

Acoustic Source Localization Algorithm Based on Four-Elements Cross Microphone Array*

WANG, Yong^{1*}, WANG, Qinyun²

(1. School of Information and Electrical Engineering, College of Industrial Technology, Xuzhou Jiangsu 221140 China;

2. School of Communication Engineering, Institute of Nanjing Technology, Nanjing 211167, China)

Abstract: A sound source location system based on four-elements cross microphone array is realized. Four-elements cross array is selected as the microphone array and the formulas of sound source localization algorithm based on four-elements cross microphone array are also derived. To solve the question that the robustness of cross-correlation time delay estimation method is poor in low signal-to-noise ratio and reverberation environment, the system uses generalized cross correlation algorithm to estimate the location of the delay, and the cortex-A8 embedded platform to achieve a robust sound source localization system.

Key words: microphone array; sound source localization; embedded system

EEACC: 7820

doi: 10.3969/j.issn.1005-9490.2017.05.033

一种基于四元十字麦克风阵的声源定位算法*

王 永^{1*}, 王青云²

(1. 徐州工业职业技术学院信息与电气工程学院, 江苏 徐州 221140; 2. 南京工程学院通信工程学院, 南京 211167)

摘 要: 实现了一种基于四元十字麦克风阵列的声源定位系统。选取四元十字阵作为麦克风阵列的阵型, 推导了基于四元十字麦克风阵列的声源定位算法的公式。针对传统互相关时延估计算法在低信噪比、混响大的环境下鲁棒性较差的问题, 系统采用广义互相关算法来进行定位的时延估计, 并使用 Cortex-A8 嵌入式平台实现了鲁棒的声源定位系统。

关键词: 麦克风阵列; 声源定位; 嵌入式系统

中图分类号: TN912.34

文献标识码: A

文章编号: 1005-9490(2017)05-1224-05

麦克风是最简单的语音信号采集设备, 是语音信号处理系统的入口。随着技术的发展以及人们的需求日益增长, 单个麦克风已经不能满足现代语音信号处理系统的需求^[1]。如果改变声源位置, 就需要重新选择麦克风的位置, 以拾取较高质量的信号; 环境中的噪声和混响都会大大降低单麦克风采集信号的质量, 影响处理的效果。因此, 用多个麦克风组成的阵列来处理语音信号的技术^{[2][3]}受到了人们的广泛关注与重视。目前, 麦克风阵列主要应用于远程电话会议^[4-5]、语音识别^[6-7]、机器人听觉^[8]、汽车环境中语音拾取^[9-10]、回响环境中声音拾取^[11-12]、大房间中的会议录音^[13]和助听设备^[14]等。

声源定位技术在基于麦克风阵列的语音信号处理中处于核心地位。其不但可以提供位置信息, 而且对基于麦克风阵列的语音增强技术具有重要的作用。此外, 利用声源定位技术, 还可以改

善基本的盲源分离技术在声音混迭的环境的性能, 解决语音信号的重构问题, 显著改善恢复的语音信号的质量。声源定位是多年来国内外研究的重点课题, 刚开始是应用在军事上, 但随着时代的发展, 声源定位技术也逐渐地应用到了日常生活当中, 比如机器人眼追踪^[15], 视频会议系统通过定位发言者的位置, 自动将摄像头对准发言者等。基于麦克风阵列的声源定位技术已经成为一大研究热点, 对其开展研究具有很强的理论意义和实践价值^[16]。现有的大部分定位系统使用的算法的计算量较大, 实时性较差, 不能满足实时性要求高的场合。所以提高声源定位的实时性, 对于提高基于麦克风阵列的语音信号处理的响应速度具有非常重要的意义。本文提出了一种基于广义互相关时延估计的四元十字阵实时声源定位算法, 并使用 Cortex-A8 嵌入式平台实现了这个声源定位

项目来源: 国家自然科学基金项目(61375028)

收稿日期: 2016-08-02

修改日期: 2016-09-26

系统。仿真实验表明,提出的算法的鲁棒性优于传统互相关法的声源定位算法。

1 声源定位系统硬件框图

声源定位系统的主要功能就是对采集到的声音信号进行处理,给出定位结果。在设计实际的系统时,我们既要有效地采集声源信号,又要准确地实现声源的定位。在选择处理器时,即要考虑处理器的成本,又要考虑其运算能力以及未来方便升级算法等因素。本文经过综合比较分析后,决定选择使用三星的 S5PV210 作为处理器^[17],选择使用 WM8960 来实现音频编码。系统的硬件框图如图 1 所示。

