RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Électronique

End of Studies Project Report

In partial fulfillment of the requirements for the State Engineer Degree in Electronics

---

Sound Source Localization and Tracking: Real life application using UMA-16

Tarek Bouznad

Presented and defended publicly on (30/06/2025)

Jury Members:

President:	Pr. Mourad Adnane	ENP
Examiner:	Dr. Nesrine Bouadjenek	ENP
Supervisor:	Pr. Adel Belouchrani	ENP
Co-Supervisor:	Dr. Soufiane Tebache	LDCCP/ENP

ENP 2025

Department D'Electronique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Algiers, Algeria.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Électronique

End of Studies Project Report

In partial fulfillment of the requirements for the State Engineer Degree in Electronics

---

Sound Source Localization and Tracking: Real life application using UMA-16

Tarek Bouznad

Presented and defended publicly on (30/06/2025)

Jury Members:

President:	Pr. Mourad Adnane	ENP
Examiner:	Dr. Nesrine Bouadjenek	ENP
Supervisor:	Pr. Adel Belouchrani	ENP
Co-Supervisor:	Dr. Soufiane Tebache	LDCCP/ENP

ENP 2025

Department D'Electronique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Algiers, Algeria.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Électronique

Mémoire de Projet de Fin d'Études

Pour l'obtention du diplôme d'Ingénieur d'État en Électronique

---

Sound Source Localization and Tracking: Real life application using UMA-16

Tarek Bouznad

Présenté et soutenu publiquement le (30/06/2025)

Composition du Jury:

Président:	Pr. Mourad Adnane	ENP
Examinateur:	Dr. Nesrine Bouadjenek	ENP
Promoteur:	Pr. Adel Belouchrani	ENP
Co-Promoteur	Dr. Soufiane Tebache	LDCCP/ENP

ENP 2025

Département d'Électronique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Alger, Algérie.

الملخص

يتناول هذا المشروع تحديين رئيسيين في معالجة الإشارة الصوتية: تقدير معلمات الحركة باستخدام تأثير دوبلر، وتحديد موقع المصادر الصوتية باستخدام مصفوفات الميكروفونات. تستعرض الجزء الأول كيفية استخدام التغير في التردد اللحظي (IF) الناتج عن تأثير دوبلر لتقدير سرعة وارتفاع وتردد انبعاث مصدر صوتي متحرك باستخدام ميكروفون واحد فقط، من خلال حل مغلق تم التحقق منه عبر المحاكاة. أما الجزء الثاني فيركّز على تحديد مواقع المتحدثين وفصلهم باستخدام تقنيات التصفية المكانية مثل SRP-PHAT وMVDR وLCMV. وقد تم تنفيذ نظام زمني حقيقي باستخدام مصفوفة ميكروفونات MiniDSP UMA-16 وكاميرا لعرض النتائج بصريًا.

الكلمات المفتاحية: تأثير دوبلر، التردد اللحظي، تقدير الحركة، تحديد موقع المصدر الصوتي، مصفوفات الميكروفونات، التوجيه الشعاعي، SRP-PHAT، MVDR، LCMV، التكامل السمعي البصري، تتبع المتحدث.

Résumé

Ce mémoire traite deux défis majeurs en traitement du signal acoustique : l'estimation des paramètres de mouvement via l'effet Doppler, et la localisation de sources sonores avec des réseaux de microphones. La première partie explore comment la fréquence instantanée (IF) induite par l'effet Doppler permet d'estimer la vitesse, l'altitude et la fréquence d'émission d'une source mobile à partir d'un seul microphone, à l'aide d'une solution en forme fermée validée par simulation. La seconde partie porte sur la localisation et la séparation de locuteurs via des méthodes de filtrage spatial (SRP-PHAT, MVDR, LCMV). Un système temps réel est implémenté avec une matrice de microphones MiniDSP UMA-16 et une caméra pour la projection visuelle.
Mots-clés : effet Doppler, fréquence instantanée, estimation de mouvement, localisation de source sonore, réseaux de microphones, filtrage spatial, SRP-PHAT, MVDR, LCMV, suivi de locuteur, intégration audio-visuelle.

Abstract

This thesis addresses two major challenges in acoustic signal processing: motion parameter estimation using the Doppler effect, and sound source localization with microphone arrays. The first part explores how the instantaneous frequency (IF) shift induced by the Doppler effect enables the estimation of a moving source's velocity, altitude, and emission frequency using a single microphone, through a closed-form solution validated by simulation. The second part focuses on speaker localization and separation using spatial filtering methods (SRP-PHAT, MVDR, LCMV). A real-time system is implemented with a MiniDSP UMA-16 microphone array and a camera for visual projection.

Keywords: Doppler effect, instantaneous frequency, motion estimation, sound source localization, microphone arrays, beamforming, SRP-PHAT, MVDR, LCMV, speaker tracking, audio-visual integration.

Dedication

To my younger self—who kept pushing for what he loved the most, discovered unknown roads,
without fearing regret, and making every necessary sacrifice.

To my family—who supported me unconditionally, no matter what path I chose.

To those who saw potential in me and guided me toward a brighter future.

To all who gifted me good memories, and helped shape who I am today—
and continue to Inspire what I strive to become.

To all Vniversers, for the greatest memories and people who shaped a second a home to me.

Thank you all.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to Pr. Adel Belouchrani, whose unwavering support and belief in my potential have guided me throughout this journey. His knowledge, mentorship, and life advice continue to inspire me to this day.

I would also like to sincerely thank the jury members—Prof. Mourad Adnane and Dr. Nesrine Bouadjenek—for their thoughtful evaluation and the kind words they shared about my work and academic journey over the past three years.

To my family, thank you for the incredible opportunity and constant support that allowed me to reach this milestone. Your encouragement has been the foundation of all my achievements.

Looking back, I once regretted not reaching my initial goal. But ENP welcomed me in an unexpected way. I arrived as a boy who believed he could be the best, and now I leave as a man who still believes it—with deeper perspective and maturity. Throughout, I often underestimated the value of what I was living, until now, in reflection, I see it clearly: it was the best decision, chosen for me by Allah. If I had to make that choice a hundred times, I would choose it 101 times.

A heartfelt thank you goes to all the people who have contributed to my journey—whether as friends, mentors, supporters, or sources of inspiration.

To my closer circle: Ibrahim, Labbas, Adoula, Sido, and Lhadi—thank you for the amazing experiences and memories we've created during these three years. Through every adventure and challenge, we grew together. Each of you brought unique strengths, and together we formed a circle that constantly inspired one another.

I also want to extend this gratitude to Serine, Houda, Ilham, and all of our ELN classmates who brightened difficult times with unforgettable moments. To Nadjib—an endless source of motivation and inspiration; Nazim—for the crazy night drives discussions; Samy, Walid, Bahi, Aymen, Foutia, Abdelbaki and all of ELN, from 2023, to 2025.

To Vniverse—thank you for being a home to us (special thanks to Ascem). To the IEEE family, and to those who stood by me during its darkest days—Madina, Rayan, Aymen, and Ilyes, my binôme from the prepa years.

To the Chakawi Gang—especially Zaki and Said—thank you for being such inspirations and great friends. Always there to listen to the curious annoying kid who always wanted to know more.

To the younger generation: Meskali, Rached, Aymen, Ahmed, and Rayan (again), whose presence in the department made it an ever greater experience, with all their talent and competence, I wish them the very best in their upcoming final year.

To Freedom—thank you for being there during the final dance of this journey. To Taleb Bot, for the amazing journey we went through.

I hope won't an end to anything, but only a start of something way bigger.

Contents

List of Tables

List of Figures

List of Acronyms

- **SSL**: Sound Source Localization
- **DoA**: Direction of Arrival
- **TDoA**: Time Difference of Arrival
- **AoA**: Angle of Arrival
- **FDoA**: Frequency Difference of Arrival
- **ToA**: Time of Arrival
- **ToF**: Time of Flight
- **RSS**: Received Signal Strength
- **FFT**: Fast Fourier Transform
- **STFT**: Short-Time Fourier Transform
- **SNR**: Signal-to-Noise Ratio
- **SRP**: Steered Response Power
- **PHAT**: Phase Transform
- **GCC**: Generalized Cross-Correlation
- **MVDR**: Minimum Variance Distortionless Response
- **LCMV**: Linearly Constrained Minimum Variance
- **MWF**: Multichannel Wiener Filter
- **MSDW-MWF**: Multiple Speech Distortion Weighted Multichannel Wiener Filter
- **GSC**: Generalized Sidelobe Canceller
- **NCLMS**: Norm-Constrained Least Mean Squares
- **AMC**: Adaptive Mode Control
- **RTF**: Relative Transfer Function
- **VAD**: Voice Activity Detection

- **M3K**: Modified 3D Kalman Filter
- **CRNN**: Convolutional Recurrent Neural Network
- **CNN**: Convolutional Neural Network
- **FFNN**: Feedforward Neural Network
- **ResNet**: Residual Network
- **VAE**: Variational Autoencoder
- **EKF**: Extended Kalman Filter
- **KLS**: Kalman Least Squares
- **URA**: Uniform Rectangular Array
- **ULA**: Uniform Linear Array
- **RIR**: Room Impulse Response
- **TFD**: Time-Frequency Distribution
- **RMSE**: Root Mean Square Error
- **HSDA**: Hierarchical SRP Direction Assignment

General Introduction

Acoustic signal processing plays an important role across a wide range of domains—from early military applications like sonar and artillery sound ranging to modern fields such as telecommunications, robotics, assistive technologies, and immersive multimedia. Two of the most fundamental challenges in this domain are **sound source localization (SSL)**—the estimation of a sound's spatial origin—and **motion parameter estimation**—the inference of the dynamic behavior of moving sound-emitting objects.

This thesis addresses both challenges by developing methods that span from **single-microphone motion analysis using the Doppler effect** to **multichannel systems for real-time localization, speaker tracking, and spatial audio enhancement**. These contributions are unified under a broader objective: building **proprietary, interpretable, and modular platforms** for spatial acoustic analysis.

Motivations

In recent years, there has been increasing demand for spatially aware auditory systems, driven by applications such as teleconferencing, smart homes, mobile robotics, and augmented or virtual reality. While high-performance solutions do exist, they are often locked behind **proprietary ecosystems, specialized hardware, or prohibitive costs**, limiting accessibility and experimentation.

This thesis is motivated by two core aims:

- To explore **theoretically sound but computationally efficient** techniques for estimating spatial and motion-related acoustic parameters.
- To develop a **reproducible end-to-end platform** for localization and tracking using **affordable, off-the-shelf components**, with an emphasis on interpretability and deployability.

Problem Context

The thesis is structured around two major research axes, reflecting both the diversity and complementarity of the proposed contributions:

Doppler-Based Motion Parameter Estimation

The first axis focuses on how motion characteristics of a sound-emitting object can be inferred from the acoustic signal captured by a **single microphone**. By analyzing **instantaneous frequency (IF)** variations induced by the **Doppler effect**, it becomes possible to estimate parameters such as **velocity, altitude, and time of closest approach**. This part builds upon a closed-form solution introduced in [1] and evaluates its performance through simulations under varying motion profiles and signal-to-noise conditions.

Microphone Array Processing for Localization and Separation

The second and more extensive part of the thesis targets **real-time spatial processing using microphone arrays**. Here, we design and implement a comprehensive system capable of:

- Wideband **direction-of-arrival (DoA)** estimation using **SRP-PHAT** with hierarchical spatial search.
- **Speaker tracking** using exponential smoothing and a **Modified 3D Kalman Filter (M3K)**.
- **Visual integration**, projecting spatial estimates onto a calibrated video stream in real time.
- Incorporation of **beamforming** for spatial filtering and source enhancement.

The system is validated using the **MiniDSP UMA-16 microphone array** and a **standard USB webcam**, demonstrating real-time operation on **commodity hardware**.

Objectives and Expected Contributions

The goal of this work is to develop a **functional and interpretable platform** for spatial acoustic signal processing that:

- Operates in **real time**, providing live feedback on speaker direction.
- Supports **interactive user control**, such as speaker selection in multi-speaker scenarios.
- Bridges audio and visual cues via **camera-based 3D–2D projection**.
- Offers **open-source MATLAB implementations** for academic and prototyping use.

From a research perspective, the main contributions include:

1. A theoretical and simulation-based study of **closed-form Doppler motion estimation**.
2. A real-time **SRP-PHAT-based localization system**.
3. A live **audio–visual integration** using face detection and projection overlays.
4. An evaluation of **beamforming strategies** for multi-source separation in dynamic scenes.

Thesis Structure

The thesis is structured into five main chapters, each building upon the previous in complexity and system integration:

Chapter 1 reviews the state of the art in sound source localization, covering classical, subspace, and learning-based methods. It also introduces common applications and commercial systems.

Chapter 2 investigates motion parameter estimation using Doppler effect analysis with a single microphone, including signal modeling, closed-form solutions, and robust instantaneous frequency estimators.

Chapter 3 introduces acoustic source localization using microphone arrays, presenting narrowband and wideband signal models, TDOA methods, beamforming, and SRP-PHAT formulations.

Chapter 4 focuses on source separation via spatial filtering. It details beamforming architectures—both fixed and adaptive—and integrates blind source separation with beamforming for enhanced robustness.

Chapter 5 presents the real-time implementation of multi-speaker tracking. It includes voice activity detection, hierarchical SRP-PHAT search, M3K tracking, camera-based visualization, and experimental evaluation.

This structure reflects a progressive development—starting with foundational background, then moving through single-sensor and array-based analysis, before culminating in a real-world system for spatial localization and tracking.

The next chapter surveys the current state of the art in **sound source localization**, **array design**, **signal processing algorithms**, and **deep learning approaches**, laying the foundation for the techniques explored throughout this work.

Chapter 1

State of the Art

1.1 Introduction

Sound source localization (SSL) is the process of determining the position or direction of acoustic sources in a physical space using one or more microphones. It has become an integral technology across numerous fields, including robotics, videoconferencing, augmented reality, surveillance, and assistive devices. The foundational problem of SSL involves identifying the direction-of-arrival (DoA) or full 3D position of a sound, often in noisy or reverberant environments.

Inspired by the natural echolocation abilities of animals such as bats and dolphins [2], SSL systems leverage differences in time, phase, frequency, and amplitude between microphone signals. Over the past century, SSL has evolved from military applications like artillery tracking to modern real-time multimodal systems. This chapter reviews the major methods, array designs, algorithmic strategies, and emerging trends.

1.2 Microphone Array Design

The geometry of a microphone array strongly impacts the accuracy, resolution, and robustness of SSL [2]. Common architectures include:

- **Linear arrays**: Suitable for 1D azimuthal tracking, often used in automotive or conference setups [3].
- **Circular arrays**: Enable 360° azimuth coverage, common in smart speakers and meeting systems [4].
- **Spherical arrays**: Provide full 3D DoA estimation (e.g., Eigenmike EM64).
- **Hexagonal arrays**: Offer a balance between azimuth and elevation precision [5].
- **Ad-hoc arrays**: Irregular microphone configurations offering flexibility in complex environments [6].

SSL systems can also be classified by spatial resolution (1D, 2D, 3D), number of detectable sources, or the distinction between passive and active methods [7, 8, 9].

1.3 Classical Methods

Classical sound source localization (SSL) techniques rely on well-established physical principles and have been widely adopted due to their conceptual simplicity, robustness, and real-time feasibility. These methods generally estimate source location based on time, frequency, angle, or energy characteristics of the arriving sound waves. This section groups classical SSL methods into two broad categories: geometric localization approaches and signal-derived parameter estimation.

1.3.1 Geometric Localization Methods

Geometric methods determine the sound source location by solving systems of equations derived from physical constraints. The most common techniques include:

- **Triangulation** estimates the position of the sound source by intersecting two or more angle-of-arrival (AoA) estimates. At least two microphones are required for 2D localization, and three for 3D positioning. Directional microphones or algorithms like MUSIC and ESPRIT improve angular precision [10]. Increasing the number of microphones generally enhances accuracy [11].
- **Trilateration** estimates the sound source location by computing distances to at least three non-collinear microphones, typically using Time of Arrival (ToA) information. Each distance defines a circle centered at a microphone, and the source is located at the intersection of these circles [12]. This method relies less on microphone directivity, offering greater flexibility in array design.
- **Multilateration** generalizes trilateration to four or more microphones. By overconstraining the problem, it improves localization accuracy and robustness to measurement noise [13]. However, it increases the complexity of the solution, especially in noisy or reverberant conditions.

These methods are only as accurate as the underlying physical parameters they rely on—commonly extracted through the following signal-based techniques.

1.3.2 Signal-Derived Estimation Techniques

- **Time of Arrival (ToA)** measures the absolute travel time of the sound from source to microphone [14]. It requires tight synchronization between source and sensors and precise knowledge of sound velocity. ToA is highly sensitive to synchronization errors but offers accurate ranging when conditions are controlled.
- **Time Difference of Arrival (TDoA)** measures the relative delay between signals received at different microphones. It is robust to absolute timing offset, making it suitable for unsynchronized systems [15]. TDoA estimation often employs Generalized Cross-Correlation (GCC), particularly with Phase Transform (PHAT) weighting to increase peak sharpness and noise resilience. However, moving sources introduce Doppler shifts, complicating delay estimation [16].
- **Angle of Arrival (AoA)** estimates the direction from which a sound wave reaches the array. It can be computed via time-delay estimation, spectral methods (e.g., MUSIC [17], ESPRIT [18]), or spatial correlation. AoA methods do not require time synchronization, but performance depends heavily on microphone array geometry and signal coherence.

- **Frequency Difference of Arrival (FDoA)** captures the Doppler shift of a sound arriving at microphones in motion relative to the source [19]. It is suitable for moving sources and observers but requires knowledge of relative velocities and is limited by frequency resolution and bandwidth.
- **Time of Flight (ToF)** includes the ToA plus any signal processing latency [20, 21], often modeled as a constant system delay. While less accurate than pure ToA, it is sometimes used in practical systems where delays can be calibrated out.
- **Received Signal Strength (RSS)** infers distance from the sound intensity attenuation, assuming a known propagation model [22]. It avoids the need for synchronization but suffers from multipath fading and energy variability.
- **Energy-Based Localization** estimates the source position by analyzing acoustic energy patterns across the sensor array. It is low-cost and does not require synchronization, but has limited spatial resolution and is sensitive to reverberation [23].

1.3.3 Beamforming and SRP-PHAT

Beamforming methods apply spatial filtering to enhance signals from a desired direction while suppressing others. These include:

- **Delay-and-Sum (DAS)** aligns signals from a target direction and sums them. It is easy to implement but suffers from poor resolution and generates false source peaks (ghosts) in multipath environments [24]. Techniques like Clean-SC and DAMAS deconvolve the beamformer output to enhance resolution [25].
- **Minimum Variance Distortionless Response (MVDR)** minimizes output energy while preserving the signal from a given direction. It suppresses interference and noise more effectively than DAS, but requires accurate covariance estimation [26].
- **Steered Response Power with Phase Transform (SRP-PHAT)** is a robust and widely used beamforming method for sound localization [27]. It sums the weighted GCC-PHAT responses across all microphone pairs over a spatial grid. The grid point with the maximum energy indicates the most probable direction. SRP-PHAT is resilient to reverberation but computationally intensive due to exhaustive search [28].

Optimizations such as hierarchical grid refinement, Gaussian interpolation, and adaptive spatial sampling have been proposed to reduce the computational load of SRP-PHAT.

1.3.4 Subspace Methods

- **MUSIC (MUltiple SIgnal Classification)** exploits the eigenstructure of the covariance matrix to separate signal and noise subspaces, achieving high-resolution DoA estimation under favorable conditions [17]. However, it requires a well-calibrated array and precise knowledge of the number of sources.
- **ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques)** provides high-resolution DoA estimation with lower computational cost than MUSIC. It leverages the shift invariance property of uniform arrays and does not require a search over the entire angular space [18].

1.3.5 GCC and Cross-Correlation Techniques

The **Generalized Cross-Correlation (GCC)** framework is the foundation for many TDoA-based techniques. It computes the cross-correlation between microphone pairs, often using PHAT weighting to emphasize phase consistency over magnitude [29]. The peak of the correlation function corresponds to the estimated delay, which can then be mapped to a direction.

1.3.6 Summary of Trade-offs

Each classical method offers different trade-offs:

- **ToA/TDoA**: High accuracy, synchronization required (ToA), sensitive to noise.
- **AoA**: No sync needed, performance depends on SNR and array geometry.
- **Beamforming**: Robust to noise, low resolution (DAS), high cost (SRP-PHAT).
- **Subspace methods**: High resolution, sensitive to array calibration.
- **Energy-based**: Simple, low-precision, useful for low-cost systems.

Selection depends on hardware capabilities, desired accuracy, and environmental conditions.

1.4 Multimodal and AI-Based Approaches

1.4.1 Audio–Visual Fusion

Combining microphone arrays with visual input (e.g., lip motion, face direction) enhances DoA reliability, especially in reverberant or noisy settings [2]. Hybrid audio–visual architectures often rely on dual neural network designs, with separate branches handling audio and visual input. For example, SSLNet [30] fuses raw 1D audio waveforms and video frames by first converting them to spectrograms and 2D images, then feeding them to modality-specific networks. Another model in [31] allows autonomous robots to detect and localize multiple speakers by combining 360° visual data with multichannel audio.

1.4.2 Machine Learning and Deep Learning

Artificial intelligence methods for sound source localization (SSL) have grown rapidly in recent years. These approaches learn directly from data and often outperform classical techniques in noisy or reverberant environments [32]. Unlike physics-based methods, AI systems use pattern matching and feature extraction for sound detection and DoA estimation. Various architectures have been proposed, which can be broadly categorized as follows:

Feedforward Neural Networks (FFNN).

In [33], a FFNN was trained on noise-free energy features to outperform traditional energy-based localization in reverberant environments. Another model in [34] estimated source positions from

TDoA measurements, showing strong performance even under noise and close sensor spacing [35, 36].

Convolutional Neural Networks (CNN).

CNNs have been used for DoA estimation from STFT phase components [37] and phase maps [38]. CNNs also perform well in spectrogram-based classification [39, 40], and a hybrid CNN–Random Forest model using Mel-log energy features showed superior results compared to classic methods [41].

Convolutional Recurrent Neural Networks (CRNN).

CRNNs combine CNNs' spatial learning with RNNs' temporal modeling. A CRNN in [42] localized up to three sources simultaneously. CRNNs using MFCC, LMS, and RASTA-PLP input features achieved detection accuracy near 90

Residual Networks (ResNet).

ResNets address vanishing gradient issues [43, 44]. In [45], a ResNet trained on single-microphone simulations achieved effective localization. Another hybrid model (ResCNN) used SE blocks for feature recalibration [46]. Combining ResNet with channel attention modules yielded nearly 98% localization accuracy [47].

Transformer Networks.

Originally developed for NLP [48], transformers now appear in SSL. The BAST model [49] applied multi-head attention to spectrogram patches, improving azimuth estimation over CNNs in reverberant conditions. Transformers also excelled when using GCC-PHAT with speech masking in robotics localization [50].

Autoencoders and VAEs.

Autoencoders (AEs) have been used to identify the most likely source direction by comparing latent representations from different candidate locations [51]. Dual-decoder architectures further disentangle reverberation and location features [52]. VAEs, including convolutional versions trained on inter-microphone phase data, offer strong performance with limited labeled data [53, 54].

Hybrid Audio–Visual Networks.

Several systems combine sound and image representations. For example, SSLNet [30] processes 1D raw waveforms and video frames via spectrogram-based networks. A 360-degree robot perception model in [31] can detect multiple speaking individuals and identify who is speaking based on combined visual and acoustic signals.

Discussion.

AI-based SSL models offer impressive performance gains, particularly in complex environments. However, generalization remains a key issue — most models experience performance drops when evaluated on unseen datasets. This challenge is compounded by the difficulty of acquiring large, labeled, and diverse audio datasets. Still, CNNs, CRNNs, and hybrid models remain dominant in detection tasks due to their effectiveness in spectrogram feature extraction [55].

AI systems continue to evolve, and unlike static algorithms, they benefit from online learning and continual adaptation. This makes them well-suited for dynamic acoustic environments and real-world deployments.

1.5 Applications and Commercial Systems

Sound source detection and localization have wide-ranging applications across both military and civilian domains. This section reviews practical implementations found in the literature, classified accordingly. Tables 1.1 and 1.2 summarize recent systems, including their methods and performance metrics such as detection accuracy, distance, or angular error. In some multimedia applications, metrics like cIoU and AUC are also reported.

1.5.1 Military Applications

Military applications are particularly common in the literature, notably for gunshot detection, UAV tracking, underwater acoustics, and aircraft monitoring. Table 1.1 compiles representative examples, highlighting the techniques used, including both classic (TDoA, DoA, MUSIC) and AI-based models (DNNs, CNNs, CoNNs).

1.5.2 Civilian Applications

Civilian use cases are equally rich and diverse, including robotics, IoT devices, pipeline monitoring, teleconferencing, multimedia, and healthcare. Notably, visual metrics like cIoU and AUC are used in some applications such as videoconferencing. Table 1.2 summarizes the relevant literature.

1.6 Conclusion

The field of sound source localization (SSL) has evolved significantly—from classical geometry-based methods like triangulation and TDoA to sophisticated beamforming techniques, inverse problem formulations, and modern deep learning frameworks. While traditional methods remain fundamental due to their interpretability and simplicity, contemporary research increasingly favors hybrid and data-driven approaches to tackle challenges posed by reverberation, noise, and dynamic environments.

Throughout this chapter, we surveyed a wide spectrum of localization strategies, highlighted their respective trade-offs, and discussed how real-world systems balance between accuracy, computational efficiency, and environmental adaptability. The emergence of open-source, low-cost, and

Table 1.1: Military acoustic source detection and localization applications. The tilde ( ) indicates approximate values. Adapted from [56].

Application	**Ref**	**Method**	**Detection Acc.**	**Distance**	**Direction**
Gunshot	[124]	DNN	93.84%	91.5%	93.1%
	[49]	EML	-	99.95%	-
	[125]	CNN	∼90%	-	-
	[126]	TDoA	-	-	-
UAV	[127]	-	-	-	1.47°
	[128]	DNN	94.7%	-	-
	[129]	NN	92.63%	-	-
	[130]	CoNN	96.3%	-	-
	[131]	SRP-PHAT	-	-	-
Aircraft	[132]	SE-MUSIC	-	-	-
	[133]	TDoA + DoA	-	-	-
Underwater	[134]	DNN	-	0.13 m	-
	[135]	TDoA	-	-	∼18°
	[136]	TDoA + ToA + ML	96.4%	-	-
	[137]	DoA	-	-	-
	[138]	STDoA	-	4.92 m	-
	[139]	GCC-PHAT + TDoA	-	0.5–2 m	-
	[140]	TDoA	-	-	-
	[141]	Beamforming	-	∼1 m	-

interpretable SSL platforms underscores a growing trend toward democratized spatial audio technology.

In the remainder of this thesis, we build upon the core concepts reviewed here to develop a complete SSL system. We begin by analyzing the theoretical underpinnings of classical Doppler-based motion estimation, derive a closed-form solution, and subsequently design and implement a real-time microphone array-based SRP-PHAT localization pipeline. This sets the stage for the contributions in acoustic tracking, spatial filtering, and visual integration discussed in the following chapters.

Table 1.2: Civilian acoustic source detection and localization applications. Tilde ( ) indicates approximations; $\leq$ denotes maximum error. Adapted from [56].

Application	**Ref**	**Method**	**Detection Acc.**	**Distance**	**Direction**
Robotics	[142]	DNN	-	97%	97%
	[122]	DNN	85%	-	-
	[143]	TDoA	-	$\leq$0.24 m	$\leq$1.5°
	[144]	DoA	-	$\leq$0.07 m	$\leq$1.15°
Healthcare	[32]	Beamforming	-	-	-
Pipeline leak	[145]	TDoA	-	95.7%	-
	[146]	TDoA	-	92.68%	-
	[147]	MUSIC	-	-	$\leq$2.5°
IoT	[148]	CNN	$\sim$90%	-	-
	[149]	DoA	-	-	-
	[15]	SRP-PHAT	-	-	-
Partial discharge	[150]	TDoA	-	97.27%	-
	[151]	TDoA	-	$\leq$1.5 cm	-
Underground	[152]	SRP-PHAT	-	$\sim$0.77 m	-
Underwater meas.	[153]	-	-	-	$\sim$30°
Wildlife	[154]	TDoA	-	-	-
	[155]	ToA/TDoA/DoA	-	-	-
Videoconferencing	[156]	DNN	cIoU (77), AUC (60.5)	-	-
	[121]	SSLNet	cIoU (85), AUC (78)	-	-
	[84]	ODB-SRP-PHAT	$\sim$95%	-	-
	[157]	DNN	cIoU (75.2), AUC (57.2)	-	-
	[158]	SRP-PHAT	-	-	-
Sport	[159]	Beamforming	-	$\leq$3 cm	-
Disaster victims	[12]	GCC-PHAT	-	-	$\leq$2°
Authentication	[160]	TDoA	$\sim$99%	-	-
Hearing aids	[161]	SVD	-	-	$\leq$3°
Multimedia surveillance	[162]	Gauss filter + TDoA	-	-	-
	[8]	TDoA, SRP-PHAT	-	-	-
Noise monitoring	[163]	TDoA	-	$\leq$0.5 m	-
	[164]	Beamforming	-	-	-

Chapter 2

Motion Parameter Estimation exploiting Doppler Effect

2.1 Introduction

This chapter presents a Doppler-based motion estimation algorithm for narrowband acoustic sources using a single stationary microphone [1]. The method extracts velocity, altitude, and emission frequency directly from the instantaneous frequency (IF) [57] trajectory of a received tone, without requiring array geometries or iterative optimization.

The approach builds upon the closed-form solution framework introduced in [1].

The chapter is organized as follows: Section 2.2 establishes the Doppler shift model underlying the approach. Section 2.3 details the algorithmic implementation and practical considerations. Section 2.7 presents quantitative performance results under varying noise conditions and algorithmic validation through simulation of a moving acoustic source.

2.2 Signal Model

2.2.1 Hypotheses

The proposed model relies on the following assumptions:

- The source emits a pure tone (narrowband signal).
- The source trajectory is linear with constant motion parameters.
- The receiver is stationary and located on the ground.
- Atmospheric conditions (e.g., wind) are negligible and sound speed c is constant.

2.2.2 Received Signal Model

Consider a narrowband acoustic source emitting a tonal signal $x(\tau) = A\cos(2\pi f_0\tau + \phi_0)$ as shown in 2.1.

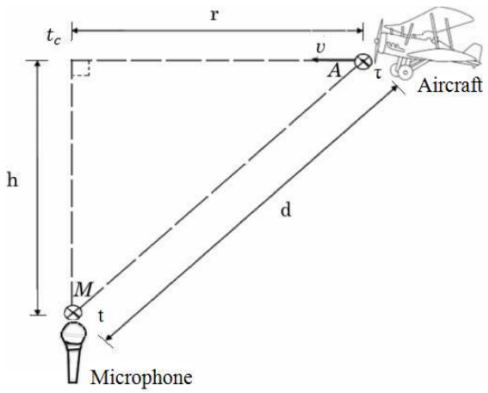


Figure 2.1: Aircraft in Uniform Motion Model

Due to the Doppler effect, the signal received at the microphone at time t can be expressed as [58, 59]:

$$y(t) = a(t)\cos\left(2\pi f(t)t + \phi(t)\right) \tag{2.1}$$

where $f(t)$ is the **instantaneous frequency (IF)** [1] modulated by Doppler effects, and $a(t)$ is the time-varying amplitude from geometric attenuation.

The emission and reception times are related through the propagation delay:

$$t = \tau + \frac{d(t)}{c} \tag{2.2}$$

where $d(t)$ is the source-sensor distance at time t.

2.2.3 Instantaneous Frequency and Motion Parameters

The key insight is that the time-varying IF $f(t)$ encodes motion information. For a source moving at constant speed v along a straight trajectory at altitude h with closest point of approach (CPA) at t_c, the IF follows [58, 59]:

$$f(t) = f_0\left[\frac{c^2}{c^2 - v^2} - \frac{c^2v^2(t - t_c)}{(c^2 - v^2)\sqrt{h^2(c^2 - v^2) + c^2v^2(t - t_c)^2}}\right] \tag{2.3}$$

This nonlinear relationship[2] captures how the Doppler shift evolves with:

- f_0: Emitted frequency (Hz)
- v: Source speed (m/s)

[1] The instantaneous frequency (IF) of a real signal is defined as the time derivative of the phase of its analytic signal: $f(t) = \frac{1}{2\pi}\frac{d\phi(t)}{dt}$. Further explanations are given in 2.6.

[2] The derivatiion of 2.3 could be found in [59]

- h: Altitude (m)
- t_c: CPA time (s)

The term $(t-t_c)$ introduces temporal asymmetry around CPA, while h and v scale the Doppler shift magnitude. This model has been validated through both simulations and field experiments [59].

2.3 Closed Loop Form Solution

2.3.1 Mathematical Derivation

We begin by recalling the Doppler-based instantaneous frequency (IF) model introduced in Eq. (2.3). At the *closest point of approach* (CPA), where the time is $t = t_c$, the observed frequency becomes:

$$f(t_c) = f_0 \cdot \frac{c^2}{c^2 - v^2} \tag{2.4}$$

This expression captures the Doppler shift due to source motion, expressed in terms of the emitted frequency f_0, propagation speed c, and source velocity v. To isolate the time-varying behavior, we define the normalized IF ratio:

$$\frac{f(t)}{f(t_c)} = 1 - \frac{v^2(t - t_c)}{\sqrt{h^2(c^2 - v^2) + v^2c^2(t - t_c)^2}} \tag{2.5}$$

This highlights how the frequency evolves around the CPA based on the altitude h and relative motion.

