits and Systems (ISCAS). IEEE, 2012, pp. 2813–2816.

[33] G. Zhou, Z. Yang, S. Xie, and J.-M. Yang, "Online blind source separation using incremental nonnegative matrix factorization with volume constraint," *IEEE transactions on neural networks*, vol. 22, no. 4, pp. 550–560, 2011.

[34] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, "A real-time blind source separation scheme and its application to reverberant and noisy acoustic environments," *Signal Processing*, vol. 86, no. 6, pp. 1260–1277, 2006.

[35] N. Grbic, X.-J. Tao, S. E. Nordholm, and I. Claesson, "Blind signal separation using overcomplete subband representation," *IEEE transactions on speech and audio processing*, vol. 9, no. 5, pp. 524–533, 2001.

[36] H. Buchner, R. Aichner, and W. Kellermann, "A generalization of a class of blind source separation algorithms for convolutive mixtures," in *Proc. ICA*. Citeseer, 2003, pp. 945–950.

[37] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, "Real-time convolutive blind source separation based on a broadband approach," in *International Conference on Independent Component Analysis and Signal Separation*. Springer, 2004, pp. 840–848.

[38] X.-L. Zhu, X.-D. Zhang, Z.-Z. Ding, and Y. Jia, "Adaptive nonlinear PCA algorithms for blind source separation without prewhitening," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 3, pp. 745–753, 2006.

[39] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 246–250.

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

[41] D. Kounades-Bastian, L. Girin, X. Alameda-Pineda, S. Gannot, and R. Horaud, "A variational EM algorithm for the separation of time-varying convolutive audio mixtures," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 8, pp. 1408–1423, 2016.

[42] J. Nikunen, A. Diment, and T. Virtanen, "Separation of moving sound sources using multichannel NMF and acoustic tracking," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 281–295, 2017.

[43] A. Masnadi-Shirazi and B. D. Rao, "An ICA-SCT-PHD filter approach for tracking and separation of unknown time-varying number of sources," *IEEE Transactions on audio, speech, and language processing*, vol. 21, no. 4, pp. 828–841, 2012.

[44] M. Taseska and E. A. Habets, "Blind source separation of moving sources using sparsity-based source detection and tracking," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 657–670, 2017.

[45] S. Chan, Z. Zhu, K. Ng, C. Wang, S. Zhang, and Z. Zhang, "A movable image-based rendering system and its application to multiview audio-visual conferencing," in *2010 10th International Symposium on Communications and Information Technologies*. IEEE, 2010, pp. 1142–1145.

[46] H. K. Maganti, D. Gatica-Perez, and I. McCowan, "Speech enhancement and recognition in meetings with an audio–visual sensor array," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2257–2269, 2007.

[47] T. Yoshioka, I. Abramovski, C. Aksoylar, Z. Chen, M. David, D. Dimitriadis, Y. Gong, I. Gurvich, X. Huang, Y. Huang *et al.*, "Advances in online audio-visual meeting transcription," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 276–283.

[48] Y. Liu, V. Kılıç, J. Guan, and W. Wang, "Audio-visual particle flow SMC-PHD filtering for multi-speaker tracking," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 934–948, 2019.

[49] X. Qian, A. Brutti, O. Lanz, M. Omologo, and A. Cavallaro, "Audio-visual tracking of concurrent speakers," *IEEE Transactions on Multimedia*, 2021.

[50] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, 2008.

[51] M. Beard, B. T. Vo, and B.-N. Vo, "A solution for large-scale multi-object tracking," *IEEE Trans. on Signal Process.*, vol. 68, pp. 2754–2769, 2020.

[52] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 229–238, 2007.

[53] J. Ong, B. T. Vo, and S. E. Nordholm, "Blind separation for multiple moving sources with labeled random finite sets," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[54] S. L. Dockstader and A. M. Tekalp, "Multiple camera fusion for multi-object tracking," in *Proc. 2001 IEEE Workshop on Multi-Object Tracking*. IEEE, 2001, pp. 95–102.

[55] T. Chavdarova and F. Fleuret, "Deep multi-camera people detection," in *2017 16th IEEE Int. Conf. Mac. Learning and Appplicat. (ICMLA)*. IEEE, 2017, pp. 848–853.

[56] P. Baqué, F. Fleuret, and P. Fua, "Deep occlusion reasoning for multi-camera multi-target detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 271–279.

[57] T. Chavdarova *et al.*, "WILDTRACK: A multi-camera HD dataset for dense unscripted pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5030–5039.

[58] J. Domke, "Learning graphical model parameters with approximate marginal inference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 10, pp. 2454–2467, 2013.

[59] J. Ong, B.-T. Vo, B.-N. Vo, D. Y. Kim, and S. Nordholm, "A Bayesian filter for multi-view 3D multi-object tracking with occlusion handling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2246–2263, 2022.

[60] S. E. Nordholm, H. H. Dam, C. C. Lai, and E. A. Lehmann, "Broadband beamforming and optimization," in *Academic Press Library in Signal Processing*. Elsevier, 2014, vol. 3, pp. 553–598.

[61] ©2022 IEEE Reprinted with permission from, J. Ong, B. T. Vo, S. Nordholm, B.-N. Vo, D. Moratuwage, and C. Shim, "Audio-visual based online multi-source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1219–1234, 2022.

[62] J. Benesty, J. Chen, and Y. Huang, *Microphone array signal processing*. Springer Science & Business Media, 2008, vol. 1.

[63] M. Brandstein and D. Ward, *Microphone arrays: signal processing techniques and applications*. Springer Science & Business Media, 2013.

[64] H. Buchner, R. Aichner, and W. Kellermann, "TRINICON: A versatile framework for multichannel blind signal processing," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3. IEEE, 2004, pp. iii–889.

[65] S. Y. Low, S. Nordholm, and R. Togneri, "Convolutive blind signal separation with post-processing," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 539–548, 2004.

[66] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, "Audio source separation by activity probability detection with maximum correlation and simplex geometry," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2021, no. 1, pp. 1–16, 2021.

[67] M. S. Brandstein and H. F. Silverman, "A practical methodology for speech source localization with microphone arrays," *Computer Speech & Language*, vol. 11, no. 2, pp. 91–126, 1997.

[68] M. Swartling, B. Sällberg, and N. Grbić, "Source localization for multiple speech sources using low complexity non-parametric source separation and clustering," *Signal Processing*, vol. 91, no. 8, pp. 1781–1788, 2011.

[69] D. B. Ward, E. A. Lehmann, and R. C. Williamson, "Particle filtering algorithms for tracking an acoustic source in a reverberant environment," *IEEE Transactions on speech and audio processing*, vol. 11, no. 6, pp. 826–836, 2003.

[70] H. Do, H. F. Silverman, and Y. Yu, "A real-time SRP-PHAT source location implementation using stochastic region contraction (src) on a large-aperture microphone array," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 1. IEEE, 2007, pp. I–121.

[71] E. A. Lehmann and R. C. Williamson, "Particle filter design using importance sampling for acoustic source localisation and tracking in reverberant environments," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, pp. 1–9, 2006.

[72] W.-K. Ma, B.-N. Vo, S. S. Singh, and A. Baddeley, "Tracking an unknown time-varying number of speakers using TDOA measurements: A random finite set approach," *IEEE Transactions on Signal Processing*, vol. 54, no. 9, pp. 3291–3304, 2006.

[73] X. Zhong and J. R. Hopgood, "Particle filtering for TDOA based acoustic source tracking: Nonconcurrent multiple talkers," *Signal processing*, vol. 96, pp. 382–394, 2014.

[74] T. Adali and S. Haykin, *Adaptive signal processing: next generation solutions*. John Wiley & Sons, 2010, vol. 55.

[75] A. Hyvärinen, "Independent component analysis: recent advances," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 371, no. 1984, p. 20110534, 2013.

[76] K. Nordhausen and H. Oja, "Independent component analysis: A statistical perspective," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 10, no. 5, p. e1440, 2018.

[77] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.

[78] A. Mirzal, "NMF versus ICA for blind source separation," *Advances in Data Analysis and Classification*, vol. 11, no. 1, pp. 25–48, 2017.

[79] J. Bobin, J. Rapin, A. Larue, and J.-L. Starck, "Sparsity and adaptivity for the blind separation of partially correlated sources," *IEEE Transactions on Signal Processing*, vol. 63, no. 5, pp. 1199–1213, 2015.

