

局部搜索的音频数据检索

李 应

(福州大学 数学与计算机科学学院, 福建 福州 350108)

摘 要:根据多媒体音频数据的特点,提出一种适用于快速音频数据检索的局部搜索数据结构,即局部搜索树(local search tree, LS-tree).在局部搜索树中,分别以音频数据小波变换系数的过零率和平均幅度作为主、次关键码,基于局部范围对作为索引的其他系数进行组织.其次,基于局部搜索树,提出采用小波包最好基小波塔型算法实现音频数据检索.最后,把采用局部搜索树的小波包最好基一小波塔型算法的搜索和基于小波不同级系数的检索方法相比较,结果表明,这种方法对音频数据检索的快速和有效性.

关键词:音频数据检索;局部搜索树;小波包最好基;塔型算法

中图分类号: TP391.3; TP311.12 **文献标识码:** A **文章编号:** 1673-4785(2008)03-0259-06

An audio data retrieval method based on local search trees

LI Ying

(College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China)

Abstract: For fast audio data retrieval, a local search tree (LS- k_d tree) structure is proposed according to the characteristics of multimedia audio data. In an LS- k_d tree, the zero- k_d crossing rate and the average magnitude of the wavelet transform coefficients in the audio data are taken as main and secondary key codes respectively, and the other coefficients used for the index are organized in the local range. On the basis of this LS- k_d tree, an audio data retrieval approach is presented using the wavelet packet best base and the wavelet pyramidal algorithm. Finally, the research results obtained from the proposed approach are compared with those obtained using the different- k_d level wavelet transform coefficients. It is found that the proposed approach is effective and fast for audio data retrieval.

Keywords: audio data retrieval; local search tree (LS- k_d tree); wavelet packet best base; pyramidal algorithm

多媒体音频数据,已经成为网络、信息时代信息的重要组成部分.目前,对于基于例子的音频数据分类、索引、检索、组织和管理方法,已经引起广泛的研究.在文献[1]中, Samet对多维数据结构与多媒体数据管理中的常用算法进行了全面的描述.其中包括 K-D tree、点4分树、MX-4分树、R-tree的插入、删除、搜索和范围查询等方法.而 R-tree实际上已经成为目前多媒体数据组织和管理的基础,尤其,它对于在磁盘中存储大量图像数据的情况特别有用^[2]. R-tree结构的大致特点是:每个 R-tree有一个值为整数 K 的序.每一个非叶的 R-tree 结点包含一个矩形集合,这个集合最多有 K 个、至少有 $\lceil K/2 \rceil$ 个矩形(根结点例外).对于叶结点,每个结点包含一个实

矩形,而非叶结点包含一组矩形.在 R-tree 中进行插入一个矩形操作时,遵循原则:结点包含的矩形数 n 应满足 $n \leq \lceil K/2 \rceil, K$;当插入矩形时,必须使得与根结点相关的矩形所需要的扩张最少(以面积计算).当进行一个矩形的搜索操作时,从根结点开始,根据所查找矩形的坐标,递归地选择子结点,直到找到叶结点为止.与此相对照,在 R-tree 中删除一个对象,必须避免可能引起的 R-tree 中的结点“下溢”,确保结点至少包含 $\lceil K/2 \rceil$ 个矩形.

以 R-tree 为基础,衍生出来的 TV-tree,主要根据正被检索的数据,动态和柔性地决定怎样分支的问题^[3].这种数据结构多用于多媒体文档的搜索.同样衍生于 R-tree 的贞段树(frame segment tree)和 R 段树(R-Segment tree or RS-tree)适用于变长度的分段^[1].它常用于视频数据的搜索.近年来,又提出了适用于各种多媒体应用场合的其他空间数据结

收稿日期:2007-11-05.

基金项目:福建省自然科学基金资助项目(A0510006).

通讯作者:李 应. E-mail: fj_liying@sohu.com.

