文章编号: 1001-2486 (2000) 04-0045-06

虚拟战场的实时四声道立体声合成*

曾亮, 李思昆

(国防科技大学计算机学院, 湖南 长沙 410073)

摘 要: 面向虚拟战场提出了实现实时四声道立体声合成的简化算法。针对四声道立体声的特殊性, 首先探讨了四声道立体声系统中扬声器的配置方案。在四声道立体声系统扬声器配置方案的基础上, 解决了单声源的表示、声强衰减及实时立体声化问题, 然后提出一种评价混声效果的标准及一个评价声源对整个声音所作贡献的公式, 并由此提出一个混声公式, 从而解决了多声源环境的实时混声问题。该套算法用于应用系统, 取得了较好的效果。

关键词: 四声道立体声; 虚拟战场; 虚拟现实
中图分类号: TP391.9 文献标识码: A

Realizing a Real-time 4-Channel Stereo Sound Environment for Virtual Battlefield

ZENG Liang, LI Si-kun
(College of Computerm, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: Real-time algorithms are given to generate 4-channel stereo sound for virtual battlefield. For the special case of 4-channel stereo sound, first we studied the speaker layout scheme of 4-channel stereo sound system. Based on the given speaker layout, we dealt with the attenuation of sound intensity and turned a mono sound into a stereo one. Then, a criterion for evaluating the effect of sound mixing is proposed. An algorithm based on this criterion is also given to mix sounds randomly in real-time. The algorithm has been successfully applied in one of our applications to realize a real-time 4-Channel Stereo Sound environment for virtual battlefield. The effect is quite good.

Key words: 4-channel stereo sound; virtual battlefield; virtual reality

分布式虚拟战场系统是典型的多用户虚拟现实应用, 它通过局域网或者广域网络由分布的各种交互设备将成员引入由计算机合成的虚拟战场环境中, 对作战单位成员进行训练和执行相关课题的研究任务。虚拟战场环境下成员的沉浸感, 来自于良好的人机接口界面, 并为仿真的可信度和可用性提供保障。虚拟战场环境下, 成员与环境之间信息的交互在很大程度上依赖于视觉通道和听觉通道。真实的合成空间声音将大大增强环境的沉浸感。听觉信息一方面为视景伴音, 对视觉效果进行增强和渲染, 另一方面独立提供重要信息。由于人对环境中声音信息的接收没有像接收视觉信息那样受角度和方位的限制, 通过声音可以获得不可见事件或实体的行为、状态信息, 无疑是对视觉信息的重要补充, 这在战场环境下更有特殊重要的意义。随着计算机软硬件技术的发展以及对合成环境逼真度要求的提高, 三维立体声的合成与播放技术已经成为必需和可行。

面向虚拟战场的 VR 环境中, 由于多通道交互的相互辅助, 对立体声精确性的要求适度降低, 我们由此提出一个实现实时四声道立体声合成的简化算法。这套算法针对四声道立体声的特殊性, 首先解决了四声道立体声系统中扬声器的配置方案。在此基础上通过对 WAV 格式的声音文件的处理, 具体解决了单声源声强的衰减、实时立体声化和多声源环境的实时混声等几个关键问题。

1 扬声器的配置方案

在利用 PC 机实现立体声的方案中, 往往采取双声道耳机来实现立体声的输出。采用双声道耳机

* 收稿日期: 2000-02-17
基金项目: 国家 863 计划资助项目 (863-306-ZD10-02-3)
作者简介: 曾亮 (1970-), 男, 博士后。

的好处在于实现简单，对于单人立体声环境也确实起到较好的效果，故而在PC机的三维虚拟现实游戏中，通常采用双声道耳机作为假定的输出，如“三角洲部队”等。显而易见，双声道耳机系统只能针对单人环境，对于多人环境，则需要采用扩张式的扬声器系统。采用扬声器系统所要面临的两个主要问题是，扬声器的配置以及原音场的再现问题（即立体声的实时合成与播放问题）。

