

北京交通大学

本科毕业设计（论文）

高速声学照相机的设计与实现

Design and implementation of high speed acoustic camera

学 院： 电子信息工程学院

专 业： 通信工程专业

学生姓名： 谢亚希

学 号： 14211127

指导教师： 余晶晶

北京交通大学

2018 年 6 月

学士论文版权使用授权书

本学士论文作者完全了解北京交通大学有关保留、使用学士论文的规定。特授权北京交通大学可以将学士论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：

指导教师签名：

签字日期： 年 月 日

签字日期： 年 月 日

中文摘要

声学照相机广泛应用于飞机、潜水艇、汽车等大型设备噪声源定位和故障分析。对声学照相机的改进研究集中于三方面：麦克风阵列设计、声源定位算法研究、以及高速运动中多普勒现象的解决。

本文通过研究设计一种新型声学照相机，实现日常场景的噪声源定位和彩色声学图谱生成，实现噪声源快速定位和探测；并基于高速运动物体进行改进算法，实现运动声源定位分析。本项目基于麦克风阵列和 NI 数据采集器设计搭建多通道声音采集系统，利用可控响应功率声源定位算法，通过 MATLAB 和 LABVIEW 形成彩色声学图谱，实现声源定位。针对高速运动物体多普勒频移问题，设计基于非线性插值的解多普勒改进算法。经过实验得到 SRCP 较 SRP 算法而言对于声源定位的精度没有影响，但声谱图效果更好，并且大幅度地减少了运算的时间成本；麦克风阵元数越多，定位效果越好，并且环绕型麦克风阵列比线型麦克风阵列定位效果好；白化参数较大、时间窗采样区间小，声谱图效果更好，定位误差小；定性分析了解多普勒算法的可行性。定性分析了解多普勒算法的可行性。

本文的研究结果可广泛应用于基于麦克风阵列多通道实时采集系统实现的声源定位和声谱图生成中，并且对于运动中物体多普勒频移现象提出了改进算法，实验结果具有一定的推广价值。

关键词：声学照相机；SRCP；解多普勒；声源定位；波束成形

ABSTRACT

Acoustic cameras are widely used in noise source location and fault analysis of large equipment such as aircraft, submarines and automobiles. The improvement of the acoustic camera focuses on three aspects: the microphone array design, the sound source localization algorithm and the solution of Doppler phenomenon in high-speed motion.

In this paper, a new type of acoustic camera is designed to realize the noise source location of the daily scene and the generation of color acoustic atlas, to realize the fast location and detection of the noise source, and to make an improvement algorithm based on the high-speed moving object to realize the analysis of the motion sound source. This project is based on a microphone array and NI data collector to set up a multi-channel sound acquisition system, and use the Steered Response Coherent Power (SRCP) algorithm to form the color acoustic atlas through MATLAB and LABVIEW. Aiming at the Doppler shift problem of high-speed moving objects, an improved Doppler algorithm based on nonlinear interpolation is designed. In the experiment, when the whitening parameters are large, the sampling interval is small, and the radial size of the microphone array is large, the sound spectrum is better and the location error is smaller. SRCP is better than the SRP algorithm, and the time consuming is obviously reduced. The feasibility of the Doppler Resolving algorithm is qualitatively analyzed.

The results of this paper can be widely used in the sound source localization and sonogram generation based on the multi-channel real-time acquisition system of microphone array, and an improved algorithm is proposed for the Doppler shift. The experimental results have a certain popularization value.

KEYWORDS: Acoustic Camera; SRCP; Doppler Resolving Algorithm; Sound source localization; Beamforming Algorithm

目 录

中文摘要.....	I
ABSTRACT.....	II
目 录.....	III
1 绪论	1
1.1 研究背景	1
1.2 国内外研究现状	1
1.2.1 麦克风阵列的研究	2
1.2.2 声源定位算法的研究	4
1.2.3 高速运动物体的声源定位算法的研究	5
1.3 本文主要工作与结构安排	6
2 基于麦克风阵列的声源定位算法	8
2.1 时延估计算法	8
2.2 基于高分辨率谱估计算法	9
2.3 基于可控波束成形的声源定位算法	9
2.3.1 波束成形	9
2.3.2 SRP 算法	11
2.3.3 利用 SRP 和 SRCP 生成声谱图	13
3 高速声学照相机	16
3.1 高速运动物体声源定位问题	16
3.2 解多普勒算法	16
4 总体系统设计与搭建	20
4.1 硬件系统方案设计	20
4.2 总体设计框图及硬件选型	21
4.2.1 总体设计框图	21
4.2.2 硬件设计与选型	22
4.2.3 NI 人机交互界面	25
5 结论与分析	28
5.1 声学实验过程及分析记录	28
5.2 结果与分析	32
5.2.1 物体内部噪声源定位	32
5.2.2 人声定位:	34
5.2.3 SRP 和 SRCP 算法影响:	36
5.2.4 麦克风阵列影响:	37
5.2.5 处理参数影响	38

6	结论及未来工作	43
6.1	结论	43
6.2	未来工作及展望	43
参考文献		45
致 谢		47
附 录		48

1 绪论

1.1 研究背景

随着越来越多的交通工具、大型机械、空中运输在人们日常生活中的逐渐普及，声音和噪声在我们的日常生活中越来越普遍，噪音污染已经逐渐成为了影响人类正常生活的重要原因。有研究显示，人们过长时间处于高于 65db 的噪声下时，心肌梗死的可能性会很高，有 2% 的心肌梗死人群是由于交通噪声所导致。汽车，飞机等等我们不曾注意的大型设备所携带的噪声，往往会影响人的潜意识，进而影响睡眠和人们的身心健康。如何有效地分析噪声的位置，快速地进行噪声源定位已经成为了解决日常生活中噪音污染的前提，只有有效且详细地获取噪声源信号，并且精确地进行噪声源定位，才能让工程师们更加高效便捷地选取合适的方法降低噪音污染，改善人们的生活。另外，机械和车间设备存在的故障和缺陷常常可以通过噪声探测来发现。一个有经验的技术工人，是可以通过耳朵听出机械上的故障的。于是人们开始研究一种更加客观的自动化的测试方法，通过快速识别机械上的噪音源位置，来帮助机械师们处理和解决机械故障。这种通过噪声源定位来发现机械故障的方法，将会大大降低传统方法的时间成本，具有重要意义。人眼区别于其他器官，是收集信息最快适用性也最强的，研究者们提出将噪声源的位置直观地显示在实物图上，用可视化的声学图像显示噪声源定位将会具有革命性的意义。声学照相机便于此诞生。

声学相机是一种噪声源测试和定位系统，基于数字麦克风传感器阵列，将声音可视化地以彩色图谱方式显现，可以以此测量声场分布，形成的声谱图类似热影像仪对物体温度的探测和显示。声学照相机具备能快速实时对高速运动物体进行可视化噪声源定位，因此得到了广泛的应用，目前主要用于喷气式飞机、潜水艇、磁悬浮列车、行驶中的汽车等大型设备噪声源定位和故障分析。

本文论述了声学照相机的主要技术热点，包括麦克风阵列的设计与研究、声源定位算法研究、高速运动声源的多普勒频移现象，并进行了硬件搭建和声学实验。

1.2 国内外研究现状

目前已有许多国内外公司开发出了成型的声学照相机。西门子发布的 LMS 声学照相机，是一款可以快速进行声源定位的多功能系统，可用于增强声音设计与品牌认知，并且可以帮助相关行业实现快速声学故障诊断，该声学照相机无论在实验室或室外场景，

利用配备的专用软件和高品质数字麦克风阵列，都可以实现可视化的即时、准确地对噪音位置的定位；挪威 Norsonic 公司研发的 Nor848 声学照相机系统则在原有声音照相机技术上创新，使用了大量的麦克风去避免假象讯号，跟一般传统式的声音影像化的技术相比，Nor848 声学照相机具有更高的指向性，在声源位置查找方面更精准，误判更少。国内也有声学相机的实例。北京朗德科技的声学照相机，具有完整的软硬件配套设备。硬件部分有摄像机的麦克风阵列、数据记录装置和笔记本搭配软件部分采用的 Noise Image，实现声源定位，并且可以同时进行对于信号的时间，频率，转速和空间选择性的分析。该系统的软件部分更加人性化，由客户决定真正需要的功能，生成特制的软件系统。如下图 1-1 中，展示了利用声学照相机进行噪声源定位的实验过程图。

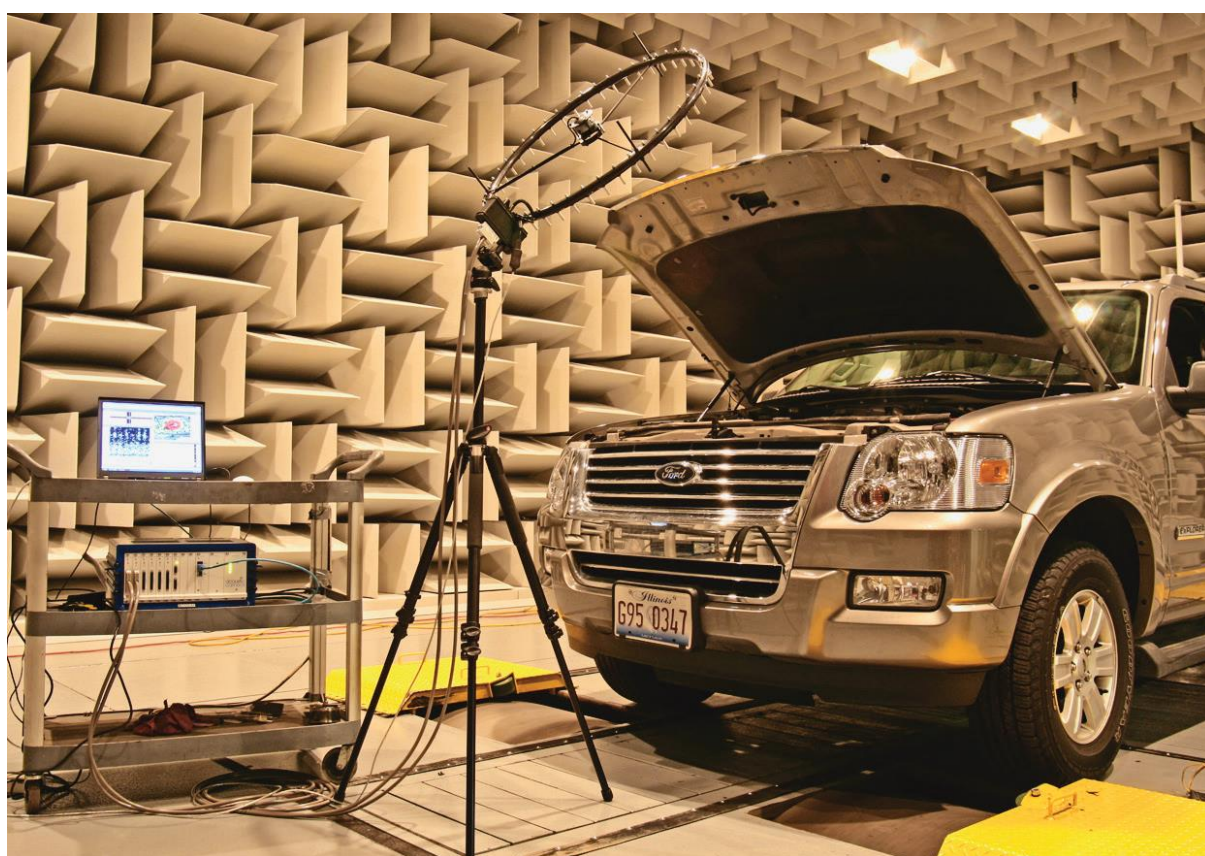


图 1-1 声学照相机噪声源定位的实验及结果^[2]

随着声学照相机研究的日益进展和应用领域上的不断扩充。声学照相机也逐渐在面临更多更新的难点和挑战。经过调研，目前国内外对声学照相机的研究方面和热点，主要是在三个方面：不同类型的麦克风阵列对于系统性能影响的研究，声源定位算法的研究以及针对高速运动物体的声源定位算法的研究。

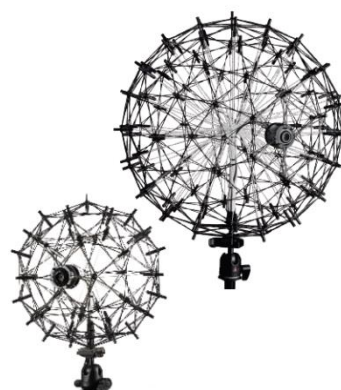
1.2.1 麦克风阵列的研究

研究发现麦克风阵列的设计直接影响声学照相机的结果^[1]。麦克风阵列的设计与声源定位结果相关性的研究是声学照相机的研究热点。声学照相机的研究中常采用复杂麦克风阵列来增加声学相机的精度和适用范围。本文所提的麦克风阵列特指在语音处理中运用的，按照某种形式排列的多麦克风组成的阵列系统，下文简称为麦克风阵列。

有研究表明，声谱图效果与麦克风数量和阵列形状均有联系。普通情况下，麦克风数量越多，声谱图的分辨率也就越高。目前常见的麦克风阵列有十字形、矩形阵列、螺旋形、星型阵列、圆型阵列等，但除了矩形阵列、十字形与分辨率有简单的解析关系之外，其他形状的阵列与分辨率之间都不是直接相关的，目前研究表明螺旋形麦克风对于声谱图的结果呈现效果是最佳的。另外还有分布式麦克风阵列，分布式麦克风阵列是规则麦克风阵列的发展，它没有特定的集合拓扑结构，可以自由摆放麦克风位置。其优势在于使得阵列的建立更加灵活可变，但同时它也存在某些问题，比如在要求严格麦克风阵列的算法中便不能使用。分布式麦克风阵列常用于波束成形算法中^[4]。如下图 1-2 和图 1-3 中，展示了部分常见的麦克风阵列，分别是环形麦克风阵列和球型麦克风阵列。

麦克风阵列相较于单个麦克风具备了更多优点：麦克风阵列克服了单个麦克风信息量少的缺点，取样更加丰富而不受限制；具有空间选择性；可以自动跟踪识别声源，而不需要改变其物理位置；可以同时识别多个声源；可用于近场测距而单麦克风无法实现。基于麦克风的声源定位研究从上世纪八十年代开始就在国际上受到了越来越多的关注，我国起步较晚^[4]。

目前麦克风阵列的研究趋势：阵列小型化的研究，因为阵列受限于半波长理论限制，所以口径较大，如何将阵列小型化，又不会影响系统定位的效果，是一个重点的研究方向；另一个就是阵列低成本化的研究，效能优良的麦克风成本还是较高的，也因为这个原因影响了阵列应用的普及，但是低成本化并不只是更换芯片器件，有的时候是针对整个系统结构的改进性设计，有时也包括算法、器件、芯片和云端；再有就是对阵列形状的研究，除了上文提及的已有的相关结论外，关于阵列形状对于结果影响的研究仍然在进行，因此，本项目基于 7 麦克风的分布式阵列，尝试了多种规则和不规则的麦克风阵列以实现声学照相机。

图 1-2 环形 32 麦克风阵列和环形 48 麦克风阵列^[2]图 1-3 球型 48 麦克风阵列和球型 120 麦克风阵列^[2]

1.2.2 声源定位算法的研究

声源定位算法作为声学照相机的核心算法始终是声学照相机研究的重点，包括如何克服混响和背景噪声的影响，满足实时处理要求、提高定位的精度，减少运算复杂度，等等。随声源定位算法应用的领域不断拓展，对于定位算法的理论研究也取得了长足的进步。目前常见的声源定位算法主要有三种：时延估计算法、高分辨率谱估计技术算法、可控波束成形算法。

时延估计算法，通过计算声源到达空间域两个传感器之间时间差来进行声源定位的方法。时延定位算法的优点是计算量小、定位精度相对高，可以实现实时定位，因此比较常用。但该方法仅适用于单个声源的定位，对于多声源定位，性能便会严重下降，因此该方法在多声源定位中也受到了极大限制。时延估计算法主要分为：最小均方自适应滤波器法（LMS）、基于高阶统计量的双谱时延估计法、广义互相关函数法（GCC）和互功率谱相位法（CSP）^[6]。

基于高分辨率谱估计的定位算法，主要是通过求解传感器接受到的信号间相关矩阵的方法来进行声源定位。其阵列信号处理的主要核心算法是空间谱估计^[7]。但该算法需

要通过增大阵列孔径来提高空间处理精度，但在许多实际应用中，这是不现实的。

基于最大输出功率的波束成形算法，又称为可控波束成形的定位算法，它是以波束成形算法为基础思想实现的。基本思想是对于传感器采集到的信号进行一系列处理（包括延迟、加权、求和等），以此来直接控制波束，以达到指向有最大输出功率方向的目的，是一种形成空间指向性的方法。波束成形允许感兴趣方向上的信号通过，对其形成增益的同时抑制不感兴趣方向上的信号，即自适应波束成形算法以及自适应零点算法。这两个方向组成了阵列信号处理的一个重要部分——自适应空域滤波^[5]。波束成形算法可以分为基于训练信号或者参考信号的方法；基于方向估计的自适应算法和基于信号结构的波束成形方法。依据是否需要发射参考信号，还可以将算法分为非盲算法和盲算法。该算法稳定性较高，且广泛应用于多声源定位，而时延估计算法只能定位得出声源位置所在的双曲线的一支，但该方法可以直接获得声源较为精确的位置。由于该方法需要进行全局搜索，运算量大，因此如何有效减少运算量成为波束成形的重要研究方向之一。

本文将在下文详细介绍声源定位算法及改进算法，将会详细重点集中于介绍基于波束成形的声源定位算法，并在其基础上，介绍可控响应功率 SRP 算法和 SRCP 算法。

1.2.3 高速运动物体的声源定位算法的研究

随着声学照相机在飞机、汽车、铁路等方面的需求增加，如何解决高速运动物体本身的多普勒频移问题，也成为声学照相机的另一个研究难点。目前多普勒估计技术有基于傅里叶变换的多普勒补偿方法，在此基础上有人提出了分数阶的傅里叶变换的多普勒系数估计方法，FFT 的多普勒频移估计算法等等。

多普勒是指：当移动台以恒定的速率沿某一方向移动时，由于传播路程差的原因，会造成相位和频率的变化，通常这种变化就被称为多普勒频移。多普勒效应造成的发射和接收的频率之差称为多普勒频移。它揭示了波的属性在运动中发生变化的规律。具体是指：物体辐射的波长因为波源和观测者的相对运动而产生变化。在运动的波源前面，波被压缩，波长变得较短，频率变得较高；当运动在波源后面时，会产生相反的效应。波长变得较长，频率变得较低。如下图 1-4 中，展示了声源向右移动时的多普勒现象。

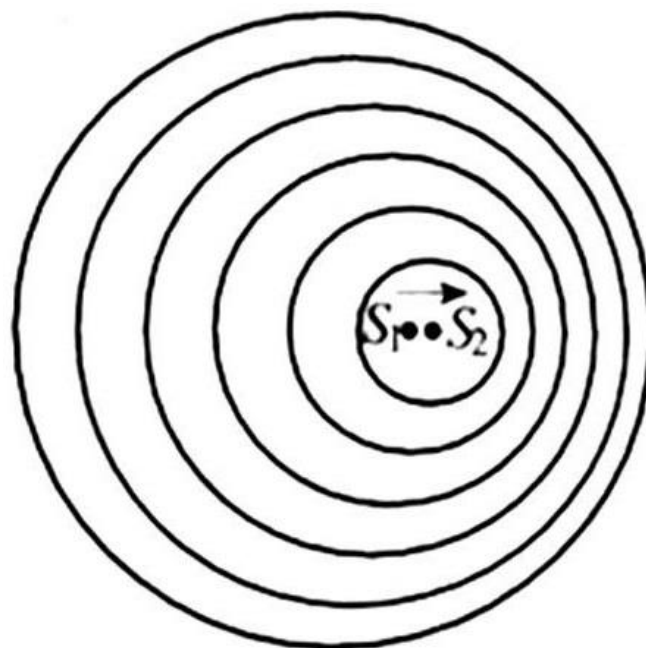


图 1-4 声源向右移动的多普勒现象

1.3 本文主要工作与结构安排

综合上述的各种研究热点和尚存在的研究问题，基于任务书，确定自己的研究内容：通过讨论和研究设计一种基于麦克风阵列的多通道声音采集系统，设计和实现日常场景的声学相机，研究并选择合适的声源定位算法，通过直观形成彩色声学图谱，实现噪声源快速定位和探测。针对高速铁路，飞机等速度大于 300km/h 的目标，针对多普勒频移问题，改进算法，实现适用于高速运动声学成像和探测设备。

本文的结构安排如下：

第一章，绪论。首先，对声学照相机产生的背景、原理与应用做了基本的介绍；其次，针对目前国内外对于声学照相机的研究现状及热点进行了调研与阐述；最后，阐述了本项目的研究内容和目标，本文的结构安排与内容。

第二章，基于麦克风阵列的声源定位算法。首先，介绍时延估计算法、高分辨率谱估计技术算法的原理和、改进技术；其次，重点介绍基于波束成形算法的可控响应功率（SRP）算法的原理，并扩展改进算法——可控响应相干功率（SRCF）算法。

第三章，高速声学相机的研究。首先介绍了高速运动物体声源定位算法中出现的新的问题，其中核心是多普勒效应的解决；然后本文阐述了一种基于非线性插值的解多普勒频移的方法。

第四章，总体系统设计与搭建。首先介绍硬件系统设计方案选择；然后根据调研方案，做出总体系统方案设计和硬件选型，包括麦克风传感器的选型、麦克风阵列的设计、

采集系统硬件选型及功能；最后，就采集系统的人机交互界面进行评价和测试。

第五章，结果与分析。首先对声学实验的过程进行简单的阐述，然后进行了人声定位、静止物体内部噪声源定位，并对结果进行分析比较，以及对于改进模块的定性分析从而得到对系统性能的评估。

第六章，结论及未来工作。对系统性能进行总结和评价，并对尚需改进的问题和未来可以拓展的工作进行阐述。

2 基于麦克风阵列的声源定位算法

2.1 时延估计算法

时延估计（TDOA）算法的基本原理是通过计算声源到达空间中不同位置的传感器的时间差来进行定位的方法。时延估计算法主要可以分为：广义互相关函数法（GCC）、最小均方自适应滤波器算法（LMS）、互功率谱相位法（CSP）和基于高阶统计量的双谱时延估计法^[8]。