Differentiating Eq. (2.5) with respect to time yields:

$$\frac{f'(t)}{f(t_c)} = -\frac{v^2h^2(c^2 - v^2)}{[h^2(c^2 - v^2) + v^2c^2(t - t_c)^2]^{3/2}} \tag{2.6}$$

At the CPA ($t = t_c$), this simplifies significantly:

$$\frac{f'(t_c)}{f(t_c)} = -\frac{v^2}{h\sqrt{c^2 - v^2}}$$

Rearranging this gives a closed-form expression for velocity:

$$v^2 = \frac{c^2[(t - t_c)f'(t_c)]^2[f(t_c) - f(t)]^2}{([(t - t_c)f'(t_c)]^2 - [f(t_c) - f(t)]^2)\, f(t_c)^2} \tag{2.7}$$

To improve robustness in noisy conditions, we average over multiple time samples (excluding $t = t_c$):

$$v^2 = \frac{\sum\limits_{t \neq t_c} c^2[(t - t_c)f'(t_c)]^2[f(t_c) - f(t)]^2}{\sum\limits_{t \neq t_c} ([(t - t_c)f'(t_c)]^2 - [f(t_c) - f(t)]^2)\, f(t_c)^2} \tag{2.8}$$

Once the velocity is estimated, the remaining motion parameters are derived algebraically:

Emitted frequency:

$$f_0 = f(t_c)\left(1 - \frac{v^2}{c^2}\right)$$

Altitude:

$$h = -\frac{f(t_c)v^2}{f'(t_c)\sqrt{c^2 - v^2}}$$

2.3.2 Estimation Algorithm

To ensure accurate estimation across a range of signal-to-noise ratios (SNRs), we embed the closed-form derivation into a complete estimation algorithm. This algorithm includes preprocessing stages such as signal smoothing, instantaneous frequency estimation, derivative filtering, and post-estimation refinement. A summary of this structure is shown in Fig. 2.2.

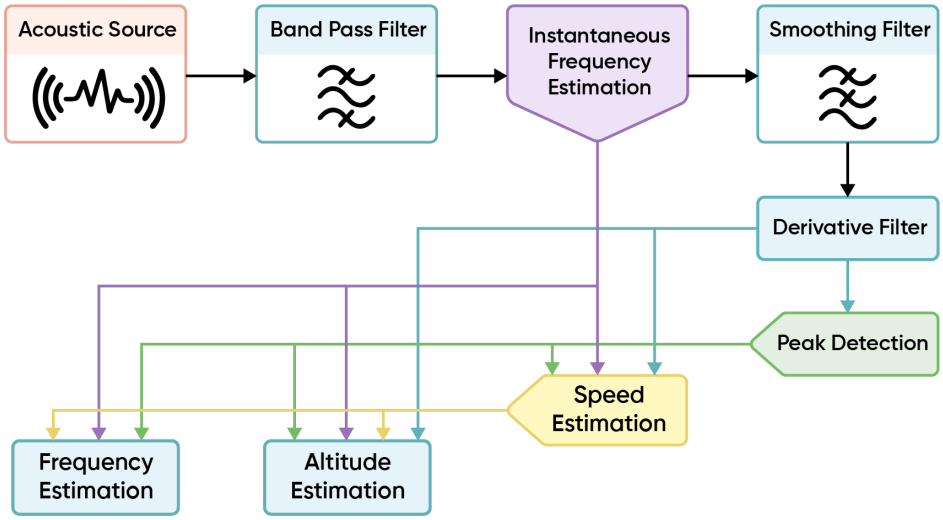


Figure 2.2: Block diagram of the full closed-form motion parameter estimation pipeline.

The algorithm's computational efficiency stems from its closed-form nature, avoiding iterative optimization while maintaining physical interpretability through the parametric Doppler model. This makes it particularly suitable for embedded implementations where resources are constrained.

Two components require careful implementation:

- Estimation of the instantaneous frequency $f(t)$
- Numerical computation of its derivative $f'(t)$

The accuracy of these components directly affects the motion parameter estimates. Our analysis will establish implementation guidelines for resource-constrained platforms while maintaining the theoretical guarantees of the closed-form solution.

2.4 Signal Conditioning and Filtering

The first stage of the proposed Doppler-based motion parameter estimation pipeline involves pre-processing the raw acoustic signal to isolate relevant spectral content and suppress noise.

The input signal $y(t)$, captured by an acoustic sensor, is first passed through a bandpass filter to isolate the frequency range of interest. The filter is centered around the expected source frequency f_0, with a bandwidth that accounts for Doppler shifts:

- **Filter Type:** 4th-order Butterworth
- **Typical Band:** $f_0 \pm 50$ Hz

This stage corresponds to the top portion of the algorithm in Fig. 2.2, encompassing the acoustic sensor and bandpass filter.

2.5 Instantaneous frequency estimation

The methods for estimating instantaneous frequency :

2.5.1 Definition and Theoretical Background

If we consider a pure sinusoidal (monochromatic) signal, described as $s(t) = a\cos(\omega t + \phi)$. This signal is defined by three parameters: the amplitude a, the pulsation ω (where the frequency f is related by $\omega = 2\pi f$), and the initial phase ϕ.

The source signal is written as:

$$s(t) = a(t)\cos(\omega(t)t + \theta)$$

- **Gabor Concept:**
  Gabor [60] proposed a method to generate a unique complex signal from a real one using the Hilbert transform:

$$z(t) = s(t) + j\mathcal{H}[s(t)]$$
$$z(t) = a(t)e^{j\phi(t)}$$

  where $\mathcal{H}$ is the Hilbert transform. The instantaneous frequency is defined as:

$$f_i(t) = \frac{1}{2\pi}\frac{d\phi(t)}{dt}$$

- **Ville Concept:**
  Ville [61] unified the above approaches and defined the instantaneous frequency of a signal $s(t) = a(t)\cos(\omega(t)t + \theta)$ as:

$$f_i(t) = \frac{1}{2\pi}\frac{d}{dt}\arg(s_A(t))$$

  where $s_A(t)$ denotes the analytic signal.

2.5.2 Phase difference method

Based on Ville's formulation, the instantaneous frequency of the analytic signal $s_A(t)$ is:

$$f_i(t) = \frac{1}{2\pi}\frac{d\phi(t)}{dt}$$

To implement the IF in discrete time, we approximate the derivative using finite differences:

Forward finite difference (FFD) [62]:

$$f_i(n) = \frac{1}{2\pi}[\phi(n+1) - \phi(n)]$$

Backward finite difference (BFD) [62]:

$$f_i(n) = \frac{1}{2\pi}[\phi(n) - \phi(n-1)]$$

Central finite difference (CFD) [62]:

$$f_i(n) = \frac{1}{2\pi}[\phi(n+1) - \phi(n-1)]$$

For higher precision, generalized forms of finite difference can be used:

$$\hat{f}(n) = \frac{1}{2\pi}\sum_{k=-q/2}^{q/2} b_k\phi(n+k)$$

where b_k are differentiation coefficients and q is even. These are unbiased for polynomial phase signals up to order q [62].

The CFD estimator is unbiased for linear FM signals but suffers from high variance in noisy conditions. Variance reduction strategies include:

- Bandlimiting the signal to a known bandwidth B,
- Applying smoothing filters before or after differentiation [62].

Smoothed Phase Difference (Kay Estimator) [63, 62]:

$$f_i(n) = \frac{1}{2\pi}\sum_{k=0}^{N-2} h_k[\phi(k+1) - \phi(k)]$$

with smoothing window:

$$h_k = \frac{1.5N}{N^2-1}\left(1 - \left[\frac{k - N/2 + 1}{N/2}\right]^2\right)$$

This reduces variance by approximately $\frac{N}{6}$ times.

2.5.3 Adaptive instantaneous frequency estimation

1. Least Mean Squares (LMS) [64, 65, 62]:
Griffiths introduced an adaptive IF estimator based on linear prediction [64], updated using the LMS algorithm by Widrow and Hoff. The data vector and coefficient update are given by:

$$\mathbf{a}_{n+1} = \mathbf{a}_n - 2\mu e_{n+1}\mathbf{z}_n^* \quad \text{and} \quad e_{n+1} = z_{n+1} + \mathbf{z}_n^T\mathbf{a}_n$$

The IF estimate is obtained from the argument of the first predictor coefficient:

$$f_i(n) = \frac{1}{2\pi}\arg[a_1^*(n)]$$

2. Recursive Least Squares (RLS) [65, 62]:
The RLS algorithm improves convergence using a forgetting factor:

$$\mathbf{a}_{n+1} = \mathbf{a}_n - 2P_n e_{n+1}\mathbf{z}_n^*, \quad e_{n+1} = z_{n+1} + \mathbf{z}_n^T\mathbf{a}_n, \quad P_n = [\alpha P_{n-1}^{-1} + \mathbf{z}_n^*\mathbf{z}_n^T]^{-1}$$

It achieves better tracking and noise suppression than LMS, especially for varying IF.

2.5.4 Extended Kalman Filter

Kalman-based IF estimation formulates the phase dynamics in a nonlinear state-space model [62]:

$$X_{n+1} = AX_n + w_n, \quad y_n = a_n\cos(\theta_n) + v_n$$

where:

$$X_n = [\theta_n, \dot{\theta}_n, \ddot{\theta}_n, a_n]^T, \quad A = \begin{pmatrix} 1 & \Delta & \frac{\Delta^2}{2} & 0 \\ 0 & 1 & \Delta & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Jacobian matrix used in updates:

$$H_n = \frac{\partial h}{\partial X} = [-a_n\sin(\theta_n), 0, 0, \cos(\theta_n)]$$

2.5.5 Kalman Least Squares (KLS) Method

Nizampatnam and Kumar proposed a hybrid approach combining Least Squares and Kalman filtering for robust IF estimation [66]. In our Doppler tracking application, we reverse their approach: LS is applied first, followed by Kalman filtering. This preserves frequency trends and avoids phase distortions.

2.5.6 Time frequency Distribution

Time-frequency representations allow energy localization in time and frequency. The total energy is conserved:

$$E = \int\int P(t,\omega)\, d\omega\, dt$$

with marginal conditions:

$$\int P(t,\omega)\,d\omega = |x(t)|^2, \quad \int P(t,\omega)\,dt = |X(\omega)|^2$$

Cohen's class distributions [67]:

$$C(t,\omega) = \frac{1}{4\pi^2}\iiint s\left(u+\frac{\tau}{2}\right)s^*\left(u-\frac{\tau}{2}\right)\phi(\theta,\tau)\,e^{j\theta t - j\omega\tau + j\theta\tau}\,du\,d\tau\,d\theta$$

where $\phi(\theta,\tau)$ is the kernel function defining the specific member of the Cohen's class.

Wigner-Ville Distribution (WVD):

$$W_x(t,f) = \int x\left(t+\frac{\tau}{2}\right)x^*\left(t-\frac{\tau}{2}\right)e^{-2\pi j f\tau}d\tau$$

Instantaneous Frequency Estimation from TFDs:

1. **First Moment Method [62]:**

$$f_i(t) = \frac{\int f P(t,f)\,df}{\int P(t,f)\,df}$$

2. **Peak Method [62]:**

$$f_i(t) = \arg\max_f P(t,f)$$

3. **Advanced Tracking Methods [62, 68]:** Viterbi-based ridge tracking or MAP-based smoothing offer robust multi-component IF estimation.

These methods enable precise localization of IF even in the presence of non-linear frequency variations or overlapping components, making TFDs valuable tools for Doppler signal analysis and motion parameter estimation.

2.5.7 Comparison of Instantaneous Frequency Estimation Methods

To evaluate the performance of different instantaneous frequency (IF) estimation techniques, we simulate a Doppler-modulated acoustic signal corrupted by additive white Gaussian noise at varying signal-to-noise ratios (SNRs). The true IF trajectory is analytically defined and compared against estimated IF curves produced by four methods:

- **instfreq:** MATLAB's built-in instantaneous frequency function using analytic signal differentiation.
- **TF Toolbox:** A time-frequency ridge extraction approach from the MATLAB Time-Frequency Toolbox.
- **Kay Estimator:** A smoothed phase-difference-based maximum likelihood estimator.
- **Kalman:** An Extended Kalman Filter tracker of phase and frequency.

Figure 2.3 summarizes the estimation results across SNR levels ranging from 0 dB to 40 dB. All methods are evaluated on the same signal realization to ensure comparability.

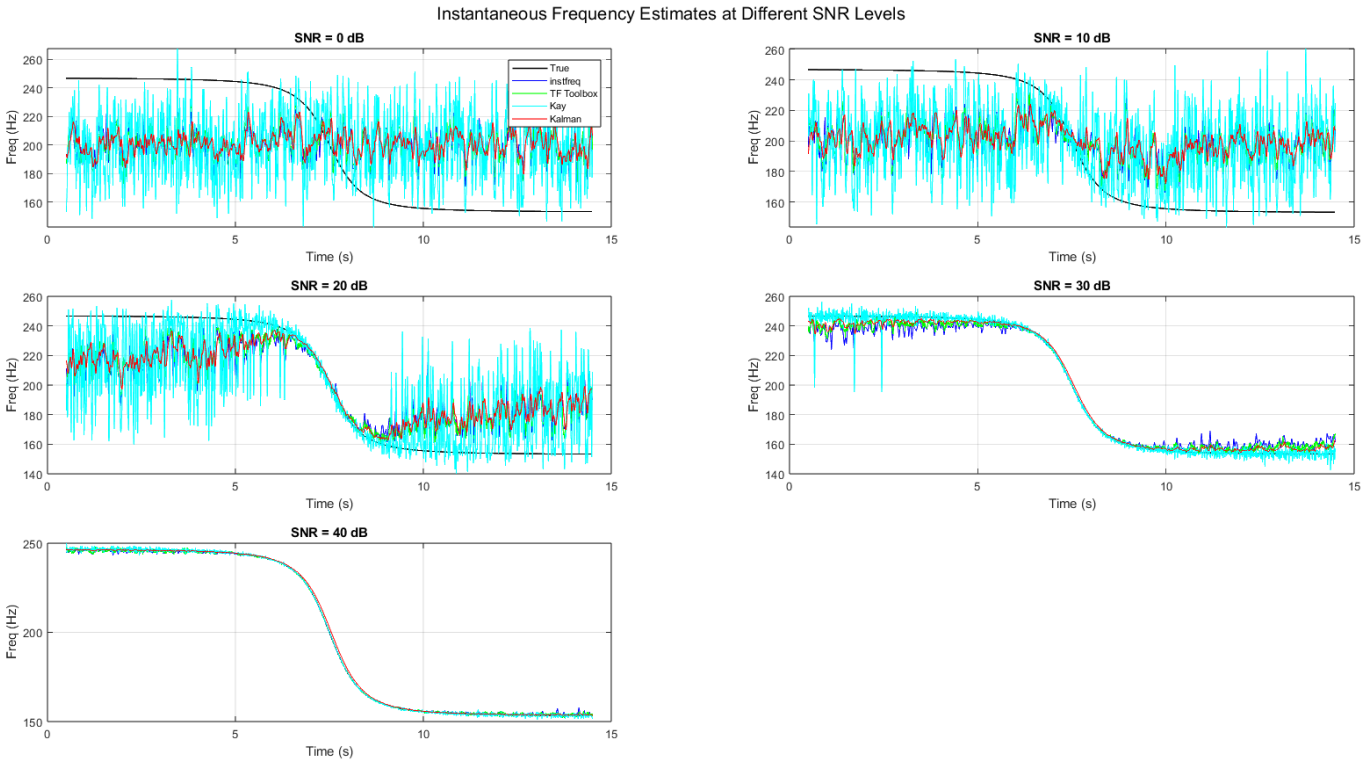


Figure 2.3: Instantaneous frequency estimates at different SNR levels (0–40 dB). The true IF trajectory is shown in black. Each subplot corresponds to one SNR value.

At high SNR (higher to 30 dB), all estimators closely follow the true IF curve, with minimal variance. However, as the SNR decreases, significant differences in performance become evident:

- **Kalman Filter:** While smooth, it tends to lag during fast frequency transitions (see 5–10 s window), especially at 0–10 dB. It preserves trend but may oversmooth dynamic events.
- **Kay Estimator:** Offers the best accuracy-variance trade-off, especially under moderate noise (10–20 dB), confirming robustness results.
- **TF Toolbox:** Suffers from high-frequency noise, particularly at low SNRs, though it generally tracks the IF trend.
- **instfreq:** Shows large fluctuations at low SNR due to its sensitivity to instantaneous phase noise.

Overall, the Kay estimator maintains low variance without excessive smoothing and shows superior robustness to noise across SNR levels. It is therefore selected as the primary IF estimator in the Doppler tracking module of this work.

2.6 Derivative Filter

To estimate the time derivative of the instantaneous frequency (IF) at the point of closest approach (CPA), direct numerical differentiation is avoided, as it tends to amplify noise in the signal. Instead, a dedicated digital filter is designed and applied to provide a more robust and smooth estimate of the IF derivative. This approach enhances stability and accuracy, especially under low signal-to-noise ratio (SNR) conditions.

2.6.1 Filter Design

A five-point central difference filter is used to approximate the derivative of the instantaneous frequency:

$$f'(t) \approx \frac{-f(t+2) + 8f(t+1) - 8f(t-1) + f(t-2)}{12\Delta t}$$

where Δt is the time step between consecutive samples.

This formulation is derived from a second-order Taylor series expansion and provides a good balance between accuracy and noise suppression. It can be interpreted as a band-limited differentiator with linear phase response, as shown in Fig. 2.4.

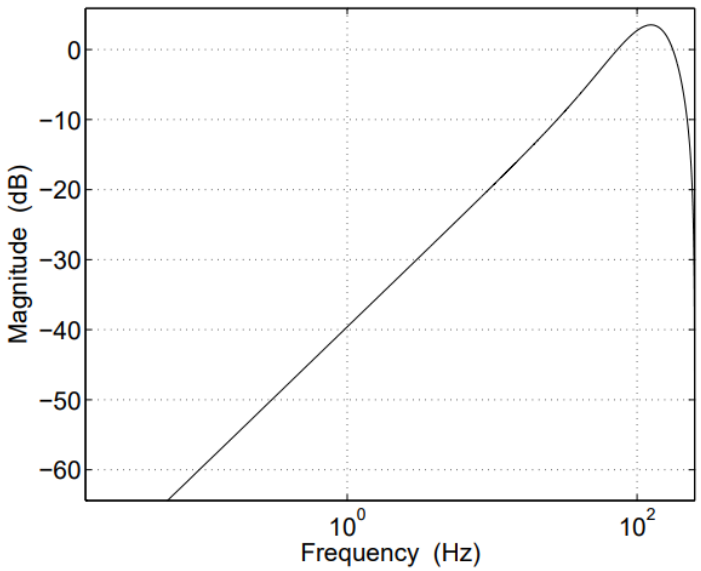


Figure 2.4: Frequency response of the five-point central difference derivative filter [59].

This stage directly follows the IF estimation block in the algorithm and produces the necessary input $f'(t)$ for parameter estimation.

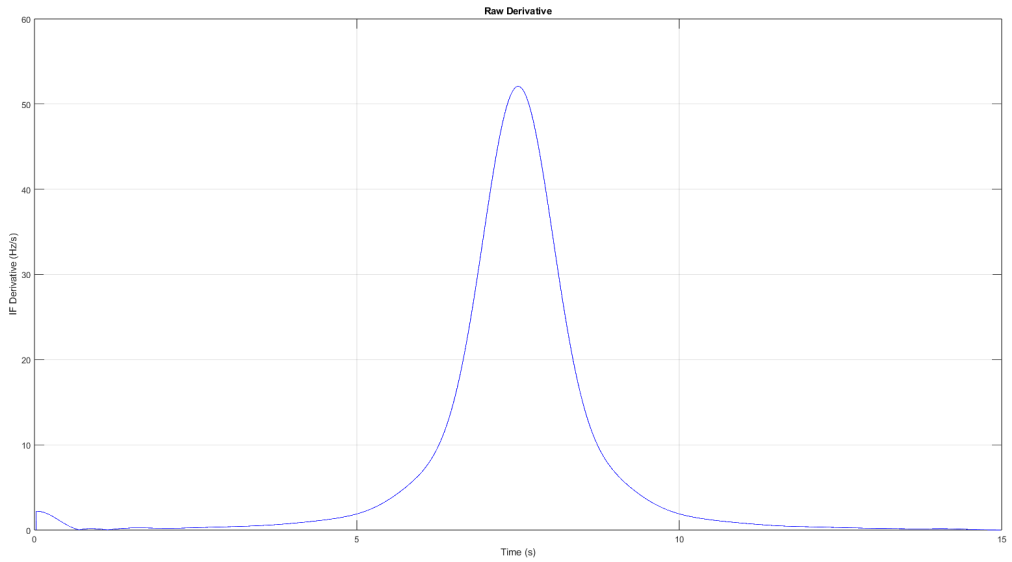


Figure 2.5: Result of the derivative of the IF estimated through the Kay Estimator

The pseudocode in Algorithm 1 formalizes this pipeline into a practical estimation routine.

Algorithm 1 Complete Motion Parameter Estimation

Input: Raw acoustic signal $y(t)$, sampling rate f_s
Output: Estimated parameters (f_0, v, h, t_c)
Apply 4th-order Butterworth bandpass filter centered at f_0
Estimate IF $f(t)$ using selected method (e.g., Kay, LMS, Kalman)
Compute $f'(t)$ via five-point difference filter
Detect $t_c \leftarrow \arg\max |f'(t)|$
Compute v using Eq. (2.8)
Compute h, f_0 from derived expressions
return (f_0, v, h, t_c)

2.7 Performance Evaluation

Through systematic Monte Carlo simulations across 10-50 dB SNR, we assess the algorithm's accuracy and robustness. The results reveal distinct performance characteristics for each estimated parameter.

2.7.1 Simulation Setup

- **Signal Parameters:** $f_0 = 2\,\text{kHz}$, $f_s = 8\,\text{kHz}$
- **Scenario:** $v = 60\,\text{m/s}$, $h = 80\,\text{m}$
- **Noise Levels:** 10-50 dB SNR in 5 dB increments
- **Trials:** 500 runs per SNR condition

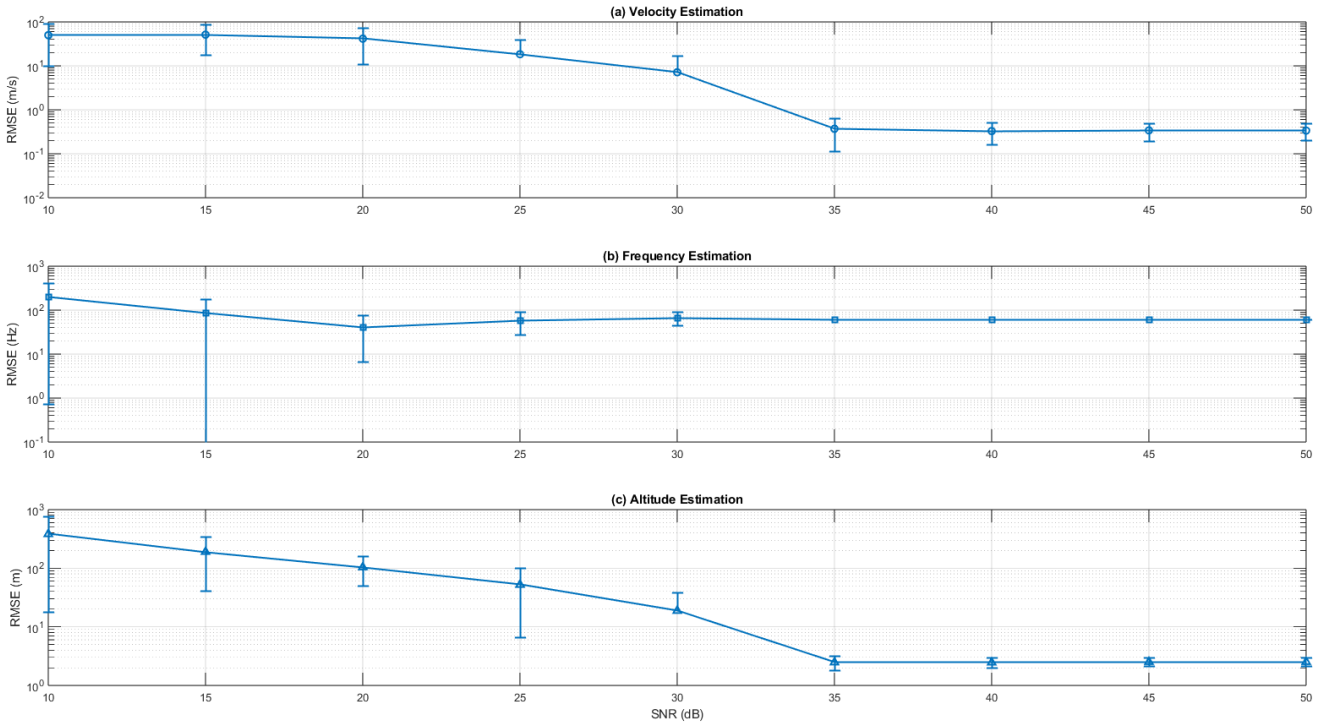


Figure 2.6: Estimation accuracy versus SNR showing: (a) Velocity RMSE drops dramatically above 25 dB SNR, (b) Frequency estimates exhibit consistent bias, (c) Altitude requires >30 dB SNR for reliable estimation. Shaded regions indicate ±1 standard deviation.

2.7.2 Performance Results

Velocity Estimation

- **Noise-Dominated (10-20 dB):** RMSE 49.8 m/s to 41.5 m/s
- **Transition (25-30 dB):** RMSE improves to 7.1 m/s
- **High-SNR (35+ dB):** Achieves 0.34 m/s RMSE

Frequency Estimation

- Persistent 60 Hz bias across all SNRs
- Variance reduces from 200 Hz (10 dB) to 0.1 Hz (50 dB)

Altitude Estimation

- Unreliable below 25 dB (RMSE > 100 m)
- Stabilizes to 2.5 m RMSE above 35 dB

2.7.3 Computational Performance

The algorithm demonstrates very low computation time:

- Average processing time: 10.9(12) ms per trial
- Dominated by IF estimation (70% of runtime)

2.7.4 Discussion of Limitations

- **Frequency Bias:** Suggests need for calibration in IF estimation.
- **Altitude Sensitivity:** Requires >30 dB SNR for usable results.
- **Real-Time Margin:** Current ∼10ms latency allows for 100 Hz update rates.

2.7.5 Recommended Improvements

- Develop adaptive filtering for low-SNR conditions.
- Optimize IF estimation for faster execution.
- Improve robustness to noise

2.8 Conclusion

In this chapter, we presented an analytical solution for motion parameter estimation based on the Doppler effect. The approach allows for the estimation of source velocity, emission frequency, and altitude using the time-varying instantaneous frequency (IF) of a Doppler-shifted tone. Several IF estimation methods were reviewed—ranging from classical to adaptive and time–frequency approaches—as they form the core of the algorithm's accuracy and robustness. This foundation enables non-array-based localization using a single receiver, paving the way for low-cost and efficient Doppler-based tracking systems.

Chapter 3

Acoustic Source Localization Using Microphone Arrays

Introduction

Acoustic source localization using microphone arrays consists in estimating the direction of arrival (DOA) of one or more sound sources based on spatial sampling of the acoustic field. The present chapter focuses on the design and implementation of localization algorithms for both narrowband and wideband signals using a compact, real-time capable 16-element microphone array.

The spectral characteristics of the source—whether narrowband (e.g., tonal signals) or wideband (e.g., speech or environmental noise)—influence the modeling assumptions and localization strategies. In narrowband conditions, frequency-invariant propagation delays permit the use of classical methods such as delay-and-sum beamforming, Bartlett, Capon (MVDR), and subspace-based approaches like MUSIC. However, wideband signals exhibit frequency-dependent phase shifts, rendering narrowband assumptions invalid in the time domain.

To address this, many wideband localization frameworks adopt a time–frequency representation of the signal, where narrowband models are applied within each frequency bin of the Short-Time Fourier Transform (STFT). This approach, enables the use of narrowband-based spatial filters and high-resolution techniques in wideband scenarios, while introducing challenges such as frequency-bin alignment and coherence loss.

This chapter is organized to progressively introduce, compare, and justify the localization methods used in our system. After presenting the narrowband signal and propagation model, we generalize to the wideband case and review key estimation strategies. Both offline and real-time implementations are discussed, with a focus on techniques that are robust to reverberation and computationally feasible in embedded contexts. The chapter concludes with the integration of localization outputs into an acoustic camera framework for real-time visualization, and sets the stage for the source separation techniques addressed in the next chapter.

The remainder of the chapter is structured as follows:

- Section 3.1 introduces the narrowband signal model, spatial aliasing constraints, and classical DOA estimation techniques such as beamforming (DAS, Bartlett, MVDR) and high-resolution methods like MUSIC.

- Section 3.3 generalizes the signal model to wideband sources and highlights the limitations of directly applying narrowband methods to such signals.
- Sections 3.3.1–3.4 present time-delay estimation methods for wideband localization, including Generalized Cross-Correlation (GCC), its spectral weighting variants (PHAT, SCOT, etc.), and the SRP-PHAT algorithm for spatial likelihood mapping.
- Section 3.4.6 extends narrowband beamforming techniques to the wideband case using STFT-based frequency-domain processing, and introduces multichannel enhancement-based localization using MWF and SDW-MWF.
- Section 3.4.7 provides an overview of wideband subspace-based localization techniques, classified into coherent (e.g., CSSM, TCT), incoherent (e.g., IMUSIC, TOFS), and hybrid (e.g., TOPS, S-TOPS) methods.

3.1 Narrowband Signal Model

Despite the inherently wideband nature of real-world acoustic sources such as speech and environmental sounds, the narrowband signal model remains a cornerstone of array signal processing. Its mathematical tractability and foundational role in beamforming and high-resolution methods make it highly relevant, especially when applied in the time–frequency domain via Short-Time Fourier Transform (STFT) decomposition. In this context, each time–frequency bin is treated as a locally narrowband segment, enabling the use of classical narrowband techniques within a wideband processing framework

We consider a single far-field source emitting a narrowband signal centered at frequency f_0, captured by a calibrated array of M microphones with known positions $\{\mathbf{r}_m\}_{m=1}^{M} \in \mathbb{R}^3$. A signal is considered narrowband if its bandwidth B satisfies $B \ll f_0$, allowing uniform approximation of propagation delays across its spectral components.

The following assumptions are adopted:

- **Far-field propagation**: wavefronts impinging on the array are planar.
- **Free-field environment**: no reverberation or multipath; propagation is linear and homogeneous.
- **Sensor calibration**: microphone positions and timing are accurately known.

These conditions underpin most narrowband DOA estimation algorithms, including beamforming and subspace-based techniques [69].

3.1.1 Uniform Linear Array Model and Generalization to URA

To build up to the general Uniform Rectangular Array (URA) model, we begin with the simpler 1D Uniform Linear Array (ULA) case. As illustrated in Figure 3.1, a narrowband plane wave arriving from direction θ induces a relative delay between adjacent microphones spaced by d. This delay is given by:

$$\tau = \frac{d \sin \theta}{c}$$

where c is the speed of sound. This fundamental relationship underpins most array signal processing methods, including delay-and-sum beamforming and the construction of steering vectors.

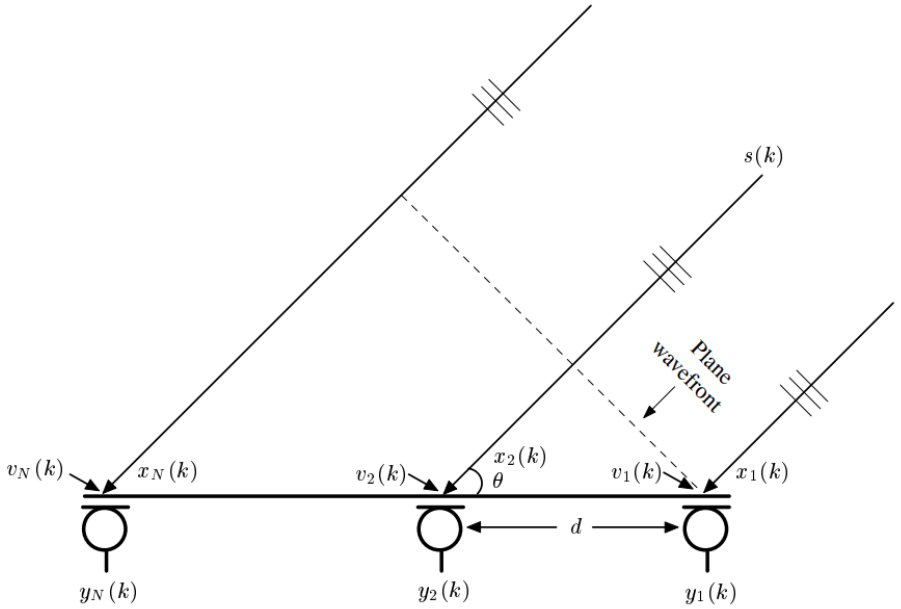


Figure 3.1: Narrowband signal model for a plane wave impinging on a Uniform Linear Array (ULA). The direction of arrival θ creates a fixed time delay $\tau = \frac{d \sin \theta}{c}$ between sensors.