[80] S. Winter, W. Kellermann, H. Sawada, and S. Makino, "MAP-based underdetermined blind source separation of convolutive mixtures by hierarchical clustering and-norm minimization," *EURASIP J. Adv. Signal Process.*, vol. 2007, no. 1, p. 024717, 2006.

[81] S. Cruces, "Bounded component analysis of linear mixtures: A criterion of minimum convex perimeter," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2141–2154, 2010.

[82] H. A. Inan and A. T. Erdogan, "A convolutive bounded component analysis framework for potentially nonstationary independent and/or dependent sources," *IEEE Transactions on Signal Processing*, vol. 63, no. 1, pp. 18–30, 2014.

[83] S. Cruces and I. Durán-Díaz, "The minimum risk principle that underlies the criteria of bounded component analysis," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 5, pp. 964–981, 2014.

[84] H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of frequency-domain blind source separation," *IEEE transactions on speech and audio processing*, vol. 12, no. 5, pp. 530–538, 2004.

[85] T. Kim, T. Eltoft, and T.-W. Lee, "Independent vector analysis: An extension of ICA to multivariate components," in *International conference on independent component analysis and signal separation*. Springer, 2006, pp. 165–172.

[86] D. Kitamura, N. Ono, H. Sawada, H. Kameoka, and H. Saruwatari, "Determined blind source separation unifying independent vector analysis and nonnegative matrix factorization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1626–1641, 2016.

[87] ——, "Determined blind source separation with independent low-rank matrix analysis," in *Audio source separation*. Springer, 2018, pp. 125–155.

[88] D. Kitamura and K. Yatabe, "Consistent independent low-rank matrix analysis for determined blind source separation," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 1, pp. 1–35, 2020.

[89] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE transactions on audio, speech, and language processing*, vol. 18, no. 3, pp. 550–563, 2009.

[90] A. Cichocki, S.-i. Amari, R. Zdunek, R. Kompass, G. Hori, and Z. He, "Extended SMART algorithms for non-negative matrix factorization," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2006, pp. 548–562.

[91] S. Rickard, "The DUET blind source separation algorithm," in *Blind speech separation*. Springer, 2007, pp. 217–241.

[92] F. Abrard and Y. Deville, "A time–frequency blind signal separation method applicable to underdetermined mixtures of dependent sources," *Signal processing*, vol. 85, no. 7, pp. 1389–1403, 2005.

[93] T. Melia and S. Rickard, "Underdetermined blind source separation in echoic environments using DESPRIT," *EURASIP Journal on advances in Signal Processing*, vol. 2007, pp. 1–19, 2006.

[94] S. Araki, H. Sawada, R. Mukai, and S. Makino, "Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors," *Signal processing*, vol. 87, no. 8, pp. 1833–1847, 2007.

[95] A. Cichocki, J. Karhunen, W. Kasprzak, and R. Vigario, "Neural networks for blind separation with unknown number of sources," *Neurocomputing*, vol. 24, no. 1-3, pp. 55–93, 1999.

[96] S. Araki, T. Nakatani, H. Sawada, and S. Makino, "Blind sparse source separation for unknown number of sources using gaussian mixture model fitting with dirichlet prior," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 33–36.

[97] F. Yin, T. Mei, and J. Wang, "Blind-source separation based on decorrelation and nonstationarity," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 5, pp. 1150–1158, 2007.

[98] U. Manmontri and P. A. Naylor, "A class of Frobenius norm-based algorithms using penalty term and natural gradient for blind signal separation," *IEEE transactions on audio, speech, and language processing*, vol. 16, no. 6, pp. 1181–1193, 2008.

[99] N. Dey and A. S. Ashour, "Microphone array principles," in *Direction of Arrival Estimation and Localization of Multi-Speech Sources*. Springer, 2018, pp. 5–22.

[100] N. Madhu, R. Martin, U. Heute, and C. Antweiler, "Acoustic source localization with microphone arrays," *Advances in Digital Speech Transmission*, pp. 135–170, 2008.

[101] J. Chen, J. Benesty, and Y. A. Huang, "Time delay estimation in room acoustic environments: an overview," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, no. 1, p. 026503, 2006.

[102] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, 1976.

[103] J. H. DiBiase, *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays*. Brown University Providence, RI, 2000.

[104] B. Berdugo, M. A. Doron, J. Rosenhouse, and H. Azhari, "On direction finding of an emitting source from time delays," *the Journal of the Acoustical Society of America*, vol. 105, no. 6, pp. 3355–3363, 1999.

[105] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereati, "Real-time passive source localization: A practical linear-correction least-squares approach," *IEEE transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 943–956, 2001.

[106] H. Cao, Y. T. Chan, and H.-C. So, "Maximum likelihood TDOA estimation from compressed sensing samples without reconstruction," *IEEE Signal Processing Letters*, vol. 24, no. 5, pp. 564–568, 2017.

[107] J. P. Dmochowski, J. Benesty, and S. Affes, "Broadband MUSIC: Opportunities and challenges for multiple source localization," in *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2007, pp. 18–21.

[108] M. Wax and T. Kailath, "Optimum localization of multiple sources by passive arrays," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 5, pp. 1210–1217, 1983.

[109] M. Cobos, A. Marti, and J. J. Lopez, "A modified SRP-PHAT functional for robust real-time sound source localization with scalable spatial sampling," *IEEE Signal Processing Letters*, vol. 18, no. 1, pp. 71–74, 2010.

[110] M. Taseska and E. A. Habets, "Spotforming: Spatial filtering with distributed arrays for position-selective sound acquisition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 7, pp. 1291–1304, 2016.

[111] H. Do and H. F. Silverman, "A fast microphone array SRP-PHAT source location implementation using coarse-to-fine region contraction (cfrc)," in *2007 IEEE workshop on applications of signal processing to audio and acoustics*. IEEE, 2007, pp. 295–298.

[112] H. Pujol, E. Bavu, and A. Garcia, "Beamlearning: an end-to-end deep learning approach for the angular localization of sound sources using raw multichannel acoustic pressure data," *The Journal of the Acoustical Society of America*, vol. 149, no. 6, pp. 4248–4263, 2021.

[113] P.-A. Grumiaux, S. Kitić, L. Girin, and A. Guérin, "A survey of sound source localization with deep learning methods," *arXiv preprint arXiv:2109.03465*, 2021.

[114] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[115] S. Chakrabarty and E. A. Habets, "Broadband DOA estimation using convolutional neural networks trained with noise signals," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2017, pp. 136–140.

[116] ——, "Multi-speaker localization using convolutional neural network trained with noise," *arXiv preprint arXiv:1712.04276*, 2017.

[117] J. M. Vera-Diaz, D. Pizarro, and J. Macias-Guarasa, "Towards end-to-end acoustic localization using deep learning: From audio signals to source position coordinates," *Sensors*, vol. 18, no. 10, p. 3418, 2018.

[118] E. Thuillier, H. Gamper, and I. J. Tashev, "Spatial audio feature discovery with convolutional neural networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6797–6801.

[119] C. Evers, E. A. Habets, S. Gannot, and P. A. Naylor, "DOA reliability for distributed acoustic tracking," *IEEE Signal Processing Letters*, 2018.

[120] X. Zhong and J. R. Hopgood, "Time-frequency masking based multiple acoustic sources tracking applying rao-blackwellised monte carlo data association," in *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*. IEEE, 2009, pp. 253–256.

[121] T. Gehrig and J. McDonough, "Tracking multiple speakers with probabilistic data association filters," in *International Evaluation Workshop on Classification of Events, Activities and Relationships*. Springer, 2006, pp. 137–150.

[122] R. P. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion*. Artech House, 2014.

[123] B.-N. Vo, S. Singh, and W. K. Ma, "Tracking multiple speakers using random sets," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2. IEEE, 2004, pp. ii–357.

[124] B.-N. Vo, W.-K. Ma, and S. Singh, "Localizing an unknown time-varying number of speakers: A Bayesian random finite set approach," in *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 4. IEEE, 2005, pp. iv–1073.

[125] N. T. Pham, W. Huang, and S. H. Ong, "Multiple sensor multiple object tracking with GMPHD filter," in *Information Fusion, 2007 10th International Conference on*. IEEE, 2007, pp. 1–7.