以下简单列举几种时延估计算法的思想：

最小均方自适应滤波器法（LMS），通过自适应地调节滤波器的权系数，使得 2 个输入信号的均方差最小以求得时间延迟；广义互相关函数法（GCC），通过计算采集信号广义互相关函数进行时间延迟估计；互功率谱相位法（CSP）通过对功率谱的归一化，消除信号幅度的信息，值保留信号相位的信息；高阶统计量的双谱时延估计法是来源于，某些实际情况中，噪声是独立的零均值的平稳高斯过程而信号可以被看作非高斯平稳随机过程的，高斯噪声在理论上是可以得到完全的抑制的，因此此时，使用高阶统计量做延迟估计会更加合理^[9]。

上述的各种算法都存在不同的优缺点，因此也适用于不同的场合：

有研究显示，最小均方自适应滤波器法（LMS），它的优点是不用依赖输入信号，可以自动化地通过调节自身结构和参数来实现最优选择，这种算法适用于统计特性的时变环境，但是缺点是其计算量很大；广义互相关函数法（GCC）在信噪比高的条件下时延估计精度比较好，但在信噪比较低的环境下，时延估计精度的效果较差，并且它常常需要大量的数据才可以分析得到准确估计结果，在实际应用中，如果只用一帧数据来得到功率谱估计的话，结果误差会比较大^[10]；互功率谱相位法（CSP），对于混响噪声都可以有比较好的抑制效果，并且算法本身也具有一定的抗噪声能力，但其延迟估计性能会明显降低随着信噪比降低而下降；高阶统计量的双谱时延估计法，在噪声信号为高斯分布，声源信号为非高斯分布时，理论上可以完全消除噪声影响，但是这种方法计算量大。

基于时延估计的声源定位算法已经有很多，但尚且仍存在许多不足，因此针对时延估计的研究始终在进行，许多研究者在基于前人的研究结果上逐渐做出改进，产生了许多新的改进算法，比如对于时延估计算法广义互相关函数法（GCC），引入一个加权函数对互相关功率谱进行调整，提出了相位变化（PHAT）的广义互相关函数（GCC-PHAT）方法等等^[11]。

时延定位算法由于计算量最小、定位精度也相对较高，并且可以实现实时定位，因

此也是最常用的一种方法。但它存在很多问题：时延估计和声源定位分成两步进行，这就存在较大的误差，因为在第二步定位中计算用到的参数是对过去时间的估计，所以这只是一种次估计的方法，不够精确^[12]；在空间环境有较强混响和噪声时，很难获取精确的时延，误差较大；仅适用于单声源定位，对于多声源定位，性能会严重下降，在多声源定位中受到了极大限制^[13]。

2.2 基于高分辨率谱估计算法

基于高分辨率谱估计算法可以分为：非参数方法和参数方法两类。非参数方法，又称为经典谱估计方法，是由一组数据估计得到自相关序列，并求得傅立叶变换，获得功率谱估计。其中典型的非参数方法是周期图，它先估计信号的自相关序列，再求自相关序列的离散傅立叶变换，获得功率谱估计。周期图所表示的功率与加窗信号的傅立叶变换幅值的平方成正比。传统周期图所加的窗是矩形窗，一般的窗函数会得到修正周期图，窗函数不同将影响功率谱的主瓣宽度和旁瓣电平。修正周期图得到功率谱估计结果并不是一致的，因此有了改进算法：平均周期图 Bartlett 法和 Welch 法，这种算法可以获得功率谱的一致估计^[5]。

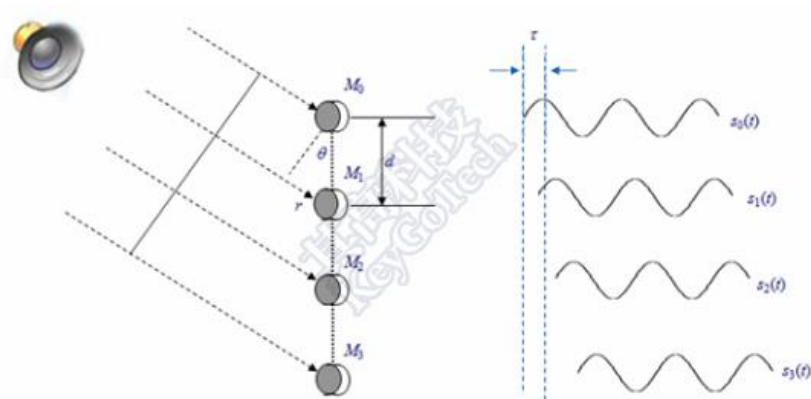
非参数方法性能主要取决于数据长度，具有频率分辨率低的缺陷，而参数谱估计方法具有频率分辨率高的优点，称为高分辨率谱估计方法。基于高分辨率谱估计声源定位算法可分为 ARMA 谱估计、最大熵谱估计、最小方差谱估计以及基于子空间的谱估计。

高分辨率谱估计的优点是理想情况下可以做到高精度的定位，但对于宽带的语音信号，运算量会很大，这需要时间平均来得到接收信号协方差矩阵的估计值，这就要求声源和噪声都是平稳的，实际中，语音只能保证暂时短时间的平稳，实际接收信号在受到噪声和混响干扰后算法性能会急剧下降，因此基于高分辨率谱估计的声源定位算法不适合实时语音定位系统^[14]。

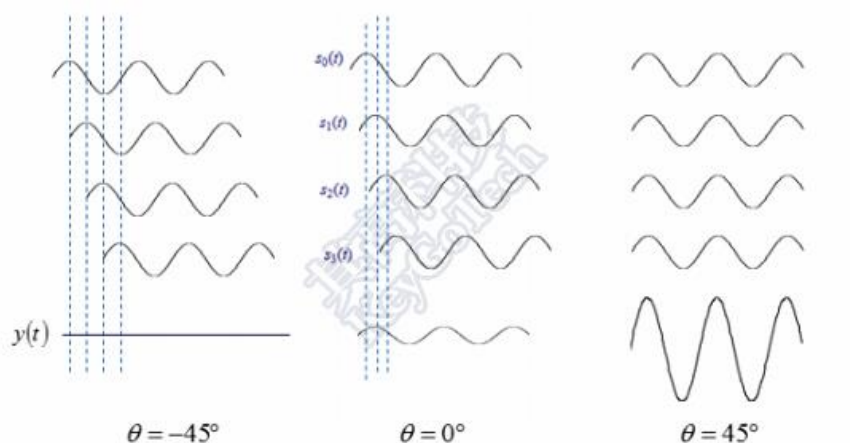
2.3 基于可控波束成形的声源定位算法

2.3.1 波束成形

波束成形（Beamforming）的基本思想是传感器阵列输出信号前，选取一个适当的加权向量以补偿各个阵元的传播时延，目的是为了使得在某一个期望方向上阵列输出可以同向叠加，使得阵列在该方向上产生增益，并可以在另一个不关心的方向上对信号进行一定程度的抑制^[15]。如下图 2-1 中，展示了波束成形的基本工作原理图如下：


 图 2-1 波束成形的基本工作原理图^[3]

使用波束成形算法，首先的条件是声源是远场声源（近场声源用 TDOA），远场声源中入射声波可以看作都是平行的；平行的声场，如果入射角度与麦克风平面垂直，那么声信号可以同时到达各个麦克风，如果不垂直，则出现如上左图的现象，声信号到每个阵元都会有不同的延时，延时大小是由入射角度决定。如下图 2-2 中，展示了不同入射角度的波形叠加后的波形图。


 图 2-2 不同入射角度叠加后的结果^[3]

从上图中可看出，根据入射角度的不同，叠加出的最终波形强度是不一样的。如 $\theta = -45^\circ$ 度，几乎没有信号， $\theta = 0^\circ$ 度，信号被衰减， $\theta = 45^\circ$ 度，信号被增强。于是可以得到结论：如果把多个原来没有极性的全向型麦克风组装成阵列后，该麦克风阵列将是具有极性的。

也就是说原来独立的全向麦克风是不具备指向性的，但多个麦克风阵列采集的信号经过加权延时求和后，可以被调整为具有指向性的。因此通过加权、延时、求和后可以将所关心的方向上声音信号增强，将不感兴趣的方向上的信号进行衰减抑制，这就是波束成形的基本原理，此时的麦克风阵列相当于一个空间滤波器。

2.3.2 SRP 算法

2.3.2.1 可控波束算法

可控波束定位算法，是一种经典定位方法。它是以波束成形为基本思想，通过控制麦克风阵列的接收方向来在整个空间域内进行扫描，并计算得出能量最大的位置就是目标声源位置。具体来说，对麦克风接收声源信号进行延时、加权、求和的一系列操作，以形成波束，引导该波束，搜索所有声源可能位置，遍历能量结果中输出功率最大的像素点即为声源位置^[16]。可控波束定位算法分为自适应波束算法和延迟累加波束算法。前者增加自适应滤波，导致运算量大，而且输出信号有一定程度的失真，在没有混响的情况下有比较好的效果。而后者信号失真小，运算量较小，但需要较多的阵元以获得较好的效果，抗噪性能较差。

可控波束成形算法被大量应用于声源定位。声源位置已知的情况下，调整每个阵元的导引时延，使每个麦克风的信号在时域上对齐，具体的做法是在每个不同的麦克风收集到的语音信号样本前补零，将麦克风阵列导引到声源的位置。然后将每通道麦克风采集处理过的信号相加，进而抑制某方向上的噪声、增强某方向上的信号。上述这种波束成形最简单实用，称为延时-求和(delay-and-sum)波束成形。还有一种较为灵活的波束成形算法，称为滤波-求和(filter-and-sum)波束成形。该算法在对每个麦克风信号作延时处理的基础上，进行滤波处理，再将信号相加。该方法比前一种方法性能好。在声源的位置未知的情况下，也就是声源定位的情况，首先假定声源处于某个位置，再调整导引时延使麦克风对准某个可能的声源位置，计算出麦克风阵被导引到该位置的响应功率，再求所有可能位置中的响应功率最大值，即为声源位置^[17]。

根据可控波束的声源定位算法的原理，可以看到，该算法需要遍历搜索声源可能的位置点，并计算每一个网格点的可控响应功率，因此计算量较大，并且计算量会随着网格数的增加，即分辨率的减小和选定研究空间域的扩大而呈现平方方式的增长。

近几年也有许多学者针对计算量大的问题提出了改进算法。如布朗大学运用随机采样的知识，提出先粗后精搜索、随机区域收缩和粒子滤波三种方法，有效降低了计算量^[5]；Perterson 提出混合算法，即将球面交叉、GCC-PHAT、SRP-PHAT 等算法进行综合运用；与波束成形结合提出的最大似然声源定位；万新旺提出基于主特征向量的改进算法等等^[17]。

除了可以利用改进算法来减小计算量，改进麦克风阵型也可以有效减少搜索计算量，并且提升系统的定位性能，研究发现开放阵的精度比严格阵要高，实验采用两种球型麦克风，一种开放阵（阵列悬于自由空间），一种严格阵（阵列固定于挡板），实验表明，当球型阵列半径变大时，性能会提升。

另外，该算法可用于实现多声源定位，但空间分辨率会较低，并且由于声源间存在干扰，强度较大的声源谱峰可能会掩盖弱声源的谱峰。因此如何提高分辨率并且提高阵列的抗混响和抗噪声能力都是目前主要的研究方向之一。如基于 SRP-PHAT 的空间梯度可控响应算法，从理论上定义了搜索空间中的一个固定点（搜索空间零位置），然后结合多通道声音激活检测技术完成声源定位，实验证明，该算法在实践中是很有效的等等。

2.3.2.2 SRP 声源定位算法

2000 年，布朗大学提出相位变换加权的可控响应功率（SRP-PHAT: Steered Response Power using Phase Transform），这是一种基于可控波束思想提出的声源定位算法。该算法的基本思想是将所研究的空间场按照所设定的分辨率划分为一个个网格，每个网格都可能是所研究的目标声源所在的位置，每一个网格对应一组阵列的导引时延，计算出阵列导引到每个网格点的输出功率即可控响应功率，取其中可控响应功率最大的值即为声源位置。SRP-PHAT 可以看作是所有可能的阵元对的 GCC-PHAT 之和，也可以看作 PHAT 加权的滤波一求和波束成形的输出功率^[18]。

可以看到，SRP-PHAT 结合了 PHAT 加权与可控波束成形的优点，该方法对比 GCC-PHAT，避免了时延估计中所带来的积累误差问题，并且运用了多对麦克风聚焦声源位置，在有混响和噪声的环境下它比 GCC-PHAT 法和可控波束成形法的鲁棒性更强、更具备稳定性和可靠性^[17]。缺点是计算量较大。

声源定位的性能也受搜索方式和顺序的影响。2004 年，Zetkin 等人提出提出空间分等级搜索方法，显著地减少了算法的计算量。该算法根据高频信号空间谱较尖锐而低频信号较宽的概念，首先将空间分为粗网格，将低通滤波后的音频信号输入 SRP-PHAT，找到声源位置所对应的网格，再将该网格细分，并且提高滤波器截止频率，再次运用 SRP-PHAT 算法，寻找到声源位置对应的细网格^[19]。运用这种方法，目标函数的计算次数远小于在整个空间区域均匀划分网格的传统方法。

2004 年，Mungamuru 和 Aarabi 提出了一种加权 SRP-PHAT 算法，用于大孔径麦克风阵列。由于大孔径阵的每个麦克风接收信号的信噪比和混响可能不一样，因此加大接收声源信号质量较好的麦克风信号对定位的贡献，即权重，可能会提高声源的定位效果。加权 SRP-PHAT 算法将声源信号到不同麦克风能量衰减的差异（与声源到麦克风的距离有关，即到达时间有关）以及声源和麦克风的指向性，构造不同的加权函数，再将该加权函数与每一对 GCC-PHAT 函数相乘后再累加得到可控响应功率，求得最大值^[20]。

SRP-PHAT 定位算法从操作上可大致为两个步骤：1）求目标函数；2）搜索声源位置。首先计算以可能声源位置为自变量的全部麦克风对的 GCC-PHAT 函数，将所有麦克风对的 GCC-PHAT 函数相加，作为 SRP-PHAT 算法的目标函数；在目标声源可能的

空间域进行搜索，目标函数值（可控响应功率）最大的位置就是估计的声源位置^[21]。

2.3.3 利用 SRP 和 SRCP 生成声谱图

根据上述对于声源定位算法的调研研究，确定本项目的算法流程，根据基于可控波束算法 SRP 及其改进算法 SRCP 实现基于麦克风阵列的声源定位，并生成声学图谱。这一节将具体阐述基于 SRP 算法创建声学图像的公式推导过程，并给出其改进算法——基于相干功率而不是总功率来创建声学图像的 SRCP 算法。该推导过程包括部分白化操作：对数据进行初步白化处理的目的是去除各观测信号之间的相关性，一般直接获得的数据之间都存在某种程度的相关性，而白化操作可以简化后续独立分量的提取过程。

具体推导过程如下：

(1)考虑麦克风个数为 P 个的麦克风阵列，其中矢量 \mathbf{r}_p 表示第 P 个麦克风的位置，其中空间域 FOV（field of view）内的声源个数为 K 个，一个 SRP 像素值是基于在有限时间帧 ΔL 上的声音事件。由第 P 个麦克风接收的波形可以写成：

$$u_p(t; \mathbf{r}_p) = \sum_{k=1}^K \int_{-\infty}^{\infty} n_k(t - \tau) d\tau \quad (2-1)$$

$n_k(t)$ —— 位于 \mathbf{r}_k 处的声源信号；

K —— 对于第 P 个麦克风接收到的声源的数目；

τ —— 声源到麦克风的传输时延。

在频域中给出了单个时间帧内，单个信道的信号：

$$\hat{U}_p(\omega; \Delta l) = \sum_{k=1}^K \hat{N}_k(\omega) \exp(-j\omega\tau_{kp}) \quad (2-2)$$

$\hat{N}_k(\omega)$ —— 声源信号在 ΔL 上的傅立叶变换。

(2)白化过程。通过 β 参数赋给每个麦克风信号，控制白化的程度，其中 β 可以在区间 $[0, 1]$ 上选择以实现不同程度的白化，其中 β 等于零表示没有白化， β 等于 1 表示全部白化，其他值导致部分白化：

$$\hat{V}_p(\omega; l) = \frac{\hat{U}_p(\omega; \Delta l)}{|\hat{U}_p(\omega; \Delta l)|^\beta} \quad (2-3)$$

β —— 白化系数；

从而得到经过白化处理的声源信号 V 如下：

$$\mathbf{V} = [\hat{V}_1(\omega; l), \hat{V}_2(\omega; l), \dots, \hat{V}_P(\omega; l)]^T \quad (2-4)$$

包含每个麦克风通道的信号信息，并以列矩阵形式表现。

(3)利用波束成形原理，获得导向向量 \mathbf{B}_i 。导向向量 \mathbf{B}_i 是声源定位算法的关键函数，也是生成声谱图的关键步骤。它是基于波束成形算法实现的，具体在该项目中，是由延迟-加权函数实现，基本过程分为两步：第一步，延迟。不同像素点达到每个麦克风所需的时间都不同，通过计算不同麦克风之间的时延，以距离声源最远的麦克风为零点，平移其余通道的信号，达到对齐信号的目的；第二步，加权。本项目中，赋予距离声源最近的麦克风最大的权重。在 MATLAB 中，分别对应于 `delayint` 和 `arweight` 函数。

它的具体表达为：

$$\mathbf{B}_i = [\hat{B}_{i1}, \hat{B}_{i2}, \dots, \hat{B}_{iP}] \quad (2-5)$$

\hat{B}_{ip} —— 小标表示对应于 \mathbf{r}_p 处的麦克风和 \mathbf{r}_i 处的像素点，表示每个麦克风此时对于该像素点的引导函数，相当于源信号数据的一个修改系数。

(4)获得每个像素点对应的可控响应功率 SRP。白化处理后的信号经过引导函数延迟-加权后，通过波束成形算法完成了声源定位，获得了每个像素点对应的可控功率响应值，由此获得了空间域 FOV 内的声场分布矩阵。 \mathbf{r}_i 表示像素点的位置，该像素点位置对应的 SRP 像素值在时间帧 L 内计算得到的信号功率如下：（其中上标 H 表示复共轭转置。）

$$S(\mathbf{r}_i, l) = \int \mathbf{B}_i \mathbf{V}(\omega, l) \mathbf{V}^H(\omega, l) \mathbf{B}_i^H d\omega \quad (2-6)$$

(5)改进的 SRCP 算法。在上述公式中，被积函数部分的结果是总功率。由于每个单独的麦克风的功率值不提供与源位置相关的信息，只有两个不同的麦克风之间互相关联的功率才提供了声源定位信息。所以每个麦克风的功率值，相当于自相关的部分，可以被去除而不会丢失空间位置的信息。这种功率的去除对于这里提出的技术是至关重要的，可以简单地计算波束形成的信号中的功率，更加高效。具体的做法是，在典型的 SRP 算法中减去每个单独的麦克风的功率。由此得到的相干功率的结果如下。

$$S_c(\mathbf{r}_i, l) = S(\mathbf{r}_i, l) - \sum_{p=1}^P \int |\hat{B}_{ip} \hat{V}_p(\omega; \Delta l)|^2 d\omega \quad (2-7)$$

至此，得到在 FOV 中的一系列网格点（即像素点）上的相干功率值，以此矩阵作为 SRCP 图像声谱图的像素值，进而得到声谱图。在 MATLAB 中，具体是通过 `imagesc` 函数实现的，结果如下图所示，具体分析将会在第五章详述。因此应用 SRCP 图谱，可

以进行声源定位、声场分析等一系列声学应用，如下图 2-3 中展示了又 7 麦克风规则阵列获得的声谱图。（具体 SRP 和 SRCP 算法对于结果的影响详见下文 5.1.2.3）

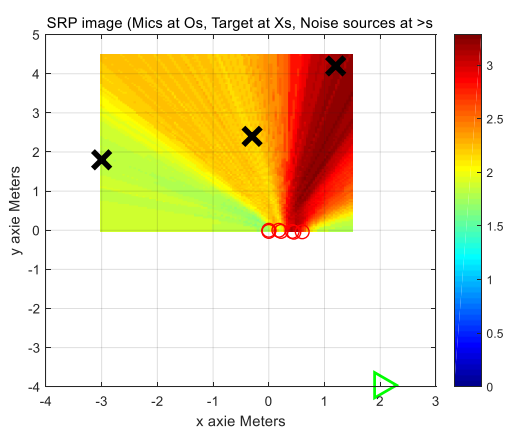


图 2-3 7 麦克风规则阵列获得的声谱图

3 高速声学照相机

3.1 高速运动物体声源定位问题

高速运动场景下的物体，区别于静态和日常环境，会出现更多不同的因素影响声源定位和声场分析。尤其现阶段飞机设计飞机噪声问题成为了重点关注的研究问题，根据美国联邦航空局的噪声适航条例，飞机噪声的重要性与飞行性能重要性同等重要。对于例如火车、汽车、飞机、高速铁路等大型机械运动设备，近场环境进行声源定位测量几乎是不可能的，因此将研究环境设置为远场条件是非常有必要的。另外，在对于高速运动中的物体各部位噪声源分布的研究中，声源的辐射噪声存在多普勒效应，如何解决和改善多普勒频移现象已成为研究高速运动物体声源定位的重点。也有许多学者进行了深入研究。刘方提出多普勒声音信号时域插值拟合校正算法，但存在对采样初始时刻的限制；徐初杰提出对水下运动目标进行多普勒测向的方法，但受限于波源做匀速圆周运动的条件等等^[22]。如下图 3-1 中，展示了多普勒效应造成的声音波形的改变。



图 3-1 多普勒效应

3.2 解多普勒算法

作为高速运动物体面临的多普勒效应问题，成为解决运动中声源定位的核心问题，针对该问题，本文进行了改进算法的研究，并提出了基于非线性插值算法的解多普勒算法，这种算法主要是补偿辐射噪声的多普勒频移和幅度衰减，进而有效消除多普勒效应对声源定位的影响^[22]，即实现解多普勒，根据测试误差分析证实了该方法有效提高了噪声源定位的准确度。

针对静止物体，可以直接采集信息，并应用 SRP 的方法获取声谱图信息，而运动声源，考虑到时变多普勒频移问题，时域的波形会被拉伸或压缩，如下图 3-2 是静止和运动声源接收信号时域图的对比^[24]：