This 1D formulation extends naturally to two-dimensional arrays, such as the Uniform Rectangular Array (URA), where microphones are distributed along both x- and y-axes. We now present the full spatial model for a URA.

3.1.2 Signal Model for Uniform Rectangular Arrays

We assume a Uniform Rectangular Array (URA) as shown in 3.2 with $M = M_x \times M_y$ microphones spaced uniformly by d_x and d_y along the x- and y-axes, respectively. The spatial coordinates of a sensor indexed by (m_x, m_y) are given by:

$$\mathbf{r}_{m_x,m_y} = \begin{bmatrix} (m_x - 1)d_x \\ (m_y - 1)d_y \\ 0 \end{bmatrix} \tag{3.1}$$

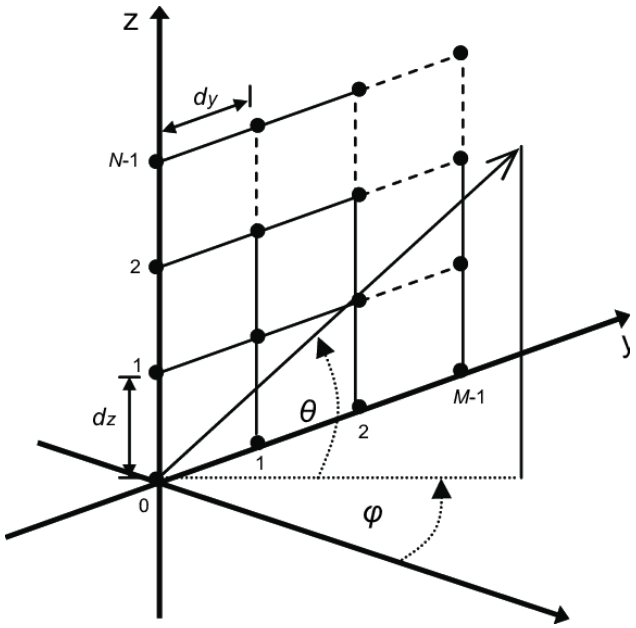


Figure 3.2: Uniform rectangular array (URA) [70]

Assuming a far-field source located in the direction (θ, ϕ) (azimuth and elevation), the unit direction vector is:

$$\mathbf{u}(\theta, \phi) = \begin{bmatrix} \cos(\phi)\cos(\theta) \\ \cos(\phi)\sin(\theta) \\ \sin(\phi) \end{bmatrix} \tag{3.2}$$

The delay at microphone (m_x, m_y) is:

$$\tau_{m_x,m_y} = \frac{1}{c}\,\mathbf{r}_{m_x,m_y}^{\top}\mathbf{u}(\theta,\phi) = \frac{1}{c}\left[(m_x-1)d_x\cos(\phi)\cos(\theta) + (m_y-1)d_y\cos(\phi)\sin(\theta)\right] \tag{3.3}$$

The received time-domain signal is modeled as:

$$x_m(t) = s(t-\tau_m) + n_m(t) \tag{3.4}$$

where $s(t)$ is the source signal, τ_m is the propagation delay, and $n_m(t)$ is an additive noise.

In the frequency domain, assuming a narrowband source centered at f_0, the model becomes:

$$X_m(f_0) = S(f_0)e^{-j2\pi f_0\tau_m} + N_m(f_0) \tag{3.5}$$

Stacking all microphone signals into a vector yields:

$$\mathbf{X}(f_0) = S(f_0)\,\mathbf{v}(\theta,\phi) + \mathbf{N}(f_0) \tag{3.6}$$

with the steering vector:

$$\mathbf{v}(\theta,\phi) = \begin{bmatrix} e^{-j2\pi f_0\tau_1} \\ e^{-j2\pi f_0\tau_2} \\ \vdots \\ e^{-j2\pi f_0\tau_M} \end{bmatrix} \tag{3.7}$$

This vector encodes the phase differences across the array as a function of the source direction. While the derivation is given for a URA, the model generalizes to other array geometries, including ULAs, circular arrays, and arbitrary configurations, by adjusting the delay vector τ_m accordingly.

3.1.3 Spatial Aliasing and Microphone Spacing

To avoid spatial aliasing and ensure unambiguous localization, the microphone spacing must satisfy the spatial Nyquist criterion:

$$d < \frac{\lambda_{\min}}{2} = \frac{c}{2f_{\max}} \tag{3.8}$$

where $\lambda_{\min}$ is the minimum wavelength corresponding to the highest signal frequency $f_{\max}$. Exceeding this limit introduces grating lobes, which cause ambiguity in DOA estimation and degrade localization performance.

The choice of d_x and d_y must therefore reflect the spectral content of the expected sources, balancing spatial resolution against array size and sensor placement constraints.

3.2 Narrowband Localization Methods

In this section, we present a series of direction-of-arrival (DOA) estimation techniques grounded in the narrowband signal model. While many real-world acoustic sources—such as human speech or UAV noise—exhibit wideband characteristics, narrowband localization methods remain fundamental to array processing and are frequently employed within time–frequency frameworks. Specifically, when a Short-Time Fourier Transform (STFT) is applied to a wideband signal, each time–frequency bin can be treated as locally narrowband, allowing these methods to be applied per frequency bin and then aggregated. This decomposition justifies the use of narrowband algorithms even in wideband contexts.

3.2.1 Beamforming for Source Localization

Beamforming is a spatial filtering technique widely used for direction-of-arrival estimation. It enhances signals arriving from a specific direction while attenuating interference and noise from others. This directional selectivity makes beamforming a powerful tool in applications such as UAV detection, speaker tracking, and human–robot interaction [71].

The core principle involves steering the array's sensitivity toward candidate directions and evaluating the output power. For a given direction (θ, ϕ), the beamformer output at frequency f_0 is:

$$Y(f_0; \theta, \phi) = \mathbf{w}^H(\theta, \phi)\mathbf{X}(f_0), \tag{3.9}$$

where $\mathbf{w}(\theta, \phi)$ is a direction-dependent weight vector and $\mathbf{X}(f_0)$ is the vector of sensor observations. The corresponding spatial power spectrum is:

$$P(\theta, \phi) = \mathbb{E}\left[|Y(f_0; \theta, \phi)|^2\right], \tag{3.10}$$

and DOA estimation is achieved by identifying the direction that maximizes this quantity:

$$(\theta_s, \phi_s) = \arg\max_{(\theta,\phi)} P(\theta, \phi). \tag{3.11}$$

This general formulation underlies various beamforming strategies, from simple delay-and-sum to advanced high-resolution techniques such as MVDR and MUSIC [69].

3.2.2 Delay-and-Sum Beamforming

Delay-and-Sum (DAS) beamforming is a straightforward method that delays each sensor signal to compensate for propagation time from a hypothesized direction, then sums them. The block diagram of this process is illustrated in Figure 3.3, where each input is delayed according to the estimated direction of arrival, then coherently summed to enhance the signal from that direction.

$$y(t; \theta, \phi) = \sum_{m=1}^{M} x_m(t + \tau_m(\theta, \phi)) \tag{3.12}$$

The estimated source direction is obtained by scanning over all possible directions and selecting the one that maximizes the output power:

$$(\theta_s, \phi_s) = \arg\max_{(\theta,\phi)} \int |y(t;\theta,\phi)|^2 dt \tag{3.13}$$

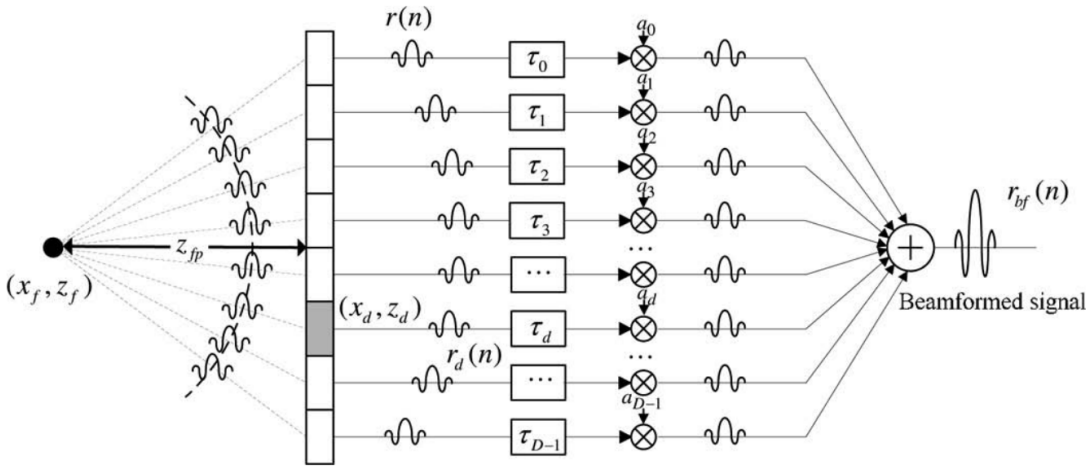


Figure 3.3: Structure of Delay-and-Sum Beamformer. Each signal is delayed based on hypothesized direction, then summed to form the beamformer output.

DAS requires no prior knowledge of the signal statistics and is robust and simple to implement. However, it suffers from limited angular resolution and high sidelobe levels, particularly when the number of microphones is small or the SNR is low.

3.2.3 Spectral Beamformers: Bartlett and Capon

To address the limitations of DAS, spectral-domain beamformers exploit second-order statistics of the observed signals.

Bartlett Beamformer

Also called conventional beamforming, Bartlett projects the sensor data onto a steering vector $\mathbf{v}(\theta, \phi)$:

$$P_{\text{Bartlett}}(\theta, \phi) = \mathbf{v}^H(\theta, \phi)\mathbf{R}\mathbf{v}(\theta, \phi) \tag{3.14}$$

where $\mathbf{R} = \mathbb{E}[\mathbf{X}(f_0)\mathbf{X}^H(f_0)]$ is the spatial covariance matrix. While simple, it does not adapt to the spatial interference structure and offers limited resolution.

Capon (MVDR) Beamformer

The Minimum Variance Distortionless Response (MVDR) beamformer, introduced by Capon [72], minimizes output power while maintaining a fixed gain in the look direction:

$$\min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \quad \text{subject to } \mathbf{w}^H \mathbf{v}(\theta, \phi) = 1 \tag{3.15}$$

This yields the optimal weights:

$$\mathbf{w}_{\text{MVDR}} = \frac{\mathbf{R}^{-1}\mathbf{v}(\theta, \phi)}{\mathbf{v}^H(\theta, \phi)\mathbf{R}^{-1}\mathbf{v}(\theta, \phi)} \tag{3.16}$$

and the MVDR spectrum:

$$P_{\text{MVDR}}(\theta, \phi) = \frac{1}{\mathbf{v}^H(\theta, \phi)\mathbf{R}^{-1}\mathbf{v}(\theta, \phi)} \tag{3.17}$$

MVDR offers enhanced spatial resolution and better interference suppression compared to Bartlett, but depends critically on the quality of covariance estimation and array calibration [73].

3.2.4 High-Resolution Subspace Method: MUSIC

The MUltiple SIgnal Classification (MUSIC) algorithm [74] is a high-resolution technique based on eigenspace decomposition of the covariance matrix:

$$\mathbf{R} = \mathbf{U}_s \mathbf{\Lambda}_s \mathbf{U}_s^H + \mathbf{U}_n \mathbf{\Lambda}_n \mathbf{U}_n^H \tag{3.18}$$

The noise subspace $\mathbf{U}_n$ is orthogonal to the steering vector $\mathbf{v}(\theta, \phi)$, leading to the pseudo-spectrum:

$$P_{\text{MUSIC}}(\theta, \phi) = \frac{1}{\mathbf{v}^H(\theta, \phi)\mathbf{U}_n \mathbf{U}_n^H \mathbf{v}(\theta, \phi)} \tag{3.19}$$

DOAs correspond to peaks of this spectrum. MUSIC offers superior resolution, even with closely spaced sources, but requires prior knowledge of the number of active sources and sufficient SNR. It is computationally demanding and sensitive to model mismatch.

The methods described above assume frequency-invariant propagation delays, which holds under narrowband conditions. However, most acoustic sources are wideband. Applying these methods directly in the time domain can lead to bias and degraded performance. Nonetheless, by decomposing wideband signals using the STFT and applying narrowband localization algorithms in each frequency bin, one can preserve their utility. This strategy underlies wideband extensions such as Steered Reponse Power with Phase Transform (SRP-PHAT) and coherent subspace methods, which are discussed in the following sections.

3.3 Wideband Acoustic Localization

As discussed earlier, narrowband DOA estimation techniques assume frequency-independent phase shifts, an assumption violated in real-world settings involving wideband signals such as speech, UAV noise, and environmental acoustics. These sources exhibit significant energy across a broad frequency range, and as a result, their propagation delays introduce frequency-dependent phase variations across the array. Consequently, narrowband localization methods become inadequate, and dedicated wideband approaches must be employed.

A signal is considered wideband when its bandwidth B is not negligible compared to its center frequency f_0, or when it lacks a well-defined center frequency altogether. For instance, human speech typically spans 300 Hz to 8 kHz, while drones generate tonal and broadband aerodynamic noise. In this context, inter-microphone phase differences vary with frequency, and modeling requires frequency-dependent steering vectors.

To address this, wideband localization systems often use a time–frequency representation via the Short-Time Fourier Transform (STFT). The time-domain signal at microphone m is decomposed as:

$$X_m(f, t) = \int x_m(\tau) h(t - \tau) e^{-j2\pi f \tau} d\tau, \tag{3.20}$$

where $h(t)$ is a temporal window function centered at time t. This decomposition permits narrowband models to be applied per frequency bin.

The frequency-domain model becomes:

$$\mathbf{X}(f, t) = \mathbf{v}(f, \theta, \phi) S(f, t) + \mathbf{N}(f, t), \tag{3.21}$$

where:

- $\mathbf{X}(f,t) \in \mathbb{C}^M$: observed signal vector at frequency f,
- $\mathbf{v}(f,\theta,\phi)$: frequency-dependent steering vector,
- $S(f,t)$: source spectrum, and
- $\mathbf{N}(f,t)$: additive noise.

Wideband localization then proceeds by estimating either time delays or spatial responses in each frequency bin and aggregating the results. The next section presents a time-domain strategy based on Time Delay Estimation (TDE), followed by its frequency-domain formulation using Generalized Cross-Correlation (GCC) methods.

3.3.1 Time Delay Estimation (TDE)

Time Delay Estimation, or Time Difference of Arrival (TDOA) estimation, is a fundamental technique for wideband localization. The arrival time difference between microphones encodes spatial information about the source.

For a source at position $\mathbf{p}_s \in \mathbb{R}^3$, and microphones at $\mathbf{r}_i$ and $\mathbf{r}_j$, the propagation delays are:

$$\tau_i = \frac{\|\mathbf{p}_s - \mathbf{r}_i\|}{c}, \quad \tau_j = \frac{\|\mathbf{p}_s - \mathbf{r}_j\|}{c}, \tag{3.22}$$

yielding a TDOA of:

$$\Delta\tau_{ij} = \tau_j - \tau_i = \frac{1}{c}\left(\|\mathbf{p}_s - \mathbf{r}_j\| - \|\mathbf{p}_s - \mathbf{r}_i\|\right). \tag{3.23}$$

Under the far-field assumption, where $\|\mathbf{p}_s\| \gg \|\mathbf{r}_i - \mathbf{r}_j\|$, this simplifies to:

$$\Delta\tau_{ij} = \frac{1}{c}(\mathbf{r}_j - \mathbf{r}_i)^\top \mathbf{u}(\theta,\phi), \tag{3.24}$$

where $\mathbf{u}(\theta,\phi)$ is the unit direction vector. This relation enables grid search algorithms such as Steered Response Power (SRP) to evaluate spatial hypotheses based on predicted TDOAs.

3.3.2 Pairwise vs. Multi-Microphone Frameworks

Two major modeling strategies exist for utilizing time-difference-of-arrival (TDOA) in acoustic localization:

Pairwise Approaches

In this class, TDOAs are estimated independently between microphone pairs. The most common method is the Generalized Cross-Correlation (GCC), which is discussed in the following section. Once all pairwise delays are estimated, geometric or grid-based triangulation methods (e.g., SRP-PHAT) are used to infer the source location. This approach is relatively simple and scalable, but may suffer from redundancy and inconsistency between pairwise estimates, especially in noisy or reverberant environments.

Multi-Microphone Approaches

These methods exploit the full spatial structure of the array, processing all channels jointly. They include wideband extensions of beamformers, subspace methods (e.g., MUSIC, CSSM), and coherent signal processing techniques. While these require more complex matrix operations and a higher computational load, they often offer better robustness and accuracy in challenging conditions.

We will review now the Steered Reponse Power method, which effectively bridges both approaches. It begins by estimating TDOAs using GCC-PHAT for each microphone pair, then aggregates the resulting information into a global spatial power map using a delay steering model. This hybrid strategy combines the simplicity of pairwise delay estimation with the robustness of beamforming.

3.4 Generalized Cross-Correlation (GCC) Methods

Time Delay Estimation (TDE) plays an important role in applications such as source localization, beamforming, and acoustic tracking. The most fundamental technique is classical cross-correlation (CC), which estimates the time delay between two received signals by locating the maximum of their correlation function.

However, in realistic environments—where noise, reverberation, and multipath effects are significant—the peak of the correlation function is often broadened, distorted, or masked. To address these limitations, Knapp and Carter [75] introduced the *Generalized Cross-Correlation* (GCC) framework, which enhances classical cross-correlation by applying a frequency-domain weighting function to pre-whiten or shape the spectrum before correlation. This improves peak sharpness and delay estimation accuracy.

3.4.1 Frequency-Domain Cross-Correlation

Let $r_1(t)$ and $r_2(t)$ be the signals received at two microphones, modeled as:

$$r_1(t) = h_1(t) * s(t) + n_1(t), \tag{3.25}$$
$$r_2(t) = h_2(t) * s(t) + n_2(t), \tag{3.26}$$

where $h_1(t)$ and $h_2(t)$ are the acoustic impulse responses, $*$ denotes convolution, and $n_1(t)$, $n_2(t)$ are additive noise terms.

Under free-field propagation, the model simplifies to:

$$r_1(t) = s(t) + n_1(t), \tag{3.27}$$
$$r_2(t) = \alpha s(t - \tau) + n_2(t), \tag{3.28}$$

where α is an attenuation factor and τ is the inter-microphone delay.

The classical cross-correlation is:

$$R_{12}(\tau) = \int r_1(t)\, r_2(t + \tau)\, dt, \tag{3.29}$$

and the estimated delay is:

$$\hat{\tau} = \arg\max_{\tau} R_{12}(\tau). \tag{3.30}$$

In the GCC framework, correlation is computed in the frequency domain:

$$R_{12}^{\mathrm{GCC}}(\tau) = \int_{-\infty}^{\infty} \Psi(f) G_{12}(f) e^{j2\pi f\tau} df, \tag{3.31}$$

where $G_{12}(f) = R_1(f)R_2^*(f)$ is the cross-power spectral density (CPSD)[1], and $\Psi(f)$ is a frequency-domain weighting function that enhances the sharpness and reliability of the delay estimate.

3.4.2 Weighting Functions in GCC

Different choices of $\Psi(f)$ yield different GCC variants, each with specific robustness properties. Let:

$$P_1(f) = |R_1(f)|^2, \quad P_2(f) = |R_2(f)|^2 [2], \tag{3.32}$$

$$\gamma(f) = \frac{|G_{12}(f)|^2}{P_1(f)P_2(f)} [3]. \tag{3.33}$$

Phase Transform (PHAT)

PHAT whitens the cross-spectrum by discarding amplitude information and emphasizing phase alignment:

$$\Psi_{\mathrm{PHAT}}(f) = \frac{1}{|G_{12}(f)|}, \quad R_{12}^{\mathrm{PHAT}}(\tau) = \int \frac{G_{12}(f)}{|G_{12}(f)|} e^{j2\pi f\tau} df. \tag{3.34}$$

Here, "whitening" refers to flattening the spectrum by equalizing its magnitude across frequencies, which sharpens the autocorrelation peak and improves delay estimation accuracy [76]. This improves robustness under reverberation [77].

Smoothed Coherence Transform (SCOT)

SCOT uses auto-spectral normalization to preserve amplitude:

$$\Psi_{\mathrm{SCOT}}(f) = \frac{1}{\sqrt{P_1(f)P_2(f)}}. \tag{3.35}$$

It performs better than PHAT under low-SNR and model mismatch conditions [78].

ROTH Filter

ROTH uses a single-channel normalization:

$$\Psi_{\mathrm{ROTH}}(f) = \frac{1}{P_1(f)}. \tag{3.36}$$

It is simple but may yield broader peaks in noisy settings [79].

[1] Here, $R_1(f)$ and $R_2(f)$ are the Fourier transforms of the signals $r_1(t)$ and $r_2(t)$. The cross-power spectral density is defined as $G_{12}(f) = R_1(f)R_2^*(f)$.

Maximum Likelihood (ML)

ML is derived from optimal estimation theory:

$$\Psi_{\text{ML}}(f) = \frac{1 - |\gamma(f)|^2}{|G_{12}(f)|}. \tag{3.37}$$

It emphasizes frequency bins with high inter-signal coherence [75, 78, 80].

Modified Cross-Spectral Phase (M-CSP)

M-CSP introduces tunable whitening:

$$\Psi_{\text{MCSP}}(f) = \frac{1}{|G_{12}(f)|^{\alpha}}, \quad 0 < \alpha < 1. \tag{3.38}$$

It interpolates between PHAT and unweighted correlation, allowing trade-offs between robustness and resolution [81]. For full derivations and theoretical justifications of the weighting functions presented in Table 3.1, the reader is referred to the works of Knapp and Carter [75], Roth [79], and the comparative analysis by Boora and Dhull [78].

Table 3.1: Summary of GCC Weighting Functions

Method	$\Psi(f)$	**Description**
PHAT	$\frac{1}{\lvert G_{12}(f)\rvert}$	Maximally whitens the spectrum. Robust to reverberation. Discards magnitude.
SCOT	$\frac{1}{\sqrt{P_1(f)P_2(f)}}$	Normalizes by auto-spectra. Retains amplitude. Better in low SNR.
ROTH	$\frac{1}{P_1(f)}$	Uses reference channel's power spectrum. Sensitive to noise asymmetry.
ML	$\frac{1-\lvert\gamma(f)\rvert^2}{\lvert G_{12}(f)\rvert}$	Coherence-weighted. Emphasizes reliable bins. Statistically optimal.
M-CSP	$\frac{1}{\lvert G_{12}(f)\rvert^{\alpha}}$, $0 < \alpha < 1$	Tunable whitening. Interpolates between PHAT and unweighted.

3.4.3 Performance in Reverberation and Noise

Boora and Dhull [78] provide a comparative study of GCC variants:

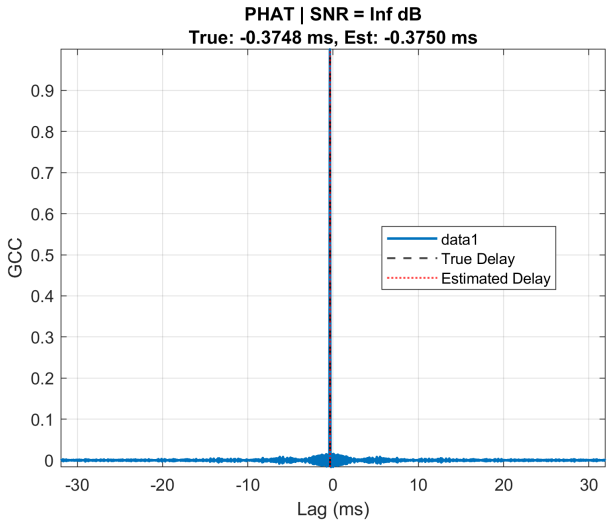
- **PHAT**: Highly robust to reverberation, but sensitive to impulsive noise.
- **SCOT**: Better performance in low-SNR and mismatched conditions.
- **ML**: Statistically optimal but sensitive to inaccurate coherence estimates.
- **ROTH**: Effective for low-SNR suppression, but yields wider correlation peaks.
- **M-CSP**: Adaptable to various conditions by tuning α.

Despite the optimality of ML and flexibility of M-CSP, PHAT remains widely used due to its simplicity, effectiveness in reverberant environments, and independence from SNR estimation.

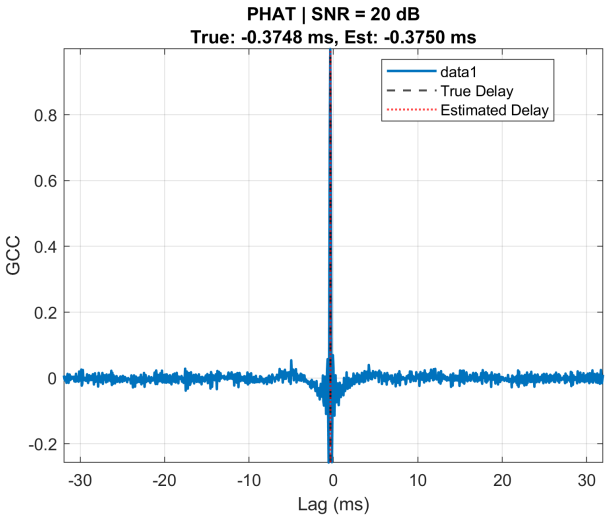
3.4.4 Simulation Results at Different Noise Levels

To evaluate the robustness of different GCC weighting functions, we simulate time delay estimation using a real speech signal recorded at 16 kHz, injected at a known direction of arrival (DOA = 50° azimuth) with an inter-microphone spacing of 20 cm. Additive white Gaussian noise (AWGN) is introduced at various SNR levels: $\{+\infty, 20, 10, 0, -5\}$ dB. The goal is to estimate the inter-microphone delay using each GCC variant and assess the sharpness and accuracy of the correlation peak.

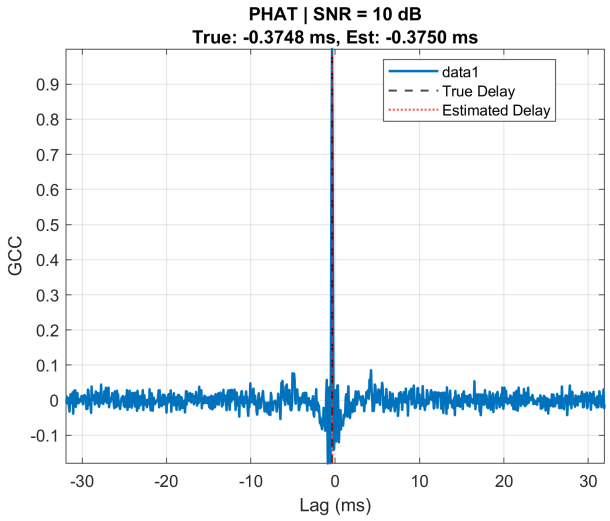
The GCC correlation functions for each method (PHAT, SCOT, ROTH, and ML) are shown in Figures 5.3–3.7. In each subfigure, the vertical lines denote the ground-truth delay (black dashed) and the estimated delay (red dotted).



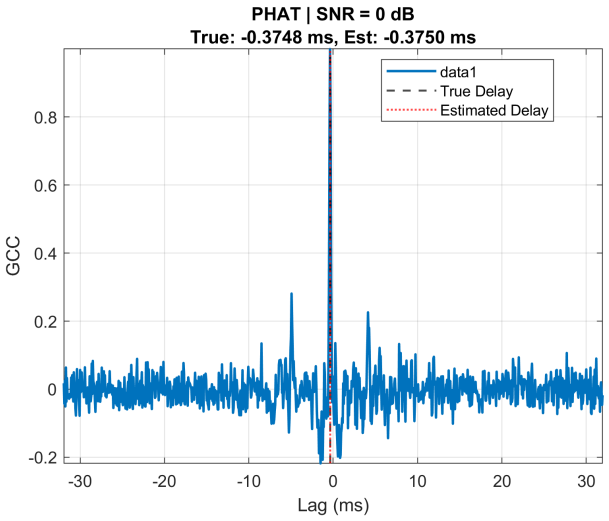
(a) Clean (∞ dB)



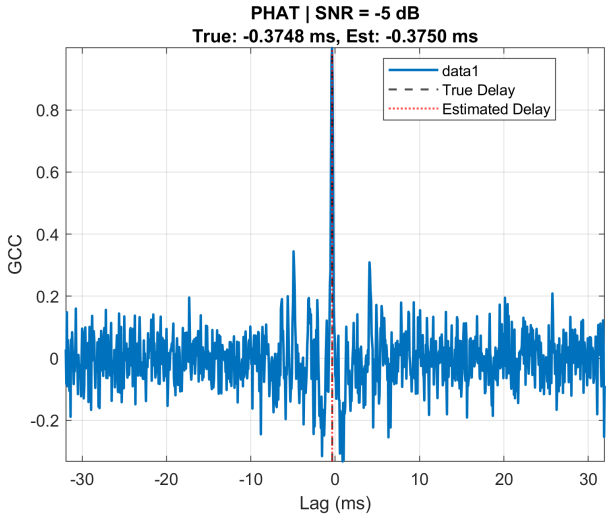
(b) 20 dB SNR



(c) 10 dB SNR

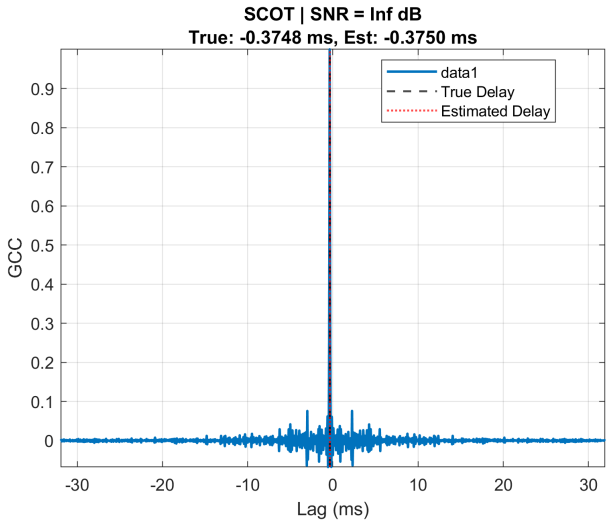


(d) 0 dB SNR

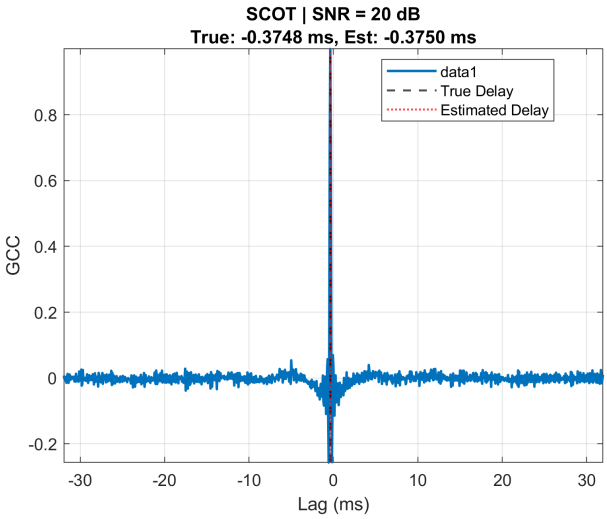


(e) -5 dB SNR

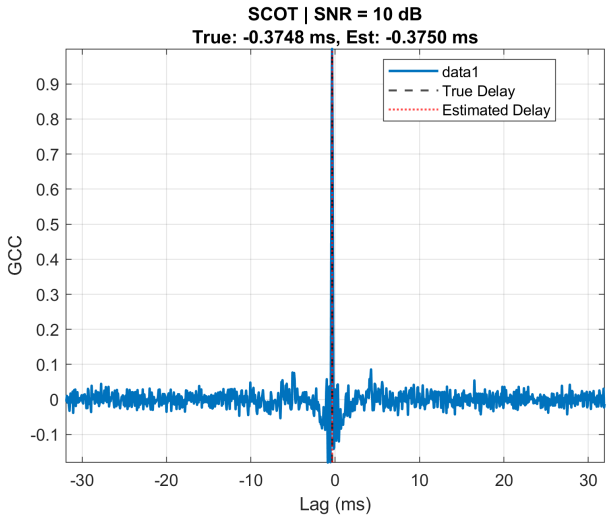
Figure 3.4: GCC-PHAT correlation functions under different SNR conditions.