[126] C. Evers, A. H. Moore, P. A. Naylor, J. Sheaffer, and B. Rafaely, "Bearing-only acoustic tracking of moving speakers for robot audition," in *2015 IEEE International Conference on Digital Signal Processing (DSP)*. IEEE, 2015, pp. 1206–1210.

[127] N. T. Pham, W. Huang, and S. Ong, "Tracking multiple speakers using CPHD filter," in *Proceedings of the 15th ACM international conference on Multimedia*, 2007, pp. 529–532.

[128] A. Plinge, D. Hauschildt, M. H. Hennecke, and G. A. Fink, "Multiple speaker tracking using a microphone array by combining auditory processing and a gaussian mixture cardinalized probability hypothesis density filter," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2476–2479.

[129] N. Chong, S. Nordholm, B. T. Vo, and I. Murray, "Tracking and separation of multiple moving speech sources via cardinality balanced multi-target multi-Bernoulli (CBMeMBer) filter and time frequency masking," in *Control, Automation and Information Sciences (ICCAIS), 2016 International Conference on*. IEEE, 2016, pp. 88–93.

[130] X. Zhong and J. R. Hopgood, "A time–frequency masking based random finite set particle filtering method for multiple acoustic source detection and tracking," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 12, pp. 2356–2370, 2015.

[131] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Robust real-time blind source separation for moving speakers in a room," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, vol. 5. IEEE, 2003, pp. V–469.

[132] ——, "Blind source separation for moving speech signals using blockwise ica and residual crosstalk subtraction," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 87, no. 8, pp. 1941–1948, 2004.

[133] J. Málek, Z. Koldovskỳ, and P. Tichavskỳ, "Semi-blind source separation based on ICA and overlapped speech detection," in *International Conference on Latent Variable Analysis and Signal Separation*. Springer, 2012, pp. 462–469.

[134] B. D. Van Veen, W. Van Drongelen, M. Yuchtman, and A. Suzuki, "Localization of brain electrical activity via linearly constrained minimum variance spatial filtering," *IEEE Transactions on biomedical engineering*, vol. 44, no. 9, pp. 867–880, 1997.

[135] R. Talmon, I. Cohen, and S. Gannot, "Convolutive transfer function generalized sidelobe canceler," *IEEE transactions on audio, speech, and language processing*, vol. 17, no. 7, pp. 1420–1434, 2009.

[136] Y. Laufer and S. Gannot, "A Bayesian hierarchical model for blind audio source separation," in *2020 28th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 276–280.

[137] K. Weisberg, B. Laufer-Goldshtein, and S. Gannot, "Simultaneous tracking and separation of multiple sources using factor graph model," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 2848–2864, 2020.

[138] S. J. Prince, *Computer vision: models, learning, and inference*. Cambridge University Press, 2012.

[139] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[140] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128 837–128 868, 2019.

[141] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[142] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[143] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

[144] ——, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[145] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[146] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE conference on computer vision and pattern recognition*. IEEE, 2008, pp. 1–8.

[147] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[148] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *2010 IEEE Computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 2241–2248.

[149] R. Girshick, P. Felzenszwalb, and D. McAllester, "Object detection with grammar models," *Advances in Neural Information Processing Systems*, vol. 24, pp. 442–450, 2011.

[150] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[151] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," 2009.

[152] P. Dollár, S. J. Belongie, and P. Perona, "The fastest pedestrian detector in the west." in *Bmvc*, vol. 2, no. 3. Citeseer, 2010, p. 7.

[153] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Process. Systems*, 2012, pp. 1097–1105.

[154] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[155] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[156] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[157] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[158] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[159] R. Girshick, "Fast R-CNN," in *Proc. of the IEEE Int. Conf. on Comput. Vis.*, 2015, pp. 1440–1448.

[160] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Advances in Neural Onformation Proc. Systems*, 2015, pp. 91–99.

[161] P. O. Pinheiro, R. Collobert, and P. Dollár, "Learning to segment object candidates," *arXiv preprint arXiv:1506.06204*, 2015.

[162] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár, "Learning to refine object segments," in *European conference on computer vision*. Springer, 2016, pp. 75–91.

[163] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: state of the art," *International Journal of Multimedia Information Retrieval*, pp. 1–19, 2020.

[164] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. of the IEEE Int. Conf. on Comput. Vis.*, 2017, pp. 2961–2969.

[165] G. Gkioxari, J. Malik, and J. Johnson, "Mesh R-CNN," in *Proc. of the IEEE Int. Conf. on Comput. Vis.*

[166] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of the IEEE Conf. Comput. Vis. and Pattern Recognit.*, 2016, pp. 779–788.

[167] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[168] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *IEEE Conf. Comput. Vis. and Pattern Recognit.*, 2017, pp. 7263–7271.

[169] ——, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[170] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen, "Repulsion loss: Detecting pedestrians in a crowd," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7774–7783.

[171] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1904–1912.

[172] W. Ouyang, H. Zhou, H. Li, Q. Li, J. Yan, and X. Wang, "Jointly learning deep features, deformable parts, occlusion and classification for pedestrian detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 8, pp. 1874–1887, 2017.

[173] S. Zhang, J. Yang, and B. Schiele, "Occluded pedestrian detection through guided attention in cnns," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6995–7003.

[174] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 267–282, 2008.

[175] P. Peng, Y. Tian, Y. Wang, J. Li, and T. Huang, "Robust multiple cameras pedestrian detection with multi-view Bayesian network," *Pattern Recognition*, vol. 48, no. 5, pp. 1760–1772, 2015.

[176] J. Berclaz, F. Fleuret, and P. Fua, "Multiple object tracking using flow linear programming," in *2009 Twelfth IEEE international workshop on performance evaluation of tracking and surveillance*. IEEE, 2009, pp. 1–8.

[177] A. Alahi, L. Jacques, Y. Boursier, and P. Vandergheynst, "Sparsity driven people localization with a heterogeneous network of cameras," *Journal of Mathematical Imaging and Vision*, vol. 41, no. 1-2, pp. 39–58, 2011.

[178] M. Golbabaee, A. Alahi, and P. Vandergheynst, "Scoop: A real-time sparsity driven people localization algorithm," *Journal of mathematical imaging and vision*, vol. 48, no. 1, pp. 160–175, 2014.

[179] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, no. 3, pp. 334–352, 2004.

[180] J. Candamo, M. Shreve, D. B. Goldgof, D. B. Sapper, and R. Kasturi, "Understanding transit scenes: A survey on human behavior-recognition algorithms," *IEEE transactions on intelligent transportation systems*, vol. 11, no. 1, pp. 206–224, 2009.

[181] W. Choi and S. Savarese, "A unified framework for multi-target tracking and collective activity recognition," in *European Conference on Computer Vision*. Springer, 2012, pp. 215–230.

[182] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1913–1921.

[183] X. Wang, "Intelligent multi-camera video surveillance: A review," *Pattern recognition letters*, vol. 34, no. 1, pp. 3–19, 2013.

[184] H. Uchiyama and E. Marchand, "Object detection and pose tracking for augmented reality: Recent approaches," in *18th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV)*, 2012.

[185] W. Hu, X. Li, W. Luo, X. Zhang, S. Maybank, and Z. Zhang, "Single and multiple object tracking using log-euclidean riemannian subspace and block-division appearance model," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 12, pp. 2420–2440, 2012.

[186] J. Zhang, L. L. Presti, and S. Sclaroff, "Online multi-person tracking by tracker hierarchy," in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*. IEEE, 2012, pp. 379–385.

[187] L. Zhang and L. van der Maaten, "Structure preserving object tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1838–1845.

[188] L. Zhang and L. Van Der Maaten, "Preserving structure in model-free tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 756–769, 2013.

[189] B. Song, T.-Y. Jeng, E. Staudt, and A. K. Roy-Chowdhury, "A stochastic graph evolution framework for robust multi-target tracking," in *European Conference on Computer Vision*. Springer, 2010, pp. 605–619.

[190] Z. Qin and C. R. Shelton, "Improving multi-target tracking via social grouping," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1972–1978.

[191] B. Yang and R. Nevatia, "Multi-target tracking by online learning of non-linear motion patterns and robust appearance models," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1918–1925.

[192] ——, "An online learned CRF model for multi-target tracking," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2034–2041.