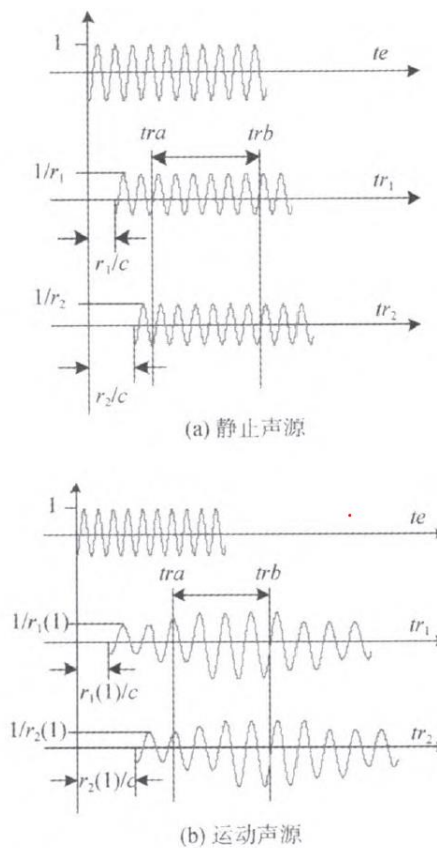


图 3-2 静止和运动声源接收信号示意图

因此这就需要根据其运动轨迹对直接接收信号解多普勒，消除多普勒效应所带来的频移和幅度增益后再利用波束成形算法声源定位，计算像素点可控响应功率，画出声谱图。解多普勒算法得到的前后结果分别是运动和静止声源信号，具体的算法推导过程如下：

首先计算声源的信号的辐射公式，下图 3-3 是运动物体 D 作直线运动过程图。

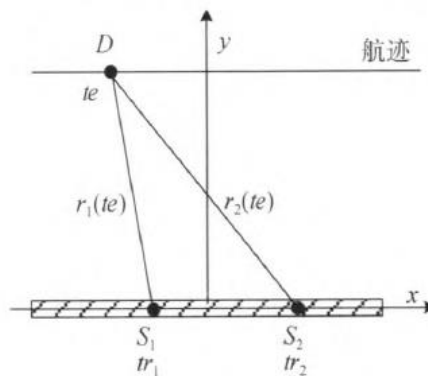


图 3-3 运动物体 D 作直线运动过程图

如上图阵元 S1, S2 是静止的麦克风阵元，用来采集声源 D 辐射的信号，声源 D 是运动声源，其中的航迹方向即 D 声源的轨迹方向，如果声源 D 的信号为 $x_e(te)$ ，那么阵元接收信号为 $x_r(tr)$ ， te, tr 分别问声源的辐射时间和阵元的接受时间，那么可以得到

阵元的接受信号：

$$x_r(tr) = x_e \left[te + \frac{r(te)}{c} \right] / r(te) \quad (3-1)$$

c ——声速；

该公式表示阵元在 tr 时刻接收到的信号，是声源在 te 时刻发出的经过了幅度衰减后的信号，其中距离矢量 $r(te)$ 随时间变化，因此接收信号的瞬时频率和幅度也在变化。

为了将接收到的运动信号通过解多普勒的方法转换成为静止情况下的声音信号，可以将式(1)进行逆过程变换，得到：

$$x_e(te) = x_r \left[tr - \frac{r(te)}{c} \right] / r(te) \quad (3-2)$$

该公式的思想在于，依照物体运动轨迹，利用逆变换，可以获得接收信号 $x_r(tr)$ 在 tr 时刻对应的声源 D 辐射出的源信号 $x_e(te)$ ，注意需补偿传播衰减。因此按照该公式获得源信号的关键在于依据运动轨迹和阵元之间的几何关系来推导 te 和 tr 之间的对应关系，：

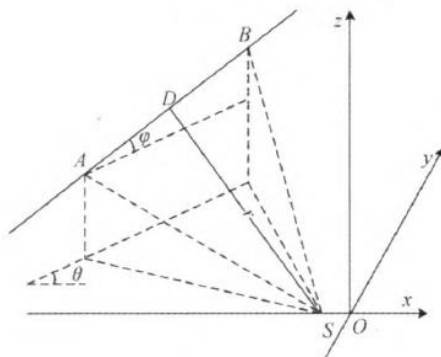


图 3-4 三维空间中的运动轨迹示意图

如上图 3-4，假设声源轨迹为 ADB，D 为声源，S 为接受麦克风阵元。已知的条件有：S 的接受时刻为 tr ，S 坐标。假设其对应的声源辐射时刻为 te ，阵元和声源之间距离为 r ，那么可以得到：

$$r = c(tr - te) \quad (3-3)$$

另外，如果已知与 D 相邻的 2 个轨迹线上的点 A 和 B，其对应的辐射时刻为 tga 和 tgb ，那么可得 AB 段矢量速度：

$$V = \frac{(B-A)}{tgb-tga} \quad (3-4)$$

则 D 坐标可以表示为 $D = A + (te - tga)V$, DS 可表示为:

$$r^2 = (D - S)(D - S)^T = te^2(VV)^T + 2(U - S)V^T te + UU^T + SS^T - 2US^T \quad (3-5)$$

其中 $U = A - tga \cdot V$ 为已知变量, 联立(3-3)、(3-5)可以得到:

$$te^2(c^2 - VV^T) + te[-2(U - S)V^T - 2c^2 tr] + c^2 tr^2 - UU^T - SS^T + 2US^T = 0 \quad (3-6)$$

$$tr^2 c^2 - 2c^2 tr + (c^2 - VV^T)te^2 - 2(U - S)V^T te - UU^T - SS^T + 2US^T = 0 \quad (3-7)$$

式(3-6)和式(3-7)是等价的, 分别是 tr 和 te 的一元二次方程, 已知其中一个即可求解出另一个。

已知了 tr 和 te 之后, 可以进入下一步插值解多普勒的运算:

如果设定所截取阵元接收到的信号, 截取时间区间为 $[tra, trb]$, 由于麦克风是等间距采样, 因此截取到的时段 $[tra, trb]$ 是等间隔时间序列, 依据式(3-6)可得到此时辐射信号对应的时间段为 $[tea, teb]$, 它是一个非等间隔的时间序列, 那么补偿传播损失后得到的信号也是非等间隔采样信号, 该信号就是解多普勒之后修正的静态信号。为改善修正后的信号为非等间隔采样信号, 采用构建 $[tea, teb]$ 内的等间隔时间序列 $te = (tea: 1/frs: teb)$ 的方法, 其中 frs 为重采样频率, 也就是解多普勒后信号的新采样率, 一般情况下, 小于接收信号采样率 fs 。同样的道理, 将 $te = (tea: 1/frs: teb)$ 带入式(3-7)就得到对应接收时刻 tr' , 这就是非线性插值拟合时间序列, 然后用最小二乘法拟合出以 $(tr, x_r(tr))$ 为插值点的插值函数, 非线性插值序列 tr' 在插值函数中对应的值就是 $x_r(tr')$, 再经过补偿传播损失后就得到了要求的 $x_e(te)$ 。还需要再添加固定传播损失以模拟静止声源发出而得到的接收信号, 得到结果:

$$x_r'(tr') = x_e(te)/r[te(1)] \quad (3-8)$$

$x_r'(tr')$ ——是解多普勒后的阵元接受到的静止声源信号, 其中:

$$tr' = te + r[te(1)]/c \quad (3-9)$$

因此本文采用非线性插值算法的解多普勒算法, 定性分析了如何消除多普勒效应对运动声源定位的影响。

4 总体系统设计与搭建

4.1 硬件系统方案设计

本项目根据确定的算法流程，进行硬件设计调研和方案确定，以实现基于麦克风阵列的多通道采集系统，并进行后期 PC 处理和声谱图的实现。主要确定了两种方案：

第一种，基于 DSP 实现的多通道采集系统，硬件系统框图如下图 4-1 所示：

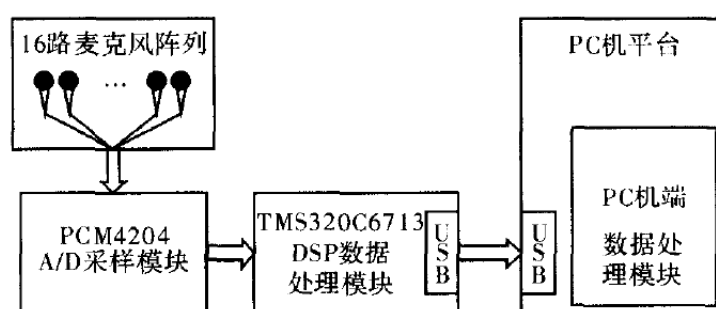


图 4-1 DSP 硬件系统设计框图

该硬件系统模块设计中:包括 16 路麦克风阵列构成的传感器采集模块、由 PCM4204、OPA1632、OPA227 构成的信号调理模块、A / D 采样模块、由 TMS320C6713 作为核心的 DSP 数据处理模块以及 PC 机^[25]。该系统的优点是：普通计算机实现通用计算，但 DSP 只负责数字信号处理（主要指音频、视频等），且再数字信号处理方面性能更强，更专注，满足本项目对于音频信号处理的要求；DSP 由于数字信号处理的专业性，因此具备强大的计算能力和运算效率；在低成本下，也可以执行复杂的信号处理工作，而无需使用昂贵的通用处理器，性价比较高；缺点是针对系统搭建而言，该系统的前置电路搭建过于复杂，需要耗费大量时间。

第二种,基于 NI 公司数据采集卡 DAQ 和 LABVIEW 实现的多通道采集系统，硬件系统框图如下图 4-2 所示：



图 4-2 NI PXI 硬件系统设计框图

该硬件系统模块设计中包括：前端传感器依然由麦克风阵列构成、数据采集模块由 NI 公司的 PXI 数据采集系统构成，处理及显示模块由 PC 机组成。该系统的优点是：NI 美国国家仪器公司在数据采集系统的研究上已经非常成熟，自 1997 年 PXI 诞生以来，

NI 一直保持着 PXI 领域的领先地位，他们设计的 PXI 数据采集系统由硬件部分：数据采集卡 PXI、机箱以及软件部分：LABVIEW 组成^[26]。该系统具备许多优点：

系统性能强大。PXIe 机箱结合 PXI 数据采集卡可以轻松实现多通道数据采集。PXI 系统中将模块化的欧式机械封装、CompactPCI 与 PCI 电气总线结合在一起，提升了同步和一些关键软件的性能^[27]。PXI 系统为机器监测、自动化、生产测试和工业测试等应用提供技术支持。其中再 PXI 中新引入的 PXI Express 技术，可以集成到背板中、可以显著提高总线带宽同时维持与现有系统的后向兼容性。PXI Express 将可用带宽提高了 45 倍多，从 132MB/s 提高到 6GB/s。并且维持与 PXI 模块间的软件、硬件兼容性。利用 NI 的数据采集系统可以实现多通道的实时同步数据采集功能。另外，NI 的 PXI 数据采集系统是一款可以与软件高度匹配结合使用的便捷操作系统，且操作简单，无需复杂编程语句，操作界面友好：利用 NIMAX 监控软件可以高效率地监控采集系统和采集卡状态，；利用 LABVIEW 图形化编程语言及 DAQ 助手函数包可以便捷地实现数据采集，只需点击开始运行，避免了使用程序化语言的复杂性，人机交互界面友好，便于学习入手。

对比上述两种设计方案，结合项目设计要求，可以得到结论：DSP 设计系统专注于数据处理功能，数据处理性能强大，性价比高，低成本，便携性好，但其前置电路制作复杂，耗时，后期软件数据采集接受部分需要运用复杂程序编程，操作复杂，并且 DSP 性能有限，后期数据处理，扩展系统功能都会受限；基于 NI 的 PXI 系统设计虽然成本较高，但系统性能完善，能够满足实时、高效、同步采集数据的基本设计要求，此系统满足多通道数据采集，可以方便扩展数据采集通道，并且采集操作简单易懂，方便实验者使用，人机界面友好，操作环境简洁，结合后期 LABVIEW 数据采集处理，可以方便实现数据预处理、存储、导出等功能，G 语言编程环境，避免了传统编程语言的复杂性，另外，LABVIEW 可以结合软件包 mathscript 和 LabWindows/CVI 可以方便地实现与 MATLAB 和 C 的互通，便于后期扩展系统功能。综合上述比较评价，最终本项目选择了基于 NI 数据采集系统的硬件设计方案。

4.2 总体设计框图及硬件选型

4.2.1 总体设计框图

由上述硬件设计得到总体设计框图如下图 4-3 所示：

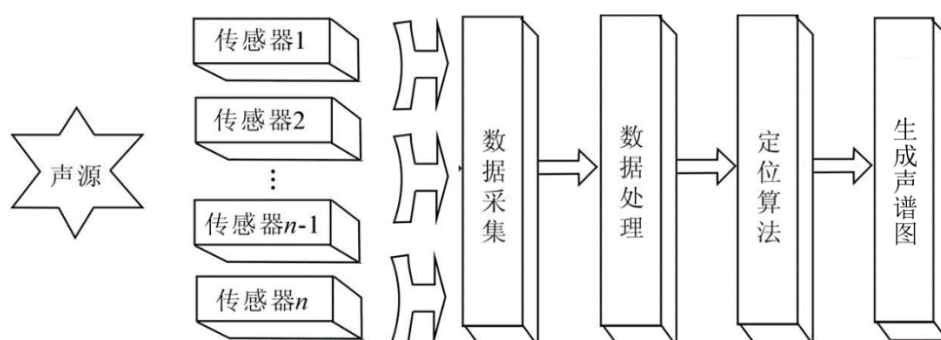


图 4-3 总体设计框图

该系统由硬件和软件组成。硬件部分：麦克风传感器阵列、NI 采集系统（NI PXIe 机箱和 NIPXI 数据采集卡）；软件部分：LABVIEW 数据采集存储模块、MATLAB 定位算法实现、MATLAB 生成声谱图。

4.2.2 硬件设计与选型

4.2.2.1 麦克风阵列选型

麦克风的选择要依据以下几个方面：首先是麦克风的基本性能，根据实验目的分析，应该选择灵敏度较高、全向型收音的麦克风，为便于后期搭建麦克风阵列还需要单个麦克风较为轻便；其次是麦克风阵列的设计和搭建，上文提到麦克风阵列对于结果将产生直接的影响，麦克风越多，结果越精确，另外阵列的形状也会影响结果。

本项目采用百灵达（Behringer）ECM8000 全指向性电容话筒，该麦克风具有以下特征：平滑的频响曲线（15 赫兹 - 20 千赫兹）、驻极体电容技术、供电（+15 伏至 +48 伏、全指向性。具体参数如下：阻抗 600 欧，灵敏度 -60dB，带宽 15Hz-20kHz，实物图如下图 4-4 所示。



图 4-4 百灵达 ECM8000 全指向性电容话筒

麦克风阵列，本项目主要设计了两种规则麦克风阵列，分别为麦克风间距较大的规则分布 7 麦克风阵列以及间距较小的 7 麦克风阵列，阵列的 3 维分布图如下图 4-5 和图 4-6 所示：

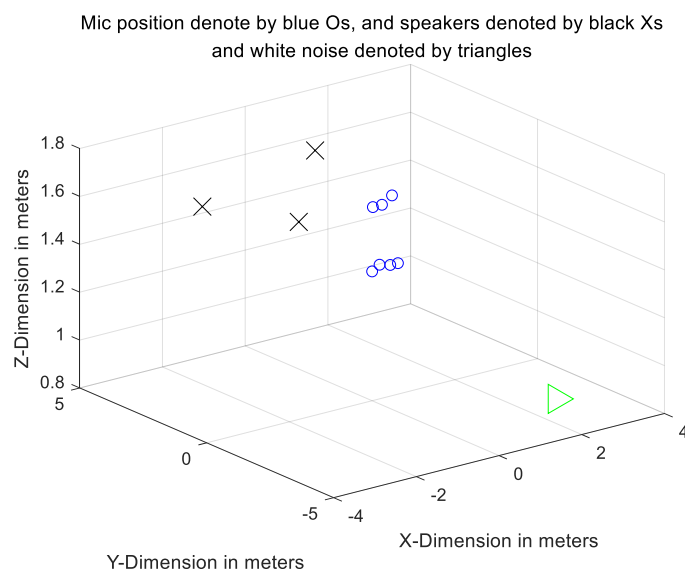


图 4-5 7 径向尺寸小的麦克风阵列分布

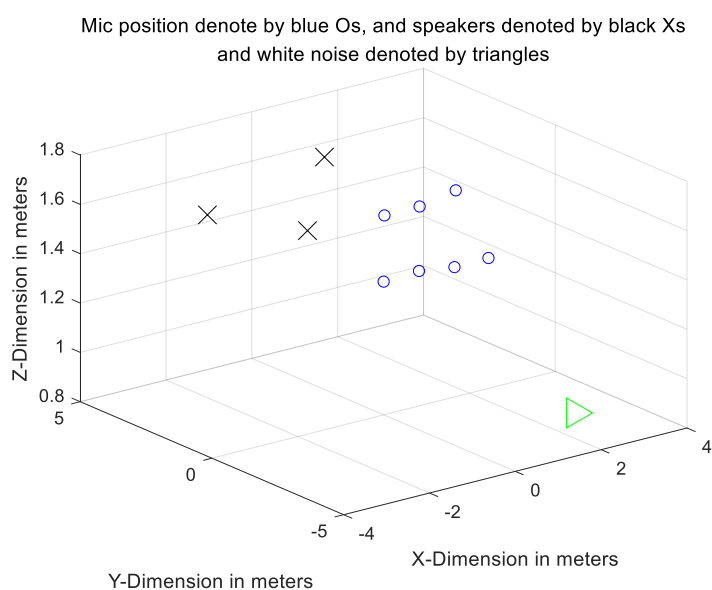


图 4-6 7 径向尺寸大的麦克风阵列分布

上两图中，圈表示麦克风阵元位置，叉表示声源位置，三角表示白噪声位置。

4.2.2.2 NI 采集系统选型：

根据本项目的设计要求，最终选择 NI 美国国家仪器公司的 PXIe1065 机箱以及 PXI 4461 和 PXI 4462 两种数据采集卡，其具体技术参数如下：

PXIe-1065 机箱是一个 18 卡槽的机箱，它包括：1 个 PXI Express 控制器插槽、9

个 PXI-1 槽、4 个 PXI 混合插槽、3 个 PXI Express 槽 和 1 个 定时插槽，具体插槽位置如下图 4-7 所示：

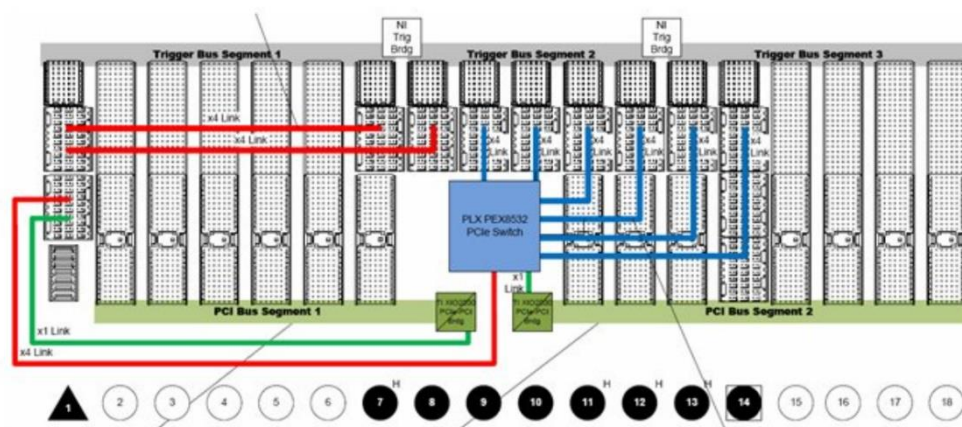


图 4-7 PXIe-1065 机箱

该机箱具备以下功能：由 PXI 总线可实现各个不同的数据采集卡之间的同步，经过实验验证，各板卡之间的数据是同步的；该机箱可提供大约 800 MB/s 上行（输入板卡执行写内存操作）带宽和 400 MB/s 的下行带宽，性能强，满足实验要求。两个独立的 PCI 总线段提供增强型的 PCI 吞吐量(相比较是其他 18 槽机箱的两倍)；机箱本身安装了 WIN7 操作系统，兼备计算机功能，具备 4 个 USB 插口，可连接无线网卡、鼠标、显示器。如下图 4-8 是 NI PXIe-1065 的实体图：

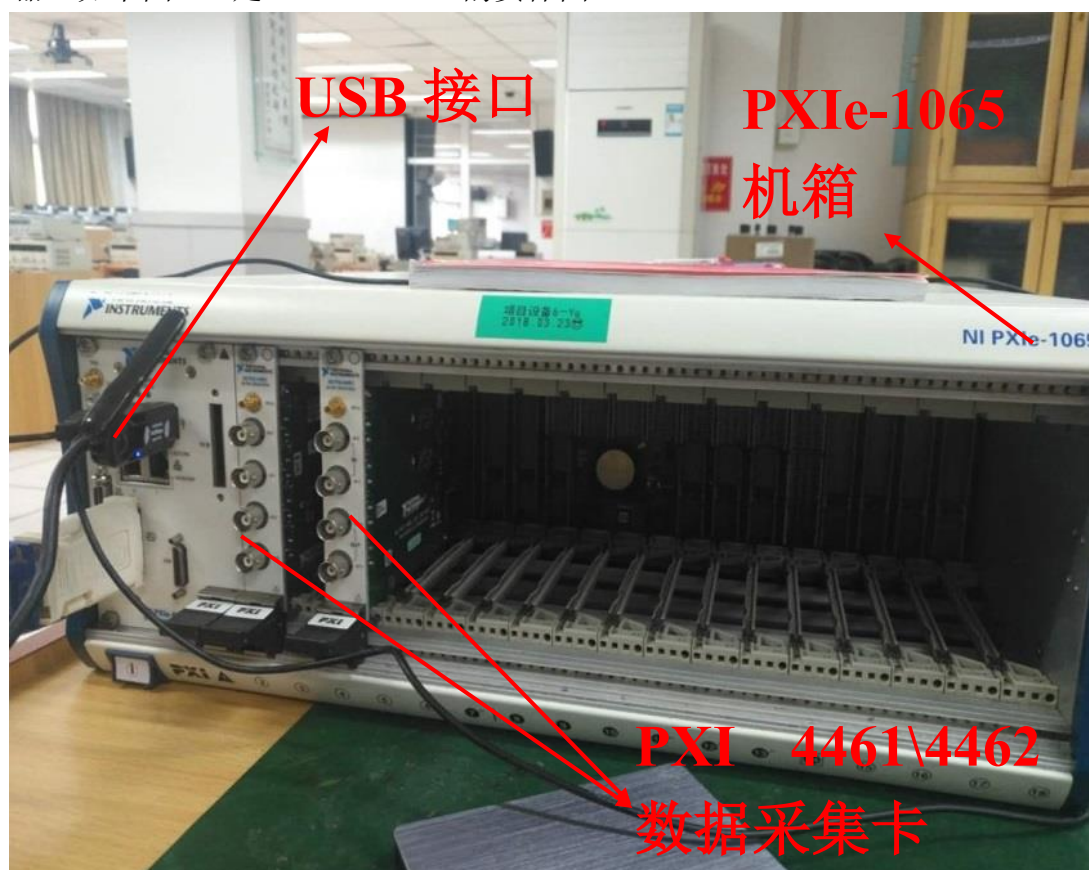


图 4-8 PXIe-1065 硬件实物图

本项目采用了两种数据采集卡，分别是 PXI 4461 和 PXI 4462，其具体参数如下：PXI 4461\4462 数据采集卡性能相似。高通滤波器截止频率 3.4Hz，最大采样率 204.8 kS/s，采用常用的 BNC 接头，支持热插拔。区别在于 PXI 4461 是两通道最大差分模拟输入，两通道模拟输出数据采集卡；PXI 4462 是四通道差分输入的数据采集卡，只具备数据输入采集功能，不具备输出功能，实物图如下图 4-9 所示：