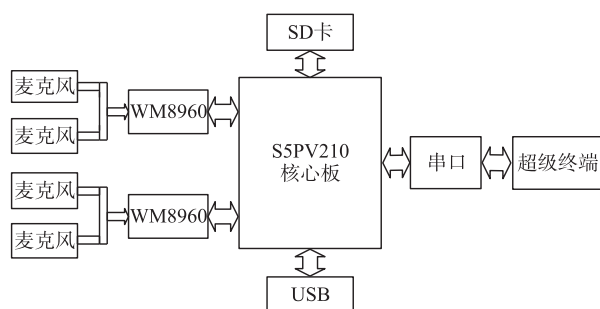


图1 声源定位系统的硬件框图

图1中的嵌入式最小硬件系统,由S5PV210处理器、SDRAM、NAND Flash、电源模块、时钟模块和复位电路组成。其主要功能是稳定可靠地运行操作系统。

系统采用4个麦克风组成四元十字阵来估计声源位置,具体的算法将在下一小节中推导。麦克风采集到模拟声信号后,送到WM8960音频编解码芯片进行编码。编码后的数字信号送往S5PV210处理,用定位算法计算出声源位置。再利用超级终端通过串口与嵌入式系统进行交互,最终结果打印在超级终端上。其中SD卡和USB模块用于嵌入式系统程序的烧写和未来系统的升级。

2 基于四元十字阵声源定位算法原理

本文采用四元十字阵声源定位算法。建立四元十字阵列分别由麦克风 M_1 、 M_2 、 M_3 和 M_4 组成,阵元间距为 L ,以阵列中心位置为原点 O ,建立如图2所示的直角坐标系。

4个阵元的直角坐标分别为 $M_1(L/2, L/2, 0)$, $M_2(-L/2, L/2, 0)$, $M_3(-L/2, -L/2, 0)$, $M_4(L/2, -L/2, 0)$ 。假设声源所在位置坐标为 $S(x, y, z)$,声源距离坐标原点的距离为 r ,仰角为 θ ,方位角为 φ 。 R_1 为声源 S 到阵元 M_1 的距离, d_{12} 、 d_{13} 、 d_{14} 为阵元之间的声程差,则根据目标和声阵的几何位置关系有:

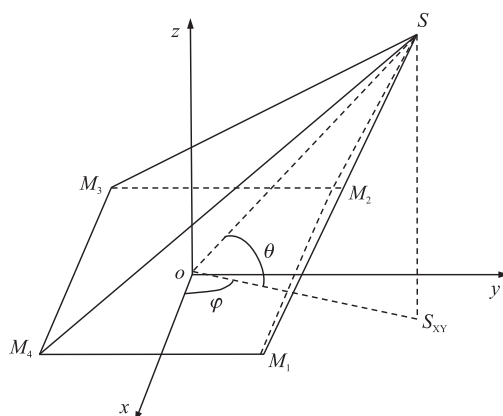


图2 四元十字阵结构图

$$\begin{aligned} x^2 + y^2 + z^2 &= r^2 \\ (x - L/2)^2 + (y - L/2)^2 + z^2 &= R_1^2 \\ \{(x + L/2)^2 + (y - L/2)^2 + z^2 &= (R_1 + d_{12})^2 \\ (x + L/2)^2 + (y + L/2)^2 + z^2 &= (R_1 + d_{13})^2 \\ \{(x - L/2)^2 + (y + L/2)^2 + z^2 &= (R_1 + d_{14})^2 \end{aligned} \quad (1)$$

将方程组整理并化简可得:

$$R_1 = \frac{d_{13}^2 - d_{12}^2 - d_{14}^2}{2(d_{12} + d_{14} - d_{13})} \quad (2)$$

将式(2)代入方程组并整理可得:

$$\begin{aligned} x^2 + y^2 + z^2 &= r^2 \\ (x - L/2)^2 + (y - L/2)^2 + z^2 &= R_1^2 \\ \{x &= (2R_1 d_{12} + d_{12}^2) / 2L \\ y &= (2R_1 d_{14} + d_{14}^2) / 2L \\ [R_1 &= (d_{13}^2 - d_{12}^2 - d_{14}^2) / [2(d_{12} + d_{14} - d_{13})] \end{aligned} \quad (3)$$

