

基于实时梯度计算的快速纹理映射体绘制

郑杰¹ 姬红兵² 杨万海¹

(西安电子科技大学工程学院 西安 710071)¹ (西安电子科技大学电子研究生院 西安 710071)²

摘要 本文分析讨论了一种基于实时梯度计算的纹理映射体绘制算法。在绘制过程中实时计算每个体素上的梯度,有效地减少了在传统基于纹理映射的体绘制中,对梯度预先计算的耗时操作,也降低了内存的消耗,加快了整个绘制过程。同时,采用三维 Sobel 算子对梯度进行计算,并进行归一化处理,有效地提高了绘制图像的质量。在实现中充分利用了目前 PC 图形硬件成熟的可编程特性,特别是 fragment program,来完成梯度的实时计算。最后对医学体数据进行绘制,得到了理想的结果。

关键词 体绘制, 三维纹理映射, 梯度计算, 通用图形处理器

Real-time Gradient Estimating in Texture-Based Volume Rendering

ZHENG Jie¹ JI Hong-Bing² YANG Wan-Hai¹

(School of Electronic Engineering, Xidian University, Xi'an 710071)¹ (Graduate School, Xidian University, Xi'an 710071)²

Abstract A novel technique for texture-based volume rendering is discussed, which estimates and normalizes the gradient on the fly. During rendering the gradient is calculated on each voxel, and normalized for the lighting computation. The new technique is presented to load the volume dataset to the 3D texture directly without the time consumptive process to pre-compute the gradient, and circumvent the huge memory consumption in previous approaches. Additionally, the 3D Sobel gradient operator is applied to the gradient estimation, which improves the quality of the rendering. The sophisticated PC graphics hardware functionality is utilized, especially the fragment program, to implement the gradient estimation. Several examples are given to illustrate the effectiveness on real clinic 3D CT datasets.

Keywords Volume rendering, 3D texture mapping, Gradient estimating, Graphics processing unit

1 引言

直接体绘制对于由各种类型的三维设备、科学仿真等产生的体数据可视化是一种非常重要的技术,它最大的优点在于不需要生成中间几何体而直接对体数据进行成像,以反映体数据中各种信息的综合分布情况,但同时要计算每个体素对成像的贡献,巨大的计算量限制了体绘制的应用。为了克服绘制中大量的计算和对带宽的极高要求,基于图形硬件纹理映射的体绘制逐步成为对规则网格体数据绘制的最为实际可行的方法。

尽管目前的 PC 图形硬件广泛提供了对三维纹理的支持,但基于三维纹理映射的体绘制仍有一些局限性。在体绘制中明暗光照是一种很有效的技术,可以增强绘制图像的空间效果,更好地理解图像的内容。梯度则是明暗光照计算中的基本要素,直接影响着绘制图像的质量。在几何体的绘制中梯度是顶点属性,而对于体绘制则是体素的属性。当使用纹理映射对体数据进行绘制时,图形硬件没有提供对梯度计算的直接支持。为了克服这个缺陷,梯度数据要预先进行计算,并和体数据一同载入图形硬件中。这样需要四倍于体数据量的纹理内存来完成绘制。由于目前图形硬件内存的限制,这就制约了对体数据的规模。同时数据预处理过程中梯度计算非常耗时,导致绘制实时性的降低。另一方面,绘制时要对归一化的梯度进行采样插值,得到的非归一化梯度则被

用于光照计算,这样会造成明暗光照的伪影,降低绘制的质量。

本文基于通用 PC 图形硬件纹理映射,提出了一种基于实时梯度计算的快速体绘制方法。在绘制时实时计算每个体素上归一化的梯度,避免了梯度的预先计算和存储,提高了对大规模体数据的绘制能力,加速了整个绘制过程。同时采用了三维 Sobel 算子进行梯度的计算,有效地提高了梯度的精度,改善了绘制图像的质量。由于充分利用了目前 PC 图形硬件成熟的可编程特性,所有绘制过程在一个绘制通道中完成,可以达到交互的绘制速度。