图 4-9 PXI-4462、4461 硬件实物图

本项目中采用了两块 PXI 4461，一块 PXI 4462 以实现多通道数据采集功能，最多可实现八通道差分模拟输入信号采集功能。

接线部分采用 BNC 转卡农 25 米专用音频屏蔽线，支持远程无失真音频传输，具体实物图如下图 4-10 所示：



图 4-10 音频线接口实物图

4.2.3 NI 人机交互界面

NI 在 PXI 技术的成熟还表现在其友好的人机交互界面，当数据通过麦克风传感器接收，并经过 NIPXI 采集系统，进入计算机后，NI 公司采用 LABVIEW 进行采集控制。具体过程如下：NI PXI 采集卡采用 LABVIEW 中的采集助手（DAQmx）程序包进行数

据采集任务，采集助手程序包主要包括以下子程序：物理通道选择、创建虚拟通道、设置采样时钟、创建任务开始触发、数据读取、清除任务、简单报错子程序。实现单通道数据采集任务的程序框图如下图 4-11 所示：

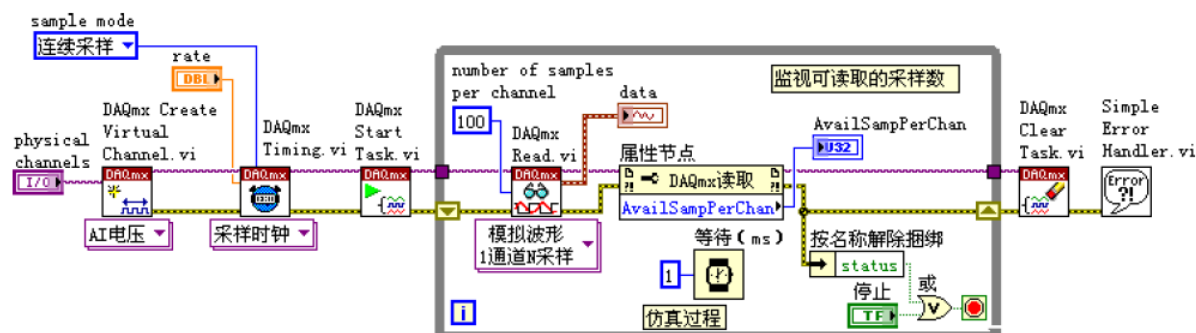


图 4-11 利用 DAQmx 实现单通道数据采集任务程序框图

然而在实际操作中，数据采集并不需要操作者一个个单独创建上述子模块来构成完整的程序框图，而是可以利用 DAQmx 程序包中的 DAQ 助手来进行数据采集，DAQ 助手实际上就是上述子程序模块创建的一种简化操作，其底层仍是上述子模块，只不过利用了人机交互界面来创建程序框图^[28]。DAQ 助手极大程度地减少了操作的复杂度，其人机界面简单易懂，非常便于理解使用。利用 DAQ 助手创建采集任务的具体过程如下：

(1) 打开 NI MAX 软件，进行数据采集卡初始化，包括自检、自校准、重置、刷新，并可以利用测试面板对数据采集卡性能进行简单测试；

(2) 打开 LABVIEW，在 Functions>>Measurement I/O>>NI DAQmx 中找到并创建 DAQ assistant；

(3) 打开 DAQ assistant，创建采集任务：选择采集信号>>模拟输入>>声压>>选择采集板卡和接口参数；

(4) 进入 DAQ 助手参数设置界面，设置采样率为 22.05KHz，采样点数为 44.1KHz，采样模式为连续采样；

(5) 创建波形显示器；

(6) 进行数据存储，在 Functions>>Express>>Output 中找到并创建 Write to Measurement File，选择存储方式.lvm（LABVIEW DAQ 助手也支持.xlsx，TDMS，TDM 形式的存储方式，经过实验.lvm 文件形式实现较为方便，功能基本满足要求），和存储位置。完整的程序框图如下图 4-12 所示：

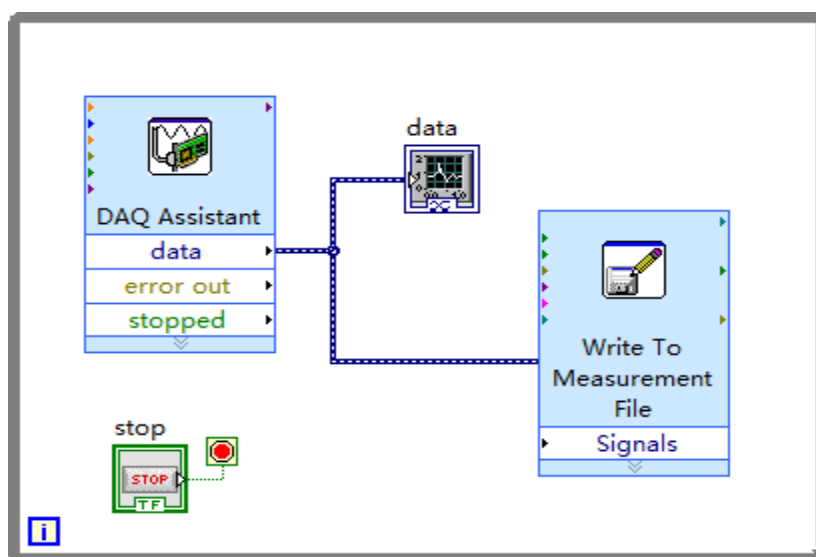


图 4-12 利用 DAQ 助手进行数据采集和存储

（7）回到控制面板点击开始进行多通道数据采集实验。具体完整的程序框图和采集到的实验波形如下图 4-13 所示：

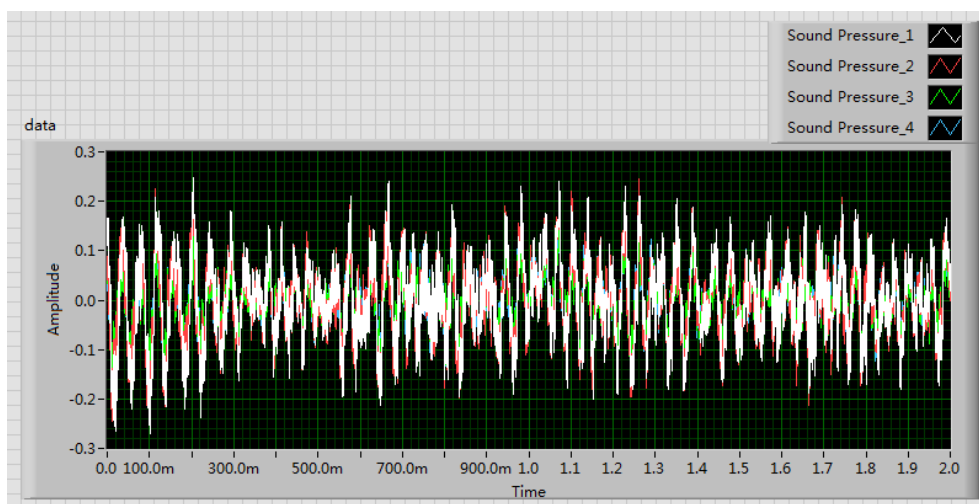


图 4-13 四通道数据同步采集波形

经过实验操作证实基于 NI 采集系统的采集过程具备很多优点：采集信号效果好，失真少，可同步实时进行多通道采样，采集过程应用 DAQ 助手人机交互界面，操作简单，功能完善，可实现多通道数据存储。

5 结论与分析

本项目为实现日常场景的噪声源定位和彩色声学图谱,采用分布式麦克风阵列和 NI 数据采集器搭建多通道分布式音频采集系统,实现音频信号采集与定位,并生成表示声场分布的声谱图。在声学相机实验中,主要采用 7mic 的规则阵,利用可控响应相干功率声源定位算法,通过 MATLAB 编程进行拾音、定位、及声谱图生成。具体的实验信息和结果分析如下。

5.1 声学实验过程及分析记录

本节简述声学实验过程如下:

(1) 搭建实验环境并测量记录实验数据。搭建麦克风阵列,测量并记录各个麦克风阵元坐标、声源坐标以及实验环境数据,包括气压、温度、湿度等(这些参数用于计算环境声速,声速对实验结果也存在影响)。如下声学实验数据记录表中,清晰地展示了各种参数的数值,以及实验环境简图,便于后续数据分析。具体数据记录表如下图 5-1 所示:

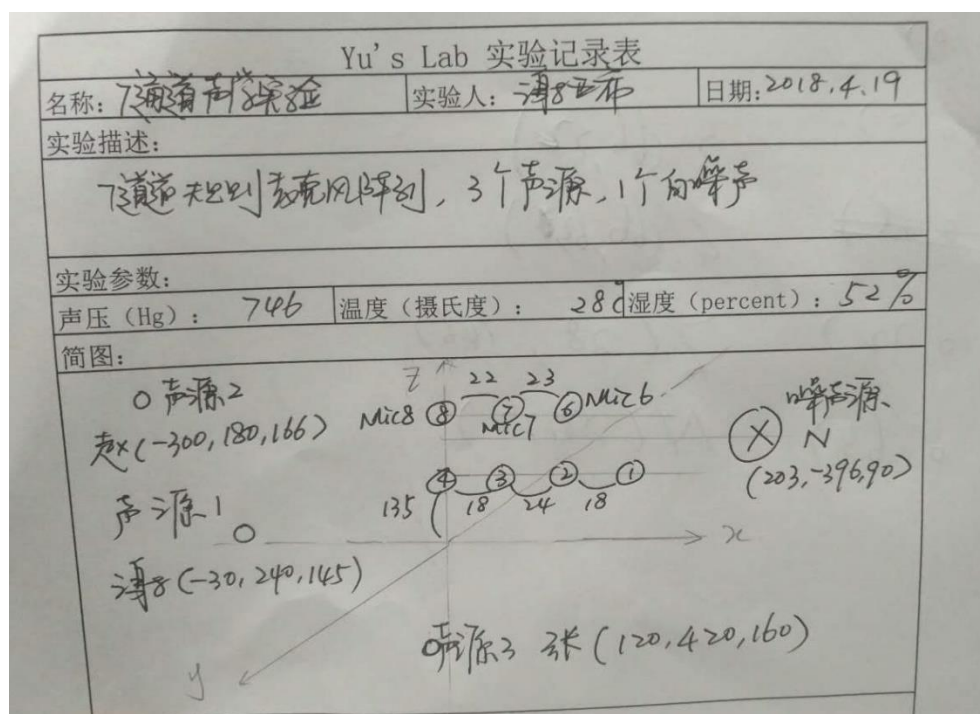


图 5-1 声学实验数据记录表 (上)

坐标:	
Mic locations:	
Mic1 (60, -4, 135)	Mic8 (0, -3, 161)
Mic2 (42, -3, 135)	
Mic3 (18, 0, 136)	
Mic4 (0, 0, 134)	
Mic5 (45, -5, 164)	
Mic7 (22, -3, 161)	
Speaker locations:	
Speaker8 (-30, 240, 145)	Speaker (120, 420, 160)
Speaker (-300, 180, 166)	
Noise locations:	
N1 (203, -396, 90)	
实验存档文件位置和名称:	
C:\Users\NZ\Desktop\xxx-test\Exprch7.lvm.	

图 5-1 声学实验数据记录表（下）

（2）进行声音信号多通道采集。利用 LABVIEW 中的 DAQ assistant，创建多通道数据采集任务，选择采集板卡和接口参数；设置采样率为 22.05KHz，采样点数为 44.1K，采样模式为连续采样；设置采集数据的存储格式为.lvm 和存储路径。LABVIEW 控制面板点击开始进行多通道数据采集实验。（详细过程请参见 4.2.3 NI 人机交互界面）。

（3）数据导入 MATLAB 工作区。将存储的.lvm 格式文件利用 MATLAB 中自带的 lvm_import 函数读取成它支持的.mat 和.dat 格式，读取的结果显示如下图 5-2 所示：

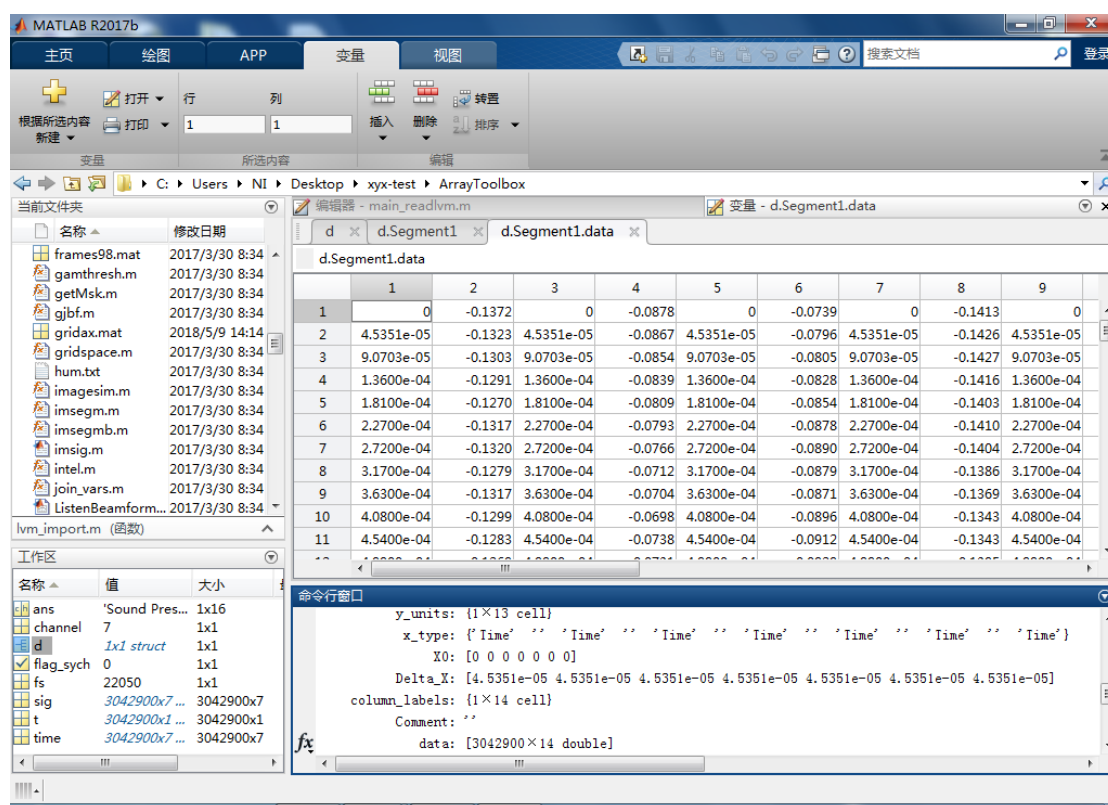


图 5-2 MATLAB 中的 7 通道音频数据读取

在 d.Segment1.data 中第 1, 3, 5, 7, 9, 11, 13 列为采样时间；第 2, 4, 6, 8, 10, 12, 14 列为 7 通道各自采集到的音频数据，可以看到数据是按照每通道一列的形式按时间存储的，并且该数据时间采样点一致，证明了 PXI 多通道数据采集是同步的；

(4) 利用编程在 MATLAB 中进行声谱图的生成和声源定位。MATLAB 处理数据的程序框图如下图 5-3 所示：

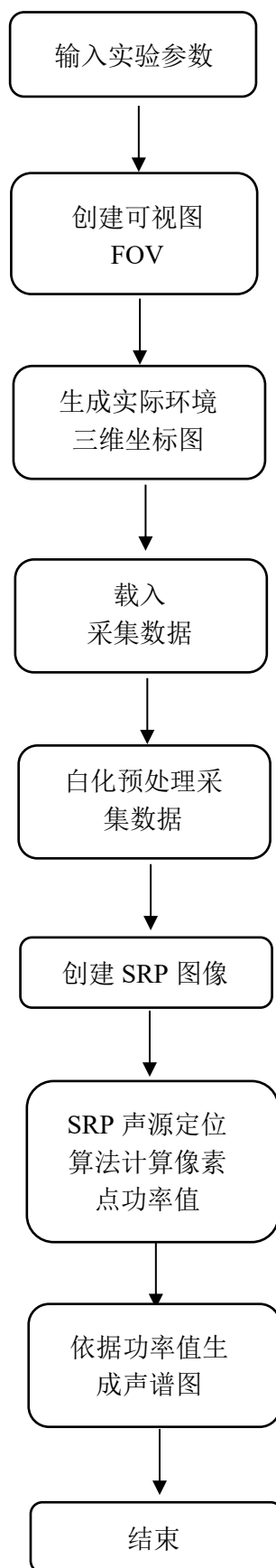


图 5-3 MATLAB 声学呈像程序框图

显示部分结果如下图 5-4 所示：

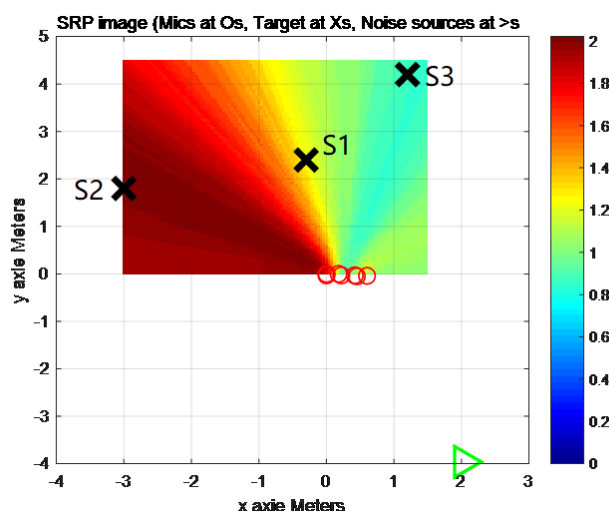


图 5-4 麦克风规则阵列声场分布顶视图

上图表示 7 麦克风规则阵列，在 3 声源环境下的声谱图。其中红圈表示 7 麦克风的实际坐标（由于是顶视图，因此部分麦克风位置重叠）；黑叉表示三个声源的实际坐标分别为 S1、S2、S3，声源为人声语音信号，该时刻仅 S2 位置的人在说话，其他位置不发声；三角表示白噪的实际坐标。颜色从蓝到红代表数值的增大，数值越大的点，可控响应相干功率值越大，即为声源位置。因此声谱图中颜色越深，越偏红色的区域，代表距离声源位置越近。该声谱图中，此刻，在 S2 附近的像素点颜色较深，表示 S2 此刻正在说话，而 S1、S3 颜色较浅，表示没有说话，与实际相符。验证了该实验可以实现声源定位的彩色声谱图的生成。

5.2 结果与分析

本节主要进行了物体内部噪声源定位和人声定位两个实验（实验具体过程详见 5.1），并进行了结果分析比较，误差分析。上节中提到，声谱图中颜色较深的区域代表声源所在区域，由此可以在声谱图中观察系统声源定位的能力。在 SRP 和 SRCP 算法中，可控响应功率最大的点，即为声源位置，由此可以得到实验声源位置坐标，与测量声源位置坐标进行比较可以得到定位误差，可以对系统的定位性能进行评估。

5.2.1 物体内部噪声源定位

物体内部噪声源定位实验中仍然采用 7 麦克风不规则环绕型阵列，对工控机箱内部的噪声源（风扇）进行声源定位，实验采用的麦克风阵列二维分布如下图 5-1，以及机箱及麦克风阵列实验实物图如下图 5-2 所示：

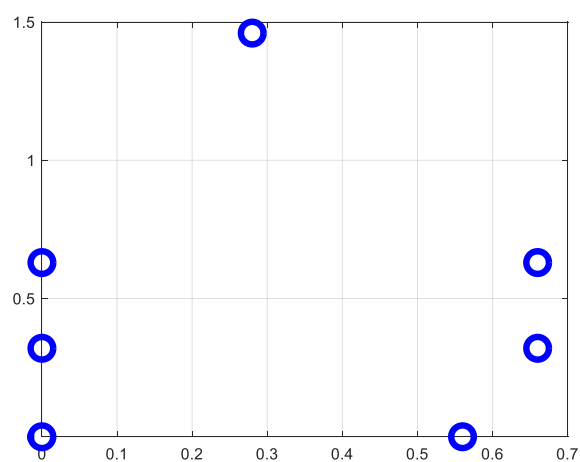


图 5-17 麦克风不规则阵列二维顶视图



图 5-2 机箱及麦克风阵列实验实物图

可以得到机箱内部噪声源定位结果如下图 5-3 所示，其中红圈代表麦克风阵元，三角表示测量的实际风扇位置，黑叉的位置表示实验得到的声源位置，有红色区域即噪声源位置，计算的平均误差为 0.3 米左右，实验证明系统噪声源定位性能良好。

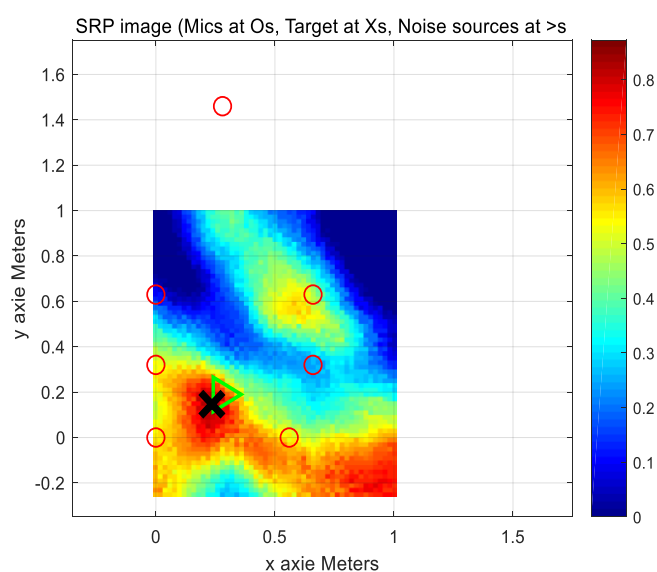


图 5-3 机箱内部噪声源定位声谱图 $t=[8,8.5]$ s

5.2.2 人声定位：

人声定位实验中主要采用 7 麦克风规则阵列，3 声源（声源采用不同性别的人声），1 白噪声点进行。本节根据不同参数对声谱图效果的影响进行了研究，并对声源定位的误差进行了分析比较。实验阵列及坐标图如下图 5-5 和图 5-6 所示：