构,其中包括: R-tree 的分支嫁接算法^[4]、e-tree 结构^[5]、RS-tree^[6]、紧凑 R-tree^[7]、SS-tree^[8]、X-tree^[9-10]、SR-tree^[11]、AB-tree^[12]、LSDh-tree^[13]、hybrid tree^[14]等等.然而,这些多维数据结构主要是针对多媒体图象数据、视频数据和文档数据的组织和管理而提出的,它们大都不适用于多媒体音频数据的组织、管理和搜索.为此,本文针对基于例子和基于内容的音频数据检索,以小波包变换和小波多分辨分析为基础,提出一种可以高效地组织、管理和搜索音频数据的局部搜索树 (LS-tree),实现音频数据检索.

1 音频数据的检索方法

对于可以检索各种音频数据的索引产生方法,文献 [15] 提出了通过短时 Fourier 变换和小波变换产生索引的方法.由于音频信号是非平稳信号,而小波变换具有多分辨分析的特点,因此,采用小波变换产生索引比基于信号统计和基于短时 Fourier 变换产生索引的方法具有更好的检索精度.对于小波多分辨率分析,它可以按照不同的尺度因子 j 把 Hilbert 空间分解为所有小波子空间 $w_j(j=Z)$ 的正交和,即 $L(R)=\bigoplus_{j=Z} W_j$. 而小波包可以对 W_j 进行进一步分解,它克服了小波多分辨分析中当时分辨率高时,频率分辨率低的缺陷,因此具有更好的音频特性.对于不同类型的音频信号,它们的频谱差别较大,因此,采用小波包来分析具有更为理想的效果.

用小波包变换来分析音频数据,是根据不同的音频数据选择不同的最好基对音频数据进行分解.曾经以小波包最好基和小波多分辨分析为基础,用小波包最好基—小波塔型算法检索音频数据.这种方法的思路是,把音频数据分解成小波包最佳树结构系数 S 和 6 级、5 级、4 级等 3 个级别的离散小波逼近系数 CA6、CA5、CA4.在检索过程中,首先用 S 对音频数据进行初步分类;然后,再分别用 6 级系数 CA6、5 级系数 CA5 和 4 级系数 CA4 通过塔形算法进行检索^[16],并最后得出结果.这个过程如图 1 所示(为了方便分析,取数据集中文件数 $n=100$,且以“A+序号”表示音频数据文件名).

在图 1 中,由于音频数据检索的非确定性,检索需要对数据集的所有文件进行计算^[16].由于 CA6、CA5、CA4 这 3 组小波系数大小不等且维数变化较大(维数变化范围 10~20 000),采用现有的索引结构很难对这些系数进行有效管理和检索.为了实现这类数据的有效管理和搜索,利用时域分析中计算过零率和平均幅度的方法^[17-18],对维数最小的系

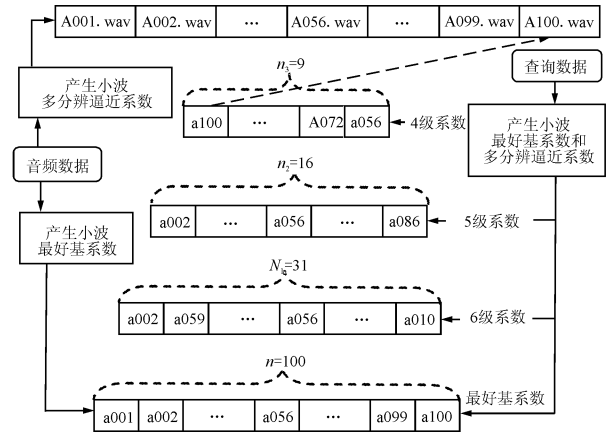


图 1 用小波最好基与塔型算法检索音频数据过程

Fig 1 The process with wavelet packet best base and pyramidal algorithm retrieval audio data

数 CA6 进行预处理,并生成 CA6 的过零率 (Z_d) 和平均幅度 (M_d). 通过这一对参数 (Z_d, M_d),提出并建立可以对 3 个级别的离散小波逼近系数 CA6、CA5、CA4 进行有效管理的局部搜索索引结构,即 LS-tree

2 关于 LS-tree

2.1 音频数据文件搜索中的问题

对于一个具有 n 个音频数据文件的集合,通过对它们相应的 6 级小波逼近系数 CA6 进行预处理,并生成相应的 n 对参数 $\{(Z_{d1}, M_{d1})\} \dots, (Z_{dn}, M_{dn})$. 如图 2 所示,以过零率 (Zero-crossing rate) 为横坐标、平均幅度 (Average magnitude) 为纵坐标,把这 n 对参数表示在二维空间上.