[193] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 58–72, 2014.

[194] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua, "Multi-commodity network flow for tracking multiple people," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1614–1627, 2014.

[195] X. Wang, E. Türetken, F. Fleuret, and P. Fua, "Tracking interacting objects using intertwined flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2312–2326, 2016.

[196] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, 2011.

[197] Y. Xu, X. Liu, Y. Liu, and S.-C. Zhu, "Multi-view people tracking via hierarchical trajectory composition," in *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4256–4265.

[198] D. Y. Kim, B.-N. Vo, B.-T. Vo, and M. Jeon, "A labeled random finite set online multi-object tracker for video data," *Pattern Recognition*, vol. 90, pp. 377–389, 2019.

[199] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1820–1833, 2010.

[200] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, 2010.

[201] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2544–2550.

[202] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, 2014.

[203] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *2008 IEEE Conf. Comput. Vis. and Pattern Recognit.* IEEE, 2008, pp. 1–8.

[204] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *Trans. Pat. Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, 2013.

[205] R. Hoseinnezhad, B.-N. Vo, and T. N. Vu, "Visual tracking of multiple targets by multi-Bernoulli filtering of background subtracted image data," in *International Conference in Swarm Intelligence*. Springer, 2011, pp. 509–518.

[206] R. Hoseinnezhad, B.-N. Vo, and B.-T. Vo, "Visual tracking in background subtracted image sequences via multi-Bernoulli filtering," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 392–397, 2013.

[207] T. Rathnayake, A. K. Gostar, R. Hoseinnezhad, and A. Bab-Hadiashar, "Labeled multi-Bernoulli track-before-detect for multi-target tracking in video," in *Information Fusion (Fusion), 2015 18th International Conference on*. IEEE, 2015, pp. 1353–1358.

[208] N. T. Pham, W. Huang, and S. H. Ong, "Tracking multiple objects using probability hypothesis density filter and color measurements," in *2007 IEEE International Conference on Multimedia and Expo*. IEEE, 2007, pp. 1511–1514.

[209] F. Poiesi, R. Mazzon, and A. Cavallaro, "Multi-target tracking on confidence maps: An application to people tracking," *COMPUT VIS IMAGE UND*, vol. 117, no. 10, pp. 1257–1272, 2013.

[210] R. Hoseinnezhad, B.-N. Vo, B.-T. Vo, and D. Suter, "Visual tracking of numerous targets via multi-bernoulli filtering of image data," *Pattern Recognition*, vol. 45, no. 10, pp. 3625–3635, 2012.

[211] M. S. Ryoo and J. K. Aggarwal, "Observe-and-explain: A new approach for multiple hypotheses tracking of humans and objects," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[212] D. Mitzel, E. Horbert, A. Ess, and B. Leibe, "Multi-person tracking with sparse detection and continuous segmentation," in *European Conference on Computer Vision*. Springer, 2010, pp. 397–410.

[213] D. Mitzel and B. Leibe, "Real-time multi-person tracking with detector assisted structure propagation," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011, pp. 974–981.

[214] T. Rathnayake, A. K. Gostar, R. Hoseinnezhad, and A. Bab-Hadiashar, "Occlusion handling for online visual tracking using labeled random set filters," in *2017 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2017, pp. 151–156.

[215] K. Otsuka and N. Mukawa, "Multiview occlusion analysis for tracking densely populated objects based on 2-d visual angles," in *Proc. of the 2004 IEEE Comput. Soc. Conf. on Comput. Vis. and Pattern Recognit.*, vol. 1. IEEE, 2004, pp. I–I.

[216] A. Osep, W. Mehner, M. Mathias, and B. Leibe, "Combined image-and world-space tracking in traffic scenes," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1988–1995.

[217] P. Li, J. Shi, and S. Shen, "Joint spatial-temporal optimization for stereo 3D object tracking," in *Proc. of the IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6877–6886.

[218] M. Pedersen, J. B. Haurum, S. H. Bengtson, and T. B. Moeslund, "3D-ZeF: A 3D zebrafish tracking benchmark dataset," in *Proc. of the IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2426–2436.

[219] M. J. Wainwright, M. I. Jordan *et al.*, "Graphical models, exponential families, and variational inference," *Foundations and Trends® in Machine Learning*, vol. 1, no. 1–2, pp. 1–305, 2008.

[220] D. Frossard and R. Urtasun, "End-to-end learning of multi-sensor 3D tracking by detection," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 635–642.

[221] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, and C. C. Loy, "Robust multi-modality multi-object tracking," in *Proceedings of the IEEE International Conf. on Comput. Vis.*, 2019, pp. 2365–2374.

[222] B. Leibe, K. Schindler, N. Cornelis, and L. Van Gool, "Coupled object detection and tracking from static cameras and moving vehicles," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1683–1698, 2008.

[223] N. T. Pham, W. Huang, and S. Ong, "Probability hypothesis density approach for multi-camera multi-object tracking," in *Asian Conference on Computer Vision*. Springer, 2007, pp. 875–884.

[224] N. T. Pham, R. Chang, K. Leman, T. W. Chua, and Y. Wang, "Random finite set for data association in multiple camera tracking," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 1357–1360.

[225] A. K. Gostar, T. Rathnayake, A. Bab-Hadiashar, G. Battistelli, L. Chisci, and R. Hoseinnezhad, "Centralized multiple-view information fusion for multi-object tracking using labeled multi-Bernoulli filters," in *2018 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2018, pp. 238–243.

[226] T. Rathnayake, R. Tennakoon, A. Khodadadian Gostar, A. Bab-Hadiashar, and R. Hoseinnezhad, "Information fusion for industrial mobile platform safety via track-before-detect labeled multi-Bernoulli filter," *Sensors*, vol. 19, no. 9, p. 2016, 2019.

[227] T. Rathnayake, A. Khodadadian Gostar, R. Hoseinnezhad, R. Tennakoon, and A. Bab-Hadiashar, "On-line visual tracking with occlusion handling," *Sensors*, vol. 20, no. 3, p. 929, 2020.

[228] J. Pu, Y. Panagakis, S. Petridis, and M. Pantic, "Audio-visual object localization and separation using low-rank and sparsity," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2901–2905.

[229] R. Hoseinnezhad, B.-N. Vo, B.-T. Vo, and D. Suter, "Bayesian integration of audio and visual information for multi-target tracking using a cb-member filter," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2300–2303.

[230] Y. Ban, X. Alameda-Pineda, L. Girin, and R. Horaud, "Variational Bayesian inference for audio-visual tracking of multiple speakers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[231] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian Bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[232] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house, 2003.

[233] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part i. Dynamic models," *IEEE Transactions on aerospace and electronic systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[234] J. Vermaak and A. Blake, "Nonlinear filtering for speaker tracking in noisy and reverberant environments," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, vol. 5. IEEE, 2001, pp. 3021–3024.

[235] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking: Part iii. Measurement models," in *Signal and Data Processing of Small Targets 2001*, vol. 4473. International Society for Optics and Photonics, 2001, pp. 423–446.

[236] I. R. Goodman, R. P. Mahler, and H. T. Nguyen, *Mathematics of data fusion*. Springer Science & Business Media, 2013, vol. 37.

[237] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Artech House, Inc., 2007.

[238] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[239] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. Ieee, 2000, pp. 153–158.

[240] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[241] A. Doucet and X. Wang, "Monte Carlo methods for signal processing: a review in the statistical signal processing context," *IEEE Signal Processing Magazine*, vol. 22, no. 6, pp. 152–170, 2005.

[242] O. Cappé, S. J. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 899–924, 2007.

[243] E. Veach, *Robust Monte Carlo methods for light transport simulation*. Stanford University, 1998.

[244] C. P. Robert and G. Casella, "Monte Carlo integration," in *Monte Carlo statistical methods*. Springer, 1999, pp. 71–138.

[245] A. Lee, C. Yau, M. B. Giles, A. Doucet, and C. C. Holmes, "On the utility of graphics cards to perform massively parallel simulation of advanced Monte Carlo methods," *Journal of computational and graphical statistics*, vol. 19, no. 4, pp. 769–789, 2010.

[246] L. M. Murray, A. Lee, and P. E. Jacob, "Parallel resampling in the particle filter," *Journal of Computational and Graphical Statistics*, vol. 25, no. 3, pp. 789–805, 2016.