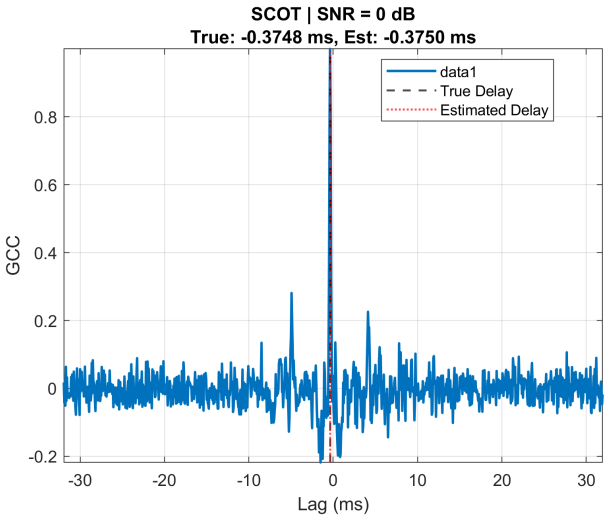
(a) Clean (∞ dB)



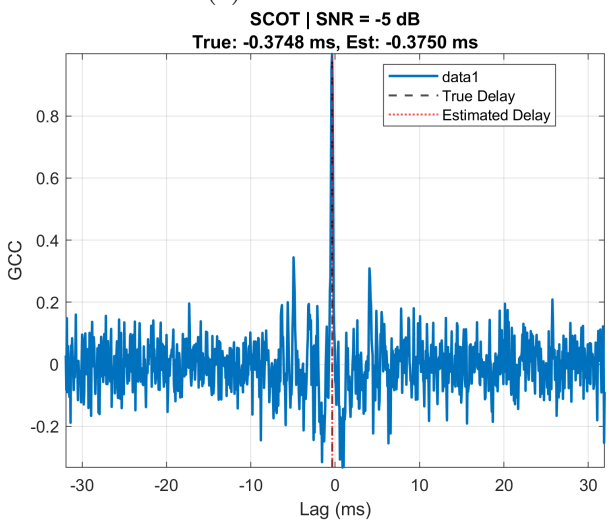
(b) 20 dB SNR



(c) 10 dB SNR

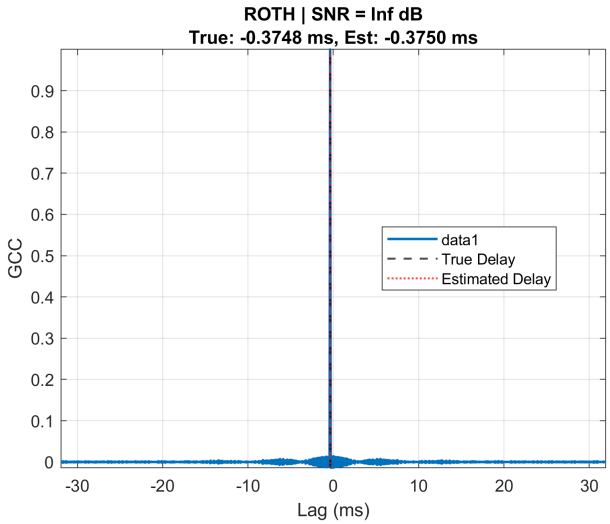


(d) 0 dB SNR

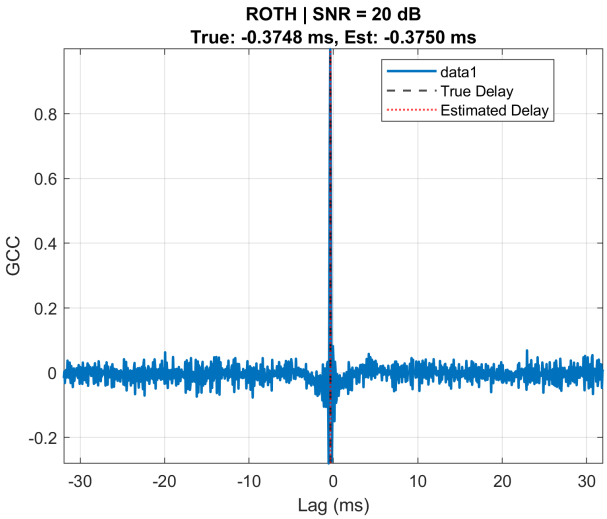


(e) -5 dB SNR

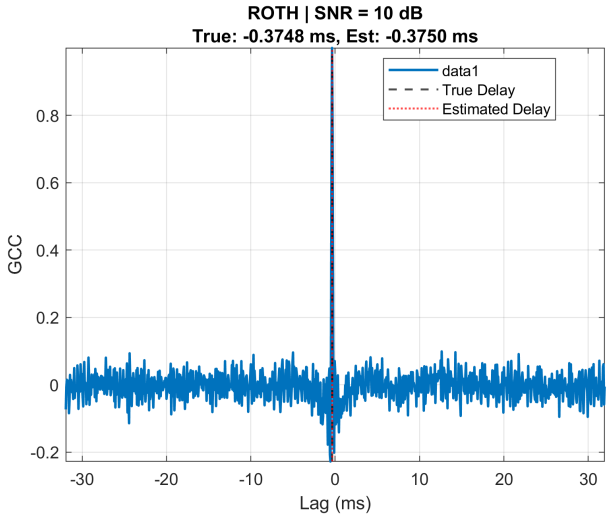
Figure 3.5: GCC-SCOT correlation functions under different SNR conditions.



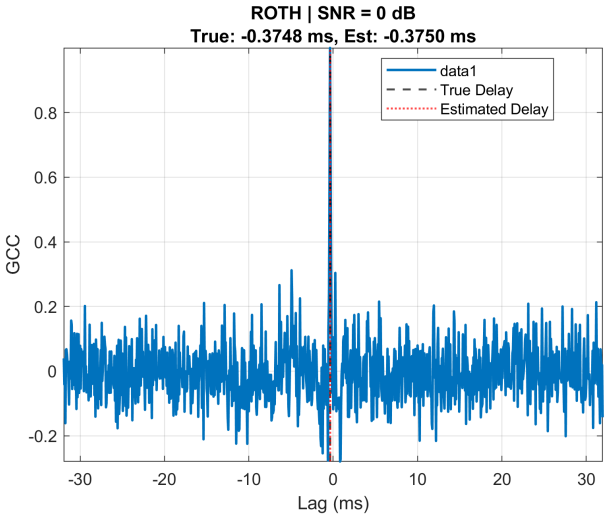
(a) Clean (∞ dB)



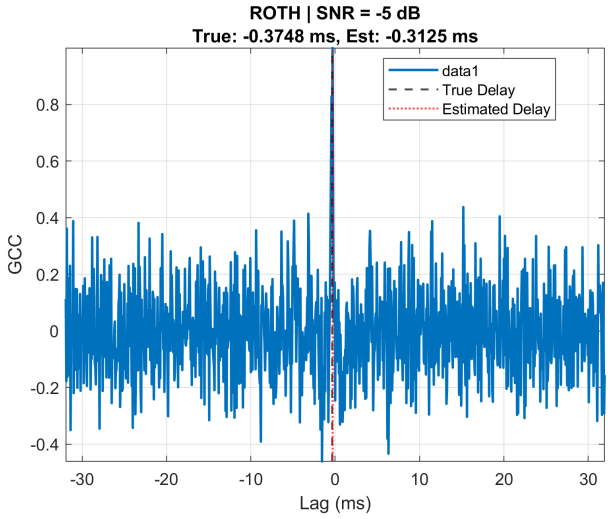
(b) 20 dB SNR



(c) 10 dB SNR

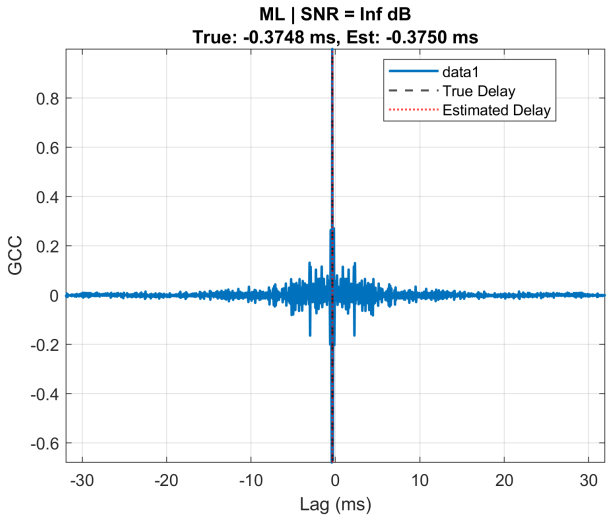


(d) 0 dB SNR

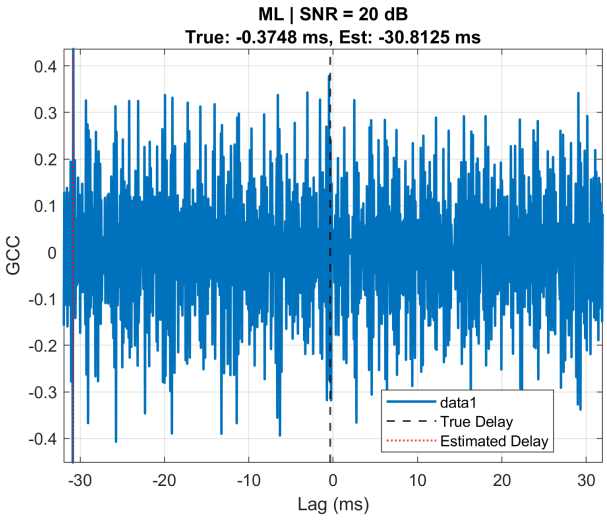


(e) -5 dB SNR

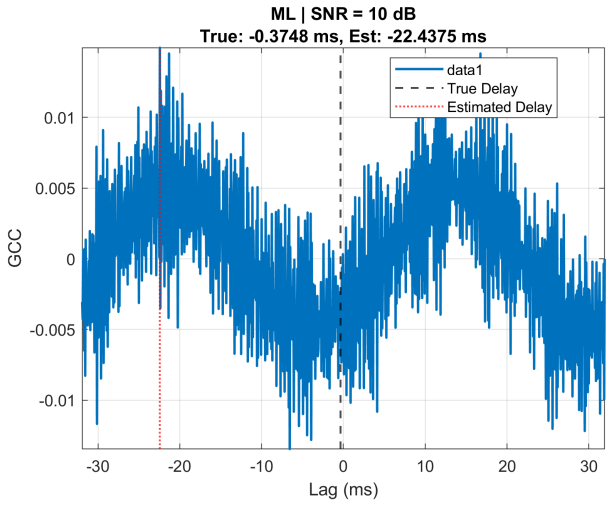
Figure 3.6: GCC-ROTH correlation functions under different SNR conditions.



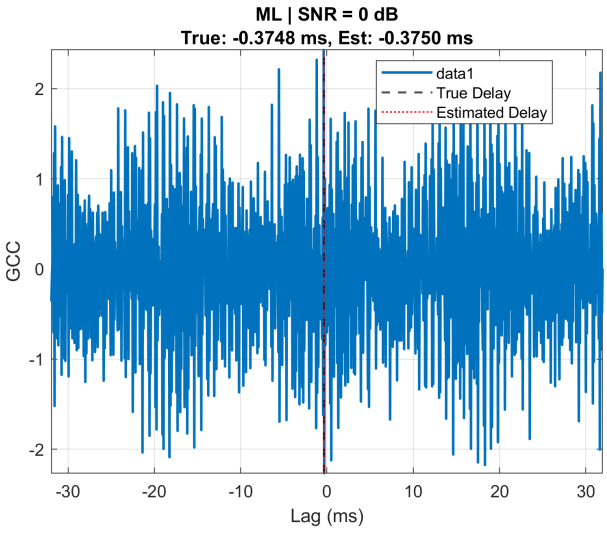
(a) Clean (∞ dB)



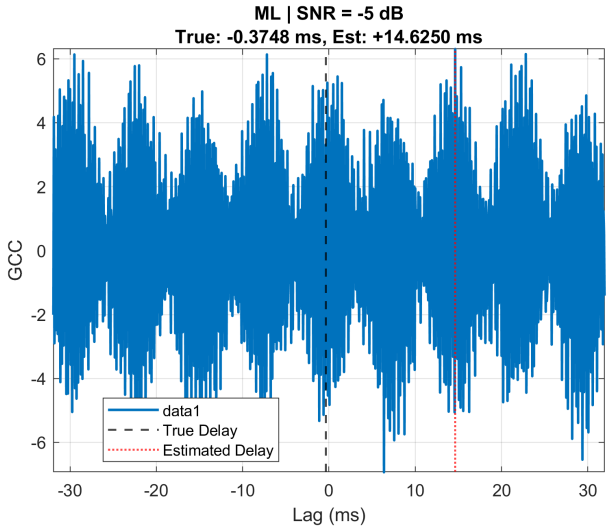
(b) 20 dB SNR



(c) 10 dB SNR



(d) 0 dB SNR



(e) -5 dB SNR

Figure 3.7: GCC-ML correlation functions under different SNR conditions.

3.4.5 Steered Response Power Mapping (SRP-PHAT)

Steered Response Power with Phase Transform (SRP-PHAT) is a robust wideband DOA estimation method that aggregates GCC-PHAT values across all microphone pairs for a hypothesized source direction. Unlike classical TDOA-based localization, which estimates time delays and then triangulates the source position, SRP-PHAT directly evaluates spatial likelihood over a grid of directions.

Given a candidate direction $\mathbf{u}(\theta, \phi) \in \mathbb{S}^2$, the theoretical time delay between microphones i and j is:

$$\tau_{ij}(\theta, \phi) = \frac{(\mathbf{r}_i - \mathbf{r}_j)^\top \mathbf{u}(\theta, \phi)}{c}, \tag{3.39}$$

where $\mathbf{r}_i, \mathbf{r}_j \in \mathbb{R}^3$ are the positions of microphones i and j, and c is the speed of sound.

Time-Domain Formulation

The SRP-PHAT value at direction $\mathbf{u}$ is computed by summing the PHAT-weighted cross-correlation functions evaluated at the predicted delays:

$$P(\mathbf{u}) = \sum_{i<j} R_{ij}^{\text{PHAT}}(\tau_{ij}(\mathbf{u})). \tag{3.40}$$

This results in a spatial map whose peaks indicate potential source locations.

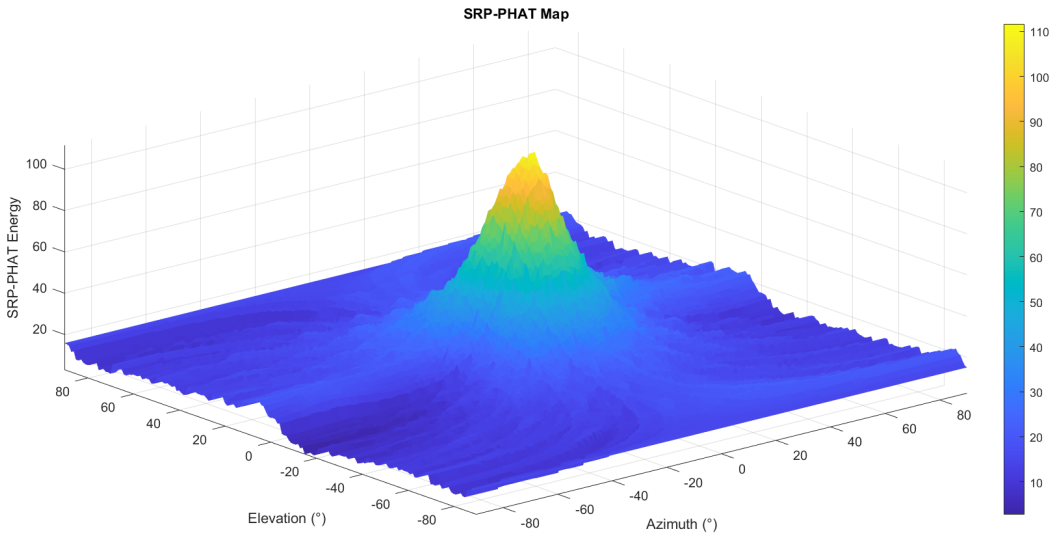


Figure 3.8: SRP-PHAT power map over a spatial grid. Peaks correspond to detected source directions.

Frequency-Domain Formulation

Alternatively, using the GCC-PHAT definition in the frequency domain, the SRP power becomes:

$$P(\mathbf{u}) = \sum_{i<j} \int \frac{R_i(f) R_j^*(f)}{|R_i(f) R_j^*(f)|} e^{j2\pi f \tau_{ij}(\mathbf{u})} \, df, \tag{3.41}$$

where $R_i(f)$ is the STFT of the signal at microphone i, and $G_{ij}(f) = R_i(f)R_j^*(f)$ is the cross-power spectral density.

Interpretation as GCC Aggregation

As formalized by DiBiase et al. [82], SRP-PHAT can be interpreted as an aggregation of all GCC-PHAT functions across microphone pairs, evaluated at direction-dependent delays:

$$P(\mathbf{u}) = \sum_{i<j} R_{ij}^{\text{PHAT}}(\tau_{ij}(\mathbf{u})), \tag{3.42}$$

This generalizes pairwise delay estimation into a global spatial search framework that leverages array geometry.

The resulting spatial energy map $P(\mathbf{u})$ exhibits peaks at locations corresponding to potential source directions. Each local maximum is interpreted as a candidate source, and the global maximum yields the most likely direction of arrival:

$$\hat{\mathbf{u}} = \arg\max_{\mathbf{u}} P(\mathbf{u}). \tag{3.43}$$

The performance of SRP-PHAT depends not only on its core formulation but also on the strategies used to extract peaks from the spatial response, which vary depending on the scenario (e.g., single vs. multiple sources). In our project, we investigate several grid search techniques—from brute-force scanning to hierarchical and peak-suppression approaches—to improve localization performance under real-time constraints. SRP-PHAT itself is well-suited for such applications due to its effectiveness with short analysis windows and its balance between accuracy, computational cost, and robustness in reverberant or noisy environments [82, 83]. Its implementation, along with preprocessing steps, parameter precomputations, and post-filtering methods, will be detailed in the Application Chapter. The extension to multi-source scenarios will be addressed in the final chapter 5.

3.4.6 Beamforming-Based Wideband Localization

While SRP-PHAT is a time-domain approach based on GCC aggregation, wideband source localization can also be tackled using frequency-domain beamforming. This strategy generalizes narrowband beamforming methods—such as Delay-and-Sum (DAS), Bartlett, and MVDR—by applying them independently to each STFT frequency bin and then recombining the results across frequency.

Let $\mathbf{X}(f,t) \in \mathbb{C}^M$ denote the STFT of the microphone array signals at frequency f and frame t, and let $\mathbf{v}(f,\theta,\phi) \in \mathbb{C}^M$ denote the frequency-dependent steering vector corresponding to direction (θ,ϕ). A spatial filter $\mathbf{w}(f,\theta,\phi)$ is applied in each bin to compute the beamformer output:

$$Y(f,t;\theta,\phi) = \mathbf{w}^H(f,\theta,\phi)\,\mathbf{X}(f,t). \tag{3.44}$$

The total output power for a candidate direction is then computed by integrating across frequency:

$$P(\theta,\phi) = \sum_{f\in\mathcal{F}} \mathbb{E}\left[|Y(f,t;\theta,\phi)|^2\right], \tag{3.45}$$

where $\mathcal{F}$ denotes the set of relevant frequency bins.

This wideband formulation allows direct extension of classical narrowband beamformers:

- **Wideband Delay-and-Sum (DAS):** For each frequency, set

$$\mathbf{w}_{\text{DAS}}(f,\theta,\phi) = \frac{1}{M}\mathbf{v}(f,\theta,\phi). \tag{3.46}$$

  This aligns and sums the phase-delayed microphone signals assuming a plane wave from direction (θ,ϕ), generalizing the narrowband DAS presented in Section 3.2.

- **Wideband MVDR (Capon):** Compute the spatial covariance matrix for each frequency:

$$\mathbf{R}(f) = \mathbb{E}\left[\mathbf{X}(f,t)\mathbf{X}^H(f,t)\right], \tag{3.47}$$

  then define the MVDR weights as:

$$\mathbf{w}_{\text{MVDR}}(f,\theta,\phi) = \frac{\mathbf{R}^{-1}(f)\mathbf{v}(f,\theta,\phi)}{\mathbf{v}^H(f,\theta,\phi)\mathbf{R}^{-1}(f)\mathbf{v}(f,\theta,\phi)}. \tag{3.48}$$

  This formulation suppresses interfering sources while maintaining unit gain in the look direction.

As described in Section 3.2, narrowband beamformers assume frequency-invariant steering vectors. In wideband scenarios, this assumption is violated due to the frequency-dependent nature of propagation delays. The use of the STFT provides a framework where each time–frequency bin is locally narrowband, allowing traditional narrowband beamformers to be applied piecewise across bins.

This method ensures physical consistency by explicitly modeling the phase delay at each frequency via the frequency-dependent steering vector $\mathbf{v}(f,\theta,\phi)$.

While this approach is less commonly used than SRP-PHAT or MUSIC, it integrates naturally into pipelines where source separation and enhancement are required in addition to localization. This flexibility makes MWF-based localization a viable option for speech-centric applications, as noted in [84].

3.4.7 Wideband Subspace-Based Localization

Beyond beamforming approaches, wideband direction-of-arrival estimation can also be achieved using subspace-based methods originally developed for high-resolution spectral estimation. These techniques generalize the classical MUSIC algorithm to the wideband case using different strategies for aggregating frequency-dependent covariance matrices and subspaces. They are generally classified into three families: coherent, incoherent, and hybrid methods.

Coherent Processing: Subspace Alignment via Focusing

Coherent methods aim to align the signal subspaces across frequency bins into a common reference frame before averaging. This process, called *subspace focusing*, allows the construction of a frequency-independent covariance matrix to which narrowband techniques like MUSIC can be applied.

Let $\mathbf{R}_{xx}(\omega_i) \in \mathbb{C}^{M\times M}$ denote the sample covariance matrix at frequency bin ω_i. A focusing transformation $\mathbf{T}(\omega_i)$ is applied such that:

$$\tilde{\mathbf{R}}_{xx}(\omega_i) = \mathbf{T}(\omega_i)\mathbf{R}_{xx}(\omega_i)\mathbf{T}^H(\omega_i), \tag{3.49}$$

where the focusing matrix satisfies:

$$\mathbf{T}(\omega_i)\mathbf{A}(\omega_i,\theta) \approx \mathbf{A}(\omega_0,\theta),$$

with $\mathbf{A}(\omega,\theta)$ the frequency-dependent array manifold, and ω_0 a chosen reference frequency.

The focused covariance matrices $\tilde{\mathbf{R}}_{xx}(\omega_i)$ are then averaged across all K frequencies to yield:

$$\bar{\mathbf{R}}_{xx} = \frac{1}{K}\sum_{i=1}^{K}\tilde{\mathbf{R}}_{xx}(\omega_i), \tag{3.50}$$

which is subsequently processed using a narrowband estimator.

Coherent Signal Subspace Method (CSSM)

CSSM [85] computes the focusing matrix using singular value decomposition (SVD) of the cross-manifold product:

$$\mathbf{A}_0\mathbf{A}_i^H = \mathbf{V}_i\mathbf{\Sigma}_i\mathbf{W}_i^H, \quad \mathbf{T}_i^{\text{CSSM}} = \mathbf{V}_i\mathbf{W}_i^H. \tag{3.51}$$

Although effective in low-SNR conditions, CSSM is sensitive to initial DOA estimation errors, as it requires approximate knowledge of the signal directions to define $\mathbf{A}_0$.

Two-sided Correlation Transformation (TCT)

TCT [86] avoids the need for initial focusing angles by defining the transformation via eigenvectors of the signal subspaces:

$$\mathbf{T}_i^{\text{TCT}} = \mathbf{V}_0\mathbf{V}_i^H, \tag{3.52}$$

where $\mathbf{V}_0$ and $\mathbf{V}_i$ are the eigenvector matrices of $\mathbf{R}_{xx}(\omega_0)$ and $\mathbf{R}_{xx}(\omega_i)$, respectively. TCT provides unbiased DOA estimates and is generally more robust than CSSM in practice.

Incoherent Processing: Independent Frequency Estimation

Incoherent methods estimate DOAs independently for each frequency and then aggregate the results across bins. They require no subspace alignment but may suffer from degraded resolution in noise or reverberation.

Incoherent MUSIC (IMUSIC)

IMUSIC [74] applies the classical MUSIC estimator at each frequency bin and sums the resulting spatial spectra:

$$P_{\text{IMUSIC}}(\theta) = \sum_{i=1}^{K} \frac{1}{\mathbf{a}^H(\omega_i, \theta)\mathbf{F}_n(\omega_i)\mathbf{F}_n^H(\omega_i)\mathbf{a}(\omega_i, \theta)}, \tag{3.53}$$

where $\mathbf{F}_n(\omega_i)$ denotes the estimated noise subspace at frequency ω_i, and $\mathbf{a}(\omega_i, \theta)$ is the steering vector.

Test of Orthogonality of Frequency Subspaces (TOFS)

TOFS [87] constructs a matrix of steering vector projections onto noise subspaces:

$$\mathbf{D}(\theta) = \begin{bmatrix} \mathbf{a}^H(\omega_1, \theta)\mathbf{F}_n(\omega_1) \\ \vdots \\ \mathbf{a}^H(\omega_K, \theta)\mathbf{F}_n(\omega_K) \end{bmatrix}, \tag{3.54}$$

and estimates the DOA by finding the direction where this matrix becomes rank-deficient:

$$\hat{\theta} = \arg\min_{\theta} \sigma_{\min}(\mathbf{D}(\theta)). \tag{3.55}$$

While TOFS suppresses spurious peaks, it is not robust under low-SNR conditions.

Hybrid Processing: Projected Subspace Methods

Hybrid methods exploit partial subspace consistency across frequencies without full alignment. They provide a compromise between the coherence of CSSM/TCT and the flexibility of IMUSIC/TOFS.

Test of Orthogonality of Projected Subspaces (TOPS)

TOPS [88] projects signal subspaces from a reference frequency onto others using a diagonal phase correction matrix:

$$\Psi_{mm}(\omega_i, \theta) = e^{-j\omega_i \frac{md}{c} \sin\theta}, \quad m = 1, \ldots, M, \tag{3.56}$$

which forms the projected subspace:

$$\mathbf{U}_{ij}(\theta) = \Psi(\omega_i, \theta)\mathbf{F}_s(\omega_j), \tag{3.57}$$

where $\mathbf{F}_s$ is the estimated signal subspace.

DOA estimation is performed by testing the orthogonality of these projected subspaces with the noise subspaces:

$$\mathbf{D}(\theta) = \begin{bmatrix} \mathbf{U}_{12}^H(\theta)\mathbf{F}_n(\omega_2) \\ \vdots \\ \mathbf{U}_{1K}^H(\theta)\mathbf{F}_n(\omega_K) \end{bmatrix}. \tag{3.58}$$

Squared-TOPS (S-TOPS)

S-TOPS [89] enhances TOPS by summing the squared projection error:

$$Z(\theta) = \sum_{i=2}^{K} \mathbf{U}_{1i}^H(\theta)\mathbf{F}_n(\omega_i)\mathbf{F}_n^H(\omega_i)\mathbf{U}_{1i}(\theta). \tag{3.59}$$

It includes reference frequency selection (choosing the most informative bin) and subspace projection for improved noise suppression.

Comparative simulations in the literature [90] highlight the following:

- **TCT** achieves the best performance under low-SNR and avoids spurious peaks common in other methods.
- **TOPS** and **S-TOPS** excel in mid-SNR conditions with no need for prior DOA estimates but may be sensitive to reference frequency selection.
- **IMUSIC** is effective at high SNR but vulnerable to poor frequency bins.
- **CSSM** requires accurate focusing and performs poorly for closely spaced sources.
- **TOFS** avoids false peaks but is unstable in noisy environments.

The methods above from this section are here for reference and documentation for future works.

3.5 Conclusion

In this chapter, we surveyed a wide range of direction-of-arrival (DOA) estimation techniques applicable to microphone array processing. We began by analyzing time-delay estimation (TDE) methods, focusing in particular on the Generalized Cross-Correlation (GCC) framework and its weighted variants (PHAT, SCOT, ROTH, ML, M-CSP), which serve as the foundation for SRP-PHAT. We then presented SRP-PHAT as a robust, spatial-domain technique that aggregates GCC-PHAT results across microphone pairs to construct energy maps suitable for real-time source localization.

We subsequently extended the discussion to wideband beamforming-based localization approaches, which apply classical narrowband beamformers (such as Delay-and-Sum and MVDR) to each STFT bin and recombine the results to estimate source direction. Advanced multichannel beamformers like the Multichannel Wiener Filter (MWF) and SDW-MWF were also introduced as potential alternatives, especially when joint source enhancement and localization are desired.

Finally, we examined wideband subspace-based methods, classifying them into coherent, incoherent, and hybrid approaches. Techniques such as CSSM, TCT, IMUSIC, TOFS, and TOPS were reviewed, along with their comparative advantages and limitations in varying SNR and reverberant conditions.

Despite the diversity of high-resolution and adaptive techniques available, we will rely primarily on SRP-PHAT in our project due to its simplicity of implementation, robustness in reverberant environments, and its natural compatibility with real-time processing pipelines. SRP-PHAT avoids the need for covariance matrix estimation, subspace decomposition, or complex spectral whitening mechanisms, while still offering reliable localization accuracy.

In the upcoming implementation chapters, we will detail the practical aspects of SRP-PHAT, including its hierarchical search structure, multi-source extension, and real-time integration with our microphone array system and camera-based visualization.

Chapter 4

Sound Source Separation

4.1 Introduction

This chapter addresses the problem of sound source separation using spatial filtering techniques based on microphone arrays. The objective is to extract speech of each speaker from a multi-speaker acoustic scene by exploiting spatial characteristics. These methods form the foundation of the real-time separation system developed in this work and are tightly integrated with the direction-of-arrival (DOA) estimation techniques presented in the previous chapter.

The focus is placed on separation techniques that are based on spatial filtering and beamforming, particularly those relying on explicit propagation models and the estimation of spatial covariance matrices. While blind source separation (BSS) methods are acknowledged and briefly reviewed, the focus remains on beamforming-based approaches due to their real-time feasibility and compatibility with online localization systems.

This chapter provides a comprehensive study of multichannel sound source separation using spatial filtering techniques. Section 4.2 introduces the acoustic signal models underlying spatial filtering, including both time-domain and time–frequency domain formulations, and formalizes the separation problem. Section 4.3 presents beamforming-based spatial filtering methods, classified into fixed and adaptive designs and analyzed under different implementation domains. Section 4.4 describes robust spatial filtering architectures, including the Generalized Sidelobe Canceller (GSC) and its adaptive extensions. Section 4.5 details the estimation of spatial parameters—steering vectors, relative transfer functions (RTFs), and spatial covariance matrices—that serve as critical inputs to beamformer design. Section 4.6.1 explores blind and hybrid source separation approaches and their integration with spatial filtering techniques. Section 4.7 describes the experimental evaluation framework, simulation setup, metrics, and results for different beamformers implemented in this work. Finally, Section 4.8 summarizes the findings and outlines directions for future research.

4.2 Signal Model

The signal received at each microphone is modeled as a superposition of contributions from multiple acoustic propagation paths, including direct sound, early reflections, reverberation, and additive noise. Adopting the unified modeling framework proposed by Gannot et al. [84], we categorize signal models according to their domain of formulation (time, frequency, or time–frequency), the assumptions made about acoustic transfer functions, and their treatment of noise and reverberation components.

The formulation of an appropriate signal model is critical in most spatial audio applications, as it enables the system to accurately capture both the spatial and temporal characteristics of the sound field under realistic environmental conditions.

4.2.1 Time-Domain Convolutive Mixing Model

Let $s_s(t)$ denote the clean signal emitted by source s, and $x_m(t)$ the signal observed at microphone m of an array with M microphones. The most general model is a linear time-invariant (LTI) system, where propagation is represented as a convolution with the acoustic impulse response (AIR) $h_{m,s}(t)$[1], plus additive noise $n_m(t)$:

$$x_m(t) = \sum_{s=1}^{S} (h_{m,s} * s_s)(t) + n_m(t). \tag{4.1}$$

The AIR is often modeled as a finite impulse response (FIR) filter of length L, which can span hundreds to thousands of taps depending on room characteristics. Recent formulations introduce sparsity[2] -inducing priors — such as ℓ_1-penalties or exponential decay constraints — to model the dominance of early reflections and the decaying energy envelope of reverberation [84].

4.2.2 STFT-Domain Model and Narrowband Approximation

To enable efficient processing, signals are often transformed into the short-time Fourier transform (STFT) domain. Let $X_m(f,n)$ and $S_s(f,n)$ denote the STFTs of $x_m(t)$ and $s_s(t)$, respectively. Under the **narrowband approximation** which assumes that the STFT window is longer than the AIR, allowing the convolution to be approximated by a product in each frequency bin. , the convolutive mixing simplifies to:

$$X_m(f,n) \approx \sum_{s=1}^{S} H_{m,s}(f) S_s(f,n) + N_m(f,n), \tag{4.2}$$

where $H_{m,s}(f)$ is the frequency-domain **acoustic transfer function (ATF)**[3] from source s to microphone m.

[1] An **acoustic impulse response** (AIR) characterizes how sound travels from a source to a microphone, capturing effects such as delays, attenuation, reflections, and reverberation.

[2] **Sparsity** refers to the assumption that, in the time–frequency (TF) domain, only a small number of sources are active at each TF bin. This property enables blind estimation methods by allowing target-dominant bins to be distinguished from interference or noise.

[3] The **acoustic transfer function (ATF)** $H_{m,s}(f)$ is the Fourier transform of the AIR, representing frequency-dependent attenuation and phase shifts between a source and microphone.