这里首先讨论四声道扬声器的配置问题。采用四个喇叭，扬声器的配置方案如图1所示。

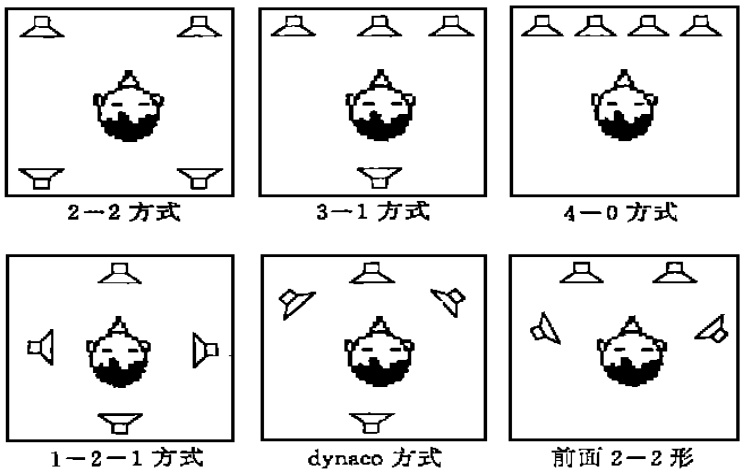

图1 四声道立体声系统喇叭配置

Fig. 1 4-channel stereo sound system speakers layout

在四声道立体声系统中，四个喇叭所放出来的声音，不再是单纯间接音，可将其更加积极地利用，通过声音移动、正面和背面声音交互配置演奏，加强回音、延迟效果等达到现实中原音场再现。上图2—2方式能够使听众有被声音包围的感觉，实现声源定位简单方便，为大多数人所接受，这也是EIA的标准方式，美国VANGUARD公司生产的磁带就采用这种方式。4—0方式则主要用于实现西洋歌剧和音乐性等舞台感受，但听众没有被声音包围的感觉，与2—2方式没有互换性，故今后不太可能被继续使用。而3—1配置，是取用2—2和4—0两种方式的折衷方式，以前面三个喇叭来获得声音定位，用后面的一个喇叭，来截取临场的感受，但在实际中不被采用。而对于菱形方式的配置（1—2—1配置），采用两个侧面和正面喇叭配置，来表现正面定位和音域，以获取临场感，dynaco公司的西洋风筝形配置，拾取3—1与1—2—1配置的折衷配置方式，这两种方式与2—2方式之间，存在有互换性。接近4—0方式的前面2—2方式，用四个喇叭所围起的内部，产生较深的舞台效果，对于对来自背后声音有排斥感的人，是一种很好的方法，由于其缺乏声音包围的感觉，我们这里不采纳。

通过上面的讨论可以看到，2—2方式声音包围的感觉好，实现声源定位简单方便，我们采用这种方式作为虚拟战场四声道立体声系统喇叭的配置方式，下面所要研究的立体声合成方法是采用此方式作为研究基础的。

2 声源的表示

我们令任一声源 $SoundSource = \langle Data, v_{max} \rangle$，其中 $Data$ 是经过标准化的WAV文件的声强数据，而 v_{max} 是这种声源最大音量。这种把声强和音量分开，并对声强数据标准化的表示方法，是本文实时立体声环境生成的基础。

声源的原始声强数据是通过录音设备采样和线性量化后得到的，并存储成PCM格式的WAV文件。为便于处理，我们统一按照22kHz的采样率采样，并存储成16位单声道的WAV文件。

原始声强数据受录音话筒位置的影响较大，为消除这种影响，我们对原始声强数据标准化。令nsamples是WAV文件的样本总数，$data[i]$ 是原始声强数据（无符号整数）的第 i 个样本，$MaxValue$

是录音格式允许的最大声强数据, 因为录音格式为 16 位, $MaxValue = 0xffff$, 则标准化方法如下:

(1) 遍历原始声强数据, 取 $wMax = \max\{data[i] \mid i = 0, \cdots, nsamples - 1\}$; 取 $wMin = \min\{data[i] \mid i = 0, \cdots, nsamples - 1\}$;

(2) 对于 $i = 0$ to $nsamples - 1$, $Data[i] = [(data[i] - wMin) \times MaxValue/(wMax - wMin)]$。

对于每类声源, 录音后根据实际情况规定它的最大音量, 也就是距离声源为 0 处没有衰减的音量, 这种音量的单位可以采用绝对的声音标准, 也可以采用应用内统一的相对音量。若使用相对音量, VR 环境中某个声源发声时, 需要首先将相对音量转换为声卡的物理音量。

3 单声源的立体声合成

作战实体在空间的某一位置发声, 声音相对于听者的位置信息就可以被感知, 定位是检验立体声合成的重要标准。在真实世界, 人对声音的空间位置的判断可以用距离和方位来表示, 受以下几个因素的影响: (1) 音量 。发声对象距离听者越近, 音量越大, 距离越远, 则音量越小, 这种现象称为衰减 (Rolloff)。(2) 接受差异。人对声音的接受器官, 即左耳和右耳, 对同一声音的接受在强度及时间上存在微弱差值——水平面内的混响时间差 (ITD) 和混响压力差 (IPD), 这是对声源进行定位的主要线索。(3) 消音效果。人外耳廓的物理构形特征使得从听者后部传来的声音与从前部传来的声音相比被轻微消音。(4) 头部相关传递函数 (HRTF) 。HRTF 是基于声音位置和考虑许多声音定位因素的一个线性函数, 用于描述人的听觉系统对不同频率声音的感知能力, 不同的人具有不同的 HRTF, 即使同一个人, 在头部周围不同的方位也具有不同的 HRTF。

就虚拟战场而言, 环境中声音的频率较低、数目较多, 声音距离听者较远; 多人参与, 使用扬声器而不是耳机来播放声音, 则 HRTF 可以不需要考虑; 由于音频较低, 混响时间差对定位起主要作用。根据这种情况, 为提高实时性, 我们通过分别计算声音在距离上的衰减和相位上的差异来实现单声源的立体声合成。

3.1 声强的衰减

声音的衰减使用户能够判断声源的距离。声音的衰减跟空气介质、风向、距离、障碍等物理因素有直接的关系。在虚拟战场中, 主要考虑距离对衰减的影响。

由于衰减, 声音在距离声源一定范围内有效, 我们称该范围为衰减范围或最大可听距离。不同声源最大可听距离不同, 超出这个距离的被认为声强为 0, 即听不到。

我们只对声源的实际播放音量按照距离进行衰减处理, 避免对声强数据逐一处理, 以便提高实时性。这个距离就是声源位置同用户头部中心位置之间的距离, 记为 r, 记 r_{max} 为最大可听距离, v_{max} 为声源的最大音量, v 为衰减后的实际播放音量, 则具体衰减公式如下:

$$\begin{cases} v = v_{max} \times (1 - r/r_{max}) & r \leqslant r_{max} \\ v = 0 & r > r_{max} \end{cases} \quad (1)$$

3.2 立体声化

对单声源立体声化, 就是要求得立体声音在每个声道上的声强分量, 通过对这些声强分量的感受, 用户能够准确判断声源方向。

如图 2 所示, 扬声器按照地球坐标系成 2—2 方式配置, D 为水平或垂直方向一对扬声器之间的距离, θ 为声源方向(这里指声源与听众的连线) 相对于地球坐标系 N 轴的夹角, 令采样率为 q 样本/s。