2 相关研究

最重要的基于纹理映射的体绘制方法由 Cabral^[2]提出,其利用高端图形工作站中的三维纹理映射特性完成了对 256³ 的体数据的交互绘制。但这种通用的纹理映射方法的主要缺点是缺乏对体数据光照的支持。为了克服这个缺陷, Van Gelder^[1]等提出采用 3~4 个参数的查找表来实现体数据的光照计算。但是图形硬件没有提供这类查找表的支持,这样当改变视线方向时都要生成新的三维纹理。

Westermann 和 Ertl^[7]通过引入一种快速的直接多通道绘制算法来显示具有光照效果的等值面,极大地扩展了基于纹理映射的体绘制方法的应用。基于这些研究成果,Meißner 等^[6]提出了一种新方法对半透明体绘制进行漫反射光照处

郑杰 博士生,主要研究领域为图像处理、计算机图形学、科学可视化等;姬红兵 博士,教授,博士生导师,主要从事信号与信息处理和目标识别方面的研究工作;杨万海 教授,博士生导师,主要从事图像融合、C3I 系统数据融合、雷达系统建模、仿真和设计技术等方面的研究。

理。然而这种通过光栅化图形硬件的多通道绘制极大地降低了绘制性能。以上这两种具有光照效果的方法采用了矩阵操作来获得漫反射光照强度, Rezak-Salama 等^[3]则利用了 NVidia GeForce2 提供的“register combiners”操作来完成。尽管得到了很好的绘制效果,但所有的明暗处理操作都是基于经过插值的非归一化的梯度进行的,造成了光照的伪影。与 Westermann 类似, Dechille 等^[4]提出利用三维纹理硬件进行插值,而用软件进行光照计算和数据分类。光照操作在 CPU 上完成以保证绘制的高质量,这样就避免了非归一化梯度的问题,但是却牺牲了绘制的实时性。Kniss 等^[5]利用通用图形硬件加速了多维传递函数的交互性。但由于多维函数自身的复杂性,带来了计算量和存储量的骤增,从而影响了绘制的实时性,同时也限制了可绘制体数据的大小。

本文的主要研究成果是利用 OpenGL 的 ARB_fragment_program 扩展采用 Sobel 算子对体绘制中光照计算所需的梯度进行实时计算和归一化处理,同时降低了 CPU 和 GPU(图形处理器)的内存消耗,得到了理想的绘制结果。因此,这种方法适用于对绘制质量有较高要求的交互式应用。

3 实时梯度计算的纹理映射体绘制

本节首先将对三维纹理映射的体绘制技术进行描述。基于对采样后体数据进行梯度计算的思想避免对梯度的预先计算,并采用归一化的梯度实现体绘制中的明暗光照效果。

3.1 三维纹理映射体绘制

体绘制的主要任务是在每个像素上对体绘制积分进行近似计算,即沿着入射光线对逐渐衰减的颜色和衰减系数进行累加。首先要对体数据进行分类来指定颜色和衰减系数。通过引入传递函数(transfer function)来将体数据强度 $s=s(\vec{x})$ 映射为颜色强度 $c(s)$ 和衰减系数 $\tau(s)$, c 一般为颜色空间中的颜色矢量。

我们假设入射光线表示为 $\vec{x}(\lambda)$, 其中 λ 为到观察点的距离, D 为最大观察距离。对空间中的任意一点 $\vec{x}$, 体绘制积分可以表示为:

$$I = \int_0^D c(s(\vec{x}(\lambda))) e^{-\int_0^\lambda \tau(s(\vec{x}(\lambda')) d\lambda'} d\lambda \quad (1)$$