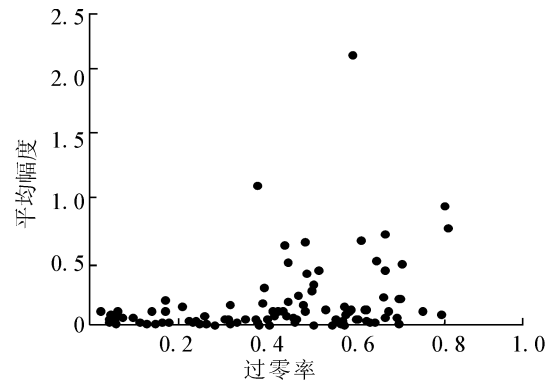


图 2 由过零率 (Z_d) 和平均幅度 (M_d) 组成的向量二维空间上的分布

Fig 2 The distributing of the vectors which are consisted of zero-crossing rates and magnitudes

为了提高检索速度和效率,必须把对整个平面的搜索,缩小到局部的搜索.以在数据集中查询按动快门的声为例,局部搜索的工作过程下.取按动快

门的声⻔片段,在含 n 个⻔频⻔件集中查询是否存在按动快门的⻔频⻔件. 这个查询过程是:首先,把待查的声⻔片段进⻔小波变换,产⻔该⻔频数据的最佳树结构系数 s 和 3 个级别的离散小波逼近系数 $cA6$ 、 $cA5$ 、 $cA4$. 然后,计算该⻔频的 $cA6$ 的过零率 Z_q 和平均幅度 M_q ,即查询数据的参数对 (Z_q, M_q) . 根据参数对 (Z_q, M_q) ,在 $\{(Z_{d_1}, \dots, M_{d_1}), \dots, (Z_{d_n}, M_{d_n})\}$ 中确定局部搜索范围 $\{(z_d, m_d) \mid ZDL \leq z_d \leq ZDU \ \& \ MDL \leq m_d \leq MDU\}$,其中, $ZDL = z_4 * Z_q$, $ZDU = z_2 * Z_q$, $MDL = m_1 * M_q$, $MDU = m_2 * M_q$,而 z_4 、 z_2 分别表示主关键码范围的下限系数和上限系数, m_1 、 m_2 分别表示次关键码范围的下限系数和上限系数. 最后利用小波包最好基—小波塔型算法来确定最匹配的⻔频数据. 例如,查询数据的参数 $(Z_q, M_q) = (0.0500, 0.1122)$,在图 3 中用 “+” 表示,则搜索范围为图 3 中的矩形包围部分,即 $\{(z_d, m_d) \mid 0.0166 \leq z_d \leq 0.0908 \ \& \ 0.0394 \leq m_d \leq 0.3245\}$,其中图 3 (b) 是图 3 (a) 的局部放大. 因此,希望用小波包最好基—小波塔型算法来确定最匹配的⻔频数据只需在矩型范围内进⻔,而不是整个平面.