图 5-5 7 麦克风规则阵列实物图

Mic position denote by blue Os, and speakers denoted by black Xs
and white noise denoted by triangles

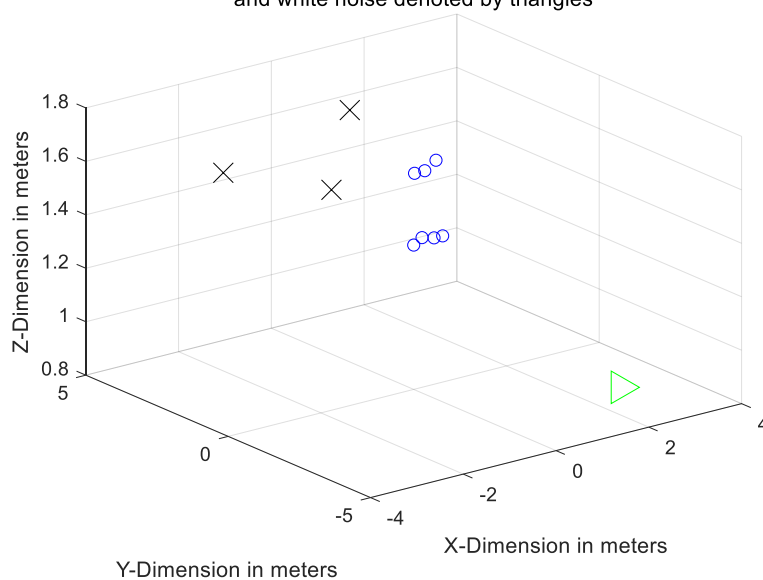


图 5-6 实验三维坐标分布

上图中，圆圈表示麦克风阵元位置，叉表示声源位置，三角表示白噪声位置。

5.2.3 SRP 和 SRCP 算法影响：

对于上文中提到的 SRP 和 SRCP 算法对于声谱图效果的影响进行了对比分析，本项目通过进行了 3 组物体内部噪声源定位实验，来分析 SRP 和 SRCP 的性能对比分析，分别是：基于 3 麦克风线型阵列、7 麦克风线型阵列，以及 7 麦克风环绕型阵列。如下图 5-7 所示是上述的 7 麦克风环绕型阵列对于物体内部噪声源定位的实验结果：

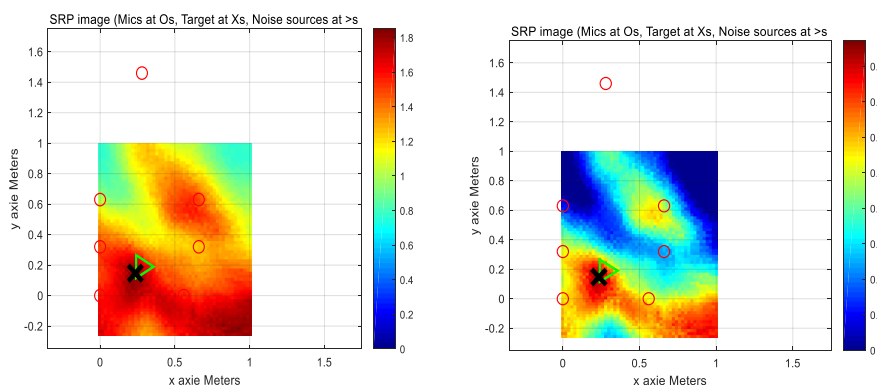


图 5-7 SRP 图像（左）和 SRCP 图像（右）

由图中可以观察到，SRP 和 SRCP 算法对于声源定位并没有不同，这可能是由于 SRCP 算法是将每个麦克风与自己相关的信号进行了消除，仅保留了与其他麦克风之间的不同成分，声源定位的结果仅与麦克风之间的不同成分有关，而 SRCP 算法并没有改变这一部分的结果，因此声源定位结果没有改变。

但从对比图中，可以看到，SRCP 声谱图的效果比 SRP 更好，是因为 SRCP 计算功率结果在 $[0,0.8]$ 之间，SRP 在 $[1,1.8]$ 之间，对于同样的颜色尺度，SRCP，显示出的颜色差别会更明显。理论上，是由于 SRCP 只造成了功率值的整体下降，而各个像素点间的差别并没有改变，因此声谱图的效果比较之下，SRCP 会更好。

7 麦克风线型阵列得到相似的结果：

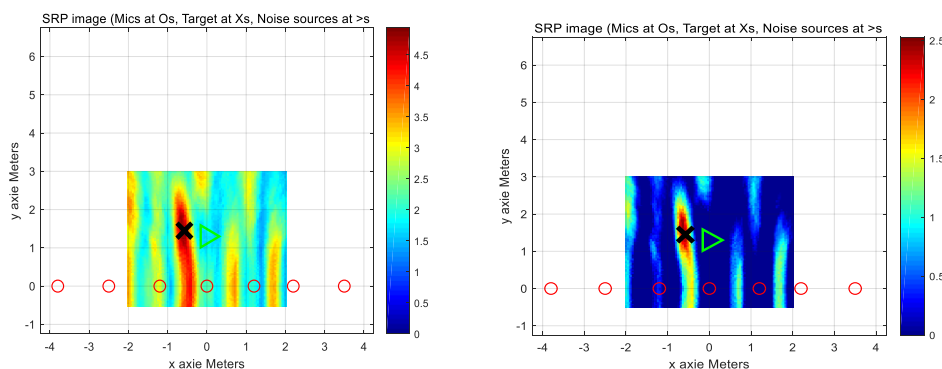


图 5-8 SRP 图像（左）和 SRCP 图像（右）

同样人声定位，也可以得到类似的结论：

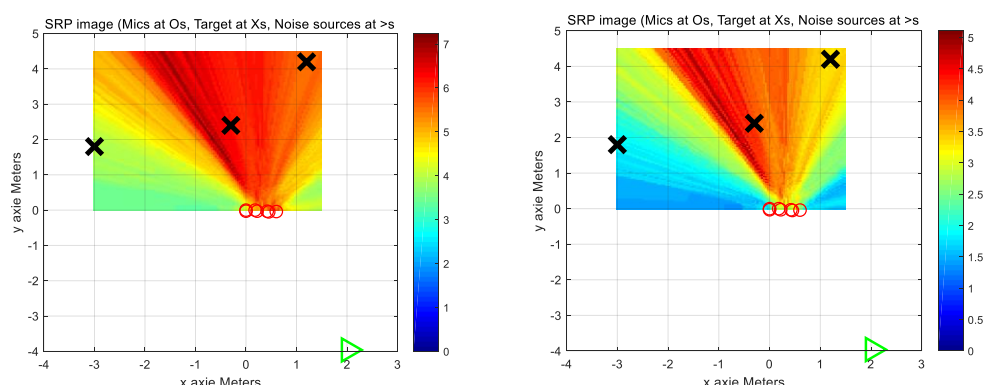


图 5-9 SRP 图像（左）和 SRCP 图像（右）

另外，对于 SRCP 和 SRP 运算时间也进行了实验比较，结果如图 5-10 所示：

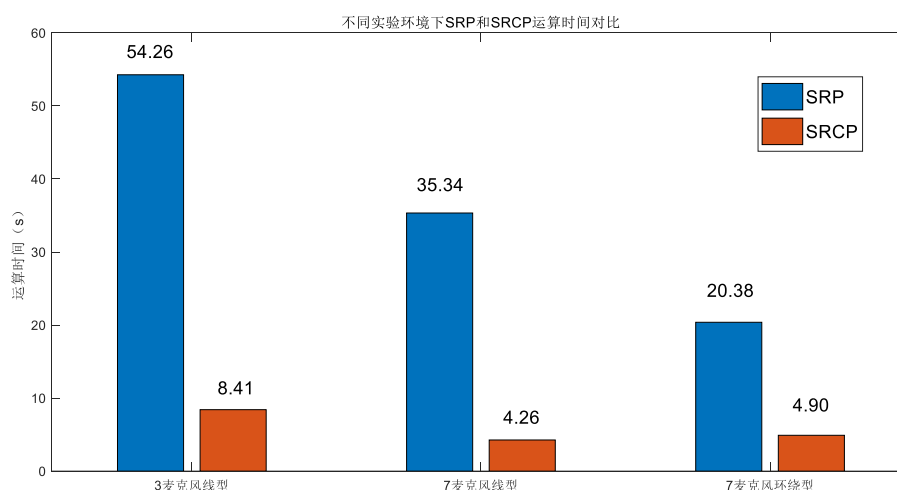


图 5-10 SRP 图像（左）和 SRCP 图像（右）

分析得到，对于不同的实验场景，SRCP 的运算能力都大大地高于 SRP，这是由于 SRCP 去除了自相关的部分，大大地减少了运算量。

5.2.4 麦克风阵列影响：

麦克风阵列也会影响定位的精确度，因此本文对麦克风阵列对结果的影响也进行了比较。实验具体采用上文提及的，针对工控机箱内部的噪声源（风扇）声源定位实验，对于 3 麦克风线型阵列、7 麦克风线型阵列和 7 麦克风环绕型阵列，进行了对比分析并得出如下结论：

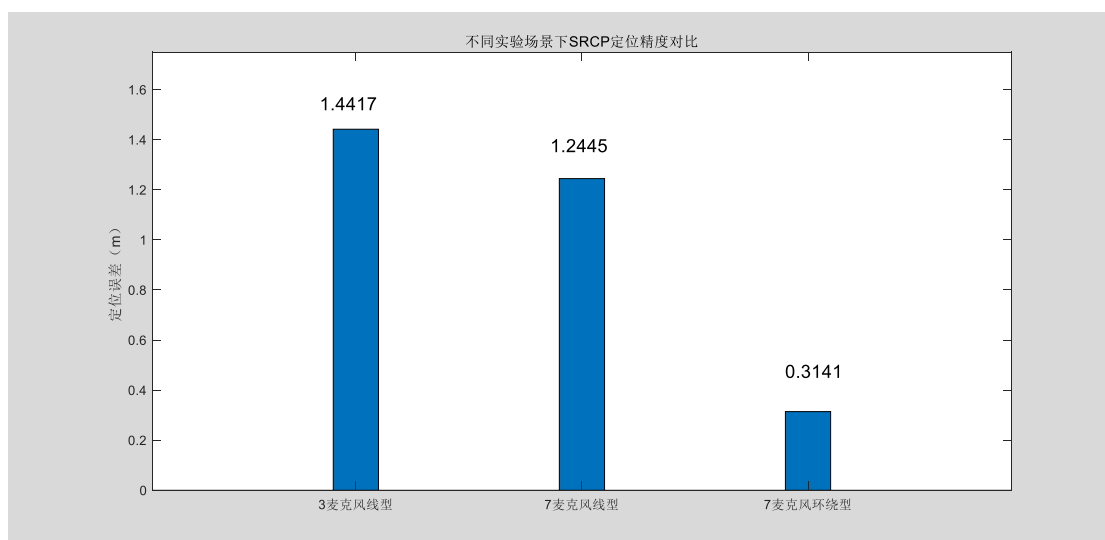


图 5-11 不同场景下 SRCP 定位精度对比

由实验可得，麦克风阵元数量越多，定位误差越小，环绕型麦克风阵列比较线型麦克风阵列效果更好。

不同麦克风阵列产生的声谱图如下图所示：

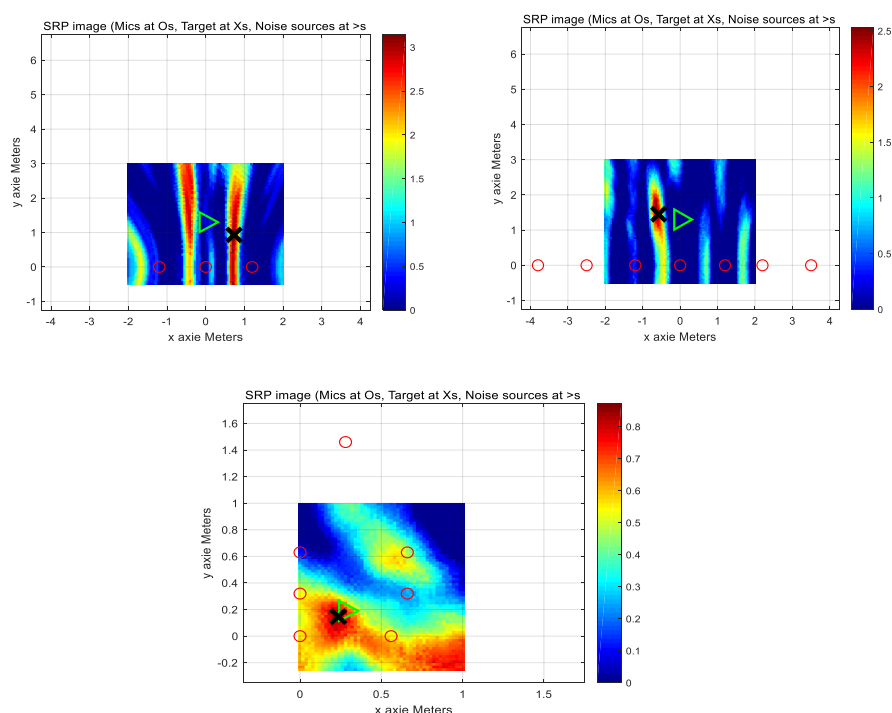


图 5-12 白化系数对比声谱图

如上二维声谱图 5-12 中，3 麦克风线型阵列（上左）、7 麦克风线型阵列（上右）和 7 麦克风环绕型阵列（下），因此根据实验可以得到白化系数 $Betar$ 越高，锐化效果的就会越明显，声谱图对于细节的展示会更加清晰，效果越好。

5.2.5 处理参数影响

5.2.5.1 白化参数影响：

上文阐述了白化的基本概念：对数据进行初步白化处理的目的是去除各观测信号之间的相关性，一般直接获得的数据之间都存在某种程度的相关性，而白化操作可以简化后续独立分量的提取过程。而白化程度的不同会影响声谱图和定位效果。

首先白化（Whitening）程度不同会对声谱图效果产生影响，因此针对白化程度进行了对比实验，如下图中叉为声源位置，圆圈为麦克风位置（其中由于二维显示，部分麦克风位置显示重叠），三角为白噪声位置。调整白化程度 Betar （取值在 $[0, 1]$ 之间，0 表示完全不白化，1 表示完全白化），观察声谱图效果：

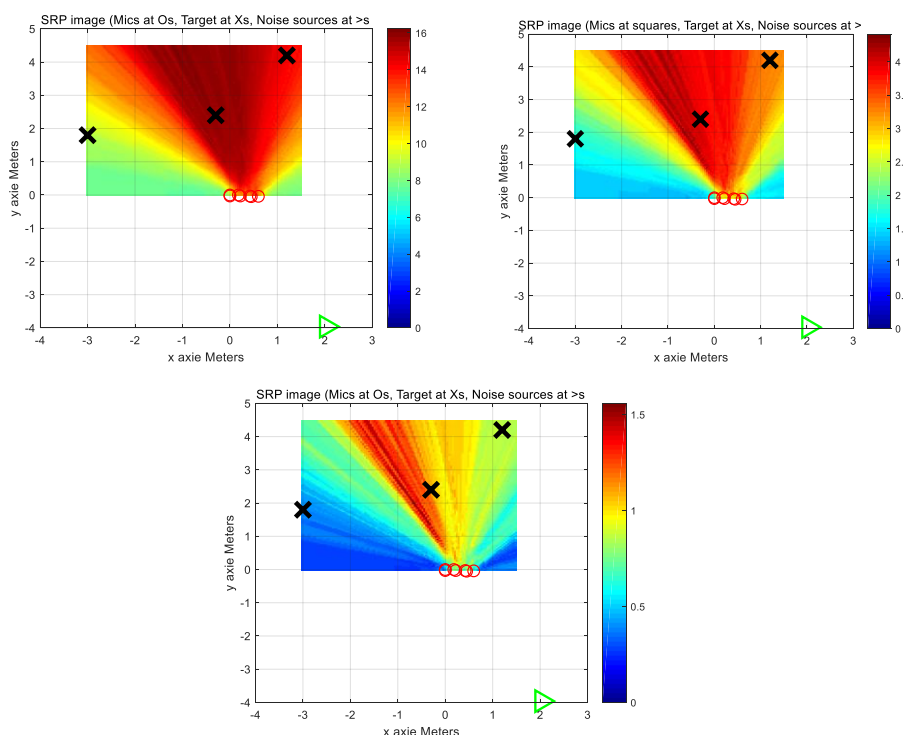


图 5-13 白化系数对比声谱图

如上二维声谱图 5-13 中，白化系数 $\text{Betar} = 0.2$ （上左）、 0.4 （上右）和 0.6 （下），因此根据实验可以得到白化系数 Betar 越高，锐化效果的就会越明显，声谱图对于细节的展示会更加清晰，效果越好。

其次，白化程度的不同对于声源定位给精确度存在影响，因此如下对于不同白化系数条件下，人声定位和物体内部噪声源定位的结果进行了比较分析，其中纵轴为相对误差。由图 5-8 可以得到，白化系数 Betar 越高，相对误差越小，说明实验结果越接近测量值，声源定位效果越好。

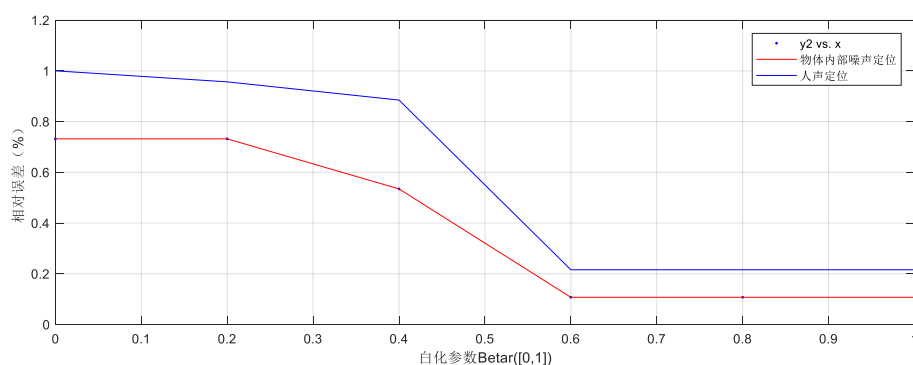


图 5-14 白化系数对声源定位给的影响

如下，针对白化系数造成的对于声源定位的影响进行进一步分析：

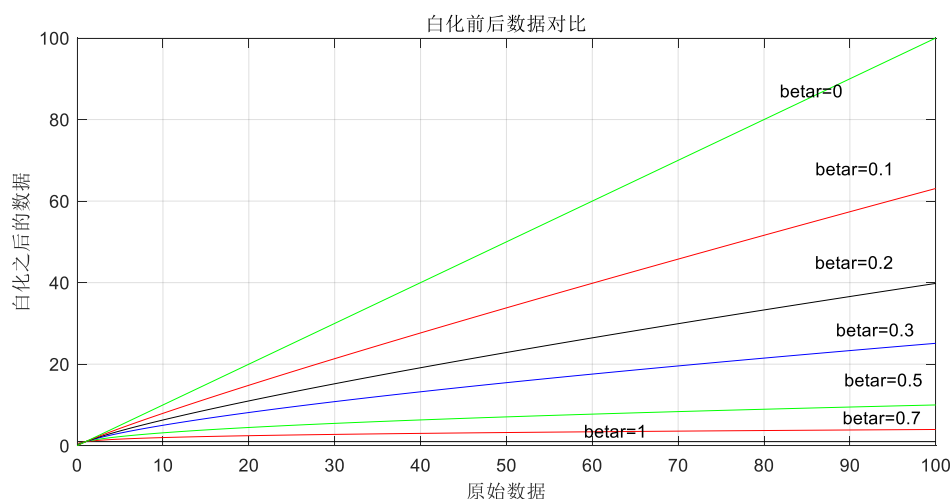


图 5-14 白化前后数据对比

上图 5-14 进行了白化前后的信号对比， betar 为 0 时，信号没有进行白化，为 1 时表示进行了全白化。由图中可以得到，白化使得信号功率普遍降低，这是由于白化抑制了部分白噪干扰。并且随着白化参数的增大，曲线趋于平滑。也就是说，白化程度越大，信号功率下降得越多，信号噪声和干扰部分得到抑制的程度也就越大，因此计算得到的信号功率越接近于真实值，所以随着白化程度的增加，声源定位的效果会越好。理论上，由于采集到的信号之间都存在不同程度的干扰，而白化预处理就是为了消除相邻像素点间的相关性，以得到更加独立的结果，这也可以从实验结果中得到证明。

另外也可以得到结论：本项目进行的物体和人声两个实验中，物体内部噪声源定位的实验结果比人声定位的实验结果相对误差普遍小，效果更好。这可能是由于物体实验中采用了环绕型麦克风阵列，而人声实验中采用了单一的行阵列。

5.2.5.2 时间窗参数影响：

在数据处理中，由于计算速度和处理工作量以及计算机存贮容量等方面的限制，实

验者常常要在实际非时限的数据中，选取有限时长的数据样本加以处理，本实验中也采取了时间窗口截取有限数据。本项目仍采用 7 麦克风规则阵列，3 人声的实验，针对时间窗大小对于声谱图效果的影响进行了实验分析，结果如下：