沿入射光线采样,对体绘制积分进行离散化,我们引入阻光度 A , 定义为:

$$A_i = 1 - e^{-\tau(s(x(id)))d} \quad (2)$$

同样,第 i 个采样光线间隔上的颜色可以近似表示为:

$$C_i = c(s(x(id)))d \quad (3)$$

这样,离散化的体绘制积分可以表示为:

$$C = \sum_{i=0}^n C_i \prod_{j=0}^{i-1} (1 - A_j) \quad (4)$$

其中 $n = \lceil D/d \rceil$, 为采样点的个数。

式(4)可以通过从前向后或从后向前进行迭代的 alpha 融合进行实现。如按照从后向前的顺序进行迭代时第 i 步可表示为:

$$C'_i = C_i + (1 - A_i) C'_{i+1} \quad (5)$$

目前的通用图形硬件采用的是以物体为序的光栅化方法。几何元素(点、线、多边形)经过扫描转化,按照像素的方式写入图像帧缓存中进行显示。然而在体数据中没有这样的几何元素,需要通过多边形切片对体数据进行采样,每个多边形的纹理可从相应的体数据获得。对这些切片多边形进行光栅化操作就可以重建出体数据。

平行于视平面的多边形切片是在基于三维纹理映射的绘

制中最常用的方法。体数据被装载到一个三维纹理中,在切片多边形内部上进行插值得到顶点的三个纹理坐标,在光栅化过程中,这些纹理坐标被用来对三维纹理进行采样,利用硬件加速的三次线性插值得到相应的体数据。另外,当使用平行投影时,在对体数据进行切片的过程中保持固定的采样间隔。这是对体绘制进行离散的一个重要假定,这样切片多边形就可以按照由前向后或由后向前的顺序正确地融合了。

在利用纹理映射绘制体数据时,通用图形硬件没有对梯度的计算提供支持。为了实现光照处理,通常要预先对体数据的表面梯度进行计算,在纹理内存中存储体数据和归一化梯度,极大地占用了图形硬件中有限的内存空间。另一方面当用切片对体数据绘制时,同时对预先计算的体数据梯度也进行了三次线性插值,插值后的非归一化梯度被用来计算光照。这样导致了明暗效果的伪影,降低了绘制图像的质量。

3.2 梯度的实时计算

本文采用对梯度进行实时计算的思想来有效地加速整个绘制过程。只需在纹理内存中存储原有传统纹理绘制 $1/4$ 的数据,同时也无需在预处理过程中进行耗时的梯度计算,直接载入体数据,提高了绘制实时性。整个绘制流程如图1所示,其中虚线框内的操作均是在图形硬件中完成的。

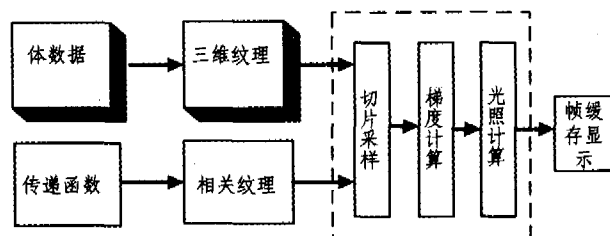


图1 绘制流程

常用的梯度计算方法,如中心差分,采用6邻域信息来描述中心位置体素的梯度。为了提高准确性,我们采用三维 Sobel 算子来计算数据场的梯度。

和中心差分相比, Sobel 梯度算子采用 26 邻域来完成梯度的计算。二维 Sobel 算子通常被用于图像的边缘检测,三维 Sobel 算子使用一个 $3 \times 3 \times 3$ 的模板来计算每个方向上的偏导数。本文采用的在 z 方向上的模板为:

$$\begin{array}{ccc} \begin{bmatrix} -1 & 0 & 1 \\ -3 & 0 & 3 \\ -1 & 0 & 1 \end{bmatrix} & \begin{bmatrix} -3 & 0 & 3 \\ -6 & 0 & 6 \\ -3 & 0 & 3 \end{bmatrix} & \begin{bmatrix} -1 & 0 & 1 \\ -3 & 0 & 3 \\ -1 & 0 & 1 \end{bmatrix} \\ z-1 & z & z+1 \end{array}$$