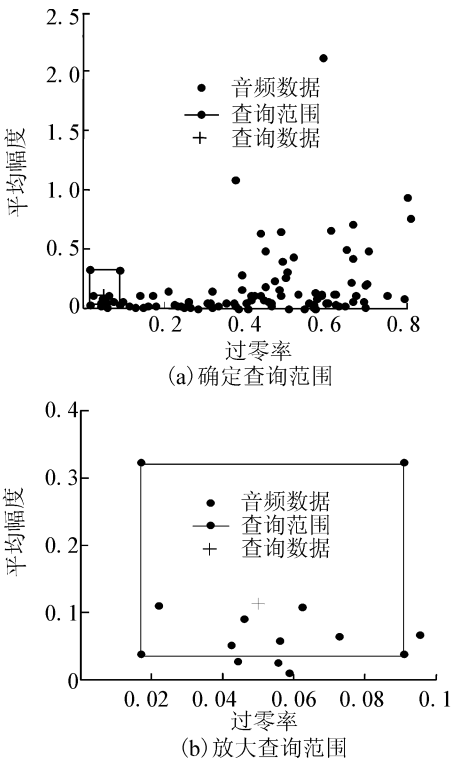


图 3 参数 $(Z_q, M_q) = (0.0500, 0.1122)$ 时的查询范围

Fig 3 The query range while the vector $(Z_q, M_q) = (0.0500, 0.1122)$

2.2 定义 LS-tree

为了实现局部搜索,定义 LS-tree 如下:

- 1) 树中每个非叶结点最多有 m 棵子树;有 n 棵子树的非叶结点有 n 对主关键码,每对主关键码指明其子树的主关键码的范围;
- 2) 除根结点外,其他的非叶结点至少有 $\lceil m/2 \rceil$ 棵子树;
- 3) 所有的叶结点都处在同一层次上,它们包含了全部主关键码、次关键码和指向相应数据对象存放地址的指针,且叶结点本身按主关键码从大到小、从右到左顺序连接;
- 4) 每个叶结点中的子树棵数 n 可以多于 m ,可以少于 m . 若设结点可容纳最大关键码数为 m_1 ,则指向对象的地址指针也有 m_1 个,因此,结点中的子树棵数 n 应满足 $n \in \lceil m_1/2 \rceil, m_1$ 在叶结点中,进⻔插入操作时,尽量使靠左的叶结点的子树棵数 $n = m_1$.
- 5) 对 LS-tree 的搜索从根结点开始,首先用主关键码范围进⻔搜索,当以主关键码范围的检索完成时,再把检索结果从左到右用查询数据的次关键码范围进⻔比较,取出符合次关键码范围的叶结点子树,根据该子树“其他参数指针 2”,进⻔小波包最好基—小波塔型算法的检索.
- 6) 其中,叶结点结构、非叶结点结构以及待查询数据的结构分别如图 4 (a)、(b) 和 (c) 所示.

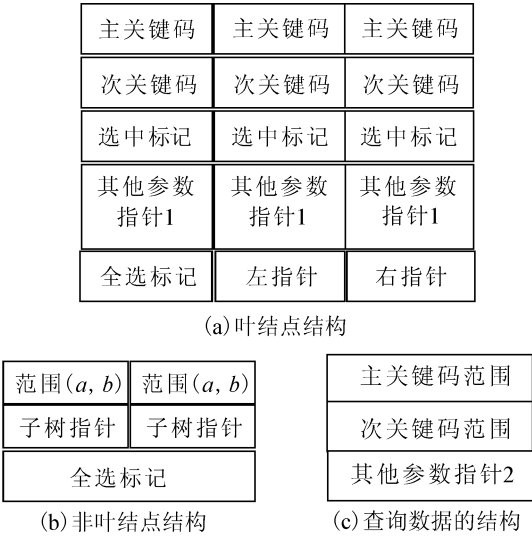


图 4 LS- k_d tree 的结点结构和查询数据的结构

Fig 4 The crunode structure of LS-tree and the structure of query data

2.3 LS-tree 的应用举例

在图 3 中取出部分分布点,形成表 1 所示的一组过零率 Z_d 和平均幅度 M_d 的参数对. 用这组数据,根据 LS-tree 的定义,可以生成图 5 所示的 LS-tree 其中叶结点上的 “*” 表示该结点上的其他部分,即图 4 (a) 中的次关键码、选中标记、其他参数指

针 1 等. 在图 5 中, 每个关键码的实际值为显示值 $\times 10^4$.

对于本例而言, 图 4(a)中“其他参数指针 1”主要包括: 音频数据的最佳树结构系数 SI 系数 CA6

的指针; 系数 CA5 的指针; 系数 CA4 的指针; 音频文件指针. 而图 4(c)中“其他参数指针 2”主要包括: 查询数据的最佳树结构系数 SI 系数 cA6 的指针; 系数 cA5 的指针; 系数 cA4 的指针.