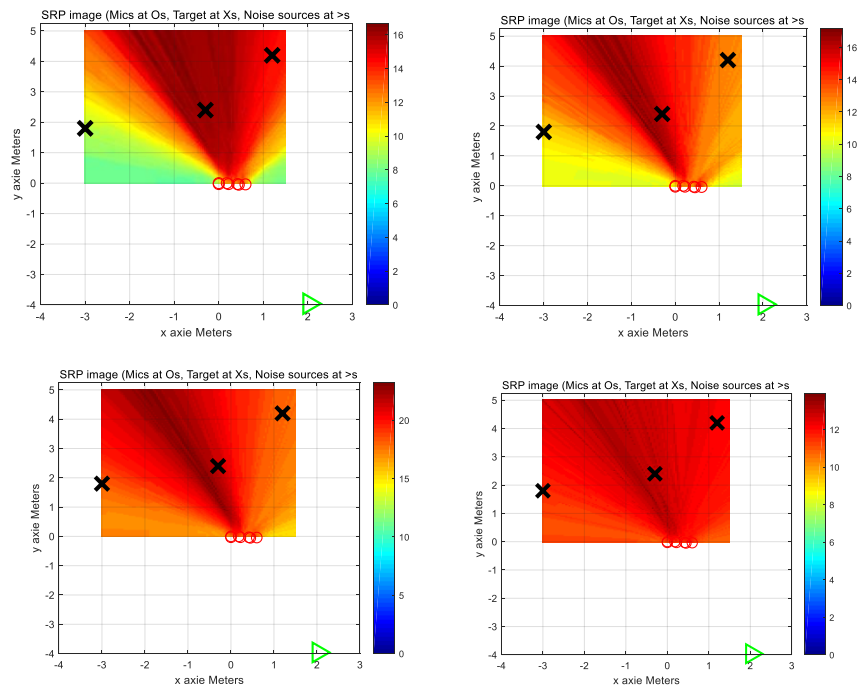


图 5-15 时间窗对比声谱图

$t_{\text{rez}}=0.02\text{s}$ （左上）、 0.06s （右上）、 0.1s （左下）、 0.3s （右下）

如上图 5-15 可以得到，时间窗越小，图像的细节越明显。这是由于在时间窗为 0.02s 的实验中，最终的像素点数据是由每通道 441 个样本数据的平均得到的结果，而在时间窗为 0.06s 的实验中，最终的像素点是由每通道 1312 个样本数据的平均得到的，对于音频信号，样本数越多，信号采样的范围可能越广，这样平均得到的结果在各个像素点之间差距不大，因此在后者的 SRP 图像中，各个像素点之间的颜色差异不大，变化没有前者明显。同理，对于 0.1s 和 0.3s 可以得到相似结论。

另外，对于不同得时间窗，系统运算耗时也不同，具体数据比较如下：

表 5-1 时间窗参数影响对比

时间窗（s）	0.02	0.04	0.06	0.08	0.1	0.3
样本数（samples）	441	882	1323	1764	2205	6615
运算时间（s）	14.96	15.03	15.37	15.91	16.2	19.18

综合上述分析可以得到，当时间窗越小，样本数据越少，可以得到的声谱图的图像越清晰，各个像素点之间的差异越大。这可能是由于，像素点的运算结果是由时间窗所截取得部分样本数据得平均得到的，因此时间窗越小，样本越少，数据会更加集中于某一时刻，因此可以刻画出此时刻的细节；时间窗越大，样本会数越多，这个时候的结果，

是一段时间内结果的均值，因此结果会越来越模糊。另外，时间窗越小，耗时越少。

本节主要进行了人声定位和物体内部噪声源定位两个实验（实验具体过程详见 5.1），并进行了结果分析比较，误差分析。上节中提到，声谱图中颜色较深的区域代表生源所在区域，由此可以在声谱图中观察系统声源定位的能力。在 SRP 和 SRCP 算法中，可控响应功率大的点，即为声源位置，由此可以精确确定声源位置坐标。

6 结论及未来工作

6.1 结论

利用 NI 数据采集系统和可控响应功率算法实现了声学照相机的基本功能。具体的实现过程如下：基于麦克风阵列，利用 NI PXI 和 LABVIEW 数据采集分析系统实现了多通道同步数据采集和存储；将存储后的数据进行白化预处理，并利用基于波束成形的可控响应功率声源定位算法在 MATLAB 中进行数据处理，计算出可视范围 FOV 中每个像素点的可控响应相干功率，并在声谱图中以类似热感图的方式显示目前的声场分布，实现声源定位；针对高速运动物体进行了解多普勒算法改进模块的设计。

SRCP 较 SRP 算法而言对于声源定位的精度没有影响，但声谱图效果更好，并且大幅度地减少了运算的时间成本；麦克风阵元数越多，定位效果越好，并且环绕型麦克风阵列比线型麦克风阵列定位效果好；白化参数较大、时间窗采样区间小，声谱图效果更好，定位误差小；定性分析了解多普勒算法的可行性。针对高速物体，本项目设计了一种理论上可行的去除多普勒频移的算法，可以针对高速运动物体的多普勒效应进行改善，但由于实验环境有限，本文仅进行了定性分析，没有进行实际验证。

本文设计的声学照相机系统应用 PXI 采集系统，数据采集效率高，可实现同步实时采样，并且利用 G 语言采样操作简便，人机交互界面友好；利用 LABVIEW 可以结合软件包 mathscript 和 LabWindows/CVI 方便地实现与 MATLAB 和 C 的互通，便于后期扩展系统功能；利用 MATLAB 处理分析数据，MATLAB 具有强大的函数库，可以高效快速地完成大数据量的分析和处理；并且实验设计算法简便，基于可控波束成形算法实现的声源定位系统信号失真小，稳定性强，并且利用计算相干功率的方法改进算法，有效减少了传统算法的计算量。

6.2 未来工作及展望

目前该系统尚存在许多待改进的问题：（1）麦克风位置测量存在误差，可以考虑研究麦克风阵列位置的校准；（2）麦克风本身存在灵敏度等方面的误差；（3）麦克风阵列的设计可以做更多尝试，由于实验条件的限制，没有进行其他球型、螺旋形麦克风阵列的尝试；（4）高速模块的设计只进行了定性分析，由于实验条件有限，系统携带不便，以及没有解决移动性电源，高速场景创建困难等问题，没有进行实验验证；（5）系统的便携性问题，目前系统还过于复杂和沉重，可以考虑简化麦克风阵列，利用小型麦克风实现系统功能；（6）进一步提高系统处理数据的效率，可以考虑利用 mathscript 和 LabWindows/CVI 实现与 MATLAB 和 C 的互通，进一步简化操作步骤。MATLAB 可以

直接提供了与 C 语言的接口，部分计算量大，用 MATLAB 实现需大量时间的程序，可以考虑用 C 语言进行，进一步提高运算速率。上述这些都是有待思考解决的问题，以及未来可以深入扩展优化系统的研究方面。论文的结论是最终的、总体的结论，不是正文中各段的小结的简单重复。结论应该准确、完整、明确、精练。如果不可能导出应有的结论，也可以没有结论而进行必要的讨论。可以在结论或讨论中提出建议、研究设想、仪器设备改进意见以及尚待解决的问题等。

参考文献

- [1] 钱世锬. 声学照相机——让我们的社区更安静[J]. 国外电子测量技术, 2009, 28(2):5-8.
- [2] GFai. 声学照相机 ---用眼睛来听画册. 北京朗德科技.
- [3] KeyCam. 声音照相机. 上海其高电子科技有限公司
- [4] Matsuo N. Microphone array: US, US6757394[P]. 2004.
- [5] 汤捷. 基于可控功率响应的声源定位快速算法研究[D]. 东南大学, 2013.
- [6] Matassoni M, Svaizer P. Efficient Time Delay Estimation based on Cross-Power Spectrum Phase[C]// Signal Processing Conference, 2006, European. IEEE, 2006:1-5.
- [7] Khanshan A H, Amindavar H, Bakhshi H. An interpolation algorithm for high resolution spectral estimation[J]. 2008:3765-3768.
- [8] 冯振宇. 基于声传感器阵列的移动目标定位技术研究[D]. 沈阳理工大学, 2013.
- [9] 王玮. 声场可视化研究[D]. 东华大学, 2011.
- [10] 杜要锋, 尹雪飞, 陈克安. 一种修正的近场声源定位时延估计方法[J]. 电声技术, 2010, 34(2):47-50.
- [11] 赵小燕, 吴镇扬. 结合激励信息和 PHAT 加权的时延估计方法[C]// 通信理论与信号处理学术年会. 2010.
- [12] LI, WanChun, WEI, et al. A robust TDOA-based location method and its performance analysis[J]. 中国科学: 辑英文版, 2009, 52(5):876-882.
- [13] 田傲. 声源定位系统中可控声源的设计与实验平台开发[D]. 河北工业大学, 2011.
- [14] 侯云山, 张新成, 金勇. 基于干扰噪声矩阵重构的自适应波束形成算法[J]. 计算机应用, 2014, 34(3):649-652.
- [15] Hoshuyama O, Sugiyama A. Robust Adaptive Beamforming[J]. IEEE Trans.acoust.speech Signal Processing, 2008, 35(10):1365-1376.
- [16] 刘真旭. 基于麦克风阵列的实时声源定位[J]. 计算机应用与软件, 2010, 27(6):43-45.
- [17] 万新旺. 基于阵列信号处理与空间听觉的声源定位算法研究[D]. 东南大学, 2011.
- [18] Dey A K, Saha S. Acoustic Beamforming: Design and Development of Steered Response Power With Phase Transformation (SRP-PHAT).[J]. 2011.
- [19] Do H, Silverman H F. SRP-PHAT methods of locating simultaneous multiple talkers using a frame of microphone array data[C]// IEEE International Conference on Acoustics Speech and Signal Processing. IEEE, 2010:125-128.
- [20] Cobos M, Marti A, Lopez J J. A Modified SRP-PHAT Functional for Robust Real-Time Sound Source Localization With Scalable Spatial Sampling[J]. IEEE Signal Processing Letters, 2010, 18(1):71-74.
- [21] 居太亮. 基于麦克风阵列的声源定位算法研究[D]. 电子科技大学, 2006.
- [22] 姜成坤, 白建斌, 陈志菲. 用于运动声源定位的解多普勒算法[J]. 水下无人系统学报, 2015, 23(3):177-182.

-
- [23] 马明卓. 基于导频的 OFDM 系统信道估计算法研究[D]. 东北大学, 2011.
- [24] 张大庆, 李翔. 确定运动物体直接反射的无线信号的多普勒频移的方法: CN107171749A[P]. 2017.
- [25] 许佳龙, 张一闻, 刘建平. 基于麦克风阵列的信号采集处理系统设计[J]. 电子科技, 2011, 24(9):124-127.
- [26] 邱正, 张永超, 沈琪.. 基于 PXI-4220 和 LabVIEW 的测力传感器数据采集系统设计[J]. 仪表技术, 2013(9):10-12.
- [27] Wang H M, Li D D, Xue P, et al. LabVIEW-based data acquisition system design[C]// 2012 international conference on measurement,information and control. 2012:689-692.
- [28] 范君乐, 王竹林. 基于 DAQmx 系统的数据采集方法[J]. 科学技术与工程, 2006, 6(16):2555-2557.

致 谢

首先要感谢我的论文指导老师北京交通大学通信工程学院的余晶晶老师。余晶晶老师对我论文的研究方向做出了指导性的意见和推荐，在过程中及时对我遇到的困难和疑惑给予悉心指点，培养我养成良好的科学研究习惯，在论文撰写过程中给予了许多指引与建议，提出了许多有益的改善性意见。我在此对余晶晶老师给予我的帮忙、关怀和教诲表示诚挚的感谢！同时，也要感谢本次项目中给予我帮助的项目组成员们，大家在百忙中帮助我完成实验，并给予了我许多帮助和启发。我在此过程也度过了一段难忘而又受益匪浅的时光。也要感谢参考文献中的作者们，正是通过他们的研究成果，使我对研究课题有了更好的思考和出发点。

最后，感谢论文评阅中辛苦工作的老师们，感谢我的朋友、家人、以及同学们，正是在他们的鼓励和支持下我才得以顺利完成此论文。论文撰写和毕业设计的过程给我带了巨大的收获，我也会以此为出发点在之后的专业生活中更加努力。

附 录

附录 A 程序代码

```
main 主函数
clear
fno = 1;
fName = 'Exprch7-1.wav'; % 读取音频文件
time_out=[38.5,39];% 选取处理信号的时间区域, s 为单位
temp = 28; % Temperature centigrade
press = 29.92; % pressure inHg
hum = 80; % humidity in percent
trez = 20e-3; % 0.02s 为一个块 Time window for frequency domain block processing 我的理解
***给频域块处理加的时间窗 s 为单位
%% 从音频文件获取参数
g = audioinfo(fName);
fs = g.SampleRate;
micnum = g.NumChannels;
% 计算实验环境声速
c = SpeedOfSound(temp,hum,press);
% 载入麦克风阵列、声源、噪声源矩阵
load('expinfo.mat');
m = expinfo.mpos; % 载入麦克风位置
sRaw = expinfo.spos; % 载入声源位置
n = expinfo.npos; % 载入噪音位置
for p=1:length(expinfo.people{1,1}) % 遍历所有人
    s.(expinfo.people{1,1}{1,p}) = sRaw(:,p); % 设置声源位置
end
% 载入各通道采集的数据
load('Exprch7-2.mat');
% Beta values for PHAT processing (PHAT processing: phase transformation 相位变换处理)
batar = .4;
%% 创建房间图像
% Opposite corner points for room, also walls are locations for noise source placement
froom = [-3.75 -3.75 0; 2.75 4.75 3.5]; % [x1, y1, z1; x2 y2 z2]
```

```

% 房间的反射系数
bs = [.8 .8 .8 .8 .7 .7];
% Field of view for reconstructing SRP image (opposite corner points)
fov = [-3 0 1.5; 1.5 4.5 1.5]';
% Room Resolution: Step through cartesian grid for mic and sound source plane
rez = .04; % In meters
% Compute grid axis for pixel of the SRP image
gridax = {[fov(1,1):rez:fov(1,2)], [fov(2,1):rez:fov(2,2)], [fov(3,1):rez:fov(3,2)]};
% 生成三维位置坐标图像
figure(fno)
plot3(m(1,:),m(2,:),m(3,:), 'bo')
hold on
plot3(s.XieYaxi(1),s.XieYaxi(2),s.XieYaxi(3),'xk', 'MarkerSize', 14)
plot3(s.ZhaoYuxiang(1),s.ZhaoYuxiang(2),s.ZhaoYuxiang(3),'xk', 'MarkerSize', 14)
plot3(s.ZhangZihao(1),s.ZhangZihao(2),s.ZhangZihao(3),'xk', 'MarkerSize', 14)
plot3(expinfo.npos(1),expinfo.npos(2),expinfo.npos(3),'g>', 'MarkerSize', 14)
hold off
hold off
%axis([min(corners(1,:)) max(corners(1,:)) min(corners(2,:)) max(corners(2,:)) min(corners(3,:))
max(corners(3,:))])
title({'Mic position denote by blue Os, and speakers denoted by black Xs', ...
      'and white noise denoted by triangles'})
xlabel('X-Dimension in meters')
ylabel('Y-Dimension in meters')
zlabel('Z-Dimension in meters')
grid on
%% 处理声源数据
% 计算选择处理数据的样本区间
sst = fs*time_out(1);
sed = fs*time_out(2);
% 白化信号
sigout = whiten(sig(sst:sed,:), batar);
%% 创建 SRP 图像
im = srpframenn(sigout, gridax, m, fs, c, trez);
%[x,y]= find(im==max(max(im)));
%x=113-x;
% Set up figure for plotting
figure(fno+1);

```

```

% Plot SRP image
imagesc(gridax{1},gridax{2},im, [0, max(max(im))]);
colormap(jet);
colorbar;
axis('xy')
axis([froom(1,1)-.25, froom(1,2)+.25, froom(2,1)-.25, froom(2,2)+.25])
hold on
plot(m(1,:),m(2,:),'or', 'MarkerSize', 10,'LineWidth', 1);
plot(s.XieYaxi(1),s.XieYaxi(2),'xk', 'MarkerSize', 18,'LineWidth', 4);
plot(s.ZhaoYuxiang(1),s.ZhaoYuxiang(2),'xk','MarkerSize', 18,'LineWidth', 4);
plot(s.ZhangZihao(1),s.ZhangZihao(2),'xk','MarkerSize', 18,'LineWidth', 4);
plot(expinfo.npos(1),expinfo.npos(2),'>g','MarkerSize', 16,'LineWidth', 2);
grid on
% Label Plot
xlabel('x axie Meters')
ylabel('y axie Meters')
title(['SRP image (Mics at Os, Target at Xs, Noise sources at >s') ])
hold off

```

srpframenn 子函数：生成 SRP\SRCP 图像

% 该函数处理来自麦克风阵列的信号，并根据声音源位置的 *引导响应相干功率(SRCP)* 算法创建声图像。

% (FOV)网格点可以在任何单位中，只要它们与下面列出的参数一致(米和秒)。

% 在 FOV 中应用了延迟和 sum beamformer，并计算了从像素位置发出的声音的相干功率。

% 相干功率与常规功率(时间的平方和归一化)不同，只有两个不同的 mics 之间的信号相互关联的功率被用于求和。

% 所以相干能量可以是负的。大的正值表明 mic 信号之间有很强的相关性或一致性。

% 一个大的负值代表一个强 180 度的相位重合(非常罕见这个数字可以超过 2 个中等大小)。

% 输出是每个像素的相干功率，表示声源的可能性值，作为空间网格的函数。

% 网格中轴的数量决定了输出可能性网格的维数。

% mleim = srpframenn(sigar, gridax, mpos, fs, c, srchwin)

% im = srpframenn(s, gridax, m, fs, prop.c, tWin);

% Inputs

% "sigar" Matrix where each column is the audio signal segment from each microphone.

% 矩阵中每一列是每个麦克风的音频信号段。

% "gridax" Cell array of grid points for x-axis, y-axis and z-axis.

% The number of elements in this cell array can be less than 3.

% If so, it will do a reduce dimension output space.

%	为 x 轴、y 轴和 z 轴的网格点单元阵列。
%	这个单元数组中的元素个数可以小于 3。
%	如果是这样，它将减少尺寸输出空间。
% "mpos"	Matrix indicating the x,y, and z cartesion positions of each mic, where the first row contains the x positions, second y positions, and third z positions.
%	If an axis is missing,only the x-y plane will be used.
%	The Dimensions should be consisent with the axes in cell array "gridax" and number of mics must be the same as columns in "sigar."
%	矩阵表示每一个麦克风的 x、y 和 z 的位置，其中第一行包含 x 位置、第二 y 位置和第三个 z 位置。
%	如果一个轴缺失，只有 x-y 平面将被使用。
%	维度应与单元阵列中的轴线一致，而 mics 的数量必须与“sigar”中的列相同。
% "fs"	Sampling frequency of the audio mic signals in "sigar"
%	“sigar”中音频 mic 信号的采样频率
% "c"	scalar is the speed of sound in the room.
%	标量是房间里声音的速度。
% "srchwin"	The window length over which coherent power is computed and represents the time resolution of acoustic image.
%	Typically, the signals in "sigar" are longer than "searchwin" so that as delays are made to each signal, actual data will be shifted into the window over which the delay and sum operation is performed.
%	计算出相干功率的窗口长度，表示声图像的时间分辨率。
%	通常情况下，“sigar”中的信号比“searchwin”要长，
%	为了当每个信号被延迟时，实际数据将被转移到执行延迟和 sum 操作的窗口中。
%	In this implementation the signal from the furthest mic is not delayed. All other signals are delayed to sync up with the max delayed signal.
%	If "sigar" is not long enough to have samples to shift into the correlation window, zero padding will occur.
%	在这个实现中，最远的麦克风发出的信号不会延迟。所有其他信号都被延迟到与最大延迟信号同步。
%	如果“sigar”不够长，无法将样本转换到相关窗口，则会发生零填充。
%	This function requires the functions: ARWEIGHTS and DELAYINT from the array toolbox to run.
%	这个函数需要函数:ARWEIGHTS 和 DELAYINT 从数组工具箱运行。
%	Written by Kevin D. Donohue (donohue@engr.uky.edu) July 2005

```

dimfov = max(size(gridax));    % Get dimensions for FOV  获得视场维度
% x-dimension
% x 维
dimx = length(gridax{1});
if dimx > 1
    xax = gridax{1}(:);
else
    xax = gridax{1}(1);
end
% y-dimension
% y 维
if dimfov >= 2    % If second dimension present  如果存在第二个维度。
    dimy = length(gridax{2});
    if dimy > 1
        yax = gridax{2}(:);
    else
        yax = gridax{2}(:);
    end
else    % If not, add a singleton to hold place  如果没有，则添加一个 singleton 来保存。
    dimy = 1;
    yax = 0;
end
% Z-axis
if dimfov == 3    % If third dimension present  如果存在第三维。
    dimz = length(gridax{3});
    if dimz > 1;
        zax = gridax{3}(:);
    else
        zax = gridax{3}(:);
    end
else    % If not, add a singleton to hold place  如果没有，则添加一个 singleton 来保存。
    dimz = 1;
    zax = 0;
end

% Obtain mic array information  获取麦克风阵列信息
[mr, mc] = size(mpos);    % Determine # number of mics = mc  确定麦克风的数量。
% Get signal dimensionality  得到信号维数

```

`[sr,sc] = size(sigar); % Determine the mic signal array size [# of samples, # of mics]` 决定麦克风信号阵列的大小

`% Test for consistency of mic and signal array` 测试 mic 和信号阵列的一致性。

`if mc ~= sc`

`error('Number of mic signal not consistent with number of mics')`

`mleim = [];`

`return`

`end`

`% extend mic coordinates to 3 dimension with zeros, if dimension less than 3`

`%如果维度小于 3，将 mic 坐标用零填满，扩展到 3 维。`

`if mr == 1`

`mpos(2:3,:) = zeros(2,mc);`

`elseif mr == 2`

`mpos(3,:) = zeros(1,mc);`

`end`

`% Find number of samples in listening window`

`%在监听窗口中找到样本数目`

`winsamp = round(sr*chwin*fs) + 1;` 四舍五入为最近的小数或整数

`temp = zeros(winsamp,sc); % 初始化信号相关矩阵,即设置全零矩阵[winsamp*sc]`

`% If listening window longer than input segment, reduce listening window to signal length.`