将上面的模板按照相应的方向旋转,可以得到 x 和 y 方向上的计算模板。Sobel 算子具有各向同性的特性,体数据中边界结构的方向不影响计算结果,可以提供较高精度的梯度。

在目前的通用图形硬件中, fragment program 特性应用在光栅化阶段,对每个像素进行处理。在对绘制切片进行纹理采样时,同时得到当前体素的邻域信息,完成梯度的实时计算以后,可以利用 fragment program 的算术运算对梯度进行归一化处理,然后进行光照计算。

我们在配备 256M 显存的 ATI Radeon X800 Pro 上利用 OpenGL 的 ARB_fragment_program 扩展^[10]和三维纹理实现了本文提出的算法。在算法开始前体数据被载入三维纹理,通过利用图形硬件的三次线性插值来计算沿视线方向的切片来绘制体数据。

为了实时地计算体数据的归一化梯度,梯度“fragment

program”应完成下面的任务:

- 根据当前的纹理坐标,得到 26 邻域的纹理坐标;
- 利用相关纹理采样得到所需的 26 个邻域体数据;
- 根据 Sobel 模板计算每个方向上的梯度分量;
- 归一化梯度。

用于梯度计算的“fragment program”使用了 89 条指令,其中 26 条为三维纹理寻址操作、32 条矢量运算和 28 条标量运算。归一化梯度使用了 3 条标量运算。所有这些对每个切片的计算只需在一个绘制通道中完成,保证了绘制的性能。

4 试验结果

我们用大量的医学临床实际数据对上述算法进行了测试实验,所有实验都是在一台 P4,2.4GHz,1GB 内存,操作系统为 Windows XP 的 PC 机上完成的,显卡为配有 256M 显存的 ATI Radeon X800 Pro。采用 C++ 和 OpenGL 对提出的算法进行了实现。

表 1 给出了对来自医学临床的 CT 扫描数据绘制的性能参数。其中对体数据为 $256 \times 256 \times 423$ 的腿部 CT 扫描数据,分别给出了采用本文提出的算法和传统的基于三维纹理映射体绘制算法进行绘制的性能参数比较,其中预处理时间包括数据梯度的预先计算和载入图形硬件的时间。

表 1 性能参数

绘制 算法	体数据	数据规模	预处理 (s)	显存占用 (M)	帧速率 (fps)
传统	Leg	$256 \times 256 \times 423$	7.89	128	15.6
本文	Leg	$256 \times 256 \times 423$	0.39	32	9.5
	Skull	$256 \times 256 \times 361$	0.38	32	9.6
	Abdomen	$256 \times 256 \times 421$	0.38	32	9.7
	Body1	$256 \times 256 \times 527$	0.43	64	7.5
	Body2	$256 \times 256 \times 957$	0.49	64	7.16

绘制窗口大小为 512×512

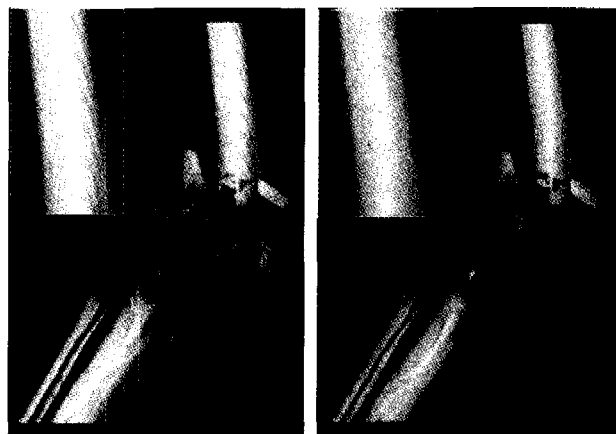


图 2 (左)传统纹理映射体绘制;
(右)本文提出的绘制方法

表 1 的性能参数对比可以看出,本文提出的算法在绘制中对梯度实时计算和归一化处理需要图形硬件进行较多的纹理采样和算术操作,造成了绘制帧速率的下降。但在预处理时不用进行梯度计算,一方面加快了数据的载入,另一方面也减少了图形硬件的内存的占用。