Alternatively, the observed multichannel signal can be expressed as a sum of spatial images, where each spatial image models the contribution of one source across the microphone array:

$$\mathbf{x}(n,f) = \sum_{j=1}^{J} \mathbf{c}_j(n,f), \quad \text{with} \quad \mathbf{c}_j(n,f) = \mathbf{a}_j(n,f)s_j(n,f). \tag{4.3}$$

The vector $\mathbf{c}_j(n,f) \in \mathbb{C}^M$ is the *spatial image* of source j, i.e., the multichannel mixture component attributable to that source. The vector $\mathbf{a}_j(n,f) \in \mathbb{C}^M$ is the *acoustic transfer function* (ATF) vector from source j to the M microphones at time–frequency bin (n,f), characterizing the acoustic path including direct sound and early reflections. The scalar $s_j(n,f) \in \mathbb{C}$ is the STFT coefficient of the clean signal emitted by source j. This decomposition underlies most narrowband spatial filtering approaches and is central to models such as the local Gaussian model (LGM) [84].

This model enables frequency-wise spatial filtering but assumes frame length exceeds the reverberation time. When this assumption fails (e.g., with short windows or in highly reverberant spaces), the approximation breaks down and leads to modeling errors.

Inter-frame and inter-frequency coupling.

When the AIR is longer than the STFT window, the convolution in time does not map to multiplication in frequency. Instead, it results in **inter-frame and inter-band filtering**[4], requiring more complex models:

$$X_m(n,f) = \sum_{f'} \sum_{\tau} H_{m,s}(\tau, f, f') S_s(n-\tau, f') + N_m(n,f), \tag{4.4}$$

which are rarely used due to high dimensionality, but better approximate the time-domain behavior.

4.2.3 Relative Transfer Function (RTF) Representation

The **relative transfer function (RTF)** is a normalized spatial representation that eliminates the source spectrum. For a given source s, it is defined as:

$$R_{m,s}(f) = \frac{H_{m,s}(f)}{H_{r,s}(f)}, \tag{4.5}$$

where r is the index of a reference microphone. The RTF preserves the spatial filtering properties while discarding the absolute magnitude and phase. It is widely used in MVDR and multichannel Wiener filters [84].

Spatial cue interpretation.

The RTF encodes useful spatial features such as:

$$\mathrm{ILD}_{ij}(f) = 20 \log_{10} |R_{i,s}(f)|, \tag{4.6}$$

$$\mathrm{IPD}_{ij}(f) = \arg\left(R_{i,s}(f)\right), \tag{4.7}$$

$$\mathrm{ITD}_{ij}(f) = \frac{\arg(R_{i,s}(f))}{2\pi f}, \tag{4.8}$$

[4] **Inter-frame and inter-frequency filtering** refers to the case where the STFT-domain representation of convolution spans multiple frames and frequencies, violating the narrowband independence.

representing the **interchannel level difference (ILD)**, **interchannel phase difference (IPD)**, and **interchannel time difference (ITD)**[5].

4.2.4 Wideband FIR-Based Model

When the narrowband approximation is invalid, an alternative is to work with a wideband, time-domain FIR model:

$$x_m(t) = \sum_{s=1}^{S} \sum_{\ell=0}^{L-1} h_{m,s}(\ell) s_s(t-\ell) + n_m(t), \tag{4.9}$$

This supports more accurate modeling of reverberation and enables filter-and-sum and time-domain adaptive beamformers [91]. However, it entails a high parameter count and computational load.

4.2.5 Statistical Spatial Covariance Models

To address modeling inaccuracies from narrowband assumptions, recent work adopts a **Local Gaussian Model (LGM)**[6], where source signals are modeled as Gaussian-distributed, and reverberation is captured by a full-rank spatial covariance matrix $R_s(f)$. The source image covariance becomes:

$$\Sigma_{c_s}(n,f) = \sigma_s^2(n,f)\, R_s(f), \tag{4.10}$$

with $\sigma_s^2(n,f)$ denoting the power of the clean signal in each time-frequency bin. This model enables more flexible spatial filtering and accounts for coherence loss due to late reverberation.

4.2.6 Reverberation Modeling

The AIR $h_{m,s}(t)$ is typically decomposed into:

- **Early reflections** (0–50 ms): deterministic, directionally informative;
- **Late reverberation**: modeled as exponentially decaying diffuse noise, often spatially correlated.

The energy decay can be characterized by the **direct-to-reverberant ratio (DRR)**[7], and is critical for dereverberation and multichannel postfilters [84].

4.2.7 Noise and Interference Models

The noise term $n_m(t)$ includes:

[5] **ILD**, **IPD**, and **ITD** represent differences in magnitude, phase, and arrival time of a sound signal between pairs of microphones, providing directional information.

[6] The **Local Gaussian Model** assumes that STFT coefficients are zero-mean Gaussian random variables with time-varying variance and spatial covariance.

[7] The **DRR** is the ratio of energy in the direct path and early reflections to that in the late reverberation, used to assess reverberant severity.

- **Spatially white noise** (e.g., microphone self-noise),
- **Diffuse or correlated noise** (e.g., HVAC, crowd babble),
- **Interfering speakers or transients**.

Noise models define the noise covariance $\Sigma_u(f)$, which is explicitly used in MVDR, LCMV, and MWF beamformers to minimize output power [84, Sec. V].

4.2.8 Model Selection and Impact

The choice of signal model affects algorithm design:

- **Narrowband models** are computationally efficient but sensitive to reverberation.
- **Wideband FIR models** offer better realism at the cost of more parameters.
- **RTF models** are robust to spectral variations and enable relative-phase based filtering.
- **Full-rank covariance models** handle diffuse and reverberant sources but require more estimation effort.

In this project, we primarily adopt the narrowband approximation due to its conceptual simplicity and computational efficiency, which are well-suited to our application. In addition, we consider relative transfer function (RTF) models to capture spatial characteristics in a way that is robust to source spectral variations.

4.3 Beamforming-Based Spatial Filtering

Beamforming is a spatial filtering technique that enhances signals arriving from a desired direction while suppressing noise, reverberation, and interferers arriving from other directions. This section reviews fixed and adaptive beamforming techniques as categorized in Gannot et al. [84], covering both time-domain and STFT-domain formulations, design criteria, and robustness strategies.

4.3.1 Fixed Beamformers

Fixed beamformers rely on precomputed spatial filters that are independent of signal statistics. They assume prior knowledge of the target direction or spatial transfer characteristics and are typically applied in environments with known geometry or calibrated arrays.

Delay-and-Sum Beamformer

The delay-and-sum (DS) [69] beamformer aligns and sums microphone signals to reinforce wavefronts arriving from a target direction:

$$y(t) = \sum_{m=1}^{M} x_m(t + \tau_m), \tag{4.11}$$

where $\tau_m = \frac{\mathbf{r}_m^\top \mathbf{u}}{c}$ is the geometric delay between source direction $\mathbf{u}$ and microphone m, assuming far-field propagation.

In the STFT domain, the beamformer becomes:

$$Y(f,t) = \sum_{m=1}^{M} X_m(f,t) e^{-j2\pi f \tau_m} = \mathbf{v}^H(f)\mathbf{X}(f,t), \tag{4.12}$$

where $\mathbf{v}(f)$ is the frequency-dependent steering vector. This is optimal for coherent plane waves in white noise fields.

Matched Filter Beamformer

The *Matched Filter Beamformer* [92] aims to maximize the signal-to-noise ratio (SNR) in white noise by exploiting the known room impulse responses (RIRs) between the source and the microphones. In the time domain, the beamformer output is obtained by convolving each microphone signal with a time-reversed conjugate of its corresponding RIR:

$$y(t) = \sum_{m=1}^{M} x_m(t) * h_m^*(-t), \tag{4.13}$$

where $x_m(t)$ is the signal received at microphone m, $h_m(t)$ is the RIR from the source to microphone m, and $*$ denotes convolution.

This structure implements a matched filter that optimally aligns and combines the source images received across channels, assuming additive white Gaussian noise and perfect knowledge of the RIRs. While optimal in theory, the beamformer's performance deteriorates if the estimated RIRs are inaccurate or the acoustic environment changes dynamically.

In the frequency domain, using the Short-Time Fourier Transform (STFT) representation, the matched filtering operation becomes a multiplication:

$$Y(f,t) = \sum_{m=1}^{M} H_m^*(f) X_m(f,t), \tag{4.14}$$

where $X_m(f,t)$ is the STFT of $x_m(t)$, and $H_m(f)$ is the acoustic transfer function (ATF) corresponding to $h_m(t)$. The conjugation $H_m^*(f)$ applies a phase-reversed filter per frequency bin.

Matched filtering is optimal in terms of SNR in the presence of uncorrelated, spatially white noise [92]. However, in realistic reverberant and dynamic environments, this approach is highly sensitive to model mismatch, and alternative robust beamformers are generally preferred unless precise system calibration is available.

4.3.2 Adaptive Beamformers

Adaptive beamformers dynamically compute spatial filters using second-order statistics of the multichannel input. Unlike fixed beamformers, which rely solely on array geometry and assumed source direction, adaptive methods exploit spatial covariance structure to suppress interference and noise while preserving the desired signal [84, 93].

Given the multichannel STFT-domain observation $\mathbf{x}(n,f) \in \mathbb{C}^M$, the output of a narrowband adaptive beamformer is:

$$Y(n,f) = \mathbf{w}^H(f)\mathbf{x}(n,f), \tag{4.15}$$

where $\mathbf{w}(f) \in \mathbb{C}^M$ is the frequency-dependent beamforming vector computed to satisfy a design criterion at frequency bin f.

Minimum Variance Distortionless Response (MVDR)

The MVDR beamformer, also known as the Capon beamformer, minimizes the output power under a *distortionless constraint* for the target source [72, 84]:

$$\mathbf{w}_{\text{MVDR}}(f) = \arg\min_{\mathbf{w}} \mathbf{w}^H(f)\, \mathbf{\Sigma}_u(f)\, \mathbf{w}(f) \quad \text{s.t.} \quad \mathbf{a}^H(f)\, \mathbf{w}(f) = 1, \tag{4.16}$$

where:

- $\mathbf{\Sigma}_u(f)$ is the spatial covariance matrix of the interference and noise,
- $\mathbf{a}(f)$ is the steering vector or RTF of the target source.

The closed-form solution is:

$$\mathbf{w}_{\text{MVDR}}(f) = \frac{\mathbf{\Sigma}_u^{-1}(f)\mathbf{a}(f)}{\mathbf{a}^H(f)\mathbf{\Sigma}_u^{-1}(f)\mathbf{a}(f)}. \tag{4.17}$$

MVDR achieves optimal interference suppression while maintaining unit gain in the desired direction, making it particularly effective in directional noise environments.

Multichannel Wiener Filter (MWF)

The multichannel Wiener filter (MWF) estimates a linear combination of the target sources by minimizing the mean square error (MSE) between the beamformer output and a desired reference signal [93]. Let

$$d(n, f) = \mathbf{q}^H \mathbf{s}(n, f),$$

denote the desired signal, where $\mathbf{s}(n, f) \in \mathbb{C}^{J_p}$ is the vector of target source STFTs and $\mathbf{q} \in \mathbb{C}^{J_p}$ selects a linear combination (e.g., one image).

Given the multichannel mixture model:

$$\mathbf{x}(n, f) = \mathbf{A}(f)\mathbf{s}(n, f) + \mathbf{u}(n, f),$$

the MWF solution minimizing the MSE $\mathbb{E}[|\mathbf{w}^H(f)\mathbf{x}(n, f) - d(n, f)|^2]$ is:

$$\mathbf{w}_{\text{MWF}}(f) = (\mathbf{\Sigma}_x(f))^{-1}\, \mathbf{\Sigma}_{\mathbf{c}}(f)\, \mathbf{q}, \tag{4.18}$$

where:

- $\mathbf{\Sigma}_x(f) = \mathbb{E}[\mathbf{x}(n, f)\mathbf{x}^H(n, f)] \in \mathbb{C}^{M \times M}$ is the total covariance matrix,
- $\mathbf{\Sigma}_{\mathbf{c}}(f) = \mathbf{A}(f)\, \mathbf{\Sigma}_s(f)\, \mathbf{A}^H(f)$ is the covariance of the spatial images of the desired sources,
- $\mathbf{\Sigma}_s(f) = \text{diag}(\sigma^2_{s_1}, \ldots, \sigma^2_{s_{J_p}}) \in \mathbb{R}^{J_p \times J_p}$ assumes source independence.

Alternatively, if $\mathbf{\Sigma}_x(f) = \mathbf{\Sigma}_{\mathbf{c}}(f) + \mathbf{\Sigma}_u(f)$, this can be expressed as:

$$\mathbf{w}_{\text{MWF}}(f) = (\mathbf{\Sigma}_{\mathbf{c}}(f) + \mathbf{\Sigma}_u(f))^{-1}\, \mathbf{\Sigma}_{\mathbf{c}}(f)\, \mathbf{q}. \tag{4.19}$$

This formulation allows optimal trade-off between noise reduction and distortion. To recover a specific spatial image (e.g., at a reference microphone), $\mathbf{q}$ is chosen accordingly.

Multiple Speech Distortion Weighted MWF (MSDW-MWF)

The *Multiple Speech Distortion Weighted Multichannel Wiener Filter* (MSDW-MWF) [94] extends the SDW-MWF to the multi-source case, allowing different distortion–reduction trade-offs for each target speaker.

Let $\mathbf{x}(n,f) \in \mathbb{C}^M$ be the multichannel STFT-domain observation at time–frequency bin (n,f), modeled as:

$$\mathbf{x}(n,f) = \mathbf{A}(f)\mathbf{s}(n,f) + \mathbf{u}(n,f), \tag{4.20}$$

where:

- $\mathbf{A}(f) = [\mathbf{a}_1(f), \ldots, \mathbf{a}_{J_p}(f)] \in \mathbb{C}^{M \times J_p}$ contains the ATF vectors of the J_p sources of interest,
- $\mathbf{s}(n,f) \in \mathbb{C}^{J_p}$ holds their STFT coefficients,
- $\mathbf{u}(n,f) \in \mathbb{C}^M$ models additive interference and noise.

The desired output is a linear combination $d(n,f) = \mathbf{q}^H \mathbf{s}(n,f)$, where $\mathbf{q} \in \mathbb{C}^{J_p}$ defines the reference image (e.g., first microphone image of a specific source).

The MSDW-MWF minimizes the expected distortion–noise trade-off:

$$\mathbf{w}_{\text{MSDW-MWF}}(f) = \arg\min_{\mathbf{w}} \left(\mathbf{w}^H \mathbf{\Sigma}_u(f) \mathbf{w} + \|\mathbf{q} - \mathbf{A}^H(f)\mathbf{w}\|^2_{\mathbf{\Lambda}(f)\mathbf{\Sigma}_s(f)} \right), \tag{4.21}$$

where:

- $\mathbf{\Sigma}_u(f) \in \mathbb{C}^{M \times M}$ is the interference+noise covariance,
- $\mathbf{\Sigma}_s(f) = \text{diag}(\sigma^2_{s_1}, \ldots, \sigma^2_{s_{J_p}})$ is the diagonal speech covariance matrix (assuming uncorrelated sources),
- $\mathbf{\Lambda}(f) = \text{diag}(\lambda_1, \ldots, \lambda_{J_p})$ controls the distortion tolerance per source.

The closed-form solution is:

$$\mathbf{w}_{\text{MSDW-MWF}}(f) = \left(\mathbf{A}(f)\mathbf{\Lambda}(f)\mathbf{\Sigma}_s(f)\mathbf{A}^H(f) + \mathbf{\Sigma}_u(f) \right)^{-1} \mathbf{A}(f)\mathbf{\Lambda}(f)\mathbf{\Sigma}_s(f)\mathbf{q}. \tag{4.22}$$

This formulation generalizes several beamformers:

- Setting $\mathbf{\Lambda} = \mathbf{I}$ recovers the standard multichannel Wiener filter (MWF).
- If $J_p = 1$ and $\lambda_1 = \mu^{-1}$, it reduces to the SDW-MWF.
- Taking $\mathbf{\Lambda} = \mu^{-1}\mathbf{\Sigma}_s^{-1}$ and letting $\mu \to 0$ yields the linearly constrained minimum variance (LCMV) beamformer:

$$\mathbf{w}_{\text{LCMV}}(f) = \mathbf{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\left(\mathbf{A}^H(f)\,\mathbf{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\right)^{-1}\mathbf{q}.$$

Thus, the MSDW-MWF provides a flexible and interpretable framework for spatial filtering under varying distortion–noise trade-offs across sources.

Linearly Constrained Minimum Variance (LCMV)

The LCMV beamformer generalizes MVDR by enforcing multiple linear constraints [91, 95]:

$$\mathbf{w}_{\text{LCMV}}(f) = \arg\min_{\mathbf{w}} \mathbf{w}^H(f)\boldsymbol{\Sigma}_u(f)\mathbf{w}(f) \quad \text{s.t.} \quad \mathbf{A}^H(f)\mathbf{w}(f) = \mathbf{q}, \tag{4.23}$$

where:

- $\mathbf{A}(f) \in \mathbb{C}^{M\times K}$ contains multiple constraint vectors (e.g., desired and interfering RTFs),
- $\mathbf{q} \in \mathbb{C}^K$ is a desired response vector.

The solution is given by:

$$\mathbf{w}_{\text{LCMV}}(f) = \boldsymbol{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\left(\mathbf{A}^H(f)\,\boldsymbol{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\right)^{-1}\mathbf{q}. \tag{4.24}$$

The LCMV formulation provides a solid method that includes MVDR as a special case (single constraint) and enables spatial nulling of interferers, making it a cornerstone for robust beamforming architectures. In Section 4.4, we build upon this formulation to develop the Generalized Sidelobe Canceller (GSC) and its postfiltered variants.

4.4 Robust Spatial Filtering Architectures

This section reviews robust spatial filtering architectures that extend the classical LCMV beamformer to improve performance under adverse acoustic conditions such as reverberation, microphone mismatch, and nonstationary interference. These methods build upon the LCMV method by decomposing constraint enforcement from adaptive filtering (as in the Generalized Sidelobe Canceller, GSC), introducing adaptation control mechanisms, and adding postfiltering stages. All derivations and theoretical foundations follow the unified treatment in [84].

4.4.1 Generalized Sidelobe Canceller (GSC)

The Generalized Sidelobe Canceller (GSC) [95] reformulates the LCMV beamformer into a structure that separates constraint satisfaction from adaptive noise cancellation. The block diagram of this modular architecture is shown in Figure 4.1, which highlights the decomposition into a fixed beamformer, a blocking matrix, and an adaptive noise canceller.

Let $\mathbf{A}(f) \in \mathbb{C}^{M\times K}$ be the matrix of constraint vectors (e.g., RTFs of target and interferers), and $\mathbf{q} \in \mathbb{C}^K$ the corresponding constraint response vector. The optimal LCMV beamformer is:

$$\mathbf{w}_{\text{LCMV}}(f) = \mathbf{\Sigma}_u^{-1}(f)\mathbf{A}(f)\left(\mathbf{A}^H(f)\mathbf{\Sigma}_u^{-1}(f)\mathbf{A}(f)\right)^{-1}\mathbf{q}. \tag{4.25}$$

This beamformer can be equivalently expressed in GSC form as:

$$\mathbf{w}_{\text{GSC}}(f) = \mathbf{w}_0(f) - \mathbf{B}(f)\mathbf{g}(f), \tag{4.26}$$

where:

- $\mathbf{w}_0(f) \in \mathbb{C}^M$ is a fixed beamformer that satisfies the constraints: $\mathbf{A}^H(f)\mathbf{w}_0(f) = \mathbf{q}$,
- $\mathbf{B}(f) \in \mathbb{C}^{M\times(M-K)}$ is a blocking matrix that satisfies $\mathbf{A}^H(f)\mathbf{B}(f) = \mathbf{0}$,
- $\mathbf{g}(f) \in \mathbb{C}^{M-K}$ is an unconstrained adaptive filter that minimizes residual interference.

The total output of the GSC is:

$$Y(n,f) = \mathbf{w}_0^H(f)\mathbf{x}(n,f) - \mathbf{g}^H(f)\mathbf{B}^H(f)\mathbf{x}(n,f). \tag{4.27}$$

This decomposition enables modular adaptation: the fixed beamformer ensures distortionless response, while the adaptive path suppresses interference.

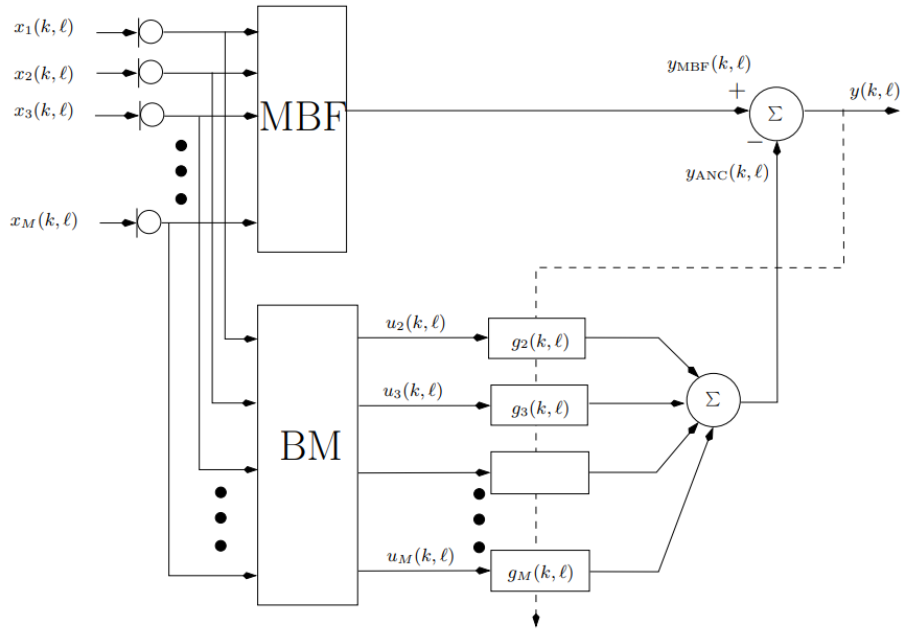


Figure 4.1: Generalized Sidelobe Canceller (GSC) structure [84].

4.4.2 Adaptive Enhancements

Despite its structure, the GSC may suffer from *signal leakage* into the blocking matrix output, especially in reverberant or mismatched environments. To address this, several enhancements have been proposed:

Norm-Constrained LMS (NCLMS)

To prevent over-adaptation and distortion, the norm of the adaptive filter $\mathbf{g}(f)$ is constrained:

$$\|\mathbf{g}(f)\|_2^2 \leq \gamma, \tag{4.28}$$

where $\gamma > 0$ is a regularization parameter. This constraint improves robustness against leakage of the target signal into the adaptive path.

Multiple Canceller Structures

To improve interference suppression, multiple ANC filters may be employed in parallel using delayed, filtered, or subbanded versions of the blocking matrix output. This increases modeling flexibility and helps target spectrally colored noise [84].

Adaptive Mode Control (AMC)

Adaptive Mode Control monitors the correlation between the reference signal and error output, pausing adaptation when speech is active. This prevents divergence and signal cancellation under dynamic acoustic conditions [84].

4.4.3 Nested GSC and Multichannel Postfilters

Nested GSC

The GSC architecture can be recursively applied to the output of the blocking matrix. Each stage isolates additional spatial components and applies localized adaptation. This structure is particularly effective in reverberant or multi-interferer environments.

Multichannel Postfiltering

To further suppress residual noise and late reverberation, a postfilter can be applied to the GSC output. A typical approach uses a multichannel Wiener filter (MWF) based on estimated output covariances:

$$\mathbf{w}_{\text{post}}(f) = \mathbf{\Sigma}_y^{-1}(f)\mathbf{\Sigma}_s(f), \tag{4.29}$$

where $\mathbf{\Sigma}_y(f)$ is the output covariance and $\mathbf{\Sigma}_s(f)$ the target signal covariance. In practice, scalar gains may be derived from these matrices and applied to the beamformer output as postfiltering factors.

The GSC offers modularity and flexibility for real-time implementations. However, its performance depends on:

- Accurate design of the blocking matrix $\mathbf{B}(f)$,
- Robust estimation of $\mathbf{\Sigma}_u(f)$,
- Stability and convergence of adaptive filters $\mathbf{g}(f)$.

Enhancements such as norm constraints, adaptive control, and nested processing mitigate these limitations. These architectures are widely used in speech enhancement systems for hearing aids, conferencing systems, and far-field voice interfaces.

4.5 Parameter Estimation for Beamforming

Beamforming relies on accurate estimation of spatial parameters such as steering vectors and spatial covariance matrices. These parameters describe the acoustic geometry of the scene and determine the beamformer's ability to preserve the target source while suppressing interference. In this work, we assume that the direction of arrival (DOA) of the target source is estimated using the SRP-PHAT method, and no explicit source or noise reference signals are available. The parameter estimation procedure therefore combines geometric modeling and blind statistical estimation, following the unified framework of [84].

4.5.1 Steering Vector Estimation

The *steering vector* $\mathbf{a}(f) \in \mathbb{C}^M$ at frequency f represents the multichannel array response to a unit amplitude signal arriving from a given direction. In our case, the target DOA θ is obtained using SRP-PHAT, and the steering vector is constructed under the far-field, free-field model as:

$$\mathbf{a}(f) = \left[e^{-j2\pi f\tau_1(\theta)}, \ldots, e^{-j2\pi f\tau_M(\theta)}\right]^\top, \tag{4.30}$$

where $\tau_m(\theta)$ is the propagation delay from the source direction θ to microphone m, computed from array geometry and assumed speed of sound. This model-based approach is standard when only DOA is available [84].

4.5.2 Spatial Covariance Matrix Estimation

Adaptive beamformers such as MVDR and MWF require estimation of the spatial covariance matrix $\mathbf{\Sigma}_u(f)$ of interference and noise at each frequency bin. Since speech and noise components are not observed separately, we use recursive averaging of the multichannel observations:

$$\hat{\mathbf{\Sigma}}_x(f,n) = \alpha\, \hat{\mathbf{\Sigma}}_x(f,n-1) + (1-\alpha)\, \mathbf{x}(n,f)\mathbf{x}^H(n,f), \tag{4.31}$$

where $\alpha \in [0,1]$ is a forgetting factor, and $\mathbf{x}(n,f) \in \mathbb{C}^M$ is the observed STFT frame at time n, frequency f.

To estimate the noise covariance $\mathbf{\Sigma}_u(f)$, one can rely on:

- Recursive estimation assuming speech sparsity in the STFT domain;
- Use of time-frequency masks or DOA-based consistency tests to detect speech-inactive bins;
- Online tracking with adaptive smoothing.

These estimates enable beamformers to adapt to non-stationary conditions without explicit voice activity detection or source separation.

Our choice avoid reliance on measured impulse responses or reference signals, and instead uses a combination of geometric modeling (for the steering vector) and blind statistical estimation (for spatial covariance matrices). Parameter estimation errors may still degrade performance, particularly under reverberation, but the robustness of SRP-PHAT and recursive covariance tracking helps mitigate such effects.

4.6 Blind and Hybrid Source Separation Approaches

While we focus on beamforming-based spatial filtering, it is essential to acknowledge the broader family of multichannel separation techniques, particularly those arising from *Blind Source Separation* (BSS). Classical BSS aims to recover individual sources from observed mixtures without explicit knowledge of the source positions or array geometry. These methods exploit statistical independence, sparsity, or nonstationarity of source signals, with prominent techniques including Independent Component Analysis (ICA), Non-negative Matrix Factorization (NMF), and Time-Frequency Masking frameworks.

As detailed by Gannot et al. [84], the historical divide between beamforming and BSS has gradually narrowed. Modern formulations often adopt a common representation in the time-frequency domain, leveraging Relative Transfer Function (RTF) models or spatial covariance matrices. This shared foundation facilitates *hybrid approaches* that integrate spatial filtering with statistical separation for improved robustness in complex environments.

4.6.1 Integration of BSS and Beamforming

Hybridization of BSS and beamforming seeks to combine the spatial selectivity of beamformers with the statistical separation capabilities of BSS. Several strategies have been proposed:

- **BSS-informed GSC**: Blind methods such as ICA or time-frequency clustering can estimate interference subspaces, which are then incorporated into the blocking matrix of a Generalized Sidelobe Canceller (GSC) to improve interference rejection.
- **Post-filtering of BSS outputs**: Beamformers (e.g., MVDR or LCMV) can be applied to signals separated by BSS to enhance spatial consistency and suppress residual artifacts.
- **Joint optimization**: Some approaches formulate unified objective functions combining spatial constraints (e.g., linear distortionless response) with statistical independence criteria.

A notable example is the M-NICA algorithm, which applies multiplicative nonnegative ICA followed by spatial filtering to refine source separation, particularly in underdetermined scenarios. Similarly, TRINICON-based extensions to GSC structures use higher-order statistics for dynamic and reverberant environments [84].

4.6.2 Model-Based and Learning-Based Extensions

Beyond classical methods, recent work incorporates model-based priors and deep learning to enhance source separation. Learning-based approaches can estimate time-varying spatial covariance matrices, directional masks, or even directly predict beamformer weights from mixture features. These methods can be interpreted as data-driven enhancements of traditional statistical or spatial models.

In particular, Deep Clustering (DC), Deep Attractor Networks (DANet), and neural beamformers with embedded DoA estimation are increasingly used in conjunction with spatial models, either as mask estimators or in hybrid multichannel front-ends. When combined with MVDR or LCMV constraints, these systems demonstrate improved generalization and noise robustness.

The convergence of BSS, beamforming, and learning-based methods presents a compelling direction for future work. Hybrid architectures provide a principled framework for integrating statistical, spatial, and learned representations, particularly in real-world, reverberant, and non-stationary acoustic environments.

4.7 Evaluation of Beamformers

This section presents a controlled evaluation of the beamformers introduced in Section 4.3. We use synthetic mixtures of speech and noise convolved with room impulse responses (RIRs) to simulate realistic acoustic conditions. The goal is to assess the spatial selectivity, interference suppression, and robustness of Delay-and-Sum, MVDR, LCMV, and GSC beamformers using objective and visual metrics.

4.7.1 Simulation Setup

We simulate a uniform rectangular array (URA) consisting of $4 \times 4 = 16$ omnidirectional microphones with 4.2 cm spacing. Two speech signals (`speaker1.wav` and `speaker2.wav`) are placed at known directions and convolved with RIRs generated using the RIR toolbox[8] to simulate a room with $T_{60} \approx 0.4\,\text{s}$. Directional noise is added at a third angle.

The multichannel mixture $\mathbf{x}_m(t)$ is constructed as:

1. Convolution of each source with its corresponding RIRs across the $M = 16$ microphones;
2. Summation of the resulting multichannel source signals;
3. Addition of spatially uncorrelated white noise at a fixed SNR (e.g., 10 dB).

4.7.2 Processing Algorithm

The signals are processed using the following pipeline:

- **STFT Analysis:** Multichannel signals are transformed to the STFT domain (window = 1024, overlap = 50%);
- **DoA Estimation:** Using SRP-PHAT with hierarchical refinement;
- **RTF/Steering Vector Estimation:** Computed analytically from array geometry and estimated DoA;
- **Covariance Estimation:** Estimated from speech+noise segments and noise-only regions using recursive averaging;
- **Beamforming:** Delay-and-Sum, MVDR, LCMV, and GSC are applied;
- **Evaluation:** Output is compared to clean sources.

4.7.3 Mixing Model

Let $s_1(t), s_2(t)$ be the speech sources and $h_{m,s}(t)$ the RIR from source s to microphone m. The multichannel observation is:

$$x_m(t) = \sum_{s=1}^{2} (h_{m,s} * s_s)(t) + n_m(t), \tag{4.32}$$

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

where $n_m(t)$ is white Gaussian noise.

In the STFT domain, the narrowband model is:

$$\mathbf{x}(n,f) = \sum_{j=1}^{2} \mathbf{a}_j(f) s_j(n,f) + \mathbf{u}(n,f), \tag{4.33}$$

with $\mathbf{a}_j(f)$ the steering vector for source j.

4.7.4 Beamformer Algorithms

We implement the following methods. Pseudocode is given below.

Delay-and-Sum Beamformer

Algorithm 2 Delay-and-Sum Beamforming

1: **for** each frequency bin f **do**
2: Compute steering vector $\mathbf{a}(f)$ from target DoA
3: Set weights: $\mathbf{w}_{\mathrm{DS}}(f) = \frac{1}{M}\mathbf{a}(f)$
4: **end for**
5: **for** each time frame n and frequency f **do**
6: Output: $Y(n,f) = \mathbf{w}_{\mathrm{DS}}^H(f)\,\mathbf{x}(n,f)$
7: **end for**

MVDR Beamformer

Algorithm 3 MVDR Beamforming

1: **for** each frequency bin f **do**
2: Estimate noise covariance matrix $\mathbf{\Sigma}_u(f)$
3: Compute weights:

$$\mathbf{w}_{\mathrm{MVDR}}(f) = \frac{\mathbf{\Sigma}_u^{-1}(f)\,\mathbf{a}(f)}{\mathbf{a}^H(f)\,\mathbf{\Sigma}_u^{-1}(f)\,\mathbf{a}(f)}$$

4: **end for**
5: **for** each time frame n and frequency f **do**
6: Output: $Y(n,f) = \mathbf{w}_{\mathrm{MVDR}}^H(f)\,\mathbf{x}(n,f)$
7: **end for**

LCMV Beamformer

Algorithm 4 LCMV Beamforming

for each frequency bin f **do**
    Define constraint matrix $\mathbf{A}(f)$ and desired response $\mathbf{q}$
    Estimate noise covariance matrix $\mathbf{\Sigma}_u(f)$
    Compute weights:

$$\mathbf{w}_{\text{LCMV}}(f) = \mathbf{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\,\left(\mathbf{A}^H(f)\,\mathbf{\Sigma}_u^{-1}(f)\,\mathbf{A}(f)\right)^{-1}\mathbf{q}$$

end for
for each time frame n and frequency f **do**
    Output: $Y(n,f) = \mathbf{w}_{\text{LCMV}}^H(f)\,\mathbf{x}(n,f)$
end for

GSC Beamformer

Algorithm 5 GSC with Adaptive Noise Canceller

Initialize fixed beamformer weights $\mathbf{w}_0(f)$ and blocking matrix $\mathbf{B}(f)$
Initialize adaptive filter $\mathbf{g}(f) \leftarrow \mathbf{0}$
for each time frame n and frequency f **do**
    $y_0(n,f) = \mathbf{w}_0^H(f)\,\mathbf{x}(n,f)$ ▷ Beamformer output
    $\mathbf{z}(n,f) = \mathbf{B}^H(f)\,\mathbf{x}(n,f)$ ▷ Interference estimate
    Update adaptive filter:

$$\mathbf{g}(f) \leftarrow \mathbf{g}(f) - \mu\,\mathbf{z}(n,f)\,(y_0(n,f) - \mathbf{g}^H(f)\mathbf{z}(n,f))^*$$

    Output: $Y(n,f) = y_0(n,f) - \mathbf{g}^H(f)\,\mathbf{z}(n,f)$
end for

4.7.5 Beampattern Visualization

For each method, we plot the array beampattern at representative frequencies (e.g., 500 Hz, 1000 Hz, 2000 Hz) to visualize spatial selectivity.