[247] J. H. Kotecha and P. M. Djuric, "Gaussian sum particle filtering," *IEEE Transactions on signal processing*, vol. 51, no. 10, pp. 2602–2612, 2003.

[248] D. Guo and X. Wang, "Quasi-Monte Carlo filtering in nonlinear dynamic systems," *IEEE transactions on signal processing*, vol. 54, no. 6, pp. 2087–2098, 2006.

[249] A. Smith, *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.

[250] S. S. Blackman, "Multiple-target tracking with radar applications," *Dedham*, 1986.

[251] Y. Bar-Shalom and T. E. Fortmann, *Tracking and data association*. Boston Academic Press, 1988. [Online]. Available: http://openlibrary.org/books/OL22317444M

[252] Y. Bar-Shalom and X.-R. Li, *Multitarget-multisensor tracking: principles and techniques*. YBs Storrs, CT, 1995, vol. 19.

[253] S. Blackman and R. Popoli, "Design and analysis of modern tracking systems (artech house radar library)," *Artech house*, 1999.

[254] D. J. Salmond, "Mixture reduction algorithms for target tracking in clutter," in *Signal and Data Processing of Small Targets 1990*, vol. 1305. International Society for Optics and Photonics, 1990, p. 434.

[255] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[256] J. Roecker and G. Phillis, "Suboptimal joint probabilistic data association," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 510–517, 1993.

[257] J. A. Roecker, "A class of near optimal JPDA algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 504–510, 1994.

[258] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for multi-target tracking," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 481–497, 2009.

[259] D. Musicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, pp. 1093–1099, 2004.

[260] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom, "A generalized SD assignment algorithm for multisensor-multitarget state estimation," *IEEE Transactions on Aerospace and Electronic systems*, vol. 33, no. 2, pp. 523–538, 1997.

[261] A. B. Poore and A. J. Robertson III, "A new lagrangian relaxation based algorithm for a class of multidimensional assignment problems," *Computational Optimization and Applications*, vol. 8, no. 2, pp. 129–150, 1997.

[262] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood method for probabilistic multihypothesis tracking," in *Signal and Data Processing of Small Targets 1994*, vol. 2235. International Society for Optics and Photonics, 1994, pp. 394–405.

[263] S. Särkkä, A. Vehtari, and J. Lampinen, "Rao-blackwellized particle filter for multiple target tracking," *Information Fusion*, vol. 8, no. 1, pp. 2–15, 2007.

[264] R. Mahler, *An introduction to multisource-multitarget statistics and applications*. Lockheed Martin, 2000.

[265] J. Goutsias, R. P. Mahler, and H. T. Nguyen, *Random sets: theory and applications*. Springer Science & Business Media, 2012, vol. 97.

[266] B.-N. Vo and B.-T. Vo, "A multi-scan labeled random finite set model for multi-object state estimation," *IEEE Transactions on Signal Processing*, vol. 67, no. 19, pp. 4948–4963, 2019.

[267] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.

[268] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.

[269] T. Vu, B.-N. Vo, and R. Evans, "A particle marginal metropolis-hastings multi-target tracker," *IEEE transactions on signal processing*, vol. 62, no. 15, pp. 3953–3964, 2014.

[270] C. Andrieu, A. Doucet, and R. Holenstein, "Particle markov chain monte carlo methods," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 269–342, 2010.

[271] P. Craciun, M. Ortner, and J. Zerubia, "Joint detection and tracking of moving objects using spatio-temporal marked point processes," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 177–184.

[272] C. Robert, *The Bayesian choice: from decision-theoretic foundations to computational implementation*. Springer Science & Business Media, 2007.

[273] H. Sidenbladh, "Multi-target particle filtering for the probability hypothesis density," *arXiv preprint cs/0303018*, 2003.

[274] T. Zajic and R. P. Mahler, "Particle-systems implementation of the PHD multitarget tracking filter," in *Signal Processing, Sensor Fusion, and Target Recognition XII*, vol. 5096. International Society for Optics and Photonics, 2003, pp. 291–299.

[275] M. Vihola, "Rao-Blackwellised particle filtering in random set multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, pp. 689–705, 2007.

[276] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4091–4104, 2006.

[277] R. Mahler, "PHD filters of higher order in target number," *IEEE Transactions on Aerospace and Electronic systems*, vol. 43, no. 4, 2007.

[278] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3553–3567, 2007.

[279] ——, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 409–423, 2008.

[280] A. M. Johansen, S. S. Singh, A. Doucet, and B.-N. Vo, "Convergence of the SMC implementation of the PHD filte," *Methodology and Computing in Applied Probability*, vol. 8, no. 2, pp. 265–291, 2006.

[281] D. Clark and B.-N. Vo, "Convergence analysis of the Gaussian mixture PHD filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1204–1212, 2007.

[282] F. Lian, C. Li, C. Han, and H. Chen, "Convergence analysis for the SMC-MeMBer and SMC-CBMeMBer filters," *Journal of Applied Mathematics*, vol. 2012, 2012.

[283] D. Eppstein, "Finding the k shortest paths," *SIAM Journal on computing*, vol. 28, no. 2, pp. 652–673, 1998.

[284] K. G. Murthy, "An algorithm for ranking all the assignments in order of increasing costs," *Operations research*, vol. 16, no. 3, pp. 682–687, 1968.

[285] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[286] T. T. D. Nguyen and D. Y. Kim, "GLMB tracker with partial smoothing," *Sensors*, vol. 19, no. 20, p. 4419, 2019.

[287] B. Wei, B. Nener, W. Liu, and L. Ma, "Centralized multi-sensor multi-target tracking with labeled random finite sets," in *2016 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2016, pp. 82–87.

[288] B. Yang, J. Wang, and H. Jiang, "An approximate implementation of multi-sensor generalized labeled multi-Bernoulli filter for multiple target tracking," in *Advanced Information Technology, Electronic and Automation Control Conf.*, vol. 5, 2021, pp. 2661–2666.

[289] S. C. J. Robertson, C. E. van Daalen, and J. A. du Preez, "Efficient approximations of the multi-sensor labelled multi-Bernoulli filter," *arXiv e-prints*, p. arXiv:2103.10396, 2021.

[290] S. Li, G. Battistelli, L. Chisci, W. Yi, B. Wang, and L. Kong, "Computationally efficient multi-agent multi-object tracking with labeled random finite sets," vol. 67, no. 1, pp. 260–275, 2019.

[291] S. Li, W. Yi, R. Hoseinnezhad, G. Battistelli, B. Wang, and L. Kong, "Robust distributed fusion with labeled random finite sets," vol. 66, no. 2, pp. 278–293, 2018.

[292] L. Gao, G. Battistelli, L. Chisci, and A. Farina, "Fusion-based multi-detection multi-target tracking with random finite sets," pp. 1–1, 2021.

[293] L. Gao, G. Battistelli, and L. Chisci, "Fusion of labeled RFS densities with minimum information loss," vol. 68, pp. 5855–5868, 2020.

[294] A.-A. Saucan and P. K. Varshney, "Distributed cross-entropy δ -GLMB filter for multi-sensor multi-target tracking," in *Int. Conf. on Information Fusion*, 2018, pp. 1559–1566.

[295] J. Y. Yu, A.-A. Saucan, M. Coates, and M. Rabbat, "Algorithms for the multi-sensor assignment problem in the δ-generalized labeled multi-Bernoulli filter," in *Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, 2017, pp. 1–5.

[296] G. Zhang, F. Lian, C. Han, H. Chen, and N. Fu, "Two novel sensor control schemes for multi-target tracking via delta generalised labelled multi-Bernoulli filtering," *IET Signal Processing*, vol. 12, no. 9, pp. 1131–1139, 2018.

[297] A. K. Gostar, R. Hoseinnezhad, A. Bab-Hadiashar, and W. Liu, "Sensor-management for multitarget filters via minimization of posterior dispersion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 6, pp. 2877–2884, 2017.

[298] H. Cai, S. Gehly, Y. Yang, R. Hoseinnezhad, R. Norman, and K. Zhang, "Multi-sensor tasking using analytical Rényi divergence in labeled multi-Bernoulli filtering," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 9, pp. 2078–2085, 2019.

