

基于矢量线强化的增强型2维流场实时绘制

周迪斌 王康健 解利军 郑耀

(浙江大学工程与科学计算研究中心,杭州 310027) (浙江大学计算机科学与技术学院,杭州 310027)

摘要 在流场绘制中,合理地结合矢量场的多种属性有助于矢量场的特征分析,据此提出了一种基于矢量线强化的增强型2维流场实时绘制算法。通过对流场的一些标量属性如大小、角度和曲率进行色彩映射,该算法不仅可以清晰显示流场运动方向,而且能显示矢量场的多种属性,有助于了解流场矢量特征分布和主要拓扑结构。该算法采用了一种矢量线强化策略,即通过对卷积纹理在垂直矢量方向上进行1维高通滤波,增加了矢量线间的对比,改善了图像质量。利用现代图形卡的可编程能力,该算法可以在微机上达到实时绘制性能。

关键词 流场可视化 实时绘制 科学可视化 高通滤波

中图分类号:TP391.7 文献标识码:A 文章编号:1006-8961(2008)09-1804-08

Enhanced Real-time 2D Flow Visualization Based on Flow Lines Enhancement

ZHOU Di-bin, WANG Kang-jian, Xie Li-jun, Zheng Yao

(Center for Engineering and Scientific Computation, Zhejiang University, Hangzhou 310027)

(College of Computer Science and Technology, Zhejiang University, Hangzhou 310027)

Abstract Considering that some helpful scalar properties to vector visualization with feature analysis, in this paper we present an enhanced realtime 2D flow visualization algorithm based on flow lines enhancement. Through the color mapping of some scalar properties, such as velocity, angle and curvature, it can not only reflect flow direction, but also display many properties of vector field which are helpful to analyze the features and topology of vector fields. In order to enhance the contrast among flow lines and improve the rendering image quality, it adopts a scheme of enhancing flow lines which applies one dimensional high-pass filter to convolution texture along the direction perpendicular to flow. By using the programmable ability of modern GPVS, the algorithm can achieve realtime rendering performance on Personal Computers.

Keywords flow visualization, realtime rendering, scientific visualization, high-pass filter

1 引言

流场可视化是科学可视化中非常重要的组成部分,特别在计算力学、流体力学和空气动力学、天气预报和地球物理模拟等领域。基于点图标法和矢量线法(如流线、迹线、时线和纹线)是目前运用非常广泛的矢量可视化方法,但这两类方法有一个共同

的缺点:太少的采样容易忽略矢量场的一些重要细节,而过多的采样又容易导致图像混乱,不易于观察流场内部细节。同时,这些方法也不利于动态矢量场的表达。基于纹理的流场可视化是近年来迅速发展的一种矢量可视化技术,其内在的致密性和高质量的纹理图像几乎能反映流场的整个细节,克服了点图标和矢量线方法的不足。同时采用基于纹理的流场动画技术不仅有助于了解矢量场的形态和拓扑

基金项目:国家杰出青年基金项目(60225009);国家自然科学基金重大研究计划项目(90405003)

收稿日期:2007-02-28;改回日期:2007-05-11

第一作者简介:周迪斌(1978~),男,浙江大学计算机学院博士研究生。主要研究方向为科学可视化和GPU通用计算。

结构,而且能表征非稳定矢量场的动态变化。同时,由于该方法需计算矢量场中所有像素点的显示纹理,因此所需的计算量要显著高于前两种方法。

早期基于纹理的流场可视化方法主要是基于CPU(center processing unit)实现,由于运算量巨大、计算耗时使得早期的算法研究主要集中在优化算法性能方面,为达到实时绘制性能往往需借助并行系统的运算能力或专用的图形硬件系统。近年来随着图形硬件的飞速发展和像素处理器的出现,一些原有的基于纹理的算法可在GPU(graphic processing unit)内完全实现。利用其内在的并行运算能力,这些方法已能在普通微机上达到实时绘制。