式中: $d_{12} = \tau_{12}C$, $d_{13} = \tau_{13}C$, $d_{14} = \tau_{14}C$ 。 τ_{12} 、 τ_{13} 、 τ_{14} 为参考阵元 M_1 与其他阵元间的时间差, C 为声速。通过方程组(3),只要估算出时间差便可求得 x 、 y 、 z 和 R_1 ,同时可以得到方位角:

$$\varphi = \arctan(y/x) \quad (4)$$

仰角:

$$\theta = \arccos\sqrt{(x^2 + y^2) / R_1^2} \quad (5)$$

在实际应用中,该阵型的几何算法还可简化为:

$$\varphi = \arctan(d_y / d_x) \quad (6)$$

$$\theta = \arccos\sqrt{(d_x^2 + d_y^2) / L^2} \quad (7)$$

式中:

$$\begin{cases} d_x = \frac{d_{21} + d_{34} + d_{31} - d_{42}}{4} \\ d_y = \frac{d_{41} + d_{32} + d_{31} + d_{42}}{4} \end{cases} \quad (8)$$

上面两式是平面四元方阵的最佳算法。

四元十字阵定位中声源的方向角和仰角的均方误差表达式是:

$$\begin{cases} \delta_\varphi = \frac{1}{2L\cos\theta}\delta_d \\ \delta_\theta = \frac{1}{2L\sin\theta}\delta_d \end{cases} \quad (9)$$

式中: δ_d 表示声程差 d_{ij} 估计的均方误差。

3 广义互相关时延估计算法

四元十字阵可以使估计方位角时精度提高,但是,该阵在估计距离时会产生一定估计误差,因此,时延估计算法的选取很重要,时延估计算法是利用时延估计来完成目标的联合测向和测距,其中时延就是声源到达各麦克风的时间差。声源的位置是由传声器阵列中各组时延值和传声器阵列的几何关系得到的。本文中我们选取广义互相关时延估计法来进行时延估计。广义互相关法以基本互相关为理论基础,通过求两信号之间的互功率谱,并在功率谱域内给予一定的加权,再反变换到时域得到两信号之间的互相关函数,最终估计出两信号之间的时延。

3.1 传统互相关法

设麦克风 1、2 接收信号的离散时间信号模型为:

$$x_1(n) = \alpha_1 s(n - \tau_1) + n_1(n) \quad (10)$$

$$x_2(n) = \alpha_2 s(n - \tau_2) + n_2(n) \quad (11)$$

式中: $s(n)$ 为声源信号, $n_1(n)$ 、 $n_2(n)$ 是高斯白噪声。 $s(n)$ 、 $n_1(n)$ 和 $n_2(n)$ 两两互不相关, τ_1 、 τ_2 分别是声波从声源到麦克风 1、麦克风 2 的传播时间, $\tau = \tau_1 - \tau_2$ 是两麦克风间的时延。 $x_1(n)$ 、 $x_2(n)$ 的互相关函数 $R_{12}(\tau)$ 可表示为:

$$R_{12}(\tau) = E[x_1(n)x_2(n - \tau)] \quad (12)$$

将式(10)和式(11)代入式(12),可得:

$$\begin{aligned} R_{12}(\tau) = & \alpha_1 \alpha_2 E[s(n - \tau_1)s(n - \tau_2 - \tau)] + \\ & \alpha_1 E[s(n - \tau_1)n_2(n - \tau)] + \\ & \alpha_2 E[s(n - \tau_2 - \tau)n_1(n)] + E[n_1(n)n_2(n - \tau)] \end{aligned} \quad (13)$$

由于 $s(n)$ 、 $n_1(n)$ 和 $n_2(n)$ 两两互不相关,式(13)变为

$$\begin{aligned} R_{12}(\tau) = & \alpha_1 \alpha_2 E[s(n - \tau_1)s(n - \tau_1 - \tau)] \\ = & \alpha_1 \alpha_2 R_{ss}(\tau - (\tau_1 - \tau_2)) \end{aligned} \quad (14)$$

根据自相关函数的性质: $R_{ss}(\tau) \leq |R_{ss}(0)|$, 可知当 $\tau - (\tau_1 - \tau_2) = 0$ 时, $R_{12}(\tau)$ 取其最大值, 因此求得 $R_{12}(\tau)$ 取最大值时对应的 τ , 就是两个麦克风之间的时延 τ_{12} 。

3.2 广义互相关法

传统互相关时延估计法原理简单, 运算量小。然而在实际环境中, 由于噪声和混响的影响, 相关函数的最大峰值会被弱化, 有时还会出现多个峰值, 这些都造成了实际峰值检测的困难^[18]。广义互相关