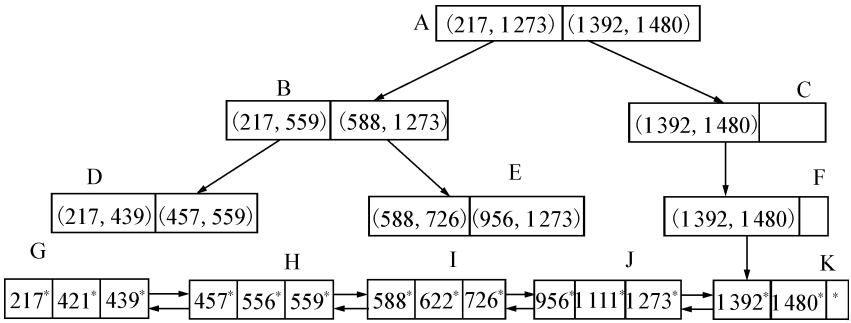


图 5 一棵 LS-tree
Fig 5 A LS-tree

表 1 一组过零率 (Z_d) 和平均 (M_d)
Table 1 A group of zero-crossing rate and average magnitude

分布点	过零率	平均幅度	分布点	过零率	平均幅度
1	0.021 7	0.011 4	8	0.095 6	0.067 2
2	0.072 6	0.065 9	9	0.058 8	0.010 7
3	0.148 0	0.011 1	10	0.043 9	0.027 6
4	0.139 2	0.111 6	11	0.127 3	0.012 9
5	0.045 7	0.091 9	12	0.111 1	0.023 1
6	0.062 2	0.109 6	13	0.055 6	0.025 1
7	0.055 9	0.058 8	14	0.042 1	0.052 5

现在, 利用基于例子的方法搜索上述中的一段 按动快门声音. 查询数据例子的参数同上 (Z_q, M_q) = (0.050 0, 0.112 2), 如果取关键码范围的下限、上限系数为: $z_1 = 0.333\ 3, z_2 = 1.815\ 2, m_1 = 0.351\ 4, m_2 = 1.815\ 2$ (z_1, z_2, m_1, m_2 通过实验确定), 那么, 主关键码范围是 $0.016\ 6 \leq z_d \leq 0.090\ 8$, 次关键码的范围是 $0.039\ 4 \leq m_d \leq 0.324\ 5$. 因此, 以 (Z_q, M_q) = (0.050 0, 0.112 2) 作为查询数据, 将返回符合 $\{ (z_d, m_d) \mid 0.016\ 6 \leq z_d \leq 0.090\ 8 \& 0.039\ 4 \leq m_d \leq 0.324\ 5 \}$ 的参数对.

要在图 2 中, 查出符合 $\{ (z_d, m_d) \mid 0.016\ 6 \leq z_d \leq 0.090\ 8 \& 0.039\ 4 \leq m_d \leq 0.324\ 5 \}$ 的点, 首先, 用主关键码范围 ($0.016\ 6 \leq z_d \leq 0.090\ 8$) 从根结点 A 开始搜索. 由于在 $[0.021\ 7, 0.127\ 3]$ 内, 所以沿左子树搜索到结点 B. 其次, 确定结果下界. 在结点 B 中, 由于 $[0.021\ 7, 0.055\ 9]$ 在 $0.016\ 6 \leq z_d \leq 0.090\ 8$ 内, 认为结点 B 的左子树之下的结点在所需