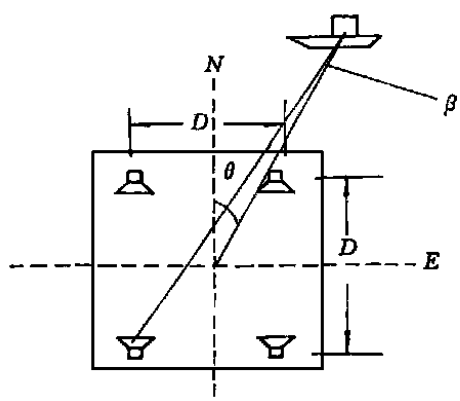

图 2 声源与扬声器的相对位置

Fig 2 The relative position of sound source and speakers

我们可以根据声源与听众在实际场景中的相对位置, 确定 θ。由 θ 所处的象限, 可以得到与声源最近的扬声器, 即最早发声的扬声器。声源到此扬声器和其它三

个扬声器的距离差分别为 $|D\sin\theta|$ (水平方向), $|D\cos\theta|$ (垂直方向), $|\sqrt{2}D\cos\beta|$ (对角方向, β 为声源方向与对角方向连线的夹角), 时间差分别为 $\Delta t_1 = |D\sin\theta|/c$, $\Delta t_2 = |D\cos\theta|/c$, $\Delta t_3 = |\sqrt{2}D\cos\beta|/c$, 其中, c 为空气中的声速。则四个扬声器相位差分别为

$$\Delta\varphi_0 = 0 \quad (\text{个样本})$$

$$\Delta\varphi_1 = \Delta t_1 \times q = |D\sin\theta| \times q/c \quad (\text{个样本})$$

$$\Delta\varphi_2 = \Delta t_2 \times q = |D\cos\theta| \times q/c \quad (\text{个样本})$$

$$\Delta\varphi_3 = \Delta t_3 \times q = |\sqrt{2}D\cos\beta| \times q/c \quad (\text{个样本})$$

令标准化声音数据缓冲区长度为 *datasize*, 样本总数为 *nsamples*, 每个样本字节数为 *bytespersample*, 第 i 个样本数据为 *Data[i]*; 对应于第 i 个样本的四个扬声器声道的数据分别为 *speakerdata[i, j]*, 样本总数为 *newnsamples*, $\Delta\varphi[j]$ 是根据上面三个公式计算出的扬声器相对于该声源的相位差, 则单声源立体声化的算法如下:

(1) 根据声源与听众在实际场景中的相对位置, 计算 θ, 由 θ 计算扬声器之间的相位差。

(2) 计算立体声数据的样本数和缓冲区长度:

i) newnsamples = nsamples + $\Delta\varphi$[j];

ii) newdatasize = 2 * newnsamples * bytespersample;

iii) 按照 newdatasize 分配缓冲区;

(3) 合成立体声数据:

i) 当 i = 0 to $|\Delta\varphi[j]| - 1$ 时, 若 $\Delta\varphi[j] = 0$, 则 speakerdata[i, j] = Data[i];
否则 speakerdata[i, j] = 0;

ii) 当 i = $|\Delta\varphi[j]| \rightarrow$ nsamples − 1 时,
若 $\Delta\varphi[j] = 0$, 则 speakerdata[i, j] = Data[i];
否则 speakerdata[i, j] = Data[i + $\Delta\varphi$[j]];

iii) 当 i = nsamples to newnsamples[m] − 1 时,
若 $\Delta\varphi[j] = 0$, 则 speakerdata[i, j] = 0;
否则 speakerdata[i, j] = Data[i + $\Delta\varphi$[j]]。

4 实时混声

4.1 已有的方法及存在的问题

虚拟战场中常常同时存在多种声源, 例如击中发生时, 爆炸声伴随解说词等, 这就需要在扬声器声道中混合多种声音。目前混声主要采用硬件合成和软件叠加两种方法。硬件合成使用多个声卡通过软开关选择要合成的声音, 如 FM、MIDI 等; 软件叠加则通过计算完成。

目前常用的软件叠加公式[1]是:

$$newdata[i] = \left(\sum_{j=0}^{n-1} sourcedate[j, i] \right) / n \qquad (2)$$

其中 *sourcedata[j, i]* 是第 j 个声源的第 i 个样本。也有根据实际情况加权叠加的。

公式 (3) 除以 n 是为了防止声强数据叠加溢出而产生噪声, 目前 WAV 声强数据一般存储为 8 位或 16 位, 为保证叠加正确性, 叠加在与 PCM 格式相同的 WAV 声强数据之间进行。