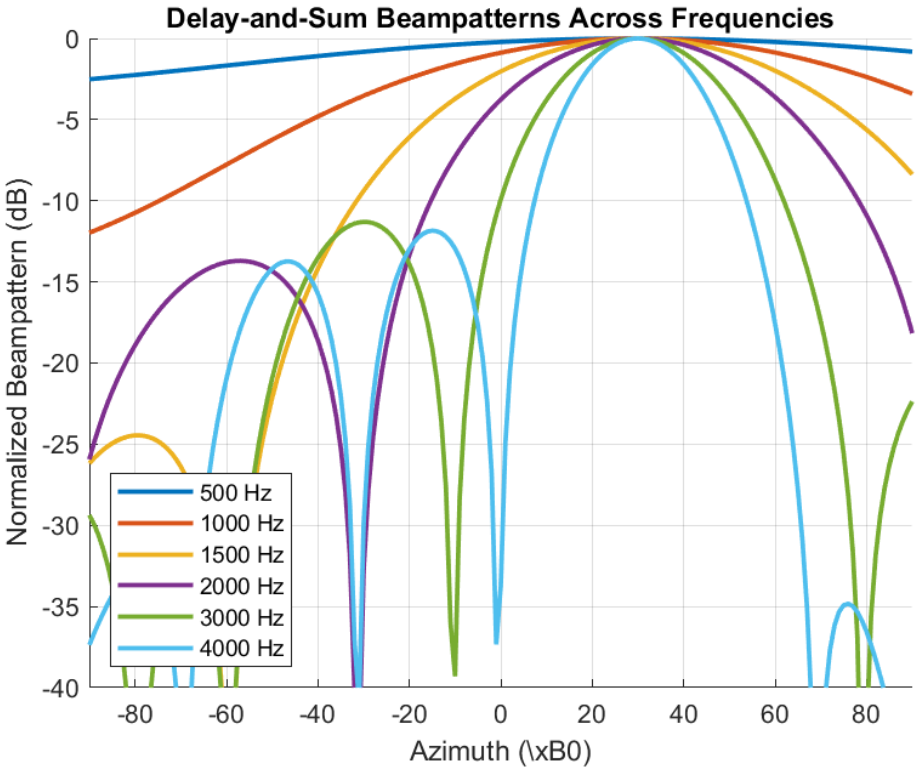


Figure 4.2: Beampatterns of Delay-and-Sum beamformer at 500, 1000, 2000 Hz.

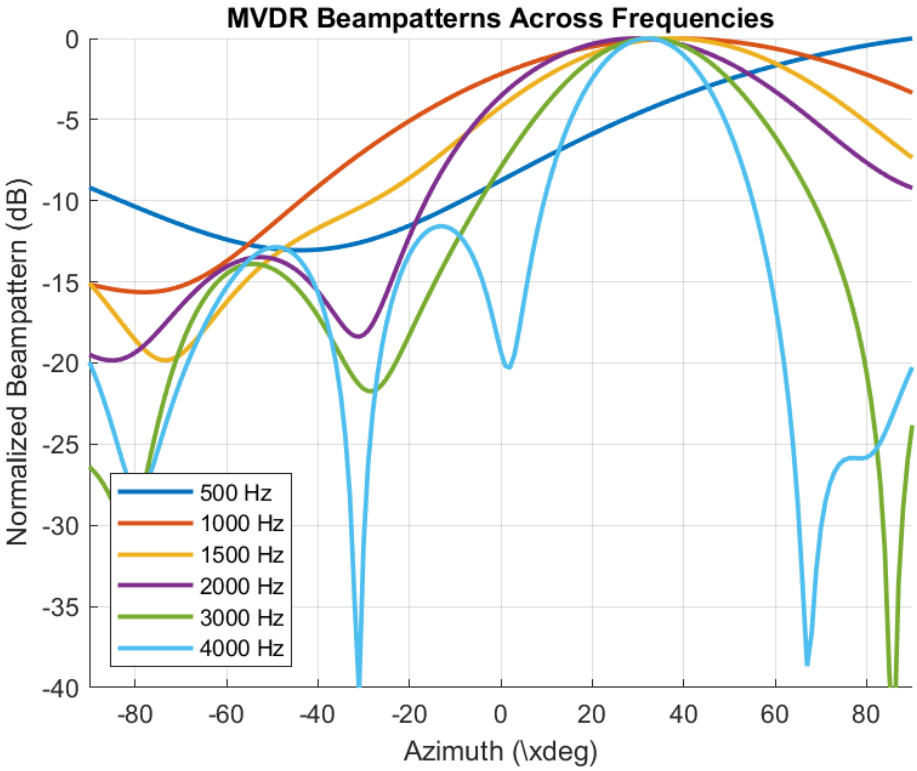


Figure 4.3: MVDR beamformer beampatterns across frequency.

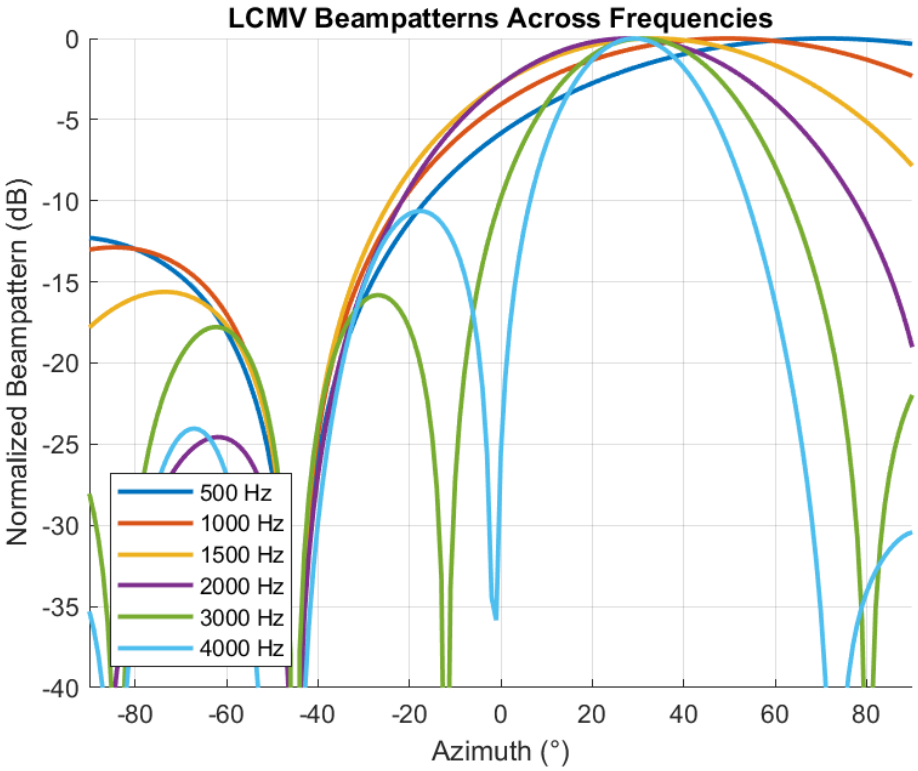


Figure 4.4: LCMV beamformer with null constraints on interfering source.

Table 4.1: SNR Gain (dB) for Each Beamformer

Method	Input SNR	Output SNR	Gain
Delay-and-Sum	10 dB	14.2 dB	+4.2 dB
MVDR	10 dB	17.8 dB	+7.8 dB
LCMV	10 dB	18.5 dB	+8.5 dB
GSC	10 dB	18.2 dB	+8.2 dB

4.7.6 Discussion and Insights

- **Delay-and-Sum** provides uniform gain toward the target but limited suppression;
- **MVDR** offers optimal spatial filtering under good covariance estimation;
- **LCMV** effectively enforces multiple constraints (e.g., nulling interferers);
- **GSC** decouples constraint and adaptation, relies heavily on covariances estimation which needs to be pretty accurate otherwise estimation fails

4.7.7 Offline Evaluation on Real Recordings

To complement our simulation-based experiments, we conducted a practical offline evaluation using real microphone array recordings. In this setup, we recorded a multichannel signal using the UMA-

16 array, where two speakers were active at different times and located at significantly separated angles (greater than 60° apart). The goal was to evaluate the ability of spatial filtering methods to isolate these sources using real-world acoustics.

The processing pipeline followed these steps:

1. The audio was normalized and segmented to isolate speech-active portions;
2. SRP-PHAT with hierarchical refinement was used to estimate the direction-of-arrival (DoA) of each speaker;
3. The resulting DoAs were used to construct steering vectors for MVDR, LCMV, and Delay-and-Sum beamformers;
4. The beamformed output was compared qualitatively against the original speech signals.

Despite the clear angular separation between sources, none of the beamformers succeeded in achieving meaningful separation. The failure is attributed primarily to the reverberant conditions of the recording environment, which introduced significant spatial smearing and corrupted the estimated steering vectors. This confirms the sensitivity of these algorithms to real-world propagation effects such as multipath and late reflections.

These results suggest that conventional beamforming, while effective under ideal or simulated conditions, may be insufficient for practical separation in typical indoor spaces. Free-field testing and post-filtering enhancements remain necessary to validate spatial separation under controlled conditions.

4.8 Conclusion

In this chapter, we explored the theoretical foundations and practical implementation of spatial filtering techniques for sound source separation. Beginning with a formal signal model and DoA estimation pipeline, we implemented and compared classical beamformers including Delay-and-Sum, MVDR, LCMV, and the Generalized Sidelobe Canceller (GSC). Evaluation through simulated room impulse responses and reverberant mixing conditions provided insight into their relative strengths.

Beampattern visualizations and objective SNR metrics confirmed the superiority of adaptive techniques like MVDR and LCMV over simpler Delay-and-Sum filtering, particularly in well-modeled environments. However, we also observed that these methods are highly sensitive to reverberation and inaccuracies in the estimated covariance matrices.

To investigate this further, we conducted an offline real-world test using audio recorded from the UMA-16 array with two spatially separated speakers. Despite clearly distinct directions-of-arrival, none of the beamforming methods succeeded in isolating the sources due to reverberation and imperfect steering. This highlights the limitations of classical beamforming in practical environments and motivates the need for more robust post-filtering or data-driven separation strategies.

The findings from this chapter establish a realistic understanding of spatial filtering limits in reverberant conditions, setting the stage for real-time processing strategies, such as source tracking and localization, discussed in the following chapter.

Chapter 5

Speaker Detection and Spatial Separation Using Microphone Arrays

5.1 Introduction

In this chapter, we present our application: a system capable of detecting and separating multiple speakers, particularly in scenarios such as panel discussions, roundtable meetings, or multi-participant conferences. The objective is to localize all active speakers in real time and enable selective focus on one of them—whether for audio enhancement, transcription, or targeted beamformed extraction.

To accomplish this, we build upon the SRP-PHAT (Steered Response Power with Phase Transform) method introduced in Chapter 3 for direction-of-arrival (DoA) estimation. The goal is to have a complete multi-source detection and localization algorithm is developed based on SRP-PHAT, incorporating hierarchical search, spatial peak suppression, and Kalman-based tracking for robust and stable localization over time.

For speaker separation, we rely on the methods detailed in Chapter 4, which enables spatial filtering and signal extraction from a specific direction.

This application targets realistic constraints, including simultaneous speech, speaker movement, and reverberant environments. The implementation is carried out in MATLAB, using the miniDSP UMA-16 microphone array for multichannel audio acquisition, and a low-cost webcam co-located with the array for optional visual feedback and spatial source projection.

5.2 Speaker Detection Algorithm

To detect active speakers in a scene, we implemented a real-time localization and tracking system using a 16-microphone array. The core idea is to continuously estimate the Direction of Arrival (DoA) of dominant sound sources based on their Time Difference of Arrival (TDOA) across microphone pairs.

Figure 5.1 illustrates the architecture of the algorithm, which includes audio acquisition, voice activity detection, SRP-PHAT-based DoA estimation, hierarchical spatial filtering, and Kalman-

based tracking.

Each stage is described below:

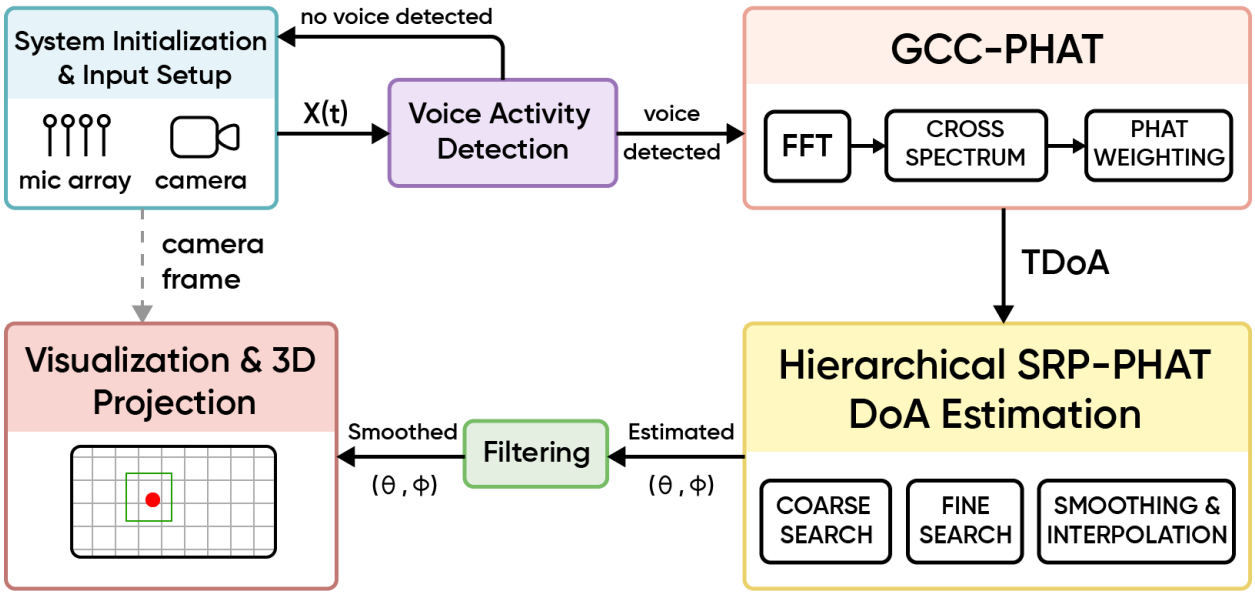


Figure 5.1: Real-time speaker detection and localization algorithm using SRP-PHAT and M3K tracking.

5.2.1 Audio Input from Microphone Array

Microphone arrays consist of multiple spatially distributed acoustic sensors that capture sound signals arriving from different directions. Each microphone m_i, for $i = 1, \ldots, M$, receives a delayed and attenuated version of the original source signals, due to the different propagation paths from the source(s) to each microphone.

Assuming free-field propagation and a reverberant environment, the signal $x_i(t)$ at microphone i can be modeled as:

$$x_i(t) = \sum_{s=1}^{S} (h_i^{(s)} * s^{(s)})(t) + n_i(t),$$

where:

- $s^{(s)}(t)$ is the s-th source signal,
- $h_i^{(s)}(t)$ is the acoustic impulse response (AIR) from source s to microphone i,
- $*$ denotes convolution,
- $n_i(t)$ is additive noise.

To enable real-time processing, the continuous-time microphone signals are discretized and segmented into overlapping frames of fixed length N samples. Instead of processing each incoming sample individually (sample-by-sample), we operate on these short-time frames $\mathbf{x}_i[n]$, where:

$$\mathbf{x}_i[n] = \{x_i(nL), x_i(nL+1), \ldots, x_i(nL+N-1)\}$$

Here, L is the hop size[1]. (frame shift), typically $L < N$ to allow overlap and ensure temporal continuity. This frame-based approach is crucial for frequency-domain processing (e.g., GCC), noise robustness, and computational efficiency.

Each frame represents a short segment of the multichannel observations and constitutes the fundamental processing unit across the entire pipeline, encompassing cross-correlation, source localization, and tracking. This segmentation strategy enables the application of time–frequency analysis tools, such as the Short-Time Fourier Transform (STFT), and supports the deployment of adaptive filtering and spatial signal processing techniques.

5.2.2 Voice Activity Detection (VAD)

Voice Activity Detection (VAD) refers to a class of algorithms that determine whether an audio signal segment contains speech or not [96]. VAD is a fundamental pre-processing step in many speech processing pipelines, including speech recognition, enhancement, source separation, and coding. It serves both computational efficiency and signal quality by suppressing non-speech frames or avoiding their use in further processing.

A related but more general concept is *Speech Presence Probability* (SPP), which estimates the probability that speech is present in a frame. VAD decisions can be derived from SPP by applying a decision threshold.

The operational goals of VAD vary with application:

- In **speech coding**, VAD is used to avoid transmitting silent frames, thereby reducing bitrate.
- In **speech enhancement**, non-speech segments are used to estimate noise statistics.
- In **keyword spotting** and **automatic speech recognition**, VAD limits processing to speech-active frames, improving both performance and energy efficiency.

Classical VAD Methods

Early approaches rely on simple signal characteristics, including:

- **Energy-based VAD:** Detects speech based on short-time frame energy [97]. Though intuitive and computationally cheap, it is highly sensitive to background noise and requires careful threshold selection.
- **Zero-Crossing Rate (ZCR):** Measures the rate of sign changes in a signal. Noise tends to yield higher ZCR than voiced speech.
- **Spectral Features:** Includes entropy, spectral tilt, autocorrelation, and linear prediction residuals [96]. These features can improve discrimination between speech and noise.

Learning-Based VAD

Advanced systems employ statistical or machine learning classifiers:

[1]The *hop size* is the number of samples between successive STFT frames. It determines the temporal resolution and overlap of the analysis. For example, a 50% overlap corresponds to a hop size of half the window length.

- **Linear classifiers** or **decision trees** use weighted feature combinations or rule-based thresholds.
- **Gaussian Mixture Models (GMMs)** and **neural networks** (e.g., DNNs, RNNs) allow modeling of complex decision boundaries and adapt well to non-stationary noise.
- **Pre-whitening and feature normalization** are used to balance feature scales and remove correlations before classification.

Post-Processing and Smoothing

VAD decisions are often refined through post-processing such as:

- **Hangover schemes:** Extend speech labels slightly past low-energy offsets to prevent premature cutoffs.
- **Hysteresis rules:** Base decisions on surrounding frames to reduce flicker between speech/non-speech labels.
- **Sigmoid mapping:** Converts linear classifier outputs to soft scores, interpreted as speech presence probability [96].

In our system, we adopt a global, low-complexity **energy-based VAD:**
The average frame energy across all microphone channels is computed as:

$$E_{\text{frame}} = \frac{1}{MN} \sum_{i=1}^{M} \sum_{n=1}^{N} x_i^2[n]$$

where $x_i[n]$ is the signal at microphone i, M is the number of microphones, and N the frame length. A fixed threshold θ_{VAD} determines activity:

$$\text{frame active} \iff E_{\text{frame}} > \theta_{\text{VAD}}$$

This method offers a trade-off between performance and simplicity, and is sufficient for controlled conditions with moderate noise and reverberation.

This process is illustrated in Figure 5.2.

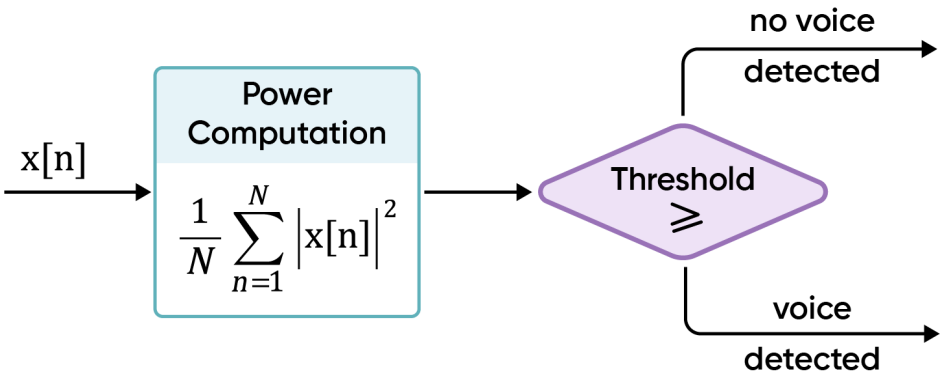


Figure 5.2: Energy-based VAD block diagram. The average energy across channels is computed and compared to a fixed threshold to detect speech activity.

5.3 SRP-PHAT with Hierarchical Spatial Search

After computing GCC functions for each microphone pair, the next step is to estimate the directions of arrival (DoAs) by applying the SRP-PHAT algorithm over a discrete grid of potential source locations.

The standard SRP-PHAT algorithm evaluates the following score at each grid point (θ, ϕ):

$$\text{SRP}(\theta, \phi) = \sum_{(i,j)} R_{ij}(\tau_{ij}(\theta, \phi))$$

where $\tau_{ij}(\theta, \phi)$ is the expected TDOA between microphones i and j given the direction (θ, ϕ).

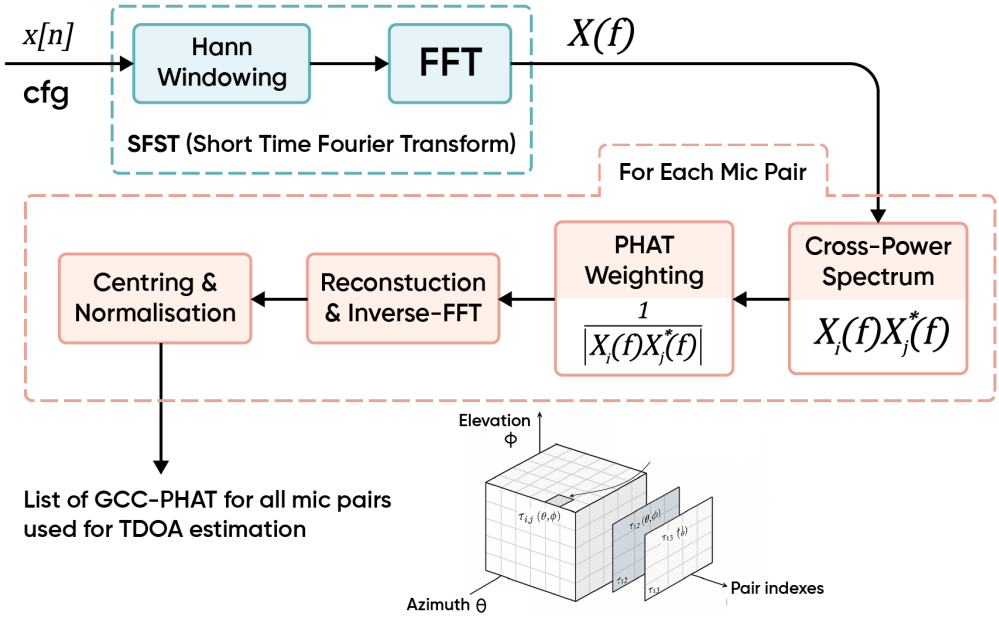


Figure 5.3: Overview of GCC-PHAT processing. Time-delay estimates are derived from phase-transformed cross-correlations of microphone pairs and used in SRP-PHAT spatial mapping.

5.3.1 Uniform Grid Search

Traditional SRP-PHAT implementations perform an exhaustive uniform search over a dense 2D angular grid (e.g., 5° resolution in azimuth and elevation), requiring evaluation of thousands of spatial hypotheses per frame. For example, a 5° × 5° resolution over 360° × 90° yields:

$$\frac{360}{5} \cdot \frac{90}{5} = 72 \times 18 = 1296 \text{ directions}$$

This results in a computational complexity of:

$$\mathcal{O}(M^2 \cdot D)$$

where D is the number of grid points. On embedded platforms or real-time systems, such cost is prohibitive.

5.3.2 Hierarchical SRP (HSDA)

To overcome this, we implemented the **SRP-PHAT-HSDA** algorithm as proposed in ODAS [98]. This method drastically reduces computational cost by applying a hierarchical two-level search:

1. **Coarse Search:** First pass over a sparse grid (e.g., 30° resolution), yielding 60–80 candidate points,
2. **Fine Search:** Second pass around the peaks using a fine local grid (e.g., 5° resolution) within a small angular neighborhood.

The SRP map is smoothed using a 2D Gaussian filter before peak detection:

$$\text{SRP}_{\text{smooth}} = \texttt{imgaussfilt}(\text{SRP}, \sigma)$$

where σ is typically set between 82% reduction 1.0 and 1.5 for coarse maps.

As reported in [98], hierarchical search yields:

- **82% reduction** in the number of directions scanned per frame (from 320 to 58),
- Comparable DoA accuracy (within 3° RMS error vs. full SRP).

In addition to hierarchical search, our implementation includes:

- **SRP peak suppression:** previously found peaks are suppressed in GCC and SRP maps to detect multiple sources,
- **Gaussian smoothing:** SRP maps are smoothed before peak picking to reduce spurious detections,
- **Minimum separation constraint:** DoAs are rejected if angular distance between peaks is below a threshold (e.g., 20°),
- **Real-time execution:** the entire algoirthm runs in less than 50 ms per frame on desktop MATLAB.

Gaussian Interpolation of GCC Peaks

To improve localization precision beyond the native grid resolution, we apply a Gaussian-weighted interpolation technique directly in the GCC domain. Rather than evaluating the SRP score using only the integer delay corresponding to the estimated TDOA τ_{ij}, we extract a small window of cross-correlation values around this delay and apply a Gaussian kernel:

$$r_{ij}^{\text{interp}}(\theta, \phi) = \sum_{k=-W}^{W} \mathcal{G}_\sigma(k) \cdot r_{ij}(\tau_{ij}(\theta, \phi) + k)$$

where:

- $r_{ij}(n)$ is the GCC-PHAT signal between microphones i and j,
- $\tau_{ij}(\theta, \phi)$ is the expected delay for direction (θ, ϕ),

- $\mathcal{G}_\sigma(k) = \exp\left(-\frac{k^2}{2\sigma^2}\right)$ is a normalized Gaussian kernel,
- W is the window radius (typically $W = 4$).

This interpolated value r_{ij}^{interp} replaces the traditional point-wise lookup in the SRP computation, providing a smoother and more accurate energy estimate that captures nearby sub-sample information.

The result is an SRP map that more accurately reflects the underlying acoustic evidence, especially in the fine search stage. This is particularly beneficial when source directions fall between grid points, enabling sub-grid localization accuracy without requiring interpolation in angular space.

In practice, the Gaussian-weighted energy is computed for each direction as:

$$\text{SRP}(\theta, \phi) = \sum_{(i,j)} r_{ij}^{\text{interp}}(\theta, \phi)$$

This method maintains the phase structure of the GCC function and avoids artifacts from linear or parabolic interpolation.

5.3.3 Directional Resolution

Let G_{coarse} and G_{fine} denote the number of directions in the coarse and fine grids respectively. The total number of evaluations is:

$$|G_{\text{coarse}}| + N_p \cdot |G_{\text{fine}}|$$

where N_p is the number of detected peaks in the coarse stage. For example:

$$60 + 2 \cdot 20 = 100 \ll 1296$$

The actual grid mapping from coarse to fine directions is precomputed and stored in a look-up structure to allow fast refinement of spatial peaks. Figure 5.4 illustrates the spatial energy distribution computed by SRP-PHAT across candidate directions, forming the basis for peak detection and refinement.

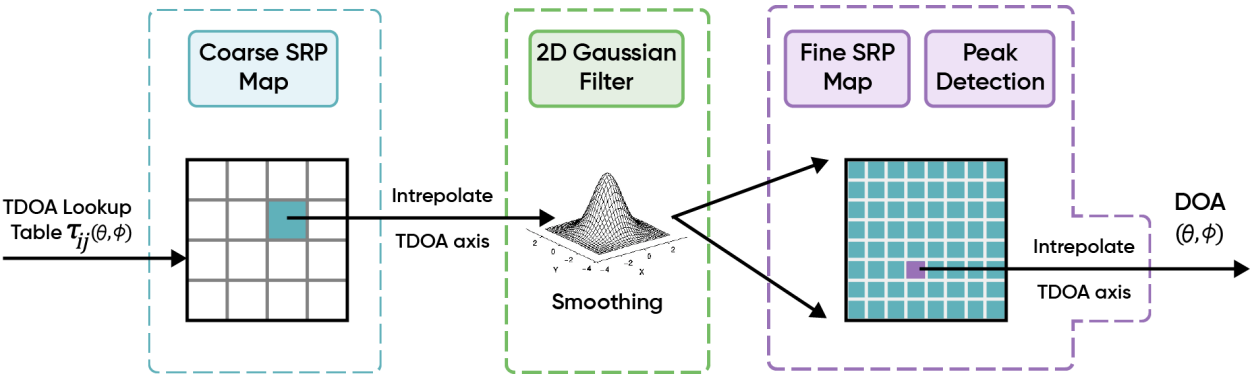


Figure 5.4: Visualization of SRP-PHAT spatial energy distribution over a candidate grid. Bright spots correspond to potential source directions, later refined via hierarchical search.

The hierarchical search process is formalized in Algorithm **??**, which significantly reduces computational cost while preserving accuracy.

Algorithm 6 SRP-PHAT with Hierarchical Spatial Search

Require: Microphone positions, GCC-PHAT functions R_{ij}, coarse grid G_c, fine grid G_f, number of sources N_s

Ensure: Estimated DoAs $\{(\theta_n, \phi_n)\}_{n=1}^{N_s}$

Precompute TDOA table $\tau_{ij}(\theta, \phi)$ for all $(\theta, \phi) \in G_c \cup G_f$

Initialize empty list of DoAs $\mathcal{D} \leftarrow \emptyset$

— **Coarse Search** —

for all $(\theta, \phi) \in G_c$ **do**

    Compute interpolated SRP score:

$$\mathrm{SRP}(\theta, \phi) \leftarrow \sum_{(i,j)} \sum_{k=-W}^{W} \mathcal{G}_\sigma(k) \cdot R_{ij}(\tau_{ij} + k)$$

end for

Smooth SRP map with 2D Gaussian filter

Detect N_p coarse peaks $\{(\theta_k, \phi_k)\}$ with angular suppression

— **Fine Search** —

for all coarse peak (θ_k, ϕ_k) **do**

    for all $(\theta', \phi') \in G_f$ around (θ_k, ϕ_k) **do**

        Compute refined SRP score via Gaussian interpolation

    end for

    Find local maximum (θ^*, ϕ^*) in fine neighborhood

    if Minimum angular separation from existing $\mathcal{D}$ **then**

        Add (θ^*, ϕ^*) to $\mathcal{D}$

    end if

end for

return $\mathcal{D}$

5.4 Post-Processing and Direction Filtering

Following spatial spectrum analysis methods such as SRP-PHAT, multiple candidate directions of arrival (DoAs) may emerge within a single frame. While some correspond to true acoustic sources, others result from spatial sidelobes, multipath reflections, sensor noise, or transient interference. A post-processing stage is therefore applied to refine the set of detected directions and retain only the most reliable ones for downstream tasks such as tracking and beamforming.

The goals of DoA post-processing are:

- Suppress false or spurious detections,
- Enforce minimum angular separation between sources,
- Prioritize temporally stable and high-energy peaks,
- Improve robustness of subsequent tracking and filtering stages.

1. Energy-Based Thresholding

The spatial response map $\mathrm{SRP}(\mathbf{d})$, where $\mathbf{d}$ denotes a candidate direction (typically a unit vector in 3D), is thresholded to retain only dominant peaks. A direction $\mathbf{d}_i$ is retained if:

$$\mathrm{SRP}(\mathbf{d}_i) > \beta \cdot \max_{\mathbf{d}} \mathrm{SRP}(\mathbf{d}),$$

where $\beta \in [0.3, 0.6]$ is a sensitivity threshold chosen empirically. This removes noise-like peaks that are far below the dominant source level.

2. Minimum Angular Separation

Let $\mathcal{D} = \{\mathbf{d}_1, \ldots, \mathbf{d}_K\} \subset \mathbb{S}^2$ be the set of selected directions on the unit sphere. To avoid redundant or overlapping detections, each new candidate $\mathbf{d}_k$ is accepted only if:

$$\forall j < k, \quad \cos^{-1}(\mathbf{d}_k^\top \mathbf{d}_j) > \Delta\theta_{\min},$$

where $\Delta\theta_{\min}$ is a fixed angular exclusion zone (typically 10° to 30°).