要的范围内, 因此这些内容不再进行搜索, 只需使相应结点 D、G 和 H 的“全选标记 = TRUE”; 同样, 也可以在结点 I 上标上“全选标记 = TRUE”. 这样, 可以确定叶结点 G、H 和 I 中的所有子树是满足 $0.016\ 6 \leq z_d \leq 0.090\ 8$ 的点. 再次, 确定次关键码搜索的内容. 根据表 1 可以知道, 这些点从右到左是: (0.072 6, 0.065 9) (0.062 2, 0.109 6) (0.058 8, 0.010 7) (0.055 9, 0.058 8) (0.055 6, 0.025 1) (0.045 7, 0.091 9) (0.043 9, 0.027 6) (0.042 1, 0.052 5) (0.021 7, 0.111 4). 最后, 用次关键码的范围 ($0.039\ 4 \leq m_d \leq 0.324\ 5$) 沿着“ \nearrow ”方向与子树中的次关键码进行比较, 并把满足次关键码的范围 $0.039\ 4 \leq m_d \leq 0.324\ 5$ 的结点选出. 它们是: (0.072 6, 0.065 9) (0.062 2, 0.109 6) (0.055 9, 0.058 8) (0.045 7, 0.091 9) (0.042 1, 0.052 5) (0.021 7, 0.111 4) 这样, 只要取出这些结点中“其他参数指针 1 所指的其他索引参数, 用小波包最好基—塔型算法进行检索, 进而减小检索范围, 提高检索效率.

3 搜索结果及不同方法的比较

3.1 基于 LS-tree 的最好基—塔型算法检索的结果

在实验中, 用 LS-tree 来管理和组织这些文件. 首先为图 2 所示的 100 个参数对 $\{ (Z_{d_1}, M_{d_1}), \dots, (Z_{d_{100}}, M_{d_{100}}) \}$ 建立起一棵 LS-tree. 然后, 以基于例子的方式对 100 个音频数据文件用小波包最佳树结构系数和小波塔型算法进行检索.

通过 LS-tree 对索引系数的管理, 使得需要进行小波包最佳树结构系数和小波塔型算法进行检索的

文件范围缩小.也就是说,在经过小波包最佳树结构系数和小波塔型算法检索之前,首先通过 LS-tree 进行搜索,只对这个搜索的结果文件进行小波包最佳树结构系数的分类.

经过小波包最佳树结构系数的分类,如果只把塔型算法的第一层检索作为结果,即把查询数据与音频数据的 6 级小波系数进行比较作检索结果.那么要分辨出 100 个音频数据文件,需要 2 582 次查询数据和音频数据 6 级小波系数的比较.如果以这层比较作为查询结果,那么查询的准确率为 71%.

如果在第 1 层检索结果的基础上,再进入第 2 层,把塔型算法的第 2 层检索作为结果,即把查询数据与音频数据的 5 级小波系数进行比较作检索结果.在我们的实验中,需要 519 次查询数据和音频数据 5 级小波系数的比较.由于 5 级小波系数的长度是 6 级小波的一倍,因此,如果以 6 级小波比较的时间衡量,那么增加的比较次数为 $519 \times 2 = 1\,038$ 次.此时查询的准确率可以达到 84%.

与第 2 层相似,如果再进入第 3 层比较,在第 2 层的基础上需要增加 257 次查询数据和音频数据的 4 级小波系数比较.这相当于增加 $257 \times 4 = 1\,028$ 次的 6 级小波系数比较的时间.经过第 3 层的搜索,准确率可以达到 94%.

3.2 不同方法的比较

对同样的 100 个音频数据文件用^[16]中的不同级小波系数检索方法进行检索.当用 6 级粗分辨逼近小波系数对音频数据集合中的数据进行搜索,并把该结果作为最终的搜索结果时,搜索准确率为 69%,并需要进行 $100 \times 100 = 10\,000$ 次 6 级粗分辨逼近小波系数的相关计算;用 5 级粗分辨逼近小波系数对音频数据集合中的数据进行检索,则准确率为 79%,需要进行相当于 $100 \times 100 \times 2 = 20\,000$ 次 6 级粗分辨逼近小波系数的相关计算;用 4 级粗分辨逼近小波系数对音频数据集合中的数据进行搜索,准确率为 89%,需要进行相当于 $100 \times 100 \times 4 = 40\,000$ 次 6 级粗分辨逼近小波系数的相关计算.

基于 LS-tree 组织和管理索引系数、用小波包最佳树—塔型算法的方法和不同级小波粗分辨逼近小波系数方法比较结果如表 2 所示.