图 2 中将本文的算法和传统的三维纹理映射方法的绘制效果质量进行了比较,采用了相同的传递函数和光照参数进

行绘制。为了清楚地表示差异,将关节上部的腿骨部分放大显示于左上角的小图。可以看出,由于传统的算法采用了非归一化的梯度进行光照计算,故影响了绘制结果(图 2 左)。而本文的算法明显地提高了绘制质量(图 2 右)。

图 3 为对头部颅骨和腹部的 CTA 体数据进行绘制的效果。在绘制结果中可以清晰地看到小的血管分支,而且在高密度部位,如骨骼上没有出现“阶梯伪影”。

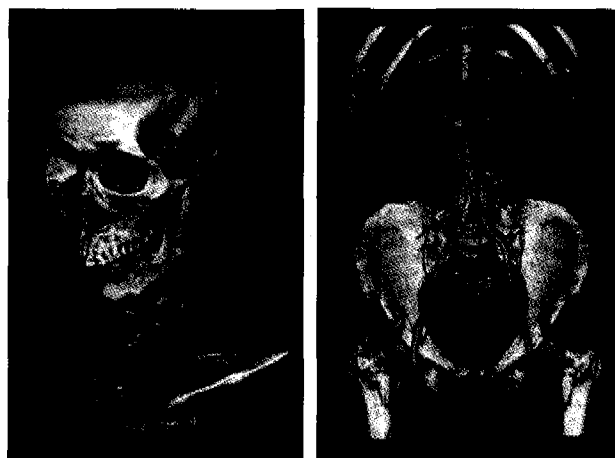


图 3 (左)半透明效果的肝脏数据;
(右)半透明效果的肾脏数据

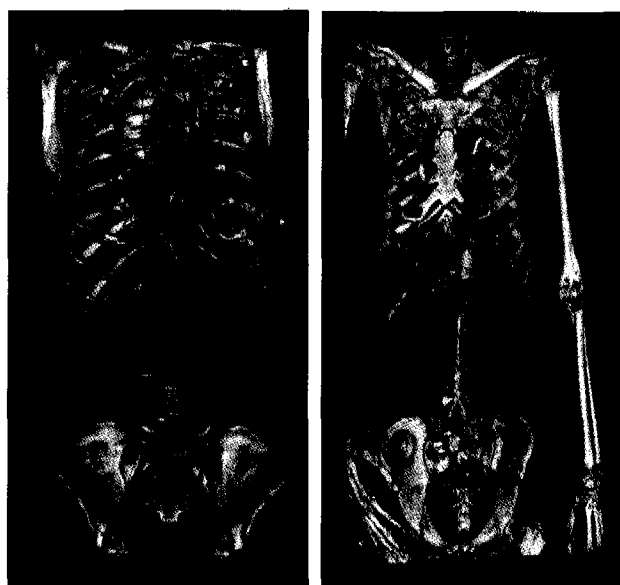


图 4 对两个人体的 CTA 数据的体绘制结果
(左)Body1;(右)Body2

图 4 为两个人体的 CTA 扫描数据,由于数据量较大,已有的三维纹理映射算法无法对其进行绘制,而本文算法的低显存占用特性提高了大规模体数据的绘制能力。而且体数据快速地被载入图形硬件,绘制结果可清楚地显示出体数据中的各种内部结构,也非常直观地展示了不同器官之间的空间关系。