法以基本互相关为理论基础, 通过求两信号之间的互功率谱, 并在功率谱域内给予一定的加权, 再反变换到时域得到两信号之间的互相关函数, 最终估计出两信号之间的时延, 其原理图如图 3 所示, 其中 $(\cdot)^*$ 表示共轭运算。

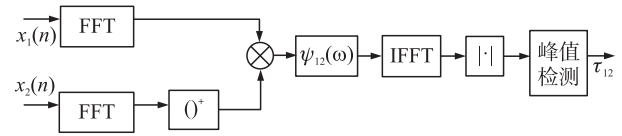


图 3 广义互相关法原理框图

根据以上分析, 我们先把两路信号的自相关函数做傅里叶变换, 得第 1 个和第 2 个麦克风接收信号的互功率谱:

$$\Phi_{x_1 x_2}(\omega) = \alpha_1 \alpha_2 \Phi_{ss}(\omega) e^{-j\omega\tau_{12}} \quad (15)$$

式中: $\Phi_{x_1 x_2}(\omega)$ 、 $\Phi_{ss}(\omega)$ 分别为 $R_{12}(\tau)$ 、 $R_{ss}(\tau)$ 对应的功率谱。对式(15)加权后做反傅里叶变换, 可得到广义互相关函数:

$$R_{x_1 x_2}(\tau) = \int_{-\infty}^{\infty} \Phi_{x_1 x_2}(\omega) \Psi_{12}(\omega) e^{j\omega\tau} d\omega \quad (16)$$

式中: $\Psi_{12}(\omega)$ 为广义加权函数, 实际应用时, 可针对不同的噪声和混响情况可以选择不同的 $\Psi_{12}(\omega)$ 。

$\Phi_{x_1 x_2}(\omega)$ 表示 $x_1(n)$ 、 $x_2(n)$ 的互功率谱。 $|\gamma_{12}(\omega)|^2$ 的定义为:

$$|\gamma_{12}(\omega)|^2 = \frac{|\Phi_{x_1 x_2}(\omega)|^2}{\Phi_{x_1 x_1}(\omega) \Phi_{x_2 x_2}(\omega)} \quad (17)$$

广义互相关法建立在非混响模型基础上的, 由于受此种模型影响, 广义互相关法不适用于有多个声源以及方向性的干扰噪声的情况。该方法在单声源、非相关噪声和低混响的环境中效果很好。另外它还具有运算量小、算法复杂度低、易于实现等优点。

4 仿真实验结果

仿真中选取一段语音段长为 1.12 s、采样频率为 44 100 Hz 的信号作为声源。首先对信号进行预处理, 再将经过端点检测的信号, 即含语音信号, 用来做时延估计, 这样降低了计算量。声源信号的波形如图 4 所示, 其中红色竖线为语音端点检测的结果。

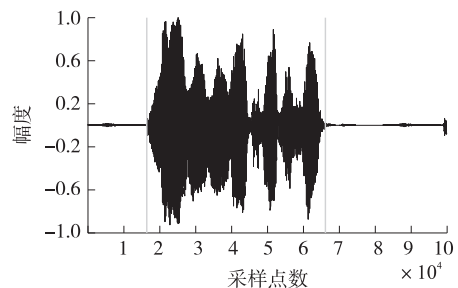


图 4 声源信号波形及端点检测结果

在仿真实验中,房间的尺寸为 $6\text{ m}\times 6\text{ m}\times 3\text{ m}$,声源坐标为 $[4\text{ m}, 4\text{ m}, 1\text{ m}]$,两个麦克风的坐标分别为 $[1\text{ m}, 1\text{ m}, 1\text{ m}]$ 、 $[4\text{ m}, 1\text{ m}, 1\text{ m}]$ 。用 Image 法产生房间脉冲响应,考虑无混响的情况下,设置房间墙

壁反射系数为 0,图 5 为声源到两麦克风的传递函数。再将声源信号与传递函数卷积,就分别得到了两个麦克风的接收信号。由图 5 可以看出,两麦克风间的时延差为 59 个点。