图 6 显示了基于 LS-tree 的最好基—塔型算法和不同级小波系数搜索时的检索精度及其花费的时间比率.其中:

$$\text{检索精度} = \frac{\text{检索到的正确文件数}}{\text{检索文件总数}},$$

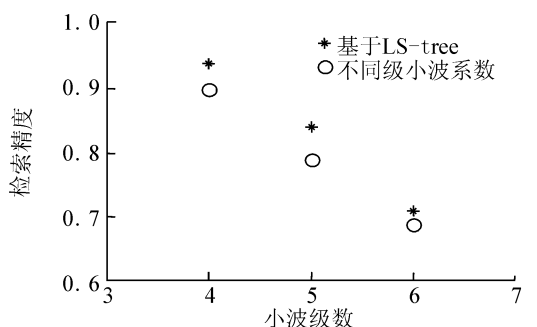
$$\text{占用时间比} = \frac{\text{该层检索所用单位时间}}{\text{4 级小波系数检索所用单位时间}}$$

表 2 基于 LS-tree 的小波最佳树—塔型算法和不同级小波系数检索的结果

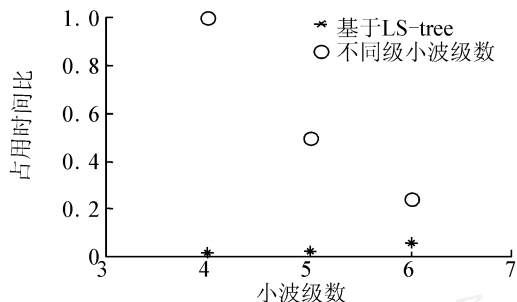
Table 2 Search result with wavelet packet best base & pyramidal algorithm base on LS-tree and different level coefficients

	错误文件	数量	搜索次数
第 1 层 最好基 塔型算法	6, 9, 10, 12, 13, 25, 26, 27, 37, 38, 43, 45, 46, 51, 52, 56, 59, 62, 63, 66, 74, 80, 85, 87, 89, 90, 94, 95, 100	29	2 582
	2, 7, 10, 12, 13, 25, 26, 27, 34, 37, 38, 45, 46, 51,		
	52, 56, 59, 62, 63, 66, 69, 74, 79, 80, 85, 87, 89, 90, 94, 95, 100		
	6 级 小波系数		
第 2 层 最好基 塔型算法	2, 6, 9, 12, 29, 31, 34, 37, 43, 51, 52, 56, 66, 80, 90, 100	16	1 038
	5 级 小波系数		
	2, 12, 14, 15, 18, 29, 31, 34, 35, 37, 53, 56, 63, 72, 80, 88, 89, 90, 92, 99, 100		
第 3 层 最好基 塔型算法	6, 9, 43, 52, 66, 88	6	1 028
	4 级 小波系数		
	12, 13, 14, 15, 21, 46, 56, 72, 83, 90		

从表 2 和图 6 (a) 中可以看出基于 LS-tree 的最好基—塔型算法检索音频数据的有效性,而且,随着搜索塔型层数的增加,精度也逐渐提高.其检索精度明显高于不同级小波系数方法.不管用几层塔型算法,从表 2 和图 6 (b) 可以看出,它所占用的时间都远低于多级小波系数的方法,而且,随着塔型层数的增加和搜索精度的提高,花费时间的增加相对较低.也就是说,在需要进行高精度搜索时,基于 LS-tree 的最好基—塔型算法花费的时间远低于不同级小波粗分辨系数搜索算法.在一般情况下,这种方法可以满足音频数据基于例子的搜索对精度和时间的要求.



(a) 基于LS-tree和不同级小波系统检索的检索精度



(b) 基于LS-tree和不同级小波系数检索的占用时间比率

图6 基于LS-tree不同级小波系数检索的占用时间比率

Fig 6 Compare with LS-tree and difference level coefficient to retrieval audio

4 结束语

正如前所述,目前,可以适用于各种多媒体数据的组织、管理和搜索的空间数据结构很多.这些空间数据结构一般都是以R-tree为基础提出来的.如,对于图像数据的组织和管理常用R*-tree,对于文档数据的组织管理采用TV-tree,对于视频数据的管理通常采用帧段树或RS-tree等.但对于多媒体音频数据的组织管理却鲜有文献.针对这种情况,本文提出并定义了LS-tree来组织管理多媒体音频数据.同时,把这种结构用于以小波包最好基-小波塔型算法为基础的音频数据检索中.实验结果表明了这种方法的有效性和实用性.可以认为,这种索引结构如果能进一步和音频数据的分段和分类技术相结合,将为实现真正的基于例子和基于内容的多媒体音频数据检索提供有效的方法和思路.