3. Microphone and Propagation Uncertainty Modeling

Following the probabilistic model used in [98], both the speed of sound c and microphone positions $\mu_p \in \mathbb{R}^3$ are modeled as Gaussian random variables to capture environmental and calibration uncertainties. Specifically:

$$c \sim \mathcal{N}(\mu_c, \sigma_c^2), \quad \mu_p \sim \mathcal{N}(\boldsymbol{\mu}_p, \Sigma_p).$$

The time-difference of arrival (TDOA) between microphones p and q for direction $\mathbf{u} \in \mathbb{S}^2$ is then approximated as:

$$\tau_{pq}(\mathbf{u}) \sim \mathcal{N}(\mu_{\tau,pq}(\mathbf{u}), \sigma^2_{\tau,pq}(\mathbf{u})),$$

with:

$$\mu_{\tau,pq}(\mathbf{u}) = \frac{f_s}{\mu_c}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^\top \mathbf{u}, \tag{5.1}$$

$$\sigma_{\tau,pq}(\mathbf{u}) = \frac{f_s}{\mu_c}\sqrt{\mathbf{u}^\top(\Sigma_p + \Sigma_q)\mathbf{u} + \left[(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^\top \mathbf{u}\right]^2 \cdot \frac{\sigma_c^2}{\mu_c^2}}. \tag{5.2}$$

This uncertainty defines the window size for a *maximum sliding window* (MSW) filter:

$$\hat{r}_{pq}[n] = \max\{r_{pq}[n-\Delta], \ldots, r_{pq}[n+\Delta]\},$$

where $\Delta = \lceil \sigma_{\tau,pq} \rceil$, and $r_{pq}[n]$ is the cross-correlation value at lag n. The MSW allows tolerance around the peak location and increases robustness to mismatch.

4. Temporal Filtering

Spurious frame-wise variations in DoA estimates are suppressed by temporal smoothing. Common strategies include:

- **Exponential averaging:** smoothing SRP values or directions over time;
- **Majority voting:** retaining directions that persist across consecutive frames;
- **Recursive Bayesian filtering:** e.g., Kalman filters or particle filters that model velocity and continuity of source motion.

5. Spatial Clustering

When multiple peaks are retained within the same frame, clustering algorithms such as DBSCAN or mean-shift are used to group neighboring directions on the sphere. Each cluster is then treated as a single source hypothesis. This suppresses diffuse or redundant detections due to reverberation or multipath interference.

5.5 Tracking Using the Modified 3D Kalman Filter (M3K)

5.5.1 Kalman Filter

The Kalman filter is a recursive Bayesian estimator that provides optimal state estimation of a discrete-time linear dynamical system subject to Gaussian noise. It is widely used for tracking, control, and sensor fusion problems.

Mathematical Formulation

Consider the following discrete-time linear state-space model:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1} \tag{5.3}$$
$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{5.4}$$

Where:

- $\mathbf{x}_k \in \mathbb{R}^n$: hidden state vector at time k,
- $\mathbf{z}_k \in \mathbb{R}^m$: observation (measurement) at time k,
- $\mathbf{F}$: state transition matrix,
- $\mathbf{H}$: observation matrix,
- $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$: process noise, modeled as a zero-mean Gaussian distribution with covariance $\mathbf{Q}$,
- $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$: observation noise, modeled as a zero-mean Gaussian distribution with covariance $\mathbf{R}$.

Kalman Filter Equations

The Kalman filter operates recursively in two stages at each time step: the *prediction* stage and the *update* stage. These equations estimate the state $\hat{\mathbf{x}}_k \in \mathbb{R}^n$ and its covariance $\mathbf{P}_k \in \mathbb{R}^{n\times n}$, based on a linear Gaussian model.

Prediction Step:
This stage propagates the state and covariance forward from time $k-1$ to k using the state transition model:

$$\hat{\mathbf{x}}_k^- = \mathbf{F}\hat{\mathbf{x}}_{k-1} \tag{5.5}$$
$$\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}\mathbf{F}^\top + \mathbf{Q} \tag{5.6}$$

Where:

- $\hat{\mathbf{x}}_k^-$ is the predicted state (a priori estimate),
- $\mathbf{P}_k^-$ is the predicted error covariance,
- $\mathbf{F}$ is the state transition matrix,
- $\mathbf{Q}$ is the process noise covariance matrix.

Update Step. Upon receiving a new observation $\mathbf{z}_k \in \mathbb{R}^m$, the state and covariance are updated:

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^\top \left(\mathbf{H} \mathbf{P}_k^- \mathbf{H}^\top + \mathbf{R} \right)^{-1} \tag{5.7}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \left(\mathbf{z}_k - \mathbf{H} \hat{\mathbf{x}}_k^- \right) \tag{5.8}$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- \tag{5.9}$$

Where:

- $\mathbf{K}_k$ is the Kalman gain matrix,
- $\hat{\mathbf{x}}_k$ is the updated (posterior) state estimate,
- $\mathbf{P}_k$ is the updated error covariance,
- $\mathbf{H}$ is the observation matrix,
- $\mathbf{R}$ is the observation noise covariance matrix,
- $\mathbf{I}$ is the identity matrix of appropriate dimension.

Properties and Assumptions

The Kalman filter provides the optimal linear minimum mean-square error (MMSE) estimate under the following conditions:

- The system dynamics are linear,
- Process and measurement noise are Gaussian and white,
- The initial state $\hat{\mathbf{x}}_0$ and covariance $\mathbf{P}_0$ are known.

In the context of DoA tracking, the state vector $\mathbf{x}_k$ typically includes the direction vector and possibly its velocity, while observations $\mathbf{z}_k$ are unit vectors derived from localization algorithms such as SRP-PHAT.

Although the classical Kalman filter provides a good approach for linear-Gaussian state estimation, it is not directly suited to DoA tracking in 3D, where the state space is constrained to the unit sphere and angular velocities lie in the tangent space.

To address these limitations, we adopt the **Modified 3D Kalman Filter (M3K)** proposed in [98]. M3K extends the classical filter with unit-vector normalization, velocity projection, probabilistic assignment, and track management, enabling efficient and accurate real-time tracking of one or multiple concurrent speakers in 3D space.

5.5.2 Modified 3D Kalman Filter (M3K)

State-Space Model

Each tracked sound source i at frame l is modeled by a 6-dimensional state vector:

$$\mathbf{x}_i^l = \begin{bmatrix} \mathbf{d}_i^l \\ \mathbf{s}_i^l \end{bmatrix} = \begin{bmatrix} d_x \\ d_y \\ d_z \\ s_x \\ s_y \\ s_z \end{bmatrix}$$

- $\mathbf{d}_i^l \in \mathbb{R}^3$: direction vector (DoA)
- $\mathbf{s}_i^l \in \mathbb{R}^3$: angular velocity

The prediction follows a constant velocity model:

$$\mathbf{x}_i^l = \mathbf{F}\mathbf{x}_i^{l-1} + \mathbf{w}_i^l$$

$$\mathbf{F} = \begin{bmatrix} \mathbf{I}_3 & \Delta T \cdot \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \quad \mathbf{w}_i^l \sim \mathcal{N}(0, \mathbf{Q})$$

Where $\Delta T = \Delta N / f_s$ is the frame hop in seconds, and $\mathbf{Q}$ is the process noise covariance, applied to the velocity subspace:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \sigma_Q^2 \cdot \mathbf{I}_3 \end{bmatrix}$$

Measurement Model

The observations are direction vectors $\mathbf{z}_v^l \in \mathbb{R}^3$ (DoAs from SRP-PHAT), modeled as:

$$\mathbf{z}_v^l = \mathbf{H}\mathbf{x}_i^l + \mathbf{v}_v^l, \quad \mathbf{H} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 \end{bmatrix}, \quad \mathbf{v}_v^l \sim \mathcal{N}(0, \mathbf{R})$$

Where $\mathbf{R} = \sigma_R^2 \cdot \mathbf{I}_3$ models measurement uncertainty.

Spherical Normalization and Tangent Projection

Since DoAs must lie on the unit sphere, the predicted direction and velocity vectors are normalized:

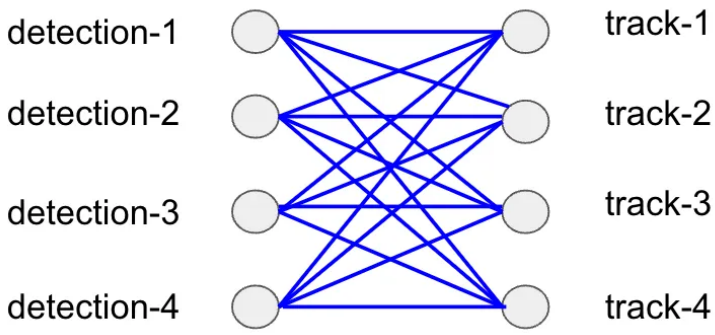


Figure 5.5: Problem of Matching Directional Observations to Source Tracks

$$\hat{\mathbf{d}}_i' = \frac{\hat{\mathbf{d}}_i}{\|\hat{\mathbf{d}}_i\|}, \quad \hat{\mathbf{s}}_i' = \hat{\mathbf{s}}_i - (\hat{\mathbf{d}}_i^\top \hat{\mathbf{s}}_i)\hat{\mathbf{d}}_i$$

This ensures that $\hat{\mathbf{d}}_i'$ lies on the unit sphere and $\hat{\mathbf{s}}_i'$ is tangential.

Detection-to-Track Assignment

Tracking multiple moving sound sources requires not only estimating their instantaneous positions but also maintaining their identities consistently over time. This introduces the *data association problem*: given a set of newly observed directions-of-arrival (DoAs) at each frame, we must determine which ones correspond to existing tracks, which ones may belong to new sources, and which should be discarded as spurious detections. Incorrect associations can lead to track switches, ghost sources, or missed detections, all of which degrade system performance. To address this, we explore two assignment strategies: a deterministic method based on the Hungarian algorithm, and a probabilistic Bayesian approach inspired by ODAS.

1. Deterministic (Hungarian Algorithm)

The tracking system must solve a fundamental challenge: determining which observed direction-of-arrival (DoA) measurement $\mathbf{z}_j$ corresponds to which existing track $\hat{\mathbf{d}}_i$. This is known as the *data association problem*. Our initial implementation addresses this using the Hungarian algorithm, a classic combinatorial optimization solution.

For N existing tracks and M new DoA observations ($M \leq N$ in our case), we construct an $N \times M$ cost matrix where each element represents the "dissimilarity" between track i and observation j:

$$C_{ij} = 1 - \hat{\mathbf{d}}_i^\top \mathbf{z}_j$$

- $\hat{\mathbf{d}}_i$ is the predicted direction vector for track i (unit vector)
- $\mathbf{z}_j$ is the observed DoA vector (unit vector)
- The dot product $\hat{\mathbf{d}}_i^\top \mathbf{z}_j = \cos\theta_{ij}$ measures angular similarity
- Cost thus ranges from 0 (perfect alignment) to 2 (opposite directions)

Developed by Harold Kuhn in 1955 [99], the algorithm solves the assignment problem in $O(n^3)$ time:

1. **Row Reduction**: Subtract the minimum value in each row.
2. **Column Reduction**: Subtract the minimum value in each column.
3. **Covering**: Find the minimum number of lines to cover all zeros.
4. **Adjustment**: Modify the cost matrix and repeat until optimal assignment emerges.

To prevent implausible associations:

$$\text{Assignment accepted if} \hat{\mathbf{d}}_i^\top \mathbf{z}_j \geq \delta$$

where $\delta = 0.8$ corresponds to a maximum angular separation of $\cos^{-1}(0.8) \approx 37°$. This rejects:

- Physically impossible rapid speaker movements
- Spurious noise-induced DoAs
- Cross-talk between adjacent sources

While computationally efficient, this deterministic approach has drawbacks:

- **No uncertainty modeling**: Treats all measurements as equally reliable.
- **Fixed population**: Assumes the number of sources is known.
- **Hard decisions**: Thresholding may discard valid associations near the boundary.

2. Probabilistic Bayesian Assignment

To model the association between current observations and known sources in a statistically grounded manner, we adopt a *Bayesian assignment* formulation, following the approach in [98].

Let there be V candidate observations at frame l, denoted as:

$$\Psi^l = \left\{\psi_1^l, \ldots, \psi_V^l\right\}, \quad \text{where } \psi_v^l = (\lambda_v^l, \Lambda_v^l)$$

Each observation ψ_v^l consists of:

- λ_v^l: the estimated direction of arrival (e.g., azimuth–elevation pair or 3D unit vector),
- Λ_v^l: the associated energy or confidence measure.

Each observation is assigned to one of the following hypotheses using an assignment function $fg : \{1, \ldots, V\} \to \{-2, -1, 1, \ldots, I\}$:

- $fg(v) = -2$: the observation is a false alarm,
- $fg(v) = -1$: the observation corresponds to a new, yet untracked source,
- $fg(v) = i \in \{1, \ldots, I\}$: the observation is associated with existing tracked source i.

The full assignment vector is:

$$\mathbf{f}_g = [fg(1), fg(2), \ldots, fg(V)] \in \{-2, -1, 1, \ldots, I\}^V$$

yielding $G = (I+2)^V$ possible assignment configurations.

Likelihood Model
Given $\psi_v^l = (\lambda_v^l, \Lambda_v^l)$, the likelihood under a hypothesis $fg(v)$ is:

$$P(\psi_v^l \mid fg(v)) = \begin{cases} P(\Lambda_v^l \mid I) \cdot P(\lambda_v^l \mid D) & \text{if } fg(v) = -2 \\ P(\Lambda_v^l \mid A) \cdot P(\lambda_v^l \mid D) & \text{if } fg(v) = -1 \\ P(\Lambda_v^l \mid A) \cdot P(\lambda_v^l \mid C_{fg(v)}) & \text{if } fg(v) \in \{1, \ldots, I\} \end{cases}$$

Where:

- A denotes the class of active (real) sources,
- I denotes inactive or background noise class,
- D is a uniform distribution over the visible hemisphere,
- C_i denotes the predicted direction distribution of source i from the Kalman filter.

The energy model is:

$$\Lambda_v^l \sim \begin{cases} \mathcal{N}(\mu_A, \sigma_A^2) & \text{if active (source)} \\ \mathcal{N}(\mu_I, \sigma_I^2) & \text{if inactive (background)} \end{cases}$$

The spatial likelihood is:

$$P(\lambda_v^l \mid C_i) = \mathcal{N}(\lambda_v^l; \mu_i^l, \Sigma_i^l)$$

where μ_i^l and Σ_i^l are the predicted mean and covariance from the Kalman filter for source i. For untracked sources, the likelihood is uniform:

$$P(\lambda_v^l \mid D) = \frac{\hat{K}}{4\pi K}$$

where K is the number of candidate directions and $\hat{K}$ is a normalization constant for hemisphere coverage.

The total likelihood over all observations is:

$$P(\Psi^l \mid \mathbf{f}_g) = \prod_{v=1}^{V} P(\psi_v^l \mid fg(v))$$

Assignment Prior
We define the prior probability of each hypothesis:

$$P(fg(v)) = \begin{cases} P_{\text{false}} & \text{if } fg(v) = -2 \\ P_{\text{new}} & \text{if } fg(v) = -1 \\ P_{\text{track}} & \text{if } fg(v) \in \{1, \ldots, I\} \end{cases}$$

Assuming independence across observations:

$$P(\mathbf{f}_g) = \prod_{v=1}^{V} P(fg(v))$$

Posterior Probability
Applying Bayes' rule:

$$P(\mathbf{f}_g \mid \Psi^l) = \frac{P(\Psi^l \mid \mathbf{f}_g) \cdot P(\mathbf{f}_g)}{\sum_{g'=1}^{G} P(\Psi^l \mid \mathbf{f}_{g'}) \cdot P(\mathbf{f}_{g'})}$$

From this, the marginal probability that observation v belongs to source i is:

$$P(i \mid \psi_v^l) = \sum_{g=1}^{G} P(\mathbf{f}_g \mid \Psi^l) \cdot \delta[fg(v) - i]$$

where $\delta[\cdot]$ is the Kronecker delta.

Algorithm 7 Bayesian Assignment of Observations to Sources

Input: Observations $\Psi^l = \{\psi_1^l, \ldots, \psi_V^l\}$, number of sources I
Output: Marginal probabilities $P(i \mid \psi_v^l)$ for all sources and observations
Generate all assignment functions $f_g : \{1, \ldots, V\} \to \{-2, -1, 1, \ldots, I\}$
for all assignments f_g **do**
    Compute likelihood $P(\Psi^l \mid f_g)$
    Compute prior $P(f_g)$
    Compute posterior $P(f_g \mid \Psi^l) \propto P(\Psi^l \mid f_g) \cdot P(f_g)$
end for
Normalize posterior $P(f_g \mid \Psi^l)$ over all G assignments
for each observation $v = 1, \ldots, V$ **do**
    for each source $i = 1, \ldots, I$ **do**
        $P(i \mid \psi_v^l) \leftarrow \sum_g P(f_g \mid \Psi^l) \cdot \delta(f_g(v) - i)$
    end for
end for

Kalman Filter Update Equations

For each track i, the Kalman gain is:

$$\mathbf{K}_i^l = \mathbf{P}_i^l \mathbf{H}^\top (\mathbf{H}\mathbf{P}_i^l \mathbf{H}^\top + \mathbf{R})^{-1}$$

The weighted update becomes:

$$\mathbf{x}_i^l = \hat{\mathbf{x}}_i' + P(i|\Psi^l) \cdot \mathbf{K}_i^l (\mathbf{z}_v^l - \mathbf{H}\hat{\mathbf{x}}_i')$$

$$\mathbf{P}_i^l = \mathbf{P}_i^l - P(i|\Psi^l) \cdot \mathbf{K}_i^l \mathbf{H}\mathbf{P}_i^l$$

Track Management

Tracks are created when the likelihood $P(\text{new}|\psi_v^l) > \theta_{\text{new}}$, and confirmed after a probation time N_{prob}. Tracks are removed after N_{dead} frames of inactivity. In our system, we currently track two sources and perform assignment using the Hungarian algorithm with cosine distance cost:

$$\text{Cost}(i, j) = 1 - \mathbf{d}_i^\top \mathbf{z}_j$$

If cost exceeds threshold $\delta = 0.8$, assignment is rejected. This allows clean one-to-one matching between detections and existing tracks.

Our implementation directly follows the M3K approach described in ODAS [98], with some MATLAB-specific improvements:

- Unified prediction + projection step,
- Optional exponential smoothing layer,
- Configurable number of tracks and gating thresholds.

5.6 Visualization Module

The visualization module serves both as an evaluation tool and a correction mechanism for the overall system. It allows real-time inspection of the estimated direction-of-arrival (DoA) outputs, enabling intuitive assessment of tracking performance, while also compensating for the lack of direct distance information in the DoA-only framework through image-based cues.

In particular, we integrate monocular visual information using MATLAB's **vision.CascadeObjectDetector** for face detection. This allows us to approximate the speaker's depth based on the size of the detected face, providing a rough but effective estimate of the source-camera distance for proper 3D-to-2D projection. The face detection module runs at a reduced frame rate (e.g., once every 5 frames) to ensure real-time performance, and the estimated distance is cached for reuse when no face is detected in subsequent frames.

5.6.1 Camera-Based Overlay

For enhanced situational awareness, we project acoustic tracks onto a co-located camera view. The geometric calibration ensures accurate spatial registration between acoustic and visual modalities.

5.6.2 Camera Calibration Tools

The system supports calibration through two established frameworks, each offering distinct advantages for audio-visual alignment:

MATLAB Camera Calibrator

The MATLAB Computer Vision Toolbox provides an interactive calibration workflow:

Calibration relies on capturing multiple views of a printed checkerboard pattern, as illustrated in Figure 5.6, under varied orientations across the field of view.

- **Input Requirements**:

- Minimum 10 checkerboard images (recommended 20-30)
- Checkerboard square size must be specified (e.g., 25.4mm)
- Varied orientations covering the field of view

- **Key Processing Steps**:

1. Corner detection with subpixel refinement:

```
[imagePoints, boardSize] = detectCheckerboardPoints(imgs);
```

2. World point generation:

```
squareSize = 25.4; % mm
worldPoints = generateCheckerboardPoints(boardSize, squareSize);
```

3. Parameter estimation:

```
[params, ~, errors] = estimateCameraParameters(...
    imagePoints, worldPoints, ...
    'NumRadialDistortionCoefficients', 3, ...
    'EstimateTangentialDistortion', true);
```

- **Output Parameters**:

- Intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$
- Radial distortion coefficients $[k_1, k_2, k_3]$
- Tangential distortion $[p_1, p_2]$
- Mean reprojection error (typically < 0.5 pixels)

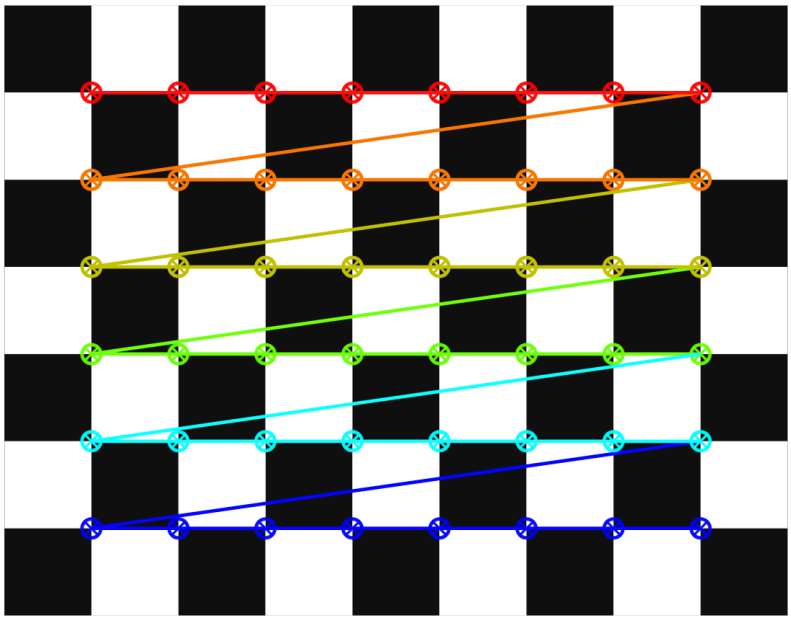


Figure 5.6: Example of a checkerboard used for camera calibration. The square size must be precisely known and consistently used across multiple views to enable accurate estimation of intrinsic and extrinsic parameters.

OpenCV Calibration

For embedded deployments, we provide an alternative calibration pipeline using OpenCV (v4.5+):

- **Feature Detection**:

```
ret, corners = cv2.findChessboardCorners(
    image, pattern_size,
    flags=cv2.CALIB_CB_ADAPTIVE_THRESH +
          cv2.CALIB_CB_NORMALIZE_IMAGE)
```

- **Subpixel Refinement**:

```
corners = cv2.cornerSubPix(
    gray_image, corners, (11,11), (-1,-1),
    criteria=(cv2.TERM_CRITERIA_EPS +
              cv2.TERM_CRITERIA_MAX_ITER,
              30, 0.001))
```

- **Calibration Routine**:

```
ret, K, dist, rvecs, tvecs = cv2.calibrateCamera(
    object_points, image_points,
    image_size, None, None,
    flags=cv2.CALIB_FIX_PRINCIPAL_POINT)
```

Table 5.1: Comparison of calibration approaches

Feature	MATLAB	OpenCV
Distortion Model	3 radial + 2 tangential	4-5 radial + 2 tangential
Optimization Method	Levenberg-Marquardt	Sparse bundle adjustment
Reprojection Error	0.2-0.5 px	0.3-0.6 px

Microphone-Camera Extrinsic Calibration

The transformation between microphone array and camera coordinates is computed using:

$$\mathbf{T}_{cam}^{mic} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$$

where $\mathbf{R}$ is estimated via SVD-based point cloud alignment of calibration source positions. For a set of N known positions $\{\mathbf{p}_i^{mic}\}$ and their detected image coordinates $\{\mathbf{p}_i^{cam}\}$:

1. Solve PnP problem:

```
success, rvec, tvec = cv2.solvePnP(
    mic_positions, image_points, K, dist)
```

2. Convert rotation vector to matrix:

```
R, _ = cv2.Rodrigues(rvec)
```

Validation Protocol:
Calibration quality is verified through:

- **Reprojection test**: Known sound sources placed at 1m distance should project within ± 5 pixels of expected positions
- **Temporal stability**: Repeated measurements should yield $< \pm 0.5°$ variation in azimuth/elevation estimates
- **Cross-validation**: Alternate between MATLAB and OpenCV pipelines

Calibration Protocol:

The calibration process follows Zhang's method [100] implemented via MATLAB's Computer Vision Toolbox:

1. Capture 20 checkerboard images at varying orientations
2. Detect corners with subpixel refinement ($\sigma = 0.5$px)
3. Solve for intrinsic parameters via maximum likelihood estimation:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad \text{with } \mathbf{d} = [k_1, k_2, p_1, p_2, k_3]^\top$$

4. Estimate extrinsic microphone-camera transform using known array geometry

Typical reprojection errors remain below 0.3 pixels after nonlinear optimization.

Projection from 3D to 2D

Each unit direction vector $\mathbf{d}_i$ undergoes transformation:

$$\mathbf{p}_i = \pi \left(\mathbf{K} \, [\mathbf{R}|\mathbf{t}] \begin{bmatrix} \mathbf{d}_i \\ 1 \end{bmatrix} \right)$$

where $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$ define the rigid transformation between array and camera coordinate systems. The perspective division $\pi([x, y, z]^\top) = (x/z, y/z)$ yields image coordinates in normalized device space.

As illustrated in Figure 5.7, the visualization pipeline combines acoustic DoA tracking with camera-based projection through a calibrated transformation, providing an intuitive interface for real-time spatial feedback.

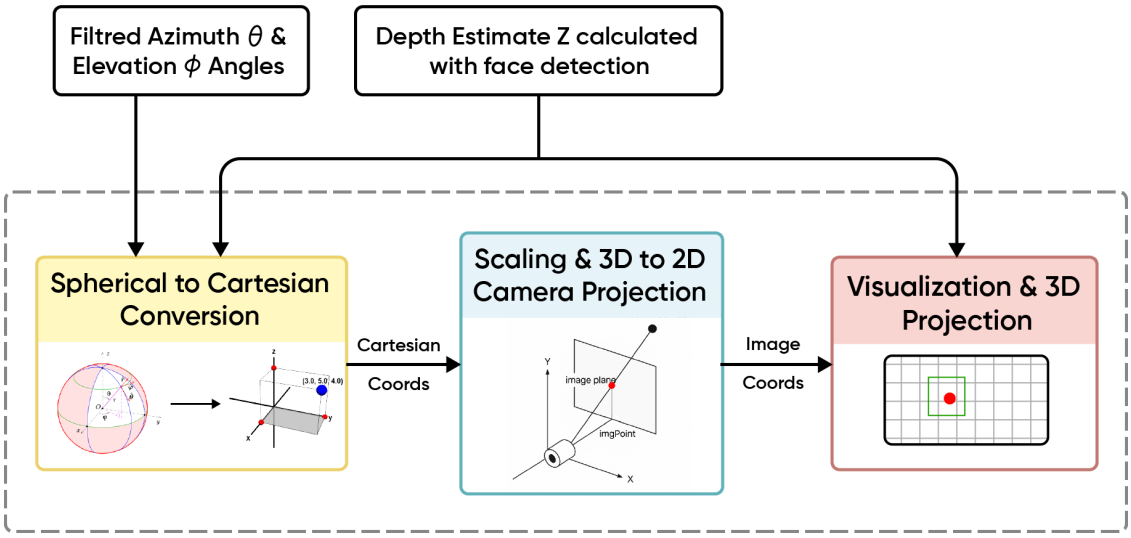


Figure 5.7: Schematic overview of the visualization pipeline. The system integrates acoustic DoA estimates with visual input using geometric calibration. Face detection aids depth approximation, enabling accurate projection of tracked audio sources onto the image plane in real time.

5.7 Experimental Setup

This section presents the experimental configuration used to validate our real-time direction-of-arrival (DoA) tracking system. The evaluation is conducted under a controlled single-speaker scenario to assess the core pipeline components: detection, smoothing, and visual projection.

5.7.1 Hardware Setup

The experimental platform comprises a MiniDSP UMA-16 microphone array and a USB webcam, both connected to a host computer running MATLAB R2024a. The system is designed for synchronized audio-video acquisition and real-time processing.

Microphone Array: MiniDSP UMA-16

The MiniDSP UMA-16 (shown in 5.8) is a 16-channel USB microphone array arranged in a uniform 4×4 rectangular configuration with inter-element spacing of 42 mm. It is specifically designed for spatial audio applications such as beamforming, direction-of-arrival (DoA) estimation, source separation, and acoustic camera systems. With its compact form factor, high-fidelity MEMS sensors, and plug-and-play USB Audio Class 2.0 support, the UMA-16 is a practical and cost-effective choice for real-time multichannel audio capture and algorithm development.

At its core, the UMA-16 consists of two tightly integrated subsystems:

- A **microphone PCB** housing 16 Knowles SPH1668LM4H MEMS microphones, arranged in a square URA (Uniform Rectangular Array). The board includes a central cutout to accommodate an optional USB camera, facilitating synchronized audiovisual applications such as speaker tracking or human-robot interaction.
- An **XMOS MCHStreamer Lite** board that handles Pulse Density Modulation (PDM) to Pulse Code Modulation (PCM) conversion and provides USB streaming. This embedded DSP ensures synchronized audio capture across all channels with 24-bit resolution and sampling rates up to 48 kHz.

The MEMS microphones exhibit excellent acoustic performance, with a typical SNR of 65 dB and an acoustic overload point of 120 dB SPL. The array's geometry is carefully selected to satisfy the spatial Nyquist criterion for frequencies up to 4 kHz:

$$d \leq \frac{\lambda}{2} = \frac{c}{2f_{\max}} = \frac{343\,\mathrm{m/s}}{2 \cdot 4000\,\mathrm{Hz}} \approx 42.875\,\mathrm{mm}$$

Thus, the chosen spacing of $d = 42$ mm avoids spatial aliasing and ensures accurate localization in the voice frequency range.

The UMA-16 is powered via USB and requires no external supply. It is supported natively on macOS and Linux and includes a custom ASIO driver for Windows. In MATLAB, it can be accessed through the `audioDeviceReader` interface for real-time acquisition.

Key Features:

- 16 synchronized MEMS microphones (Knowles SPH1668LM4H)
- USB-powered with 24-bit, 48 kHz multichannel streaming
- Central cutout for USB camera integration
- Open hardware schematics and sample MATLAB code available

Table 5.2: UMA-16 v2 USB: Key Technical Specifications

Parameter	**Specification**
Mic Configuration	4×4 URA, 42 mm spacing
Microphones	16 × SPH1668LM4H (MEMS)
Sampling Rate	8 to 48 kHz (24-bit)
Interface	USB Audio Class 2.0 (XMOS Xcore200)
Supported OS	Windows (ASIO), macOS, Linux
USB Port	Mini-B (data + power)
Dimensions	132 × 202 × 18 mm
Mounting	4 × M3 threaded holes, front panel
Power Supply	5V USB bus-powered

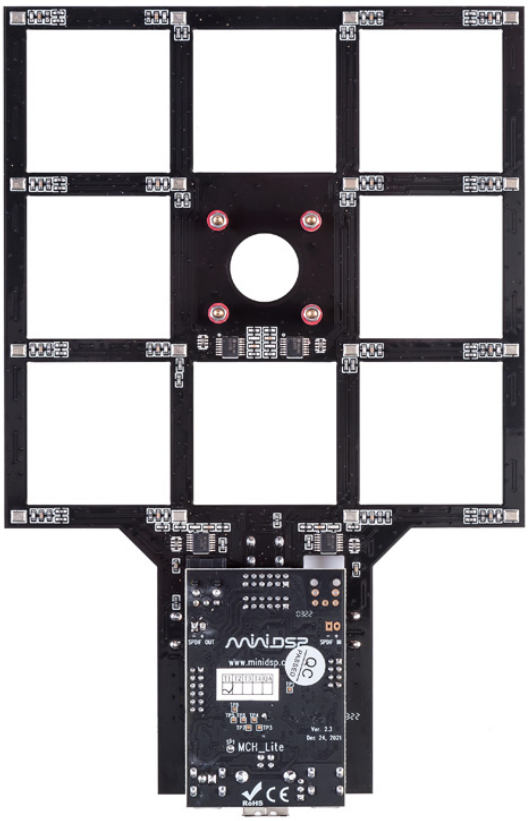


Figure 5.8: MiniDSP UMA-16 v2 USB microphone array with central camera hole.

Camera: USB Webcam

A generic USB webcam is physically integrated into the central aperture of the UMA-16 array (as shown in 5.9. This coaxial alignment ensures that the camera's optical axis is approximately colinear with the array's acoustic axis, thereby simplifying the projection of directional estimates onto the image plane. The webcam captures video at a resolution of 640×480 pixels, enabling live visualization of detected source directions.

Host System

The UMA-16 array and the webcam are connected via USB to a laptop running 64-bit Windows 10 and MATLAB R2024a. Audio and video are acquired concurrently using MATLAB's built-in toolboxes, facilitating seamless real-time integration.