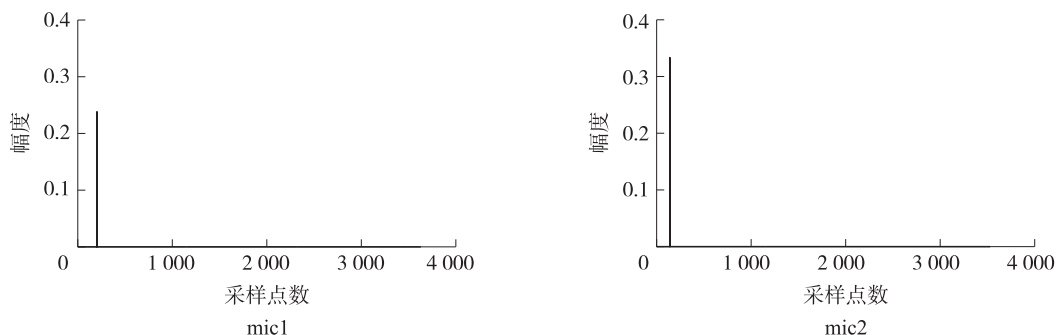


图 5 无混响时声源到两个麦克风的传递函数

实验分别在信噪比为 10 dB、0 dB、-5 dB 情况下分别对传统相关算法和广义互相关算法进行了对比。该实验中帧长为 2 048,使用的声源波形如图 6 所示。

从图 6 可以看出,在无混响、高信噪比的情况下,广义互相关算法都有较高的稳定性。随着信噪比的降低,两种互相关算法的波动都有所增加,广义互相关算法相对传统互相关算法较为稳定。