[299] N. Ishtiaq, S. Panicker, A. K. Gostar, A. Bab-Hadiashar, and R. Hoseinnezhad, "Selective sensor control via Cauchy Schwarz divergence," in *Smart Trends in Computing and Communications: Proceedings of SmartCom 2020*. Springer Singapore, 2021, pp. 113–124.

[300] S. Panicker, A. K. Gostar, A. Bab-Hadiashar, and R. Hoseinnezhad, "Tracking of targets of interest using labeled multi-Bernoulli filter with multi-sensor control," *Signal Processing*, vol. 171, p. 107451, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016516841930502X

[301] A. K. Gostar, T. Rathnayake, R. Tennakoon, A. Bab-Hadiashar, G. Battistelli, L. Chisci, and R. Hoseinnezhad, "Cooperative sensor fusion in centralized sensor

networks using Cauchy-Schwarz divergence," *Signal Processing*, vol. 167, p. 107278, 2020.

[302] ——, "Centralized cooperative sensor fusion for dynamic sensor network with limited field-of-view via labeled multi-Bernoulli filter," vol. 69, pp. 878–891, 2021.

[303] Y. Zhu, "Efficient sensor management for multitarget tracking in passive sensor networks via Cauchy-Schwarz divergence," *arXiv e-prints*, p. arXiv:2011.08976, 2020.

[304] R. Mahler, "A GLMB filter for unified multitarget multisensor management," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVIII*, vol. 11018, International Society for Optics and Photonics. SPIE, 2019, pp. 123–134. [Online]. Available: https://doi.org/10.1117/12.2520129

[305] Y. Punchihewa, B.-N. Vo, and B.-T. Vo, "A generalized labeled multi-Bernoulli filter for maneuvering targets," in *2016 19th International Conference on Information Fusion (FUSION)*. IEEE, 2016, pp. 980–986.

[306] W. Yi, M. Jiang, and R. Hoseinnezhad, "The multiple model Vo–Vo filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 2, pp. 1045–1054, 2017.

[307] C. Cao and Y. Zhao, "A multiple-model generalized labeled multi-Bernoulli filter based on blocked Gibbs sampling for tracking maneuvering targets," *Signal Processing*, vol. 186, p. 108119, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165168421001572

[308] S. Reuter, A. Scheel, and K. Dietmayer, "The multiple model labeled multi-Bernoulli filter," in *Int. Conf. on Information Fusion*, 2015, pp. 1574–1580.

[309] A. K. Gostar, T. Rathnayake, C. Fu, A. Bab-Hadiashar, G. Battistelli, L. Chisci, and R. Hoseinnezhad, "Interactive multiple-target tracking via labeled multi-Bernoulli filters," in *2019 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2019, pp. 1–6.

[310] F. Papi, B.-N. Vo, B.-T. Vo, C. Fantacci, and M. Beard, "Generalized labeled multi-Bernoulli approximation of multi-object densities," *IEEE Transactions on Signal Processing*, vol. 63, no. 20, pp. 5487–5497, 2015.

[311] S. Li, W. Yi, R. Hoseinnezhad, B. Wang, and L. Kong, "Multiobject tracking for generic observation model using labeled random finite sets," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 368–383, 2018.

[312] R. Mahler, "A fast labeled multi-Bernoulli filter for superpositional sensors," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*, vol. 10646, International Society for Optics and Photonics. SPIE, 2018, pp. 113–124. [Online]. Available: https://doi.org/10.1117/12.2305465

[313] M. Beard, S. Reuter, K. Granström, B.-T. Vo, B.-N. Vo, and A. Scheel, "Multiple extended target tracking with labeled random finite sets." *IEEE Trans. Signal Processing*, vol. 64, no. 7, pp. 1638–1653, 2016.

[314] M. Beard, B.-T. Vo, and B.-N. Vo, "Bayesian multi-target tracking with merged measurements using labelled random finite sets." *IEEE Trans. Signal Processing*, vol. 63, no. 6, pp. 1433–1447, 2015.

[315] B.-N. Vo, B.-T. Vo, N.-T. Pham, and D. Suter, "Joint detection and estimation of multiple objects from image observations," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5129–5141, 2010.

[316] R. Zhu, Y. Long, J. Yang, and W. An, "Multi-sensor multi-object joint detection and tracking from image observations using labeled multi-Bernoulli densities," in *Progress In Electromagnetics Research Symp.*, 2017, pp. 3067–3071.

[317] K. A. LeGrand and K. J. DeMars, "The data-driven δ-generalized labeled multi-Bernoulli tracker for automatic birth initialization," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*, vol. 10646, International Society for Optics and Photonics. SPIE, 2018, pp. 36–55.

[318] S. Lin, B.-T. Vo, and S. E. Nordholm, "Measurement driven birth model for the generalized labeled multi-Bernoulli filter," in *Int. Conf. on Control, Automation and Information Sciences*, 2016, pp. 94–99.

[319] W. J. Park and C. G. Park, "Multi-target tracking based on Gaussian mixture labeled multi-Bernoulli filter with adaptive gating," in *Int. Symp. on Instrumentation, Control, Artificial Intelligence, and Robotics*, 2019, pp. 226–229.

[320] Y. G. Punchihewa, B.-T. Vo, B.-N. Vo, and D. Y. Kim, "Multiple object tracking in unknown backgrounds with labeled random finite sets," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 3040–3055, 2018.

[321] C.-T. Do, T. T. D. Nguyen, and H. V. Nguyen, "Robust multi-sensor GLMB filter: An application to multi-target tracking with bearing-only sensors," *arXiv e-prints*, p. arXiv:2106.00208, 2021.

[322] C. Li, Z. Fan, and R. Shi, "A generalized labelled multi-Bernoulli filter for extended targets with unknown clutter rate and detection profile," vol. 8, pp. 213 772–213 782, 2020.

[323] C.-T. Do, T. T. D. Nguyen, and D. Moratuwage, "Multi-target tracking with an adaptive $\delta-$GLMB filter," *arXiv e-prints*, p. arXiv:2008.00413, 2020.

[324] C.-T. Do, T. T. D. Nguyen, and W. Liu, "Tracking multiple marine ships via multiple sensors with unknown backgrounds," *Sensors*, vol. 19, no. 22, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/22/5025

[325] D. Y. Kim, "Visual multiple-object tracking for unknown clutter rate," *IET Computer Vision*, vol. 12, no. 5, pp. 728–734, 2018.

[326] R. Mahler, "A clutter-agnostic generalized labeled multi-Bernoulli filter," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*, vol. 10646, International Society for Optics and Photonics. SPIE, 2018, pp. 101–112. [Online]. Available: https://doi.org/10.1117/12.2305464

[327] W. Liu, Y. Chen, H. Cui, and C. Wen, "A nonuniform clutter intensity estimation algorithm for random finite set filters," vol. 54, no. 6, pp. 2911–2925, 2018.

[328] M. I. Hossain, A. K. Gostar, A. Bab-Hadiashar, and R. Hoseinnezhad, "Visual mitosis detection and cell tracking using labeled multi-Bernoulli filter," in *Int. Conf. on Information Fusion*, 2018, pp. 1–5.

[329] D. Y. Kim, B.-N. Vo, A. Thian, and Y. S. Choi, "A generalized labeled multi-Bernoulli tracker for time lapse cell migration," in *Int. Conf. on Control, Automation and Information Sciences*, 2017, pp. 20–25.

[330] W. J. Hadden, J. L. Young, A. W. Holle, M. L. McFetridge, D. Y. Kim, P. Wijesinghe, H. Taylor-Weiner, J. H. Wen, A. R. Lee, and K. Bieback, "Stem cell migration and mechanotransduction on linear stiffness gradient hydrogels," *Proc. of the National Academy of Sciences*, vol. 114, no. 22, pp. 5647–5652, 2017.

[331] T. T. D. Nguyen and D. Y. Kim, "On-line tracking of cells and their lineage from time lapse video data," in *Int. Conf. on Control, Automation and Information Sciences*, Oct 2018, pp. 291–296.

[332] T. T. D. Nguyen, B.-N. Vo, B.-T. Vo, D. Y. Kim, and Y. S. Choi, "Tracking cells and their lineages via labeled random finite sets," *arXiv preprint arXiv:2104.10964*, 2021.