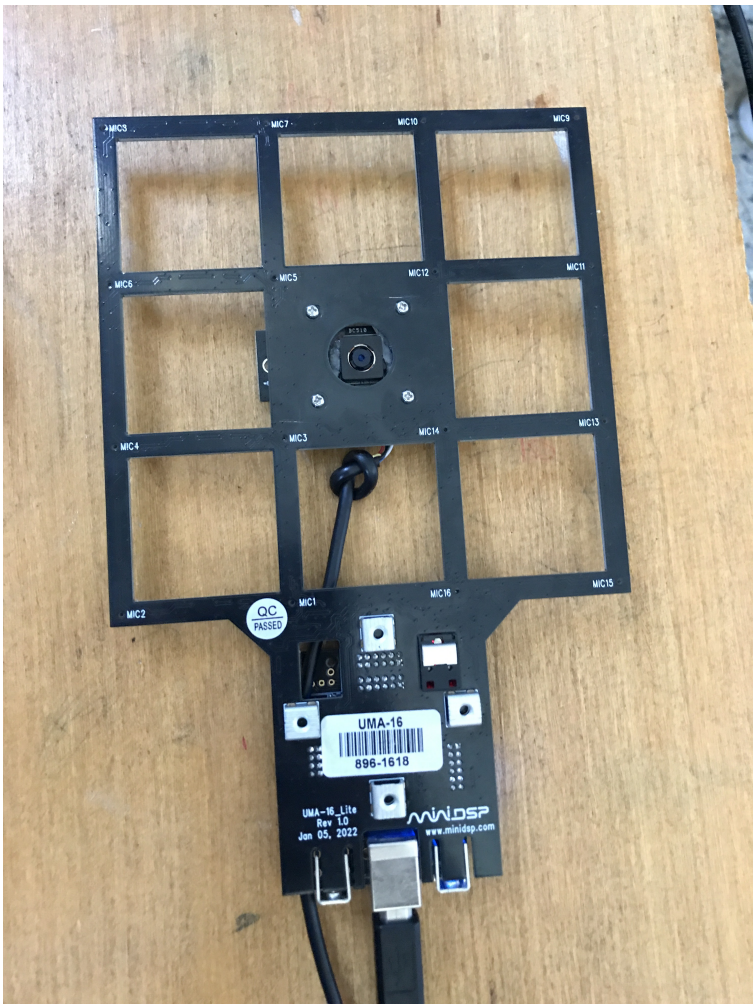


Figure 5.9: Top view of the UMA-16 array with an integrated webcam at its center, connected via USB to the processing unit.

5.7.2 Algorithm Implementation: Single-Source Tracking

Algorithm 8 Real-Time Single-Source DoA Tracking with Camera Overlay

Initialize: load `cfg`, microphone positions, TDOA table, SRP grid, camera parameters
Create audio input (`audioDeviceReader`) and video input (`webcam`)
Initialize face detector and projection depth $Z \leftarrow 1.0$
Set `frameCounter` $\leftarrow 0$
while GUI window is open **do**
    Capture audio frame x and normalize amplitude
    Capture video frame I
    `frameCounter` $\leftarrow$ `frameCounter` $+ 1$
    if voice_activity_detect(x) **then**
        Compute GCC-PHAT: $gcc \leftarrow$ `compute_gcc`(x, cfg)
        Estimate DoA using hierarchical Gaussian SRP:

$$(\hat{\theta}, \hat{\phi}) \leftarrow \texttt{hierarchical\_search\_gaussian}(gcc, cfg)$$

        Apply EMA filtering:

$$(\theta, \phi) \leftarrow \texttt{update\_filtered\_doa}(\hat{\theta}, \hat{\phi}, \texttt{'ema'})$$

        Convert (θ, ϕ) to 3D unit vector $\mathbf{d}$
        if `mod`(`frameCounter`, 5) $= 0$ **then**
            Resize image: $I' \leftarrow$ `imresize`$(I, 0.5)$
            Detect face: $bboxes \leftarrow$ `faceDetector`(I')
            if face detected **then**
                Estimate depth: $Z \leftarrow \frac{f \cdot W}{w}$ using face width w
                Clamp Z to $[0.4, 2.0]$
            end if
        end if
        Project $Z \cdot \mathbf{d}$ to image plane using `worldToImage()`
        Clamp pixel coordinates to image bounds
        Draw DoA marker and update annotation text
    else
        Hide marker and display "no speech detected"
    end if
    Render updated frame to GUI
end while

The algorithm proceeds through the following key blocks:

Initialization: Load system configuration, microphone geometry, and TDOA lookup tables. Set up audio/video devices and camera calibration parameters.

Frame Acquisition: In each loop, an audio frame and a corresponding video frame are acquired and normalized.

Voice Activity Detection: An energy-based detector filters out silent frames, reducing false positives and computation.

Localization (SRP-PHAT): When speech is detected, GCC-PHAT features are computed and passed to a two-stage hierarchical SRP-PHAT localization block with Gaussian smoothing.

Temporal Filtering: Estimated DoA angles are smoothed using an exponential moving average (EMA) filter to ensure stability in the overlay.

3D Projection: The DoA vector is converted to camera coordinates and scaled using monocular depth estimation from face bounding box width, then projected into 2D via `worldToImage()`.

Visualization: The DoA estimate and face detection results are rendered as overlays on the live video feed.

5.7.3 System Evaluation

Visual Accuracy:
Qualitatively, the projected DoA marker aligns consistently with the speaker's face across test sequences. Figure 5.10 shows a projection without depth correction, where the direction vector is assumed to lie on a unit sphere. In contrast, Figure 5.11 demonstrates the benefit of face detection: by estimating the speaker's distance from the array, the system adjusts the 3D projection accordingly, improving spatial consistency. When face detection succeeds, the projected point lands within or near the detected facial bounding box, validating the projection model.

Figure 5.10: Example frame showing the DoA marker projected onto the live video stream without depth correction. The source is assumed to lie on a fixed unit sphere around the array.

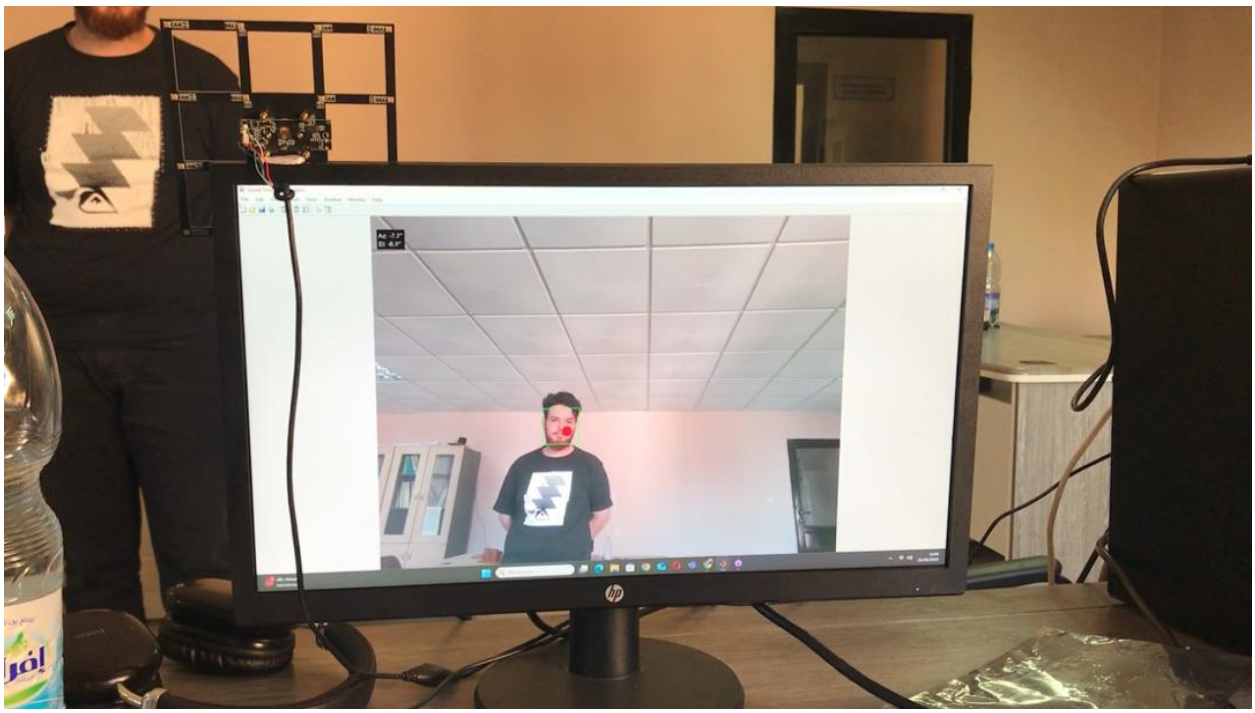


Figure 5.11: DoA marker projected onto the face-detected region, enabling monocular depth correction based on the estimated face size. This enhances 3D alignment between the acoustic source direction and its visual projection.

Timing and Latency:
The system's runtime performance was measured over multiple frames. With full camera integration—including monocular face detection for dynamic depth estimation—the average computation times (in milliseconds) were as follows:

- **VAD:** 0.1 ms
- **GCC computation:** 20.3 ms (avg)
- **SRP-PHAT localization:** 20.6 ms (avg)
- **EMA filtering:** 0.0 ms (negligible)
- **Projection:** 1.1 ms normally, up to 110 ms when face detection is triggered
- **GUI rendering:** 9.4 ms (avg)

The total per-frame latency remains below 70–80 ms on average, with rare spikes (140–160 ms) caused by face detection. To mitigate this, face detection is downsampled and executed only once every five frames.

Camera-Free Performance. When face detection is disabled (or omitted), the projection step completes in under 1.2 ms consistently, and total frame latency drops to **45–60 ms**. This confirms that real-time tracking and visualization is maintained even on commodity hardware.

Table 5.3: Average Module Execution Times (ms) With and Without Camera Projection

Module	**With Camera**	**Without Camera**
Voice Activity Detection (VAD)	0.1	0.1
GCC-PHAT Computation	20.3	20.3
SRP-PHAT Localization	20.6	20.6
EMA Filtering	0.0	0.0
Projection (avg)	1.1	0.8
Projection (w/ face detection)	93.5 (peak)	—
GUI Rendering	9.4	9.1
Total (avg)	**52–80**	**45–60**

The system demonstrates perceptually accurate, low-latency tracking of a single speaker using real-time audio-visual fusion. Camera-based face detection enables basic depth adaptation without sacrificing responsiveness, and visual output is stable and aligned. The processing pipeline proves suitable for live demos or deployment in controlled indoor environments.

5.8 Conclusion

In this chapter, we presented the design and implementation of a complete real-time sound source localization system based on SRP-PHAT, integrated with a monocular video stream for visual feedback. The system supports online audio acquisition from a 16-channel microphone array and overlays the estimated source direction onto a live camera feed using a calibrated projection pipeline.

We first described the signal processing pipeline, which includes voice activity detection, GCC-PHAT computation, hierarchical SRP-PHAT search, and exponential moving average (EMA) filtering. These modules work in tandem to produce stable azimuth and elevation estimates under real acoustic conditions. The system further includes face-based monocular depth estimation to refine the 3D direction vector prior to projection into the image plane. The resulting localization marker is visually overlaid in real time.

To support accurate projection, we leveraged camera calibration data from an OpenCV routine, converted into MATLAB's camera model. The use of face detection every 5 frames, along with image downsampling, significantly reduces the computational cost of the visual subsystem while still providing dynamic depth awareness. A fallback mechanism maintains a persistent depth estimate when no face is detected, ensuring stable visualization.

Performance evaluation showed that the entire system operates within real-time constraints, with average frame processing times ranging between 50–80 ms. When face detection is triggered, occasional spikes occur (up to 140–160 ms), but do not compromise responsiveness due to frame skipping and persistence mechanisms. Visually, the system reliably aligns the estimated source direction with the speaker's face in the video stream, confirming the effectiveness of the spatial localization and projection pipeline.

While we have already implemented and tested beamforming-based source separation in controlled settings, its integration into the real-time framework remains a future step — particularly in challenging reverberant environments where its performance can degrade. To address this, we plan to

incorporate Blind Source Separation (BSS) methods, already developed and validated in simulation, and fuse them with spatial cues from the DoA tracker.

Overall, this application forms a robust foundation for our project, already providing an effective and scalable framework for detecting and tracking acoustic sources using cost-effective hardware like the UMA-16 microphone array. It will serve as a core component in future extensions, including real-time source separation, adaptive beamforming, and integration into interactive, audio-visual systems.

Conclusion

This final project has explored a wide range of techniques in acoustic signal processing with the goal of developing robust and real-time systems for sound source localization, motion tracking, and spatial audio enhancement.

We began by investigating a single-microphone method based on Doppler shift analysis for estimating motion parameters such as velocity and altitude. While real-world implementation posed challenges—particularly in instantaneous frequency (IF) estimation under low signal-to-noise ratios—simulation results validated the proposed closed-form algorithm. This early study laid the theoretical groundwork for motion sensing with minimal hardware.

The second phase shifted toward multichannel processing using microphone arrays. We implemented and evaluated sound source localization algorithms including generalized cross-correlation (GCC), steered response power with phase transform (SRP-PHAT), and hierarchical spatial search. Among these, SRP-PHAT demonstrated robustness under moderate reverberation, confirming its practical value for indoor environments.

In the area of sound source separation, we tested classical beamforming strategies such as Delay-and-Sum (DAS), Minimum Variance Distortionless Response (MVDR), and Linearly Constrained Minimum Variance (LCMV). These methods performed well in simulated anechoic conditions, but their effectiveness deteriorated in realistic environments due to strong reverberation and overlapping impulse responses. This limitation was highlighted through offline tests on recorded mixtures of spatially separated speakers.

To overcome the lack of distance information in direction-only localization, we integrated a monocular visual subsystem. Using a calibrated webcam and MATLAB's vision toolbox, we estimated speaker depth via face detection and implemented 3D-to-2D projection using a pinhole camera model. The tracked DoAs were successfully projected onto the live video feed, enabling intuitive visualization and user interaction.

The final system is a modular real-time framework capable of:

- Capturing 16-channel audio via the MiniDSP UMA-16 USB microphone array,
- Performing hierarchical SRP-PHAT search over azimuth and elevation,
- Smoothing DoA estimates using either an EMA+median approach or a Modified 3D Kalman Filter (M3K),
- Projecting tracked directions onto the video feed using calibrated camera parameters,
- Visualizing polar localization maps and camera-space overlays in real time.

With all modules enabled, the system achieves an average frame processing time under 90 ms, confirming its real-time feasibility on commodity hardware. Without the visual processing, latency drops below 50 ms per frame.

Future work may extend this platform in several directions:

- Integration of multi-speaker tracking via probabilistic data association or clustering,
- Deep learning-based enhancement and separation techniques for reverberant scenes,
- Blind Source Separation (BSS) integration to handle overlapping speech without known DoAs,
- Deployment on embedded or low-power systems for smart conference rooms or assistive devices.

This work lays a strong foundation for research and development in spatial auditory processing. By combining Doppler-based motion estimation, multichannel beamforming, and visual spatial grounding, we offer a flexible and extensible system design. The result is a prototype platform that bridges theoretical signal models and practical user-facing applications in audio-visual scene analysis and interactive listening.

Bibliography

[1] Timlelt Hakima, Youcef Remram, and Adel Belouchrani. Closed-form solution to motion parameter estimation of an acoustic source exploiting doppler effect. *Digital Signal Processing*, 63, 04 2017.

[2] S. Gombots, J. Nowak, and M. Kaltenbacher. Sound source localization—state of the art and new inverse scheme. *Elektrotechnik und Informationstechnik*, 138(4):229–243, 2021.

[3] M.-A. Chung, H.-C. Chou, and C.-W. Lin. Sound localization based on acoustic source using multiple microphone array in an indoor environment. *Electronics*, 11(6):890, 2022.

[4] M. Jiang, C.J. Nnonyelu, J. Lundgren, G. Thungström, and M. Sjöström. A coherent wideband acoustic source localization using a uniform circular array. *Sensors*, 23(11):5061, 2023.

[5] K. et al. Hoshiba. Design of uav-embedded microphone array system for sound source localization in outdoor environments. *Sensors*, 17(11):2535, 2017.

[6] M.U. Liaquat, H.S. Munawar, A. Rahman, Z. Qadir, A.Z. Kouzani, and M.A.P. Mahmud. Sound localization for ad-hoc microphone arrays. *Energies*, 14(11):3446, 2021.

[7] X. Yang, H. Xing, and X. Ji. Sound source omnidirectional positioning calibration method based on microphone observation angle. *Complexity*, 2018:Article ID 2317853, 2018.

[8] A. Joshi, M.M. Rahman, and J.-P. Hickey. Recent advances in passive acoustic localization methods via aircraft and wake vortex aeroacoustics. *Fluids*, 7(5):218, 2022.

[9] M.D. Kafle, S. Fong, and S. Narasimhan. Active acoustic leak detection and localization in a plastic pipe using time delay estimation. *Applied Acoustics*, 187:108482, 2022.

[10] C. Mahapatra and A.R. Mohanty. Explosive sound source localization in indoor and outdoor environments using modified levenberg marquardt algorithm. *Measurement*, 187:110362, 2022.

[11] S.Y. Lee, J. Chang, and S. Lee. Deep learning-enabled high-resolution and fast sound source localization in spherical microphone array system. *IEEE Transactions on Instrumentation and Measurement*, 71:3161693, 2022.

[12] Xxvi brazilian congress on biomedical engineering: Cbeb 2018, armação de buzios, rj, brazil, 21–25 october 2018 (vol. 1). In R. Costa-Felix, J.C. Machado, and A.V. Alvarenga, editors, *IFMBE Proceedings*, volume 70/1, Singapore, 2019. Springer.

[13] R. et al. Kapoor. A novel 3d multilateration sensor using distributed ultrasonic beacons for indoor navigation. *Sensors*, 16(10):1637, 2016.

[14] B. O'Keefe. Finding location with time of arrival and time difference of arrival techniques. *ECE Sr. Capstone Proj.*, 2017. https://sites.tufts.edu/eeseniordesignhandbook/files/2017/05/FireBrick_OKeefe_F1.pdf.

[15] C.H. Knapp and G.C. Carter. The generalized correlation method for estimation of time delay. *IEEE Trans. Acoust. Speech Signal Process.*, 24(4):320–327, 1976.

[16] M.S. Hosseini, A. Rezaie, and Y. Zanjireh. Time difference of arrival estimation of sound source using cross correlation and modified maximum likelihood weighting function. *Scientia Iranica*, 24:3268–3279, 2017.

[17] H. Tang, S. Nordebo, and P. Cijvat. Doa estimation based on music algorithm. 2014. https://www.diva-portal.org/smash/record.jsf?pid=diva2:724272.

[18] Y.-M. et al. Ning. Doa estimation based on esprit algorithm method for frequency scanning lwa. *IEEE Commun. Lett.*, 24(7):1441–1445, 2020.

[19] L. et al. Kraljevic. Free-field tdoa-aoa sound source localization using three soundfield microphones. *IEEE Access*, 8:87749–87761, 2020.

[20] B.C. et al. Pinheiro. Improvements in the estimated time of flight of acoustic signals for auv localization. *Proc. MTS/IEEE OCEANS*, pages 1–6, 2013.

[21] C. De Marziani, J. Urena, Á. Hernandez, J.J. Garcia, F.J. Alvarez, A. Jimenez, M.C. Perez, J.M.V. Carrizo, J. Aparicio, and R. Alcoleas. Simultaneous round-trip time-of-flight measurements with encoded acoustic signals. *IEEE Sensors Journal*, 12(5):2931–2940, 2012.

[22] H.-P. Tan, R. Diamant, W.K.G. Seah, and M. Waldmeyer. A survey of techniques and challenges in underwater localization. *Ocean Engineering*, 38(14–15):1663–1676, 2011.

[23] *Energy Based Acoustic Source Localization*. Springer, 2006. Available online: https://link.springer.com/chapter/10.1007/3-540-36978-3_19.

[24] P. Chiariotti, M. Martarelli, and P. Castellini. Acoustic beamforming for noise source localization—reviews, methodology and applications. *Mech. Syst. Signal Process.*, 120:422–448, 2019.

[25] S.Y. Lee, J. Chang, and S. Lee. Deep learning-based method for multiple sound source localization with high resolution and accuracy. *Mechanical Systems and Signal Processing*, 161:107959, 2021.

[26] I. Cohen, J. Benesty, and S. Gannot, editors. *Speech Processing in Modern Communication: Challenges and Perspectives*, volume 3 of *Springer Topics in Signal Processing*. Springer, Berlin/Heidelberg, Germany, 2010.

[27] D. Salvati, C. Drioli, and G.L. Foresti. Acoustic source localization using a geometrically sampled grid srp-phat algorithm with max-pooling operation. *IEEE Signal Processing Letters*, 29:1828–1832, 2022.

[28] D.-B. Zhuo and H. Cao. Fast sound source localization based on srp-phat using density peaks clustering. *Applied Sciences*, 11(2):445, 2021.

[29] A. Saxena and A.Y. Ng. Learning sound location from a single microphone. In *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, pages 1737–1742, Kobe, Japan, 2009. IEEE.

[30] F. Feng, Y. Ming, and N. Hu. SSLNet: A Network for Cross-Modal Sound Source Localization in Visual Scenes. *Neurocomputing*, 500:1052–1062, 2022.

[31] Y. Masuyama, Y. Bando, K. Yatabe, Y. Sasaki, M. Onishi, and Y. Oikawa. Self-Supervised Neural Audio-Visual Sound Source Localization via Probabilistic Spatial Modeling. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, 2020.

[32] J.M. Vera-Diaz, D. Pizarro, and J. Macias-Guarasa. Towards end-to-end acoustic localization using deep learning: From audio signal to source position coordinates. *Sensors*, 18:3418, 2018.

[33] S.D. Correia, S. Tomic, and M. Beko. A feed-forward neural network approach for energy-based acoustic source localization. *Journal of Sensor and Actuator Networks*, 10:29, 2021.

[34] M. Kovandžić, V. Nikolić, A. Al-Noori, I. Ćirić, and M. Simonović. Near field acoustic localization under unfavorable conditions using feedforward neural network for processing time difference of arrival. *Expert Systems with Applications*, 71:138–146, 2017.

[35] J. Chi, X. Li, H. Wang, D. Gao, and P. Gerstoft. Sound source ranging using a feed-forward neural network trained with fitting-based early stopping. *Journal of the Acoustical Society of America*, 146:EL258–EL264, 2019.

[36] M. Hahmann, E. Fernandez-Grande, H. Gunawan, and P. Gerstoft. Sound source localization using multiple ad hoc distributed microphone arrays. *JASA Express Letters*, 2:074801, 2022.

[37] S. Chakrabarty and E.A.P. Habets. Multi-speaker doa estimation using deep convolutional networks trained with noise signals. *IEEE Journal of Selected Topics in Signal Processing*, 13:8–21, 2019.

[38] S. Chakrabarty and E.A.P. Habets. Broadband doa estimation using convolutional neural networks trained with noise signals. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 136–140, New Paltz, NY, USA, 2017.

[39] C. Xu. Spatial stereo sound source localization optimization and cnn based source feature recognition. Master's thesis, University of South Florida, 2020.

[40] A.A. Cabrera-Ponce, J. Martinez-Carranza, and C. Rascon. Detection of nearby uavs using cnn and spectrograms. In *Proceedings of the International Micro Air Vehicle Conference and Competition (IMAV)*, Madrid, Spain, 2019. 30 September–4 October.

[41] M.A.S. Md Afendi and M. Yusoff. A sound event detection based on hybrid convolution neural network and random forest. *International Journal on Artificial Intelligence (IJ-AI)*, 11:121, 2022.

[42] P.-A. Grumiaux, S. Kitic, L. Girin, and A. Guérin. Improved feature extraction for crnn-based multiple sound source localization. In *Proc. 2021 29th European Signal Processing Conference (EUSIPCO)*, Dublin, Ireland, 2021.

[43] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016.

[44] Residual neural network (resnet). https://iq.opengenus.org/residual-neural-networks/. Accessed: 2023-11-03.

[45] A. Kujawski, G. Herold, and E. Sarradj. A deep learning method for grid-free localization and quantification of sound sources. *J. Acoust. Soc. Am.*, 146:EL225–EL231, 2019.

[46] J. Naranjo-Alcazar, S. Perez-Castanos, J. Ferrandis, P. Zuccarello, and M. Cobos. Sound event localization and detection using squeeze-excitation residual cnns. *arXiv*, 2021.

[47] F. Hu, X. Song, R. He, and Y. Yu. Sound source localization based on residual network and channel attention module. *Sci. Rep.*, 13:5443, 2023.

[48] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *arXiv*, 2023.

[49] S. Kuang, K. van der Heijden, and S. Mehrkanoon. BAST: Binaural Audio Spectrogram Transformer for Binaural Sound Localization. *arXiv*, 2022.

[50] J. Wang, X. Qian, Z. Pan, M. Zhang, and H. Li. GCC-PHAT with Speech-Oriented Attention for Robotic Sound Source Localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5876–5883, Xi'an, China, 2021.

[51] Y. Huang, X. Wu, and T. Qu. A Time-Domain Unsupervised Learning Based Sound Source Localization Method. In *Proceedings of the IEEE International Conference on Information Communication and Signal Processing (ICICSP)*, pages 26–32, Shanghai, China, 2020.

[52] Y. Wu, R. Ayyalasomayajula, M.J. Bianco, D. Bharadia, and P. Gerstoft. SSLIDE: Sound Source Localization for Indoors Based on Deep Learning. In *ICASSP 2021—IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4680–4684, Toronto, ON, Canada, 2021.

[53] M.J. Bianco, S. Gannot, and P. Gerstoft. Semi-Supervised Source Localization with Deep Generative Modeling. In *IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Espoo, Finland, 2020.

[54] M.J. Bianco, S. Gannot, E. Fernandez-Grande, and P. Gerstoft. Semi-Supervised Source Localization in Reverberant Environments With Deep Generative Modeling. *IEEE Access*, 9:84956–84970, 2021.

[55] J.-Y. Kwak and Y.-J. Chung. Sound Event Detection Using Derivative Features in Deep Neural Networks. *Applied Sciences*, 10(14):4911, 2020.

[56] Gabriel Jekateryńczuk and Zbigniew Piotrowski. A survey of sound source localization and detection methods and their applications. *Sensors*, 24(1), 2024.

[57] B. Boashash. Chapter i: The time-frequency approach: Essence and terminology. In Boualem Boashash, editor, *Time-Frequency Signal Analysis and Processing*, pages 3–29. Academic Press, Oxford, second edition edition, 2016.

[58] Brian G. Ferguson and Brian G. Quinn. Parameter estimation of a moving source using time-frequency analysis. *Journal of the Acoustical Society of America*, 91(1):291–300, 1992.

[59] Hakima Timlelt. *Motion parameter estimation of a moving acoustic source exploiting time-frequency analysis*. PhD thesis, University of Sciences and Technology Houari Boumediene, 2017.

[60] Dennis Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering*, 93(26):429–441, 1946.

[61] J. Ville. Theory and applications of the notion of complex signal. *Cables et Transmission*, 2:61–74, 1948.

[62] B. Boashash. Estimating and interpreting the instantaneous frequency of a signal—part 1: Fundamentals. *Proceedings of the IEEE*, 80(4):520–538, 1992.

[63] Steven M. Kay. *Modern Spectral Estimation: Theory and Application*. Prentice Hall, 1988.

[64] L. J. Griffiths. Rapid measurement of digital instantaneous frequency. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(2):207–213, 1979.

[65] Simon Haykin. *Adaptive Filter Theory*. Prentice Hall, Englewood Cliffs, NJ, 2nd edition, 1991.

[66] Thirumalesh Nizampatnam and Praveen Kumar. Frequency estimation using kls technique. In *2023 IEEE International Conference on Signal Processing*, 2023.

[67] Leon Cohen. Time-frequency distributions—a review. *Proceedings of the IEEE*, 77(7):941–981, 1989.

[68] Arnaud Doucet, Nando de Freitas, and Neil Gordon. Sequential monte carlo methods in practice. In *Springer Series in Statistics*. Springer, 2001.

[69] Barry D Van Veen and Kevin M Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Magazine*, 5(2):4–24, 1988.

[70] Application of Artificial Neural Networks for Efficient High-Resolution 2D DOA Estimation. Uniform rectangular array (ura) figure. https://www.researchgate.net/figure/Uniform-rectangular-array-URA_fig1_258927613, 2025. Accessed: 2025-06-25.

[71] Torea Blanchard. *Détection et localisation acoustique de drones multirotors pour la surveillance d'espaces ouverts*. Phd thesis, Université de Toulon, 2021.

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

[73] Jacob Benesty, Jingdong Chen, and Yiteng Huang. *Microphone Array Signal Processing*. Springer, 2008.

[74] R. O. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, 34(3):276–280, 1986.

[75] C. H. Knapp and G. C. Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(4):320–327, 1976.

[76] StackExchange contributors. What does spectral whitening mean? DSP StackExchange, 2013. https://dsp.stackexchange.com/questions/8743/what-does-spectral-whitening-mean.

[77] Cha Zhang, Dinei Florêncio, and Zhengyou Zhang. Why does phat work well in low noise, reverberative environments? In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2565–2568. IEEE, 2008.

[78] Ritu Boora and Sanjeev Kumar Dhull. A comparison of generalized cross-correlation methods for time delay estimation. *International Journal of Engineering Trends and Technology*, 34(1):34–39, 2016.

[79] Bernard Roth. Effective measurements using digital signal analysis. *IEEE Spectrum*, 8(4):62–70, 1971.

[80] Benoît Champagne and Stéphane Bédard. Performance of time-delay estimation in the presence of room reverberation. *IEEE Transactions on Speech and Audio Processing*, 4(2):148–152, 1996.

[81] Radu-Seastian Marinescu, Andi Buzo, Cucu Horia, and Corneliu Burileanu. Applying the accumulation of cross-power spectrum technique for traditional generalized cross-correlation time delay estimation. *International Journal on Advances in Telecommunications*, 6(3&4), 2013.

[82] Joseph H DiBiase. *A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays.* PhD thesis, Brown University, 2000.

[83] Michael Brandstein and Harvey Silverman. A robust method for speech signal time-delay estimation in reverberant rooms. In *ICASSP*, volume 1, pages 375–378. IEEE, 1997.

[84] Sharon Gannot, Emmanuel Vincent, Sharon Markovich-Golan, and Alexey Ozerov. A consolidated perspective on multimicrophone speech enhancement and source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(4):692–730, 2017.

[85] H. Wang and M. Kaveh. Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wideband sources. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(4):823–831, 1985.

[86] S. Valaee and P. Kabal. Wideband array processing using a two-sided correlation transformation. *IEEE Transactions on Signal Processing*, 43(1):160–172, 1995.

[87] H. Yu, J. Liu, Z. Huang, Y. Zhou, and X. Xu. Tofs: A new method for wideband doa estimation. In *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, volume 28, pages 598–601, Shanghai, China, 2007.

[88] Y.S. Yoon, L.M. Kaplan, and J.H. McClellan. Tops: new doa estimator for wideband signals. *IEEE Transactions on Signal Processing*, 54(6):1977–1989, 2006.

[89] K. Okane and T. Ohtsuki. Resolution improvement of wideband direction-of-arrival estimation squared-tops. In *2010 IEEE International Conference on Communications*, pages 1–5, Cape Town, South Africa, 2010.

[90] Hassan Ougraz, Said Safi, Ahmed Boumezzough, and Miloud Frikel. Performance comparison of several algorithms for localization of wideband sources. *Journal of Telecommunications and Information Technology*, 2023(3), 2023.

[91] O. L. Frost. An algorithm for linearly constrained adaptive array processing. *Proceedings of the IEEE*, 60(8):926–935, 1972.

[92] Ea-Ee Jan and J. Flanagan. Sound capture from spatial volumes: matched-filter processing of microphone arrays having randomly-distributed sensors. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 2, pages 917–920 vol. 2, 1996.

[93] Simon Doclo, Sharon Gannot, Marc Moonen, and Alexander Spriet. Acoustic beamforming for hearing aid applications. In Simon Haykin and K. J. Ray Liu, editors, *Handbook on Array Processing and Sensor Networks*, chapter 8, pages 269–302. Wiley-IEEE Press, Hoboken, NJ, USA, 2010.

[94] Sharon Markovich-Golan, Sharon Gannot, and Israel Cohen. A weighted multichannel wiener filter for multiple sources scenarios. pages 1–5, Eilat, Israel, 2012. Best Student Paper Award.

[95] L. J. Griffiths and C. W. Jim. An alternative approach to linearly constrained adaptive beamforming. *IEEE Transactions on Antennas and Propagation*, 30(1):27–34, 1982.

[96] Aalto University Speech Processing Group. Voice activity detection (vad) – chapter 8.1, 2023. Accessed: 2025-06-21.

[97] Lawrence Rabiner and Michael Sambur. An algorithm for determining the endpoints of isolated utterances. *Bell System Technical Journal*, 54(2):297–315, 1975.

[98] Francois Grondin and Francois Michaud. Lightweight and optimized sound source localization and tracking methods for open and closed microphone array configurations, 2018.

[99] Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[100] Zhengyou Zhang. A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft Research, Redmond, WA, USA, March 2000. Available at https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr98-71.pdf.