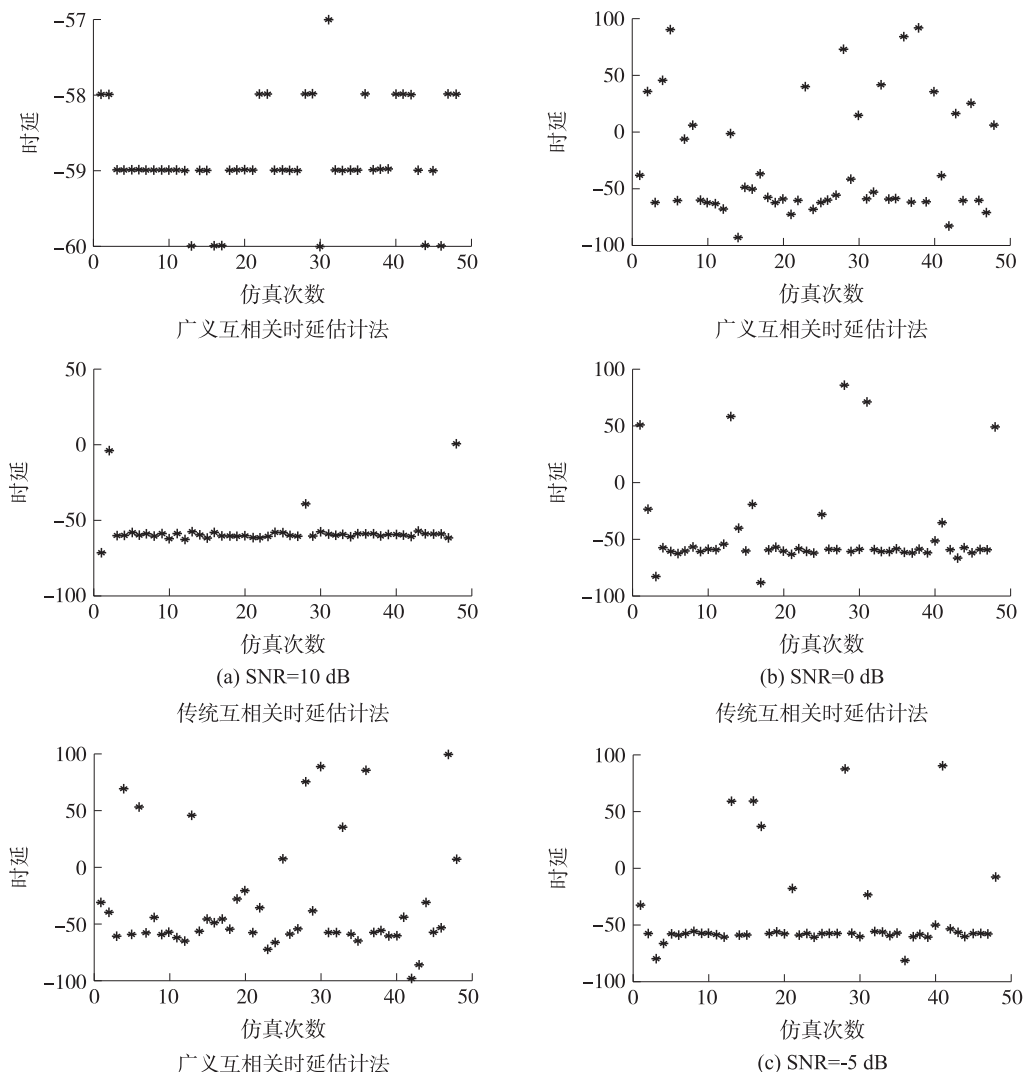


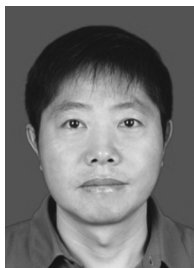
图 6 无混响情况下两种互相关算法在不同信噪比时的对比

5 结论

本文提出了一种基于广义互相关时延估计的四元十字阵的实时声源定位算法,在此基础上设计了一套实时声源定位系统。在综合考虑功耗限制,算法复杂度,数据处理能力,多路音频输入及接口限制等等因素后,选择了以 Cortex-A8 嵌入式平台的声源定位系统,并对系统硬件的设计等做了阐述。仿真实验表明,提出的算法的鲁棒性优于传统互相关法的声源定位算法。

参考文献:

- [1] 杜江,朱柯. 智能麦克风阵列语音分离和说话人跟踪技术研究[J]. 电子学报,2005,33(2):382-384.
- [2] 柯昆. 声源定位技术研究[D]. 西安:西安电子科技大学,2010.
- [3] 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.
- [4] Flanagan J L. Bandwidth Design for Speech-Seeking Microphone Arrays[C]//Proceedings of ICASSP85, Tampa, FL, 1985:732-735.
- [5] Kellerman W. A Self-Steering Digital Microphone Array[C]//Proceedings of ICASSP91, Toronto, CA, 1991:3581-3584.
- [6] Che C, Q Lin. Microphone Arrays and Neural Networks for Robust Speech Recognition[C]//Proceedings of the Human Language Technology Workshop, Plainsboro, NJ, 1994:342-347.
- [7] Giuliani D, M Omologo. Talker Localization and Speech Recognition Using a Microphone Array and a Cross-Power Spectrum Phase Analysis[C]//Proceedings of ICSLP, 1994, Volume 3:1243-1246.
- [8] Michel O, Clergeot H. Multiple Source Tracking Using a High Resolution Method[J]. Acoustics, Speech, and Signal Processing, 1991, Volume 2:1277-1280.
- [9] Grenier Y. A Microphone Array for Car Environments[C]//Proceedings of ICASSP92, San Francisco, CA, 1992, Volume I:305-309.
- [10] Oh S, V Viswanathan. Hands-Free Voice Communication in an Automobile with a Microphone Array[C]//Proceedings of ICASSP92, San Francisco, CA, 1992:Volume I:281-284.
- [11] Flanagan J, Surendran A. Spatially Selective Sound Capture for Speech and Audio Processing[J]. Speech Communication, 1993, 13(1-2):207-222.
- [12] Jan E, Svaizer P. Matched-Filter Processing of Microphone Array for Spatial Volume Selectivity[C]//Proceedings of ISCAS95, Piscataway, NJ, USA, 1995:1460-1463.
- [13] Flanagan, Johnson J D. Computer-Steered Microphone Arrays for Sound Transduction in Large Rooms[J]. Acoust Soc Amer, 1985, 78(5):1508-1518.
- [14] Greenberg J E, Zurek P M. Evaluation of an Adaptive Beamforming Method for Hearing Aids[J]. Acoust Soc Amer, 1992, 91:1992-1676.
- [15] Michel O, Clergeot H. Multiple Source Tracking using a High Resolution Method[J]. Acoustics, Speech, and Signal Processing, 1991, 2:1277-1280.
- [16] 崔玮玮,曹志刚,魏建强. 基于双麦克风的2维平面定位算法[J]. 信号处理,2008,24(2):299-302.
- [17] Samsung. S5PV210 RISC Microprocessor User's Manual Revision 1.10. 2010:1-15.
- [18] 李雪梅,陶然,王越,等. 时延估计技术研究[J]. 雷达科学与技术,2010,8(4):362-367.



王 永(1972-),男,江苏铜山人,硕士,副教授,徐州工业职业技术学院,主要研究方向为计算机应用与电子技术,wy_040618@163.com。