[333] L. Cament, M. Adams, and P. Barrios, "Space debris tracking with the Poisson labeled multi-Bernoulli filter," *Sensors*, vol. 21, no. 11, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/11/3684

[334] B. Wei and B. D. Nener, "Multi-sensor space debris tracking for space situational awareness with labeled random finite sets," vol. 7, pp. 36 991–37 003, 2019.

[335] S. Quan, D. Ding, and N. Zhaodong, "Space debris tracking via generalized labeled multi-Bernoulli random finite sets," in *Int. Conf. on Signal Processing, Communications and Computing*, 2019, pp. 1–4.

[336] D. Moratuwage, M. Adams, and L. Cament, "Space object tracking using a jump Markov system based δ-GLMB filter for space situational awareness," in *Advanced Maui Optical and Space Surveillance Technologies Conf.*, 2019.

[337] B. Wei and B. Nener, "Distributed space debris tracking with consensus labeled random finite set filtering," *Sensors*, vol. 18, no. 9, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/9/3005

[338] J. Olofsson, C. Veibäck, and G. Hendeby, "Sea ice tracking with a spatially indexed labeled multi-Bernoulli filter," in *Int. Conf. on Information Fusion*, 2017, pp. 1–8.

[339] T. Rathnayake, A. Khodadadian Gostar, R. Hoseinnezhad, R. Tennakoon, and A. Bab-Hadiashar, "On-line visual tracking with occlusion handling," *Sensors*, vol. 20, no. 3, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/3/929

[340] K. Dai, Y. Wang, J.-S. Hu, K. Nam, and C. Yin, "Intertarget occlusion handling in multiextended target tracking based on labeled multi-Bernoulli filter using laser range finder," vol. 25, no. 4, pp. 1719–1728, 2020.

[341] K. Dai, Y. Wang, Q. Ji, H. Du, and C. Yin, "Multiple vehicle tracking based on labeled multiple Bernoulli filter using pre-clustered laser range finder data," vol. 68, no. 11, pp. 10 382–10 393, 2019.

[342] J. A. Gaebler and P. Axelrad, "Identity management of clustered satellites with a generalized labeled multi-Bernoulli filter," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 11, pp. 2046–2057, 2020.

[343] J. A. Gaebler, P. Axelrad, and P. W. Schumacher, "Cubesat cluster deployment track initiation via a radar admissible region birth model," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 10, pp. 1927–1934, 2020.

[344] S. Lin, "Robust pitch estimation and tracking for speakers based on subband encoding and the generalized labeled multi-Bernoulli filter," vol. 27, no. 4, pp. 827–841, 2019.

[345] ——, "Jointly tracking and separating speech sources using multiple features and the generalized labeled multi-Bernoulli framework," in *Int. Conf. on Acoustics, Speech and Signal Processing*, 2018, pp. 3211–3215.

[346] D. Moratuwage, M. Adams, and F. Inostroza, "δ-generalized labeled multi-Bernoulli simultaneous localization and mapping with an optimal kernel-based particle filtering approach," *Sensors*, vol. 19, no. 10, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/10/2290

[347] ——, "δ-generalised labelled multi-Bernoulli simultaneous localisation and mapping," in *Int. Conf. on Control, Automation and Information Sciences*, 2018, pp. 175–182.

[348] H. Deusch, S. Reuter, and K. Dietmayer, "The labeled multi-Bernoulli SLAM filter," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1561–1565, 2015.

[349] X. Sun, R. Li, and L. Zhou, "Multidimensional information fusion in active sonar via the generalized labeled multi-Bernoulli filter," vol. 8, pp. 211 335–211 347, 2020.

[350] H. V. Nguyen, H. Rezatofighi, B.-N. Vo, and D. C. Ranasinghe, "Online UAV path planning for joint detection and tracking of multiple radio-tagged objects," vol. 67, no. 20, pp. 5365–5379, 2019.

[351] W. Wu, H. Sun, Y. Cai, and J. Xiong, "MM-GLMB filter-based sensor control for tracking multiple maneuvering targets hidden in the Doppler blind zone," vol. 68, pp. 4555–4567, 2020.

[352] J. Sun, C. Liu, Q. Li, and X. Chen, "Labelled multi-Bernoulli filter with amplitude information for tracking marine weak targets," *IET Radar, Sonar & Navigation*, vol. 13, no. 6, pp. 983–991, 2019.

[353] J. Wang, B. Yang, W. Wang, and Y. Bi, "Multiple-detection multi-target tracking with labelled random finite sets and efficient implementations," *IET Radar, Sonar & Navigation*, vol. 13, no. 2, pp. 272–282, 2019.

[354] R. Liu, H. Fan, and H. Xiao, "Labeled multi-Bernoulli filter joint detection and tracking of radar targets," *Applied Sciences*, vol. 9, no. 19, 2019.

[355] C.-T. Do and H. Van Nguyen, "Tracking multiple targets from multistatic Doppler radar with unknown probability of detection," *Sensors*, vol. 19, no. 7, 2019. [Online]. Available: http://www.mdpi.com/1424-8220/19/7/1672

[356] D. Jiang, M. Liu, Y. Gao, Y. Gao, W. Fu, and Y. Han, "Time-matching random finite set-based filter for radar multi-target tracking," *Sensors*, vol. 18, no. 12, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/12/4416

[357] C. Liu, J. Sun, P. Lei, and Y. Qi, "δ-generalized labeled multi-Bernoulli filter using amplitude information of neighboring cells," *Sensors*, vol. 18, no. 4, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/4/1153

[358] A. Scheel and K. Dietmayer, "Tracking multiple vehicles using a variational radar model," vol. 20, no. 10, pp. 3721–3736, Oct 2019.

[359] C.-T. Do and T. T. D. Nguyen, "Multiple marine ships tracking from multistatic Doppler data with unknown clutter rate," in *Int. Conf. on Control, Autom. and Inf. Sci. (ICCAIS)*. IEEE, 2019, pp. 1–6.

[360] H. Pessentheiner, "Localization characterization and tracking of harmonic sources with applications to speech signal processing," Ph.D. dissertation, Ph. D. dissertation, Graz University of Technology, 2017.

[361] E. A. Lehmann, A. M. Johansson, and S. Nordholm, "Reverberation-time prediction method for room impulse responses simulated with the image-source model," in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*. IEEE, 2007, pp. 159–162.

[362] E. A. Lehmann and A. M. Johansson, "Prediction of energy decay in room impulse responses simulated with an image-source model," *The Journal of the Acoustical Society of America*, vol. 124, no. 1, pp. 269–277, 2008.

[363] J. P. Dmochowski, J. Benesty, and S. Affes, "A generalized steered response power method for computationally viable source localization," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, no. 8, pp. 2510–2526, 2007.

[364] J. Benesty, C. Paleologu, T. Gänsler, and S. Ciochină, "Recursive least-squares algorithms," in *A perspective on stereophonic acoustic echo cancellation*. Springer, 2011, pp. 63–69.

[365] J. P. Morgan, "Time-frequency masking performance for improved intelligibility with microphone arrays," Ph.D. dissertation, Master Thesis in the College of Engineering at the University of Kentucky, 2017.

[366] I. Cohen, J. Benesty, and S. Gannot, *Speech processing in modern communication: Challenges and perspectives*. Springer Science & Business Media, 2009, vol. 3.

[367] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *IEEE International Conference on Acoust., Speech, and Signal Process. Proc. (Cat. No. 01CH37221)*, vol. 2. IEEE, 2001, pp. 749–752.

[368] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[369] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, "Subjective and objective quality assessment of audio source separation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2046–2057, 2011.

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

[371] A. Andriyenko, S. Roth, and K. Schindler, "An analytical formulation of global occlusion reasoning for multi-target tracking," in *2011 IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*. IEEE, 2011, pp. 1839–1846.

[372] S. Scheidegger, J. Benjaminsson, E. Rosenberg, A. Krishnan, and K. Granström, "Mono-camera 3D multi-object tracking using deep learning detections and PMBM filtering," in *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2018, pp. 433–440.

[373] H.-N. Hu, Q.-Z. Cai, D. Wang, J. Lin, M. Sun, P. Krahenbuhl, T. Darrell, and F. Yu, "Joint monocular 3D vehicle detection and tracking," in *Proceedings of the IEEE International Conf. on Comput. Vis.*, 2019, pp. 5390–5399.

[374] E. Maggio, M. Taj, and A. Cavallaro, "Efficient multitarget visual tracking using random finite sets," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 18, no. 8, pp. 1016–1027, 2008.