`% 如果监听窗口长于输入段，则减少监听窗口的长度。`

`if winsamp > sr`

`winsamp = sr;`

`end`

`% Initialize acoustic image array` 初始化声像矩阵

`mleim = zeros(dimy, dimx, dimz);`

`% Loop through every point in FOV (循环 LOOP) 遍历 FOV 中的每一个点。`

`for kz=1:dimz % Z-Dimension Loop 遍历 Z 维 (循环 Z 维)`

`for ky=1:dimy % Y-Dimension Loop 遍历 Y 维`

`for kx = 1:dimx % X-Dimension Loop 遍历 X 维`

`% Distance of FOV position from all microphones`

`% 从所有麦克风到 FOV 位置的距离。`

`ds = mpos - [xax(kx); yax(ky); zax(kz)]*ones(1,mc);`

`% Convert distances to time`

`% 把距离转换成时间`

`rst = sqrt(ds(1,:).^2 + ds(2,:).^2 + ds(3,:).^2) /c;`

```

% Create shading values to weight mic inputs as function of distance, giving closest
mic the most weight.
% 为重量 mic 输入创建阴影值作为距离的函数，给予最接近的 mic 最大重量。
at = arweights(rst); %加给每个麦克风上的权重
% Find mic with maximum delay to FOV
%找到最大延迟到 FOV 的 mic。
md = max(rst);
% Implement all other delays with respect to the furthest mic.
%在最远的麦克风上执行所有其他的延迟。
rstref = md - rst; % 每个麦克风延迟的时间 S 单位
sd = delayint(sigar,fs,rstref,sr/fs);
% Time domain sample interval shifting 时间域样本间隔移动。
% 延迟后的样本矩阵 sd
% Load up matrix with aligned mic signal corresponding to mic position and apply
shading weights
% 负载矩阵与 mic 位置相对应的 mic 信号，并应用阴影权值。
for ksig = 1:sc
    temp(:,ksig) = at(ksig)*sd(sr-winsamp+1:sr,ksig);
end
% Beamform: Sum coherent power in array 波束状:阵列中一致的功率。
dumval = sum(temp,2).^2 - (sum(temp.^2,2));
% sum(temp,2),temp 矩阵每一行总和的列向量；.^2 每个元素的平方
% dumval = (a1+a2+a3+....)^2-(a1^2+a2^2+a3^2+....)
%          .....+b3+.... - .....+b3^2+....
%          .....
%          .....
%          .....
%          .....

mleim(ky, kx, kz) = mean(dumval)*fs; % mean(dumval):dumval 向量的均值

end % End X-Dimension Loop 结束 Z 维度循环
end % End Y-Dimension Loop
end % End Z-Dimension Loop
mleim = squeeze(mleim); % Remove singleton dimensions 删除单维度

```

```
delayint 延迟子函数
function [sd, tn] = delayint(s,fs,d,ns)
% 这个函数会延迟 f 采样的 S 信号 D 秒。
% 如果在输入参数中出现 NS，输出信号的长度要与这个长度相等，将以秒为单位(通过截断或
零填充)。
% 如果不存在，输出信号的长度将根据延迟而变化(它将是原始信号的长度，加上最大的延迟
集中到下一个样本)。
% [sd, tn] = delayint(s,fs,d,ns)
%
% 延迟版本的信号 SD 开始于同一时间点(D 不应该为负值，否则会出现错误)。
% 这个版本在时间域上使用一个矢量替换来执行延迟，在这个过程中，最接近的值被舍入到
采样间隔的最近整数倍数上。
% 如果在输出中存在 TN，则为输出数组创建一个时间轴，其中 0 表示 S 的原始起始点。
% 如果 S 是一个矩阵，程序将假设每一列是一个信号，D 中的元素个数必须等于列数
% Find length and dimension of input signal 查找输入信号的采样点数和通道数。
[r,c] = size(s);

% Covert to column vectors for processing 隐藏到列向量进行处理。
if r == 1;
    s = s.'; % 行列颠倒
    [rn,c] = size(s);
elseif c == 1
    [rn,c] = size(s); % Otherwise rn = r indicating it came in as a column vector 否则 rn = r 表示它以
列向量的形式出现。
else % If multi column and row then process each column as its own signal 如果多列和行然后
将每个列处理为它自己的信号。
    [rn,c] = size(s); % Otherwise input was columnwise matrix 否则输入为列式矩阵。
    if length(d) ~= c
        error('Number of columns in signal matrix must equal number of elements in delay vector')
    end
end

if min(d) < 0;
    error('Delay cannot be negative')
end

% Compute requested delay in sample points rounded off to nearest sample
% 计算请求的延迟是多少个样本点，向上取整
nd = ceil(d*fs);
```

```

% Determine final output length
% 确定最终输出长度
if nargin == 4
    slen = ceil(ns*fs);
else
    slen = max(nd)+rn; % 延迟点数要求最多的通道为标准，其他通道也延迟这么长
    % 得出最终的输出长度 slen
end
sdd = zeros(slen,1); % Initialize dummy vector for storing integer shift 为存储整型移位而输入的假
矢量。
sd = zeros(slen,c); % Initialize output matrix 初始化输出矩阵

% Loop through each row of signal matrix and apply delay
% 遍历每一个通道信号矩阵并应用延时。
for k=1:c
    % Shift integer sample component of delay
    % 延迟的移位整型样本组件。
    id = fix(d(k)*fs)+1;
    sdd(id:min([slen,(rn+id-1)])) = s(1:min([slen-id+1, rn]),k);
    % 用源信号前补零的方式来进行延迟
    sd(:,k) = sdd(1:slen);
end

% Restore dimension of signal vector to original orientation
% 将信号矢量的维数恢复到原来的方向。
if rn == r % If input was originally a column or multi-signal matrix we are done 如果输入最初
是一个列或多信号矩阵，我们就完成了。
    if nargout == 2 % Create time axis if requested 如果需要，创建时间轴。
        tn = [0:slen-1]/fs;
    end
else % If input was originally a row vector, take transpose 如果输入原是行向量，取转置。
    sd = (sd. ');
    if nargout == 2 % Create time axis if requested 如果需要，创建时间轴。
        tn = [0:slen-1]/fs;
    end
end
end

```


增白，也可以不加

```
% The rest of the frequencies are set to zero.
% 剩下的频率分量会被设为 0
% The output W will be the same size as S and is the time domain whitened
% signal. If S is a matrix, this program will perform a columnwise
% whitening of the signal. 如果 S 是个矩阵，那么这个程序会执行列优先的准则
[r,c] = size(s);

if c == 1;
    s = s.'; % Ensure it is a row vector
    [r,cn] = size(s);
elseif r == 1
    [r,cn] = size(s); % Otherwise cn = c indicating it came in as a row vector
else % If multi column and row then process each column as its own signal
    s = s.';
    [r,cn] = size(s); % Otherwise cn = c indicating it came in as a row vector
end
if pw < 0 || pw > 1
    error('Whitening parameter must be between 0 and 1')
end

nfft = 2^nextpow2(2*cn); % pad with zeros to prevent circular convolution and make power of 2
w = zeros(r,cn) + 1j*zeros(r,cn); % Initialize output array

fw = zeros(r,nfft) + 1j*zeros(r,nfft); % Initialize frequency array

if nargin == 3
    lowindp = round(flim(1)*(nfft/2))+1;
    highindp = round(flim(2)*(nfft/2))+1;
    lowindn = (nfft)-highindp+2;
    highindn = (nfft)-lowindp+2;
    frq = [lowindp:highindp, lowindn:highindn];
else
    frq = (1:nfft);
end
% Loop to take FFT of each row and apply whitening operation
% for all non-zero frequency components
for k=1:r
```



```
fs = fft(s(k,:),nfft); % Take FFT 傅里叶变换
fw(k,frq) = fs(frq);
nzt = find(abs(fw(k,:)) ~= 0); % Find non-zero spectral values 找到不为零的值，F(S)不可
以为零
fw(k,nzt) = fw(k,nzt) ./ abs(fw(k,nzt)).^pw; % Normalize magnitudes for non-zero values
% 归一化不为零的值的梯度 即上面所述的公式，进行增白的核心步骤
dum = ifft(fw(k,:)); % return to time domain 反傅里叶变换
w(k,:) = dum(1:cn); % Remove zero padded samples
end

% Restore dimension of signal vector to original orientation
if cn == c && r == 1
    w = w;
else
    w = (w. ');
end
% If original signal was real, make output real
if isreal(s)
    w = real(w);
end
```

附录 B 外文文献翻译

基于麦克风阵列的采用分级搜索的声源定位的可控响应功率算法

作者: Leonardo O. Nunes, 学生会员, IEEE, Wallace A. Martins, IEEE 会员, Markus VS Lima, IEEE 会员, Luiz WP Biscainho, IEEE 会员, Maurício Vos Costa, Felipe M. Gonçalves, Amir Said, 研究员, IEEE 和 Bowon Lee, IEEE 高级会员

摘要: 扬声器在封闭环境中的本地化通常通过实时处理由一组麦克风捕获的多个音频信号来实现。声源定位的主要相关方法之一是转向响应功率 (SRP), 它在空间网格上搜索最大功率点。高精度的本地化需要密集的网格和/或许多麦克风, 这往往会不实际地增加计算需求。本文提出了一种新的声源定位方法 (称为 H-SRP), 该方法将 SRP 方法应用于空间区域而不是网格点。这种安排为分层搜索的使用提供了空间, 这种分层搜索的灵感来自于分支定界范式, 它保证能在无回声环境中找到全局最大值, 并通过实验显示在混响条件下也能工作。H-SRP 除了受益于逐点搜索的混响搜索与混响效果相比的改进的鲁棒性, H-SRP 具有高度的性能和易于管理的复杂性。特别在典型的演示室中使用 16 个麦克风阵列的实验产生了 7cm 量级的定位误差, 并且对于给定的固定复杂度, 竞争方法的误差是两倍到三倍大。

关键词: 声源定位, 导向响应功率, 麦克风阵列, 计算复杂度, 分层搜索, 分支定界。

第一章 简介

声源定位 (SSL) 可用于从通信 (例如电话会议系统) 到医疗应用 (例如助听器) 等各种实际系统中。但为了定位声源, 必须依赖某种空间信息, 例如由麦克风阵列提供的空间信息在针对 MA 设计的 SSL 技术中, 两种算法通常是普遍的选择[2]: 第一种明确地基于到达时间差 (TDOA), 而第二种依赖于最大化引导响应功率 (SRP) 的波束形成器。基于 TDOA 的方法, 其中最流行的技术使用由广义互相关 (GCC) [5] - [7]估计的 TDOAs, 与基于 SRP 的算法相比, 需要相对较少的数值操作来定位源。然而, 基于 TDOA 的方法的性能受噪声和混响的影响很大[2], [8], 这可能会阻碍它们在实际应用中的使用。在这种情况下, 基于 SRP 的方法 (其经典版本 (以下称为 C-SRP) [2] [8]) 应用最广泛, 它们更适合于它们对应用环境固有的声学问题的稳健性。

为了估计源位置, 将 C-SRP 方法应用于表示源位置候选的预定义空间点的网格上。高定位精度只能通过增加网格点的数量或麦克风的数量来实现 (通常, 捕获的声音信号的数量越多, 获得的空间分集越高)。因此, 逐点搜索源位置的负担增加了 C-SRP 算法的计算复杂度, 其增加会导致实时操作变得不切实际, 从而使得该算法在大多数感兴趣的应用中无用[9]。

为了解决这个问题, 已经提出了几种修改搜索过程的方法。例如, 在文献[10]中, 作者基于随机区域收缩 (SRC) 算法设计了 C-SRP 的搜索策略, 该算法能够估计源位置, 而不需要评估与 C-SRP 相关的目标函数网格点。与 SRC 方法类似, 粗到细区域收缩 (CFRC) [11]试图通过根据一组启发式逐步减少搜索空间来找到源位置。文献[12]提出了一种基于粒子滤波的 SRC 方法的改进。

所有先前的方法通过避免评估搜索网格中每个点的 C-SRP 目标函数来减少算术运算的次数。这最终意味着有时可能会错过真正的来源地位，因为没有正式保证在搜索过程中被忽视的地点不适合来源地点。在混响存在的情况下，这种缺点变得更加明显，其在相关方法的目标函数中引起许多局部最大值。

本文提出了一种新的基于 SRP 的单源定位方法，与 C-SRP 相比，它能够降低搜索阶段的计算复杂度，并且在理论上保证收敛到全局最大值（表示在分区三维搜索空间，该子区域包含源位置）在消声条件下。同时在实际非消声环境中保持良好性能。关键的想法是对部分搜索空间的隐式探索。所提出的搜索策略，这里称为分层搜索，通过定义一个与声学活动相关的目标函数来适应分支定界（B&B）范式[14]，以解决 SSL 问题，其目标函数的计算值为给定的三维空间区域从不小于其理想条件下的任何分区域。在实际混响条件下的仿真结果表明，所提出的方法（简称 H-SRP）能够以较低的计算负担实现良好的精度。应该指出的是，使用分级搜索的体积型 SRP 的想法虽然有不同的表述，但最初是在[13]中提出的。

本文的结构如下。第二部分概述了提议的分级搜索，描述了 B&B 模式如何为 SSL 应用量身定制；强调边界函数所起的作用，以及它必须满足的条件才能保证收敛到相关目标函数的全局最大值。在第三节中讨论了满足无声环境中这种条件的边界函数的选择，其中还包括对混响效应的简要讨论。第四节讨论了实施 H-SRP 方法的实际情况，该方法与第五节中的先前工作进行了比较。使用人工生成和记录信号的实验见第六节。第七节提出了结论。三个定理的证明由附录 A，B 和 C。

第二章 B&B 支持的分层搜索

如前所述，C-SRP 的大部分计算负担是由于其搜索过程所致，这需要搜索空间被分成一个网格点，每个点必须被访问一次。此外，对于给定的 MA 和预定义的采样频率，只能通过增加网格内点的数量来提高位置估计的准确度，即将其变得更密集。有些方法试图通过避免访问所有网格点来避免这个问题[9]，[10]。然而，由于 C-SRP 目标函数可能表现出多个局部最大值，因此从确定性角度来看，这种方法无法保证收敛于全局最大值。因此，出现以下两难局面：为了确保与实际源头位置相接近，不应忽视栅格点；另一方面，通过网格的穷举搜索通常对于实时应用来说太复杂。解决这个问题的一种自然方法是对搜索空间进行隐式探索[13]。最初为离散和组合优化问题开发的分支定界（B&B）范式[14]，[15]似乎是为此目的精心设计的，因为它保证了收敛到其相应目标函数的全局最大值。

基于 B&B 的算法与搜索空间一起工作，搜索空间可以分为嵌套子空间，每个子空间在动态生成的树结构中被看作是一个节点[14] 基于 B&B 的算法不是评估树中所有可能节点的基本目标函数，而是使用边界函数 2，它可以帮助您决定如何通过分支过程生成新节点。总之，一般 B&B 算法的主要组成部分是[14]：(i) 选择要处理的节点，(ii) 边界函数计算，和 (iii) 节点分支。通过适当的塑造（以及它们对 SSL 问题的适应），本文提出了建议的分层搜索。

首先，所提出的分层搜索认为节点将对应于 3-D 空间区域。在这种情况下，根节点是整个欧几里得搜索空间（例如会议室）。边界过程使用边界函数，这在第三节提出，与给定空间区域内声源的存在（或不存在）相关联。在这里，目标函数和边界函数是相同的。至于分支过程，它就是为了生成新的子区域（新节点）而节点被细分的方式。因此，所提出的分层搜索通过将搜索空间（根节

点）划分成更小的区域（节点）（其构成分支过程）并然后计算每个空间区域的边界函数的值（其构成边界过程）来操作。

边界函数的作用是允许隐含地评估相关搜索空间的部分，即可以在没有明确地访问每个点的情况下探索整个搜索空间。在数学上，所提出的分级搜索依赖于适当的边界函数 $b: \mathcal{V} \rightarrow \mathbb{R}_+$ ，这其中 $\mathcal{V} \subset \mathcal{P}(\mathcal{F})$ 是整个搜索集合。一个辅助的 \mathcal{V} 技术条件是紧密的和连通的集合的元素。

分级搜索时按顺序将搜索空间 \mathcal{F} 细分为子区域，直至预定义的 $V_{\min} \in \mathbb{R}_+$ 最小，这与搜索算法的停止准则有关。在这个过程中，边界函数在丢弃不需要任何进一步细分的集合中起关键作用。用 \mathbf{L} 来表示一系列序列 $[\mathcal{V}, b(\mathcal{V})]$ 属于要被估计的次区域 $\mathcal{V} \in \mathcal{V}$ 的。以及它们的相关边界值 $b(\mathcal{V}) \geq 0$ ，并且 b_{\max} 是该算法的给定迭代处找到的最大边界值， \mathcal{V}^* 是其相应的子区域（即 $b(\mathcal{V}^*) = b_{\max}$ ），所提出的分层搜索如下。

- (1) 初始化，让 $\mathbf{L} \leftarrow \{[\mathcal{F}, b(\mathcal{F})]\}$, $b_{\max} \leftarrow -1$,
- (2) 如果 $\mathbf{L} = \emptyset$ ，则停止：搜索已经完成并且 \mathcal{V}^* 大小不大于最大的子区域边界值 V_{\min} 。
- (3) 令 $\mathbf{L} \leftarrow \mathbf{L} \setminus \{[\mathcal{V}, b(\mathcal{V})]\}$ 。
- (4) 如果 $b(\mathcal{V}) < b_{\max}$ 那么回到第 2 步
- (5) 如果 $b(\mathcal{V}) \leq V_{\min}$ ，令 $b_{\max} \leftarrow b(\mathcal{V})$
- (6) 把 \mathcal{V} 分成 D 小块
- (7) 令 $\mathbf{L} \leftarrow \mathbf{L} \cup \{[\mathcal{V}_1, b(\mathcal{V}_1)], [\mathcal{V}_2, b(\mathcal{V}_2)], \dots, [\mathcal{V}_D, b(\mathcal{V}_D)]\}$ ，然后回到第 2 步。

图 1 说明了分级搜索如何操作。算法的第一步包括细分搜索空间和计算每个子区域的边界函数（图 1 (a)）。然后，选择具有最大边界值的子区域并进一步细分直至达到大小/的区域（图 1 (b)）。最后，为了保证当前最大值对应于全局最大值，任何具有大于（或等于）当前最大值的边界值的其他区域必须被细分，直到找到新的最大值或所有边界值低于当前最大值（图 1 (c)）。或者，该算法可以以树形格式表示，其中每个节点是搜索空间的细分（在每个区域被细分为两个区域的情况下，其中一个将具有二叉树）。在这种情况下，当树的所有树叶的边界值都低于当前最大值时，会找到全局最大值。

在本节中，显示了所提出的分层搜索的基础，但没有说明边界功能。目前为止，任何功能满足以下两个性能足以保证收敛到全局最大值：

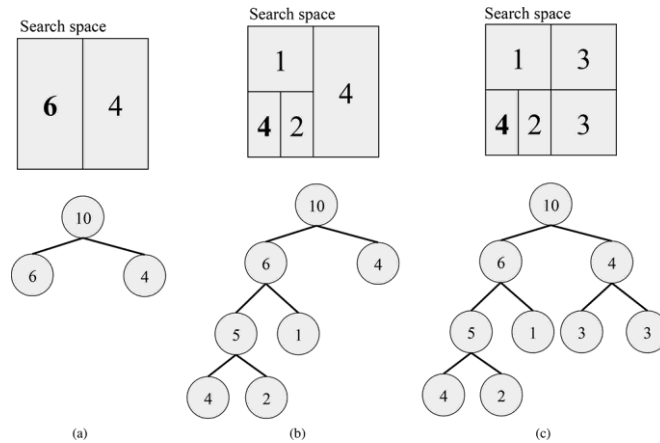


图 1.用于二维搜索空间的 B&B 算法的三个不同阶段的插图。 还提供了每个阶段的等效树形表

示。根据“最佳第一搜索”策略访问节点。每个区域/节点内的数字代表给定区域的边界值（边界函数的输出）。

第三章 绑定功能

在描述提出的边界函数之前，首先需要介绍一些自然导致所提出的边界函数的 C-SRP 方法的已知特性。因此，这里的想法首先是展示 C-SRP 技术如何解决发射信号源的问题，该信号源在消声环境中最大限度地集中在空间和时间域中。事实证明，这个简单的玩具例子的解释对于进一步的发展将是非常有益的。在讨论之后，将会展示如何修改 C-SRP 方法来处理时间不集中的声学信号，例如宽带/语音信号。这样的表示顺序将指出“计数 T doAs”确实是构建分层搜索的边界函数的关键方面。

C-SRP 方法将麦克风阵列波束引导到许多位置以搜索声源位置。该搜索基于最大化波束形成器的输出信号的功率。因此，C-SRP 最大化以下目标函数：

$$W(\mathbf{x}) \triangleq \sum_{n \in \mathbb{Z}} \left| \sum_{m=0}^{M-1} s_m[n + k_m(\mathbf{x})] \right|^2, \quad (1)$$

也可以重写为：

$$W(\mathbf{x}) = \frac{1}{2\pi} \sum_{m_1=0}^{M-1} \sum_{m_2=0}^{M-1} \int_{-\pi}^{\pi} S_{m_1}(e^{j\omega}) \times S_{m_2}^*(e^{j\omega}) e^{j\omega k_{m_1, m_2}(\mathbf{x})} d\omega, \quad (2)$$

现在，考虑将 C-SRP 方法应用于两种不同的设置，即：定位脉冲并在消声环境中定位宽带/语音信号。

(1) 在消声环境中定位脉冲：假设位于 $\bar{\mathbf{x}} \in \mathbb{R}^{3 \times 1}$ 位置的单个声源发射脉冲信号 $s[n] = \delta[n]$ 。在这种情况下，考虑到消声环境 and 无视与声音传播相关的衰减因素，则：

$$W(\mathbf{x}) = \sum_{m_1=0}^{M-1} \sum_{m_2=0}^{M-1} \delta[k_{m_1, m_2}(\mathbf{x}) - k_{m_1, m_2}(\bar{\mathbf{x}})], \quad (3)$$

其中任何分立的 TDOA 可以被重写为：

$$k_{m_1, m_2}(\mathbf{x}) \triangleq \text{round} \left\{ \int_s \frac{\|\mathbf{x} - \mathbf{x}_{m_1}\|_2 - \|\mathbf{x} - \mathbf{x}_{m_2}\|_2}{c} \right\}, \quad (4)$$

在这个非常理想化但具有启发性的设置中，人们可以清楚地看到 C-SRP 方法如何聚合与每个麦克风对相关 TDOA，以便估计真实的声源位置。事实上，(3) 表明，对于三维欧几里得空间中的每个位置，C-SRP 方法只是将一个数字关联起来，该数字量化了与真实源位置相关的实际 TDOA 源头。换句话说，计算 TDOAs 是这里的底线。