结论 在传统的基于三维纹理映射的体绘制方法上,本文提出了一种新的体绘制算法,充分地利用了目前 PC 通用图形硬件的可编程特性,提高了绘制的实时性和绘制图像的质量。

本文提出的方法利用 OpenGL 的 fragment program 扩展实时计算体数据梯度,并使用归一化的梯度实现了体绘制的

明暗光照。这种方法明显地减少了传统的基于三维纹理映射的体绘制方法的巨大内存占用,并且加快了整个绘制过程。由实验结果可见,利用目前通用图形硬件灵活的纹理操作和强大的光栅化能力,可以得到理想的绘制结果,使得体绘制更具实用和推广价值。

参考文献

- 1 Van Gelder A, Kim K. Direct Volume Rendering with Shading via Three-Dimensional Textures [C]. In: Proc. of Volume Visualization Symposium, San Francisco, CA, October 1996. 23~30
- 2 Cabral B, Cam N, Foran L. Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware [C]. In: Workshop on Volume Visualization, Washington, DC, USA, October 1994. 91~98
- 3 Rezk-Salama C, Engel K, Bauer M, et al. Interactive Volume Rendering on Standard PC Graphics Hardware Using Multi-textu-

- ring and Multi-stage Rasterization [C]. In: Proc. of Eurographics/SIGGRAPH Workshop on Graphics Hardware, Interlaken, Switzerland, August 2000. 109~118
- 4 Dachille F, Kreeger K, Chen B, et al. High-Quality Volume Rendering Using Texture Mapping Hardware [C]. In: Proc. of Eurographics/SIGGRAPH Workshop on Graphics Hardware, Lisbon, Portugal, August 1998. 69~76
- 5 Kniss J, Kindlmann G, Hansen C. Multidimensional Transfer Functions for Interactive Volume Rendering [J]. IEEE Transactions on Visualization and Computer Graphics, 2002, 8(3): 270~285
- 6 Meißner M, Hoffmann U, Straßer W. Enabling Classification and Shading for 3D Texture-based Volume Rendering Using OpenGL and Extensions [C]. In: Proc. of IEEE Visualization, San Francisco, CA, USA, October 1999. 207~214
- 7 Westermann R, Ertl T. Efficiently Using Graphics Hardware in Volume Rendering Applications [C]. In: Proc. of ACM SIGGRAPH, Orlando, FL, USA, August 1998. 169~177
- 8 ATI Corporation. ATI Developer Relation Site. Available at: <http://www.atl.com/developer>

(上接第 222 页)

0.2%及0.4%。这是由于 D 变化,而考虑了求全局频繁集合 F^+ 。从实验结果看,算法运行时间随 $|D|$ 变大而基本呈线性增长。图1、图2、图3的实验结果也表明了EUFIA算法的有效性和较好的可测量性。