虽然目前基于GPU的流场实时绘制效率得到很大提高,但一些绘制算法的流场空间相关性不够,主要反映在流场中矢量线不够明确,矢量线间强度对比不够,图像的属性强度分布区域较窄。一些已有流场绘制过程的后处理方式,如高通滤波,又没有充分考虑到矢量场区别于静态图像的内在的有向性特点;同时,考虑到矢量场的属性,特别是曲率有助于矢量场的分析和观察,首次提出一种基于矢量线强化的增强型2维流场实时绘制算法,该算法采用了一种增强型的显示方式,即利用比例混合或直接相乘的方式合成显示矢量场的卷积纹理和其他属性;同时依据矢量场内在的有向性特点,采用了一种矢量线强化策略,即通过对卷积纹理在垂直矢量方向上进行1维高通滤波。利用增强型显示方式,该算法不仅可以清晰显示流场运动方向,而且能同时显示一些流场属性,如矢量大小、角度、曲率,有助于了解流场矢量特征分布和主要拓扑结构,同时矢量线强化策略有效地增加了矢量线间的对比,改善了图像质量。

2 相关工作

随着现代图形硬件发展,一些基于纹理的流场绘制算法已能借助图形硬件内在的并行能力实现加速。Heidrich等人利用SGI(Silicon Graphic Inc.)支持的片元纹理加速了LIC(line integration convolution)卷积计算^[1],但该方法仅适合于稳定的2维矢量场。Jobard等人提出了一种基于GPU支持的纹理卷积技术^[2],实现了2维非稳定矢量场的可视化。Van提出了一种叫做IBFV^[3](image based flow visualization)流场绘制技术。该方法对图形硬件要求不高,仅利用了目前图形硬件支持的标准操

作如纹理映射和像素融合,而且能以极高的帧率模拟很多技术,如微粒、箭头、流线、时线、LIC和拓扑分析。随后IBFV算法被拓展运用到3维曲面矢量场和3维矢量场的可视化。Jobard等也提出了一种适合于动态流场的可视化的LEA^[4]算法,该算法尽管与硬件无关,但也取得很高的绘制帧率。LEA纹理法利用连续的沿迹线更新纹理混合实现动态流场绘制的时间相关性,而单帧纹理图像通过沿流线方向短距离的1维低通滤波实现了动态流场绘制的空间相关性。

最近,Laramee等人提出了一种新的基于IBFV和LEA的绘制算法ISA^[5](image space advection),该算法先将任意3维曲面的矢量场投影映射为2维平面的矢量纹理,然后利用基于平面矢量场的向后纹理卷积和连续的纹理融合实现了非稳定3维曲面流场实时绘制。Grabner和Laramee在文献[6]中详细介绍了该算法在GPU具体实现。这类绘制方法无后处理,矢量线不够清晰,空间相关性不够。

Weiskopf等人提出了一种基于通用的时间空间相关性框架的UFAC^[7]算法,该算法通过沿迹线的纹理属性流动实现时间相关性,同时通过沿即时流线方向的纹理卷积实现空间相关性。虽然该算法能模拟很多算法,但算法的时间相关性不够,帧间跳跃大。Shen和Kao提出了一种基于粒子模型的UFLIC^[8](unsteady flow LIC)算法,随后,Liu和Moorhead在AUFLIC^[9](accelerated UFLIC)算法中改进了该算法,提出了一种新的粒子管理机制,并采用了一种as-long-as possible的策略最大程度地保持流线的连续性。而Li提出了针对该算法的GPU实现,称为GPUFLIC^[10]。此类算法虽然能取得高质量的矢量线,但是其算法实现较为复杂。

在矢量场增强绘制特征方面,Okada和Lane^[11]研究了LIC纹理增强的技术,利用彩色纹理来突出流场分离和汇合边界区域;Theisel研究认为矢量的曲率图有助于矢量场的分析和观察,并着重分析了矢量曲率在关键点区域的相关变化^[12];Risquet和Theisel提出一种综合的2D矢量场技术^[13],不单可以显示流场变化,而且可以显示流场的其他属性。

本文在基于CPU的Risquet的合成算法基础上,提出一种基于GPU的增强型算法框架以达到实时绘制的效率。本文算法中所有属性映射及卷积运算均在图形硬件中实现,能实时展示流场动画,而Risquet算法只能显示静态纹理图像。同时,首次利用了一种矢量线强化策略来改善矢量线的图像质