参考文献:

- [1] SAMET H. The design and analysis of spatial data structures[D]. MA, USA: Addison-Wesley, 1989.
- [2] HJALTASON G, SAMET H. Ranking in spatial databases [C]//Advances in Spatial Databases—4th Symposium. Berlin, Germany: Springer-Verlag, 1995: 83-95.
- [3] LIN K I, JAGADISH H V, FALOUTSOS C. The TV-tree: an index structure for high-dimensional data [J]. VLDB Journal, 1994, 3(4): 517-542.

- [4] SCHRECK T, CHEN Z. Branch grafting method for R-tree implementation [J]. The Journal of Systems and Software, 2000, 53(1): 83-93.
- [5] SHIM K, SRINANT R, AGRAWAL R. High-dimensional similarity joins [J]. IEEE Trans on Knowledge and Data Engineering, 2002, 14(1): 156-171.
- [6] PARK D J, HEU S, KIM H J. The RS-tree: an efficient data structure for distance browsing queries [J]. Information Processing Letters, 2001, 80(2): 195-203.
- [7] HUANG P W, LIN P L, LIN H Y. Optimizing storage utilization in R-tree dynamic index structure for spatial databases [J]. The Journal of Systems and Software, 2007, 55(3): 291-299.
- [8] WHITE D A, JAN R. Similarity indexing with the SS-tree [C]//Proceedings of the Twelfth International Conference on Data Engineering. New Orleans, USA: Stanley Y. W. Su, 1996: 516-523.
- [9] BERCHTOLD S, KEM D A, KRIEGER H P. The X-tree: an index structure for high-dimensional data [C]//Proc Intl Con on Very Large Databases. Bombay, India: VADB, 1996: 28-39.
- [10] CAIC, MITRA S K, DING R. Smart wavelet image coding: X-tree approach [C]//Signal Processing 82 [S 1], 2002: 239-249.
- [11] COLOSSIN G, NASCIMENTO M A. Benchmarking access structures for the similarity retrieval of high-dimensional multimedia data [C]//IEEE International Conference Multimedia and Expo. New York, USA, 2000: 1215-1218.
- [12] PRAMANIK S, LI S, RUAN J. Performance analysis of AB-tree [C]//IEEE International Conference Multimedia and Expo. New York, USA, 2000: 1701-1704.
- [13] HENRICH A. The LSDh-tree: an access structure for feature vectors [C]//Data Engineering 14th International Conference. Orlando, USA, 1998: 362-369.
- [14] CHAKRABARTI K, MEHROTRA S. The hybrid tree: an index structure for high dimensional feature spaces [C]//Data Engineering, 15th International Conference. Sydney, 1999: 440-447.
- [15] SUBRAMANYA S R. Indexing and searching schemes for audio data in audio/multimedia databases (multimedia database) [D]. USA: George Washington Univ, 1999.
- [16] LI Y, HOU Y. Search audio data with the wavelet Pyramidal algorithm [J]. Information Processing Letter, 2004, 91(1): 49-55.
- [17] ZHANG T, JAY K C C. Audio content analysis for online audiovisual data segmentation and classification [J]. IEEE Trans Speech and Audio Processing, 2001, 9(4): 441-457.
- [18] JUMAPATHY K, KRISHNAN S, RAO R K. Audio signal feature extraction and classification using local discriminant bases [J]. IEEE Trans Audio, Speech and Language Processing, 2006, 15(4): 1236-1246.

作者简介:



李应,男,1964年生,副教授,博士,主要研究方向为多媒体数据检索和信息安全,发表学术论文6篇。