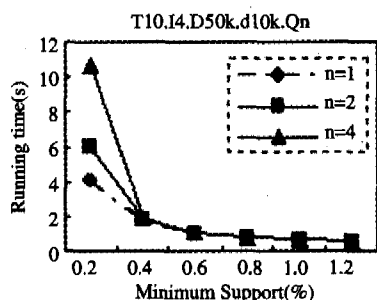


图1 最小支持度变化时 EUFIA 运行时间

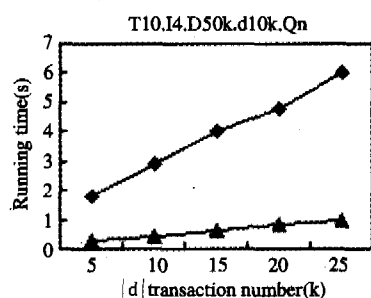


图2 $|d|$ 增大时 EUFIA 运行时间

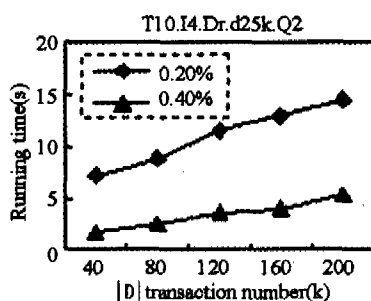


图3 $|D|$ 增大时 EUFIA 运行时间

结束语 在实际应用中,事务数据库是不断变化的,并且许多时候新增事务比旧的事务更加有价值,更值得去充分挖掘。在更新中如何保证既快速又不丢失有效的规则是一个很重要的问题。为此,本文提出了一个全面更新频繁项集的算法 EUFIA,它将新增事务划分成若干分区进行扫描,并将结

果归入不同的3个类别,从而达到了在不扫描原数据库的情况下,充分、快速地挖掘增量事务,获取频繁项集,且又不丢失有效的规则。另外,该算法的设计思想可以方便地应用到多层关联规则和多层序列模式的发现中,我们将对此作进一步的研究。

参考文献

- 1 Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Washington D C, 1993. 207~216
- 2 Agrawal R, Srikant R. Fast algorithm for mining association rules. In: Proceedings of the 20th International Conference on VLDB, Santiago, Chile, 1994. 487~499
- 3 Han J, Kamber M. Data Mining: Concepts and Techniques. Beijing: Higher Education Press, 2001
- 4 Han J, Jian P, et al. Mining frequent patterns without candidate generation. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Dallas, TX, May 2000. 1~12
- 5 Shoemaker C, Ruiz C. Association Rule Mining Algorithms for Set-valued Data. In: Proc. 4th Intl. Conf. on Intelligent Data Engineering and Automated Learning, LNCS Vol. 2690, Hong Kong, China, 2003. 669~676
- 6 Wang Jianyong, Han J, Lu Y, Tzvetkov P. TFP: an efficient algorithm for mining top-k frequent closed itemsets. Knowledge and Data Engineering, IEEE Transactions, 2005, 17(5): 652~663
- 7 颜跃进, 李舟军, 陈火旺. 频繁项集挖掘算法. 计算机科学, 2004, 31(3): 112~114
- 8 Cheung D W, et al. Maintenance of discovered association rules in large databases: an incremental updating technique. In: Proceedings of the 12th International Conference on Data Engineering, New Orleans, Louisiana, 1996. 106~114
- 9 Cheung DW, LEE SD, Kao B. A general incremental technique for maintaining discovered association rules. In: Topor RW, Tanaka K, eds. Proc. of the 5th Int'l Conf. on Database Systems for Advanced Applications. World Scientific, 1997. 185~194
- 10 Hong T P, Wang C Y, Tao Y H. A new incremental data mining algorithm using pre-large itemsets. Intelligent Data Analysis, 2001, 5(2): 111~129
- 11 Ayan N F, Tansel A U, Arkun E. An efficient algorithm to update large itemsets with early pruning. In: Proc. 5th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, San Diego, Aug. 1999. 287~291
- 12 Otey M E, Parthasarathy S, Wang C, Veloso A, Meira W Jr. Parallel and distributed methods for incremental frequent itemset mining. Systems, Man and Cybernetics, Part B, IEEE Transactions, 2004, 34(6): 2439~2450
- 13 Veloso A A, Meira W Jr, de Carvalho M B, Póças B, Parthasarathy S, Zaki M J. Mining frequent itemsets in evolving databases. In: Proc. 2nd SIAM Intl. Conf. on Data Mining, Arlington, TX, May 2002. 494~510
- 14 Lee C H, Lin C R, Chen M S. Sliding-window filtering: An efficient algorithm for incremental mining. In: Proc. 10th Intl Conf. on Information and Knowledge Management, Atlanta, Nov. 2001. 263~270
- 15 朱玉全, 孙志辉, 季小俊. 基于频繁模式树的关联规则增量式更新算法. 计算机学报, 2003, 26(1): 91~96