量。该策略本质上是垂直矢量方向的 1 维高通滤波。目前该算法主要是针对 2 维稳定矢量场,如果考虑矢量场图像空间的动态投影,该算法也可以扩展到其他非稳定的 3 维曲面矢量场。

3 算法

图 1 是算法流程图,首先根据矢量场纹理和 2D 属性纹理(灰度图)计算图像空间的纹理卷积,然后是噪声注入并混合,用以补偿图像对比度减少,其后的强化处理用以增强矢量线的对比度;同时,依据标量属性及其色彩映射表计算矢量属性映射纹理;最后是属性映射纹理和卷积纹理合成显示。

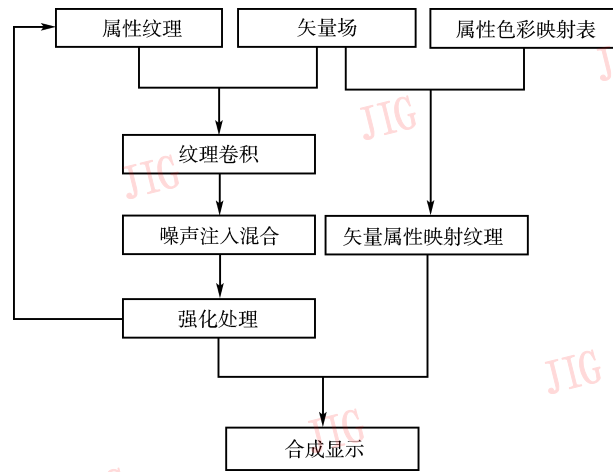


图 1 基于矢量线强化的增强型 2 维流场绘制算法流程图

Fig. 1 The workflow of the algorithm

在该算法中,首次卷积的属性纹理为随机白噪声纹理。为了保持流场绘制的时间相关性,经矢量线强化处理的卷积纹理作为下次卷积的属性纹理。以上的算法中矢量场主要是针对 2 维稳定流场,对于非稳定流场每次都需要重新计算,或者是 3 维曲面上的矢量场在观察视角变化时,每次需重新计算其投影平面的 2D 矢量场及相关的属性映射纹理。对于规则化的 2D 纹理可以直接映射为矢量纹理,而对于曲面上的矢量场或者非规则化的 2D 纹理场,可以通过 GPU 计算矢量在平面的投影,然后利用显卡的光栅化和插值能力,对其在投影平面上的投影进行采样,将采样结果映射为图形卡支持的 2D 矢量纹理。具体投影映射至 2D 矢量场的计算可以参照 ISA^[5,6] 纹理法。

3.1 强化处理

在已有的基于硬件的流场绘制算法中,有些算

法直接卷积纹理并显示,没有后处理过程,如 ISA 算法和 IBFV 相关算法;另外一种是采用 3×3 高通滤波核来强化突出流场的矢量线,如 GPUFLIC 算法。第 1 种方法显示的矢量线不够清晰,对比度弱,图像质量不高,后一种方法虽然矢量线对比增强,整体流线清晰,但还是有些区域矢量线比较杂乱,效果较基于 CPU 实现的 AUFLIC 算法的效果要差。这是因为 AUFLIC 算法采用了特殊的 as-long-as-possible 的卷积策略,能明显改善矢量线的质量,但该优化策略目前还不能在图像硬件中实现。

改善图像质量需要强化矢量线间的强度对比,而不是矢量线内部的强度对比。普通的 3×3 高通滤波缺乏方向性,不仅会影响矢量垂直方向的强度对比,而且会影响矢量方向的强度对比,考虑矢量场内在的有向性,采用了 1 维高通滤波,如图 2 所示,在矢量线垂直方向上进行高通滤波。其中, \mathbf{v} 是矢量场在 P_c 点的方向,两个垂直矢量方向的采样点分别为 P_{s1} 和 P_{s2} 。基于矢量线强化的 1 维卷积核为 $[-1, 3, -1]$,其方向正好与该矢量方向垂直,中心点 P_c 的权值为 3,而采样点 P_{s1} 和 P_{s2} 的权值均为 -1 ,采样点到中心点的距离与其速度大小线性相关,如式(1)所示。设置采样距离与速度大小线性相关,主要是考虑到矢量较小区域可能矢量线较为密集或者处于关键点区域,降低采样距离有利于降低该区域矢量线粒度,便于这些矢量线的区分。