我们曾经按照公式 (3) 进行了实时混声, 效果不好, 主要问题是声强失真。一个新的声音同已有的声音混声时, 由于新老声强按照 $n = 2$ 的方式均被削弱, 用户听起来已有声音突然变小, 随着新声音不断加入, 原有的声音越来越小; 当某个声音播放完毕时, 声强恢复, 用户听起来声音突然变大。这种失真使得稳定的声源听起来忽强忽弱, 影响虚拟战场的声音效果。

4.2 评价混声效果的标准

虚拟战场要求各声源的声强基本不失真, 每种声音都比较清晰, 不因混声而受到影响。

我们认为保证声强不失真且声音比较清晰的前提是, 保证每个声源无论在何种情况下对整个声音所作贡献都能够保持恒定。根据实践经验, 我们提出如下评价声源对整个声音所作贡献的公式:

$$contribution[j, t] = e[j, t] \times V[t]; \quad (3)$$

$j = 0, 1, \cdots, n, \cdots$ 标记第 j 个声源, t 是当前时刻, $e[j, t]$ 是 t 时刻正在播放样本的声强数据同其初始声强数据(经历标准化、立体声化后的声强数据)之间的比例, $V[t]$ 是 t 时刻系统的播放音量。只有保证任意参与混声的声源 j 的 $contribution[j, t]$ 在任何情况下保持不变, 才能获得稳定的声音效果。

容易证明, 利用公式 (3) 进行混声计算时, 随着新声源的加入, 单个已有声源对整体声音的贡献依次削弱为原来的一半, 不能保持恒定, 这也就是其混声过程中已有声源声音忽大忽小的原因。

4.3 混声方法

根据上述混声标准, 为了达到虚拟战场对混声的要求, 我们提出如下混声公式:

$$newdata[i] = \sum_{j=0}^{n-1}\left(sourcedata[j, i] \times \frac{v[j]}{V}\right), \text{ 其中 } V = \sum_{j=0}^{n-1} v[j] \quad (4)$$

$v[j]$ 是第 j 个声音的实际音量, V 是虚拟战场声音环境的实际播放音量。

令已有声音的声强数据是 $olddata$, 系统实际播放音量为 V, 那么, 当一个实际音量为 v、声强数据为 $sdata$ 的新声音加入到已有声音中去时, 播放时间重叠的样本的新声强数据和系统实际播放音量依次按照下列公式计算:

$$newdata[i] = \frac{V}{V+v} \times olddata[i] \times \frac{v}{V+v} \times sdata[i]; \quad (5)$$

$$V = V + v; \quad (6)$$

当某个声源停止发声时, 因为它已经没有声强数据参与混声, 所以由下述公式降低因其加入而增加的音量:

$$V = V - v \quad (7)$$

上述混声公式涉及音量和声强数据两个方面的叠加, 能够保证声强数据不越界, 防止引入叠加噪声; 还能够保证混声过程中声源对整个声音的贡献保持恒定。

5 面向虚拟战场的实时四声道立体声合成算法

我们根据 2, 3, 4 节的方法实现了面向虚拟战场实时合成四声道立体声的如下简化算法:

(1) 预处理:

i) 按照 22kHz 的采样率分别录制各种声源的声音, 存储为 16 位单声道 WAV 文件;

ii) 按 2 节方法对每个声音文件进行标准化, 使其声强数据范围为 [0, 0xffff];

iii) 设定每种声源播放的最大音量 v_{max} 和最大可听距离 r_{max}。

iv) 为各种声源建立如下记录:

```
struct sourcetype { char wavefilename[30] ;     // 标准化后的声音文件名。
                    double v_max;     // 最大播放音量。
                    double r_max;     // 最大可听距离。
                    double duration;     // 声音持续时间。} SoundSourceType [Num] 。
```