[375] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *European Conf. Comput. Vis.* Springer, 2016, pp. 17–35.

[376] J. Ferryman and A. Shahrokni, "Pets2009: Dataset and challenge," in *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*. IEEE, 2009, pp. 1–6.

[377] X. Alameda-Pineda, J. Staiano, R. Subramanian, L. Batrinca, E. Ricci, B. Lepri, O. Lanz, and N. Sebe, "Salsa: A novel dataset for multimodal group behavior analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1707–1720, 2015.

[378] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pat. Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, 2000.

[379] P. Schneider and D. H. Eberly, *Geometric tools for computer graphics*. Elsevier, 2002.

[380] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[381] R. P. Mahler, B.-T. Vo, and B.-N. Vo, "CPHD filtering with unknown clutter rate and detection profile," *IEEE Trans. on Signal Process.*, vol. 59, no. 8, pp. 3497–3513, 2011.

[382] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: Towards a benchmark for multi-target tracking," *arXiv preprint arXiv:1504.01942*, 2015.

[383] R. Kasturi, D. Goldgof, P. Soundararajan, V. Manohar, J. Garofolo, R. Bowers, M. Boonstra, V. Korzhova, and J. Zhang, "Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 319–336, 2008.

[384] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit.*, 2019, pp. 658–666.

[385] H. Rezatofighi, T. T. D. Nguyen, B.-N. Vo, B.-T. Vo, S. Savarese, and I. Reid, "How trustworthy are the existing performance evaluations for basic vision tasks?" *arXiv preprint arXiv:2008.03533*, 2020.

[386] A. Maksai, X. Wang, F. Fleuret, and P. Fua, "Non-Markovian globally consistent multi-object tracking," in *Proceedings of the IEEE International Conf. on Comput. Vis.*, 2017, pp. 2544–2554.

[387] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3246–3260, 2014.

[388] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein, "Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation," *arXiv preprint arXiv:1804.03619*, 2018.

[389] T. Afouras, J. S. Chung, and A. Zisserman, "The conversation: Deep audio-visual speech enhancement," *arXiv preprint arXiv:1804.04121*, 2018.

[390] R. Gao and K. Grauman, "Visualvoice: Audio-visual speech separation with cross-modal consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 495–15 505.

[391] H. Zhao, C. Gan, A. Rouditchenko, C. Vondrick, J. McDermott, and A. Torralba, "The sound of pixels," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 570–586.

[392] C. Gan, H. Zhao, P. Chen, D. Cox, and A. Torralba, "Self-supervised moving vehicle tracking with stereo sound," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7053–7062.

[393] H. Zhao, C. Gan, W.-C. Ma, and A. Torralba, "The sound of motions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1735–1744.

[394] C. Gan, D. Huang, H. Zhao, J. B. Tenenbaum, and A. Torralba, "Music gesture for visual sound separation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 478–10 487.

[395] Y. Tian, D. Hu, and C. Xu, "Cyclic co-learning of sounding object visual grounding and sound separation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2745–2754.

[396] M. Shuo, Y. Ji, X. Xu, and X. Zhu, "Vision-guided music source separation via a fine-grained cycle-separation network," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 4202–4210.

[397] C. Gan, Y. Zhang, J. Wu, B. Gong, and J. B. Tenenbaum, "Look, listen, and act: Towards audio-visual embodied navigation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9701–9707.

[398] J.-T. Lee, M. Jain, H. Park, and S. Yun, "Cross-attentional audio-visual fusion for weakly-supervised action localization," in *International Conference on Learning Representations*, 2020.

[399] G. Sterpu, C. Saam, and N. Harte, "Attention-based audio-visual fusion for robust automatic speech recognition," in *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, 2018, pp. 111–115.

[400] S. Petridis, T. Stafylakis, P. Ma, F. Cai, G. Tzimiropoulos, and M. Pantic, "End-to-end audiovisual speech recognition," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 6548–6552.

[401] F. Tao and C. Busso, "Gating neural network for large vocabulary audiovisual speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 7, pp. 1290–1302, 2018.

[402] A. Torfi, S. M. Iranmanesh, N. Nasrabadi, and J. Dawson, "3D convolutional neural networks for cross audio-visual matching recognition," *IEEE Access*, vol. 5, pp. 22 081–22 091, 2017.

[403] G. Sell, K. Duh, D. Snyder, D. Etter, and D. Garcia-Romero, "Audio-visual person recognition in multimedia data from the IARPA janus program," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 3031–3035.

[404] A. Nagrani, S. Albanie, and A. Zisserman, "Seeing voices and hearing faces: Cross-modal biometric matching," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8427–8436.

[405] S. Hörmann, A. Moiz, M. Knoche, and G. Rigoll, "Attention fusion for audio-visual person verification using multi-scale features," in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*. IEEE, 2020, pp. 281–285.

[406] X. Qian, A. Xompero, A. Cavallaro, A. Brutti, O. Lanz, and M. Omologo, "3D mouth tracking from a compact microphone array co-located with a camera," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 3071–3075.

[407] X. Qian, A. Brutti, O. Lanz, M. Omologo, and A. Cavallaro, "Multi-speaker tracking from an audio-visual sensing device," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2576–2588, 2019.

[408] H. Liu, Y. Sun, Y. Li, and B. Yang, "3D audio-visual speaker tracking with a novel particle filter," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 7343–7348.

[409] X. Qian, A. Brutti, M. Omologo, and A. Cavallaro, "3D audio-visual speaker tracking with an adaptive particle filter," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2896–2900.

[410] J. Li, Y. Wang, C. Wang, Y. Tai, J. Qian, J. Yang, C. Wang, J. Li, and F. Huang, "DSFD: dual shot face detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5060–5069.

[411] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 269–286.

[412] K. Abed-Meraim and Y. Hua, "3-D near field source localization using second order statistics," in *Conference Record of the Thirty-First Asilomar Conference on Signals, Systems and Computers (Cat. No. 97CB36136)*, vol. 2. IEEE, 1997, pp. 1307–1311.

[413] N. Kabaoglu, H. A. Çirpan, E. Cekli, and S. Paker, "Maximum likelihood 3-D near-field source localization using the em algorithm," in *Proceedings of the Eighth IEEE Symposium on Computers and Communications. ISCC 2003*. IEEE, 2003, pp. 492–497.

[414] N. Kabaoğlu, H. A. Çırpan, E. Çekli, and S. Paker, "Deterministic maximum likelihood approach for 3-D near field source localization," *AEU-International Journal of Electronics and Communications*, vol. 57, no. 5, pp. 345–350, 2003.

[415] W. Zuo, J. Xin, N. Zheng, H. Ohmori, and A. Sano, "Subspace-based near-field source localization in unknown spatially nonuniform noise environment," *IEEE Transactions on Signal Processing*, vol. 68, pp. 4713–4726, 2020.

[416] H. Chen, Z. Jiang, W. Liu, Y. Tian, and G. Wang, "Conjugate augmented decoupled 3-D parameters estimation method for near-field sources," *IEEE Transactions on Aerospace and Electronic Systems*, 2022.

[417] O. Nadiri and B. Rafaely, "Localization of multiple speakers under high reverberation using a spherical microphone array and the direct-path dominance test," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 10, pp. 1494–1505, 2014.

[418] X. Li, L. Girin, R. Horaud, and S. Gannot, "Multiple-speaker localization based on direct-path features and likelihood maximization with spatial sparsity regularization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1997–2012, 2017.

[419] X. Li, Y. Ban, L. Girin, X. Alameda-Pineda, and R. Horaud, "Online localization and tracking of multiple moving speakers in reverberant environments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 1, pp. 88–103, 2019.

[420] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 4, pp. 1–48, 2013.

[421] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*. Springer, 2016, pp. 850–865.

[422] A. He, C. Luo, X. Tian, and W. Zeng, "A twofold siamese network for real-time object tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4834–4843.

[423] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1763–1771.

[424] C. Kim, L. Fuxin, M. Alotaibi, and J. M. Rehg, "Discriminative appearance modeling with multi-track pooling for real-time multi-object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9553–9562.

[425] D. Eberly, "Perspective projection of an ellipsoid," *Geometric Tools, LLC, http://www. geometrictools, com, Created: Mar*, vol. 2, 1999.