(2) 在消声环境中定位声源 - PHAT 滤波的作用：例如，现在考虑一种更现实类型的声信号，其可以表示语音信号。为了保留在 (3) 中获得的完全相同的直观和有用的结果，应该将 (2) 中的 C-SRP 目标函数修改为以下定义 C-SRP-PHAT 方法的形式

$$W_{\text{PHAT}}(\mathbf{x}) \triangleq \frac{1}{2\pi} \sum_{m_1=0}^{M-1} \sum_{m_2=0}^{M-1} \int_{-\pi}^{\pi} \Psi_{m_1, m_2}(e^{j\omega}) S_{m_1}(e^{j\omega}) \times S_{m_2}^*(e^{j\omega}) e^{j\omega k_{m_1, m_2}(\mathbf{x})} d\omega, \quad (5)$$

其中相变（PHAT）滤波器被定义为：

$$\Psi_{m_1, m_2}(e^{j\omega}) \triangleq \frac{1}{|S_{m_1}(e^{j\omega}) S_{m_2}(e^{j\omega})|}. \quad (6)$$

通过使用这个新的目标函数，再次得到：

$$W_{\text{PHAT}}(\mathbf{x}) = \sum_{m_1=0}^{M-1} \sum_{m_2=0}^{M-1} \delta[k_{m_1, m_2}(\mathbf{x}) - k_{m_1, m_2}(\bar{\mathbf{x}})]. \quad (7)$$

因此，对于消声环境，处理通用源信号时 C-SRP-PHAT 目标函数的解释与时空定位源信号/相关的 C-SRP 的解释相同（即，计数 TDOAs 是关键的想法）。

第四章 实用考虑

在本节中，讨论关于实现所提出的算法的实际考虑事项。首先介绍分支过程，即将搜索空间划分为子区域的策略。然后，描述用于找出（11）和（8）的延时边界的算法。之后，描述分级搜索的初始化过程。该部分以提出的算法的实现摘要结束。值得一提的是，本节所讨论的主题也是这项工作的新贡献，[13]中没有提到。

在当前的实施中，只考虑长方体区域。因此，整个搜索空间被假定为长方体

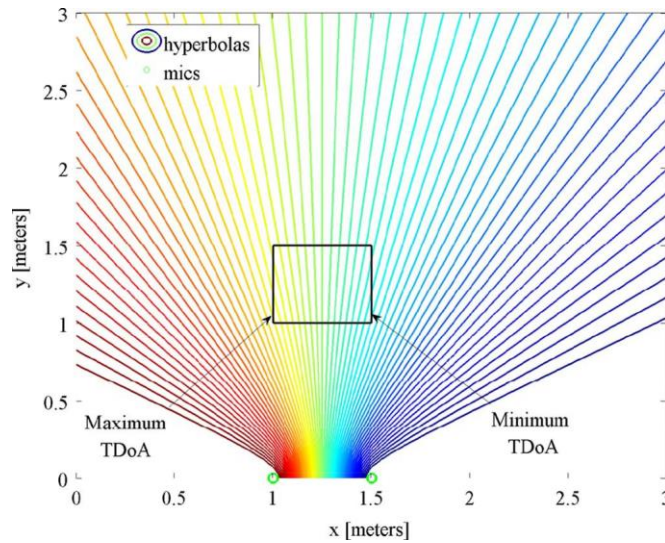


图 2 说明了二维空间，正方形区域内的最大和最小 TDOA 将位于其顶点。
在推论 1 中描述的本文中采用的简化。

推论 1: 集合 $\{\tau_{m_1, m_2}(\mathbf{x}) \mid \mathbf{x} \in \mathcal{V}\}$ 的最小值 $\tau_{m_1, m_2}^{\min}(\mathcal{V}) \in \mathbb{R}$
 其中当 \mathcal{V} 是长方体, 可近似为:

$$\tau_{m_1, m_2}^{\min}(\mathcal{V}) \approx \min_{[x \ y \ z]^T \in \mathcal{S}_{m_1, m_2}^{\mathcal{V}}(\mathcal{V})} \{\tau_{m_1, m_2}(x, y, z)\},$$

为了便于可视化, 图 2 描绘了说明 TDOA 和音量通常如何与给定麦克风对相关的 2-D 表示。实际上, 由于它是二维表示, 长方体和双曲面映射到正方形和双曲线, 每个双曲线应该被理解为导致相同 TDOA 值的点的集合。TDOA 值从右侧 (深蓝色双曲线) 向左侧 (棕色双曲线) 逐渐增加。注意到最小 TDOA $\tau_{m_1, m_2}^{\min}(\mathcal{V})$ 和最大 TDOA $\tau_{m_1, m_2}^{\max}(\mathcal{V})$ 确实在正方形的顶点。请注意, 如果广场位于不同的位置, 相同的评论仍然有效。

因此, TDOA 限定给定的音量, 并且给定的麦克风对可以通过与相关音量的顶点关联的最大和最小 TDOA 值近似。值得一提的是, 可以确定 $\tau_{m_1, m_2}^{\max}(\mathcal{V})$ 遵循前面所述的相同步骤, 但用相关表达式中的“max”运算符替换 “min”运算符。

分层搜索 初始化:

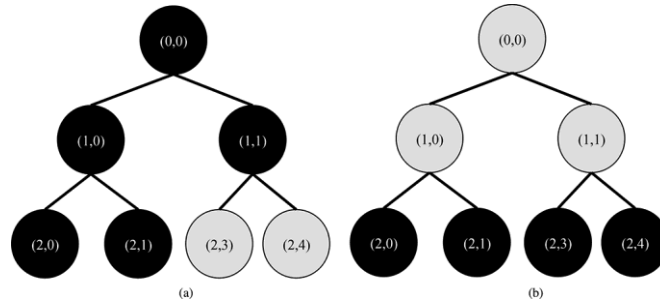


图 3: 假设节点 (2,0) 具有最大边界值, 用于分级搜索和全网格搜索的最小节点数量示例 (黑圈)。(a) 分层搜索 (b) 全网格搜索。

在本小节中, 将分级搜索的计算复杂性与等效完整进行搜索。这种比较考察了分级搜索的计算复杂性, 表明在某些情况下, 从搜索区域的细分开始搜索可能更有效。

通常, H-SRP 方法的计算复杂度可以分解为两部分: 一部分与计算每个麦克风对 9 的信号之间的互相关以及另一部分由计算边界函数在搜索期间访问的每个空间区域。在下面的讨论中, 重点将放在计算边界函数的代价上。

假设已针对所有麦克风对的所有可能的 TDOA 计算互相关, 计算给定空间区域 \mathcal{V} 的边界值的成本与 (13) 的总和中的项的数量成比例。实际上, 这个数字将取决于麦克风的数量和集合中的离散 TDOA 数量 $\mathcal{K}_{m_1, m_2}(\mathcal{V})$ 。

为了研究分层搜索的计算复杂性, 将采用其树形表示, 其中树的第 j 级的第 i 个节点将被表示为 $\mathcal{N}_{i,j}$ 。考虑到前一段的讨论, 可以看出, 计算给定节点边界所需的操作数总是小于或等于其父节点边界的操作数。那么必须回答的一个重要问题是, 如果通过分层搜索算法选择具有最高边界的节点或者通过树的给定层上的所有节点中的全部搜索来选择节点在计算上是有效的。作为一个例子, 图 3 (a) 显示了在层次搜索期间为了选择节点 $\mathcal{N}_{2,0}$ 具有最大边界而必须访问的节点的最小数目, 而

图 3 (b) 显示了在全搜索情况下必须考虑哪些节点。比较这个图中的两个图，可以容易地看到全搜索在计算上更高效，因为分层搜索计算靠近根节点的节点，其可能具有更多 TDOA 并因此具有更大的计算复杂度。另一方面，随着树的生长，分层算法可以忽略更多的节点，从而克服第一级节点的成本。

实际上，分级搜索只能从其最佳情况计算成本[10]低于全搜索的水平应用。该方案允许避免计算大型节点。通过计算计算树中每个节点的边界值[11]所需的操作次数以及与最佳情况分级搜索和全部搜索相关的成本，可以离线选择该初始级别。一旦选择了初始级别，就可以应用分级搜索 这样，避免了树的第一层的高成本边界的计算，而层次搜索仍然可以避免树的多个分支，从而降低了总体复杂度。在算法上，可以通过将初始级别的所有节点的空间区域和边界值归属到列表而不是整个搜索空间的相应数量来执行初始化。在第二部分描述的算法中。

第五章 与之前的作品比较

在[16]中，作者描述了 C-SRP 算法的修改，其中考虑了与单个点周围空间中的某个区域相关联的多个 TDoA。该方法的目标是在空间中的每个点周围定义一定的影响区域，找到与该区域相关的一组延迟，然后评估修改后的 SRP（以下称为 M-SRP）函数。在数学上，这个函数等于

$$W_{\text{MOD}}(\mathbf{x}) = \sum_{m_1=0}^{M-2} \sum_{m_2=m_1+1}^{M-1} \sum_{k=\hat{K}_{\mathbf{x},m_1,m_2}^{\min}}^{\hat{K}_{\mathbf{x},m_1,m_2}^{\max}} \mathcal{R}_{m_1,m_2}[k]. \quad (22)$$

其中值 m_1 和 m_2 首先考虑一个立方体为中心，然后执行线性近似以找到该立方体内的最小和最大延迟值，考虑麦克风和麦克风。

如果重写 (21) 为

$$W_{\text{PHAT}}(\mathcal{V}) = 2 \sum_{m_1=0}^{M-2} \sum_{m_2=m_1+1}^{M-1} \sum_{k \in \mathcal{K}_{m_1,m_2}(\mathcal{V})} \mathcal{R}_{m_1,m_2}[k] + \sum_{m=0}^{M-1} \mathcal{R}_{m,m}[0], \quad (23)$$

那么可以注意到 M-SRP 目标函数和提出的边界函数之间的相似性。两种方法都考虑到一组延迟以估计某个区域。尽管如此，可能的区域是不同的：M-SRP 的立方体和 H-SRP 的立方体。而且，最大和最小延迟仅近似于 (22) 而对于 (23)，如果某些（温和）条件成立，则它们是确切的（见第 IV.B 节）。总体而言，这些差异允许在分层搜索中使用边界函数，而 (22) 更适合用于穷举搜索，就像其最初的目的一样[14]

第六章 实验

在本节中，描述了两个实验。第一个实验使用模拟信号并显示参数如何分层搜索会影响其在不同声学条件下的性能。第二个实验使用记录的信号，并将层次搜索的性能与文献中找到的其他方法进行比较。

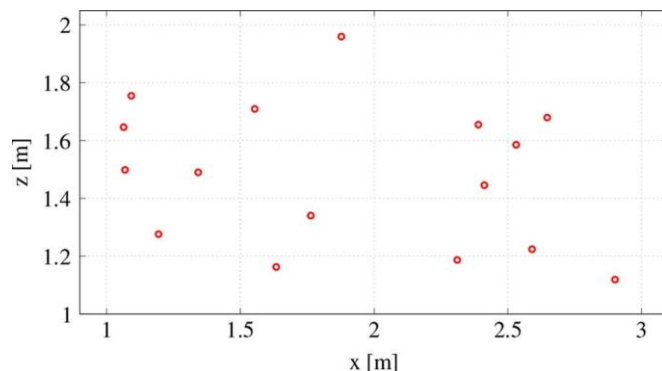


图 4.用模拟信号定位实验中使用的麦克风（圆圈）。

A.使用模拟信号进行评估

本实验使用模拟信号的目的是了解选择/影响分级搜索的性能，特别是定位精度和执行操作次数之间的折衷。分层搜索也在不同的混响条件下进行测试，以表明即使其理论公式基于消声情况，也可以在更真实的条件下使用。

（1）验设置：模拟考虑了三个房间大小为 4 米 6 米 3 米不同的声学特性。第一个是消声室，而第二个和第三个混响时间分别为 250 和 500 ms。如前所述，这些参数选择允许在理想化（消声）在其发展中使用的声场景以及混响对其性能的影响。

与[2]中的排列类似，本实验采用了一个带有 16 个麦克风的平面 16 阵列，随机放置在 2 m 1 m 的矩形区域中，如图 4 所示。在模拟中，阵列应该放置在其中一个房间墙上。

源信号是由话音活动检测器（VAD）预处理的 1 秒长的女性语音信号，以消除任何无声间隔。这一步保证了话筒上始终存在一个语音信号。实际上，在麦克风获取信号之后，VAD 步骤将被执行。语音信号以 24 位精度在 48 kHz 采样。语音源在 5 个随机选择的位置处进行模拟，如表 1 所示。选择这些位置以表示感兴趣区域的可区分性指数（DI）[20]范围。DI 是 a 的能力的数字度量

表 1 5 个模拟源头位置的坐标

Pos.	Coordinates [m]
1	[2.92, 2.18, 1.64]
2	[2.09, 1.01, 1.88]
3	[1.28, 2.13, 1.88]
4	[1.30, 0.99, 1.69]
5	[1.52, 2.36, 1.78]

麦克风阵列来区分空间中的给定点与其邻居，其中主要受到阵列几何形状和考虑中的空间区域的影响。具体而言，5 个职位涵盖了从他们领取的地区发现的 30%至 90%的 DI 范围。

(2) 算法设置：使用分层搜索算法来估计具有 50% 重叠的连续 4096 个样本长帧中的源位置。PHAT 预滤波器用于估算每个麦克风对的互相关函数。

(3) 优值：每帧的定位误差测量为实际和估计源位置之间的欧几里德距离。为了获得定位性能的整体值，可以在所有不同帧和不同位置上计算误差的中值。对于每个选定的 V_{\min} ，计算一个中值误差值 V_{\min} 。

所执行的操作次数的计算基于 (21) 的评估次数。由于搜索算法是逐帧执行的，因此操作次数可能因帧而异。然后，对于每个帧和源位置，存储 (21) 的每个评估中的总和总数，并且通过对所有帧和源位置上的这些总和进行平均来获得复杂度的总体度量。

(4) 结果：图 5 显示了三条曲线，描绘了针对三个不同房间的 6 个选择值的中值误差和操作次数的不同结果。从图中可以看出，对于给定的声学配置，随着 V_{\min} 的尺寸减小，误差随着所执行操作的数量的相关增加而减小。另一方面，增加混响时间会增加给定操作次数的误差。因此，声学场景看起来更具挑战性，为了达到给定的性能，较小的 V_{\min} 应该以增加必要的操作数为代价。尽管如此，即使在混响条件下，误差随着 V_{\min} 而下降的事实也验证了该方法。通过分层搜索实现的最小中值误差对于消声室和具有 $T60=250\text{ms}$ 的房间为 1.5cm，对于具有 $T60=500\text{ms}$ 的房间为 2.5cm。

B. 使用记录的信号进行评估

在这个实验中，进行记录是为了获得每个麦克风的信号。在这个更现实的情况下，目的是比较层次搜索的定位精度与文献中找到的其他方法的精度。所选择的方法是经典的 SRP（采用全网格搜索），[16] 中描述的修改的 SRP (M-SRP)（其也采用全网格搜索）和具有随机区域收缩的 C-SRP SRC）[10]

(1) 实验设置：录音发生在位于里约热内卢联邦大学的声学处理演示室。房间尺寸为 5.2 米 7.5 米 2.6 米，其混响时间约为 500 毫秒。

该阵列使用全向高灵敏度麦克风 17 阵列的几何形状与之前的实验一样是平面的，这次麦克风放置如图 6 所示。MA 被放置在房间的其中一个墙上，即， θ 适用于所有麦克风。

源信号是一个 5 s 长的无静音女性语音信号，采样率为 48 kHz，精度为 24 位。直径 6 厘米的扬声器 18 被用作声源。图 7 显示了 9 种不同来源的俯视图

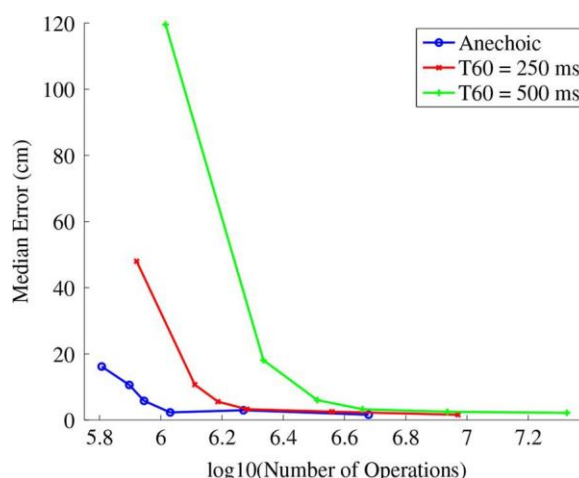


图 5.模拟信号的实验结果。 所有的结果都是指分层搜索方法的变化值为 V_{min} （最左边的点对应于所有图中的 V_0 ）。

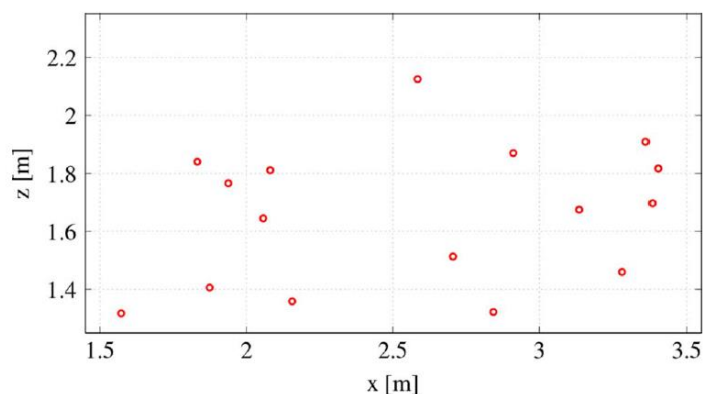


图 6.用模拟信号定位实验中使用的麦克风（圆圈）。

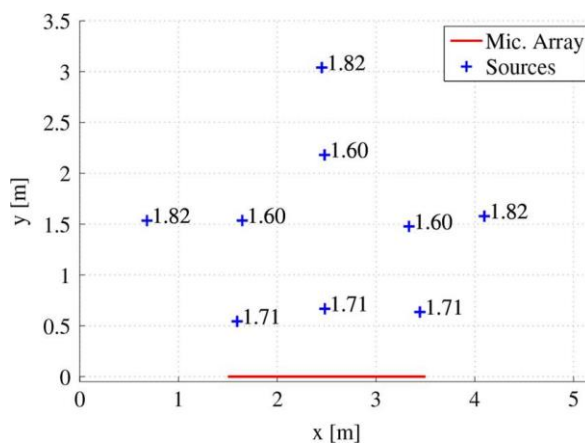


图 7.实验中记录信号源的自顶向下视图。 每个源位置旁边的数字以米为单位表示源的高度。

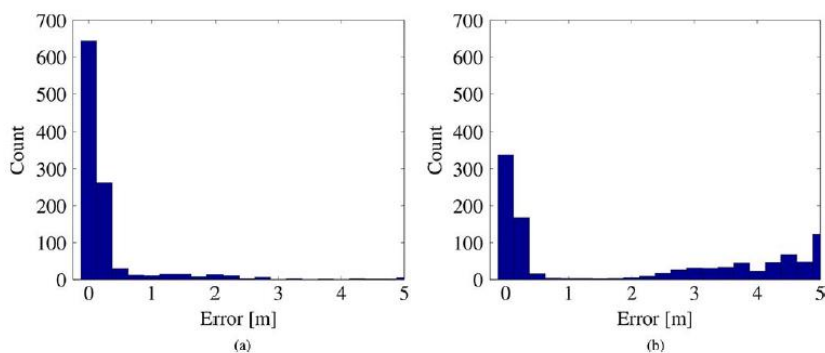


图 8.层次搜索和 SRC 算法都针对每个帧和每个源位置提交的错误的直方图。

(a) 分层搜索 (b) SRC

记录的职位。 无论源位置如何，扬声器始终面向麦克风阵列。

(2) 算法设置：所有评估算法估计 50% 重叠的连续 4096 个样本长帧中的源位置，并在计算互相关时使用 PHAT 预滤波器。

选择每种搜索方法的各个参数以产生相似数量的操作。C-SRP 算法被设置为在相邻点之间的间距等于 10cm 的网格上操作。M-SRP 方法采用间距为 20 厘米的网格。SRC 方法每次迭代使用 9000 个点，选择 300 个最佳点。分层搜索使用最小体积大小 1.0 厘米 1.5 厘米 0.5 厘米，并初始化为。

(3) 优缺点：每种方法的定位性能都是通过所有框架和位置上的实际源估计值和估算源位置之间的欧几里德距离的中位数来评估的。

另一方面，操作的数量考虑了与每种方法相关的目标函数的每次评估所进行的总和的数量。对于所有方法，没有考虑每个麦克风对的互相关函数的计算成本。此外，与空间中每个位置（针对 C-SRP 和 SRC 算法）或 TDoA 边界（针对高分辨率图像）相关联的 TDoA 的计算成本

Algorithm	Median Error [cm]	$\log_{10}(\text{Number of operations})$
H-SRP	7.2	7.1
C-SRP	15.5	7.1
M-SRP	18.9	7.0
SRC	22.3	7.1

表 2 实验结果

对于使用全网格搜索的 C-SRP 和 M-SRP，使用每帧的总和数，因为它在帧间不变化。对于 SRC 和分级搜索，每帧的操作数量会有所不同，因此将采用每帧操作的平均数量。

(4) 结果：表 II 显示了每种方法获得的中值误差和操作次数 20。按照设计，每种方法的操作次数是相似的；在这些条件下，中值误差显著变化。21 总体而言，通过所提出的分层搜索方法实现了最佳性能，其获得的误差等于小于下一个最佳方法的误差的一半。应该指出的是，扬声器膜本身具有大约 6cm 的直径，这引起源信号的确切位置的不确定性。

比较不执行全网格搜索的两种方法（分级搜索和 SRC）时，可以看到具有确定性地搜索源位置的方法的优点。图 8 显示了描述所有帧和所有源位置的层次搜索和 SRC 方法的误差分布的直方图。可以看出，两种方法之间的主要区别在于分层搜索中发现的少量错误估计，表明它通常选择与源位置关联的全局最大值。另一方面，SRC 经常选择远离实际声源的位置。

第七章 结束语

本文提出了一种新的声源定位方法 H-SRP。该方法依赖于与给定体积区域内的声学活动相对应的测量，并且采用受分支定界范例启发的分级搜索方法。经证明，所提出的方法在消声和单源情况下找到具有给定最小尺寸（停止标准）的体积，其包含 C-SRP 方法所针对的全局最大值。在模拟和实际混响室进行的测试表明，所提出的方法可以成功应用于更真实的场景。此外，结果表明 H-SRP 可以促进以比竞争的本地化方法更少的数字操作提供准确的位置估算。寻找足够的条件来保证在混响环境中收敛到包含源位置的区域，并且使用不同的麦克风阵列几何形状和语音信号执行广泛的测试，这些都是未来研究的有趣话题。