(2) 初始化

i) 初始化用于记录当前正参与播放声源的混声表, 混声表将按照 *endtime* 升序排列, 其表项结构为:

```
struct mixlist { long typenum;   // 声源的类型编号。
                 struct point location;   // 声源位置。
                 double v;   // 实际播放音量。
                 TIME endtime;   // 声音播放停止的墙上时刻。
                 struct mixlist * next; } *  MixList;
```

ii) 初始化系统播放音量 V，令 $V = 0$。

(3) 实时混声:

i) 有新声源加入播放时:

(a) 建立混声表表项:

计算声源到听者头部中心的距离 r; 按照公式(1) 对音量进行衰减, 计算实际音量 v; 按照公式(2) 计算每个扬声器的相位差, 并按照3.2节的算法将声音立体声化; 取当前墙上时间 *currenttime*; 计算 *endtime* = *currenttime* + *duration*; 按照 *endtime* 升序排列，将这个声源插入到混声表中。

(b) 取当前系统声音播放位置，抛弃已播放完毕的样本; 将剩余样本中同新声音样本播放时间重叠的样本按照公式 (5) 进行新声强数据的计算; 不重叠的部分直接拷贝，不需计算; 按照公式 (6) 计算当前 V。

(c) 如果新声源在混声表首部，则清除当前闹钟，并按照 *endtime* 重设闹钟。

(d) 停止当前声音的播放; 按照新声强数据重置播放缓冲区和播放设备，调整系统播放音量，开始播放。

ii) 处理闹钟中断:

(a) 取当前墙上时间 *currenttime*;

(b) 循环，直到混声表为空或第一个表项的 *endtime* > *currenttime*:

根据公式 (7) 和混声表的第一个表项，重新计算当前播放音量 V; 删除第一个表项;

(c) 调整系统播放音量;

(d) 若混声表为空，清除当前闹钟; 否则按照混声表第一个表项的 *endtime* 重置闹钟。

该算法的预处理步是脱机进行的，实时执行的时间复杂度集中在以爆发方式执行的第 (3) 步中。第 (3) 步的第 ii) 子步的时间开销较小，对时间复杂度不产生影响，开销较大的是第 i) 子步。令 n 为新加入声源的初始样本数，由于一个系统的扬声器数是常数，所以对该声源实时立体声化的时间复杂性为 $O(n)$。由于混声计算样本叠加的最坏时间复杂性也是 $O(n)$，则整个算法的时间复杂性为 $O(n)$。

6 结束语

我们在配置为 Intel Pentium II 266MHz 的 CPU，主存 128MB，硬盘 6.4GB，使用 4 块 PCI 声卡的微机上，利用 Microsoft 的 Visual C++ 6.0 实现了本文算法，并已经成功地将这个算法应用于虚拟战场中。该算法能够达到实时混声，立体声效果较好; 立体声环境中的每种声音都比较清晰，对同时混音的数目基本没有限制，混音数目不影响立体声效果; 由于使用 16 位声强数据，标准化和混声计算取整引入的噪声很小，基本上对音质没有影响。由于文章的篇幅，我们这里不详细给出试验数据。

参考文献:

[1] 罗福元. 王行仁. 虚拟声音的生成与仿真研究 [A]. 中国系统仿真学会学术年会论文集 [C], 1997: 757-762.

[2] 汪成为, 高文, 王行仁. 灵境 (虚拟现实) 技术的理论、实现及应用 [M]. 北京: 清华大学出版社, 1996: 46-49, 224-227, 484-486.

[3] 秦加法. 用声卡实现逼真的音响模拟 [A]. 中国系统仿真学会学术年会论文集 [C], 1997: 971-974.

[4] Wenzel E M, et al. A virtual display system for conveying three-dimensional acoustic information [A]. Proceedings of the Human Factors Society Factors Annual Meeting [C], 1988: 86-90.