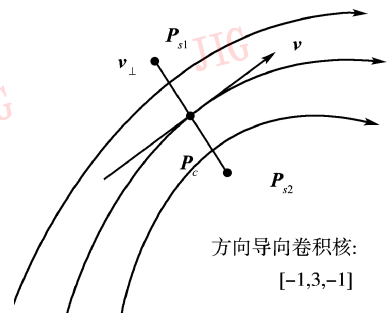


图 2 矢量线 1 维卷积示意图

Fig. 2 The illustration of 1D high-pass filtering

$$L = \|\mathbf{v}\| \cdot (1 - \alpha) + \alpha \quad (1)$$

其中, $\|\mathbf{v}\|$ 为矢量大小, α 是采样距离调节参数,其范围在 $0.0 \sim 1.0$, L 为采样距离。

从式(1)可以看出随矢量长度增加,采样距离也相应增加,采样点 P_{s1} 和 P_{s2} 的坐标由下式计算:

$$\mathbf{P}_{s_i} = \mathbf{P}_c \pm L \cdot \mathbf{v}_\perp \quad (2)$$

其中, \mathbf{v}_\perp 为垂直矢量方向的单位矢量, \mathbf{P}_c 点为中心

点坐标, P_s 为左右采样点坐标。

图 3 显示了数据 I 的纹理卷积结果,图 3(a) 是利用 FastLIC 算法绘制的结果,可以看出图像色彩比较平均,矢量线的不够清晰,对比不够,因为纹理 LIC 卷积过程本质是沿矢量方向一种低通滤波,会降低矢量线间的强度对比;图 3(b)和图 3(c)是本文算法卷积结果,其中卷积方式为向后单向卷积,卷积长度为 20。图 3(b)的卷积纹理是经过 2 维高通滤波处理,滤波核为 3×3 矩阵:

$$F = \begin{vmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{vmatrix}$$

与图 3(a)比较,其矢量线相对清晰明确,对比

度强,图像质量有所提高,但图像中存在一些显著的亮区域和暗区域,从而导致矢量线在帧间变化时出现明暗突变,这主要是高通滤波核 F 没用考虑矢量场的有向性特点。图 3(c)是经过矢量线强化处理的卷积纹理,强化处理明显地加强了矢量线间的色彩强度对比,而又不影响单个矢量线内部的色彩强度,同时也没有图 3(b)中的那些影响矢量线的连续性的明暗突变区域。

为了获得图像质量最佳的卷积纹理,矢量线强化需要选取合适的采样距离,即选取合理的 α 值,为此,分别考察了 α 值分别为 0.0、0.3、0.5、0.8 和 1.0 的 5 种情况,如图 4 所示,其中黑色框标示为放大区域。

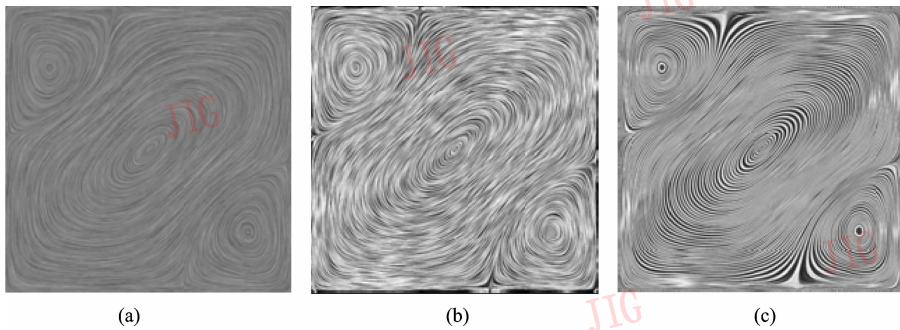


图 3 纹理卷积

Fig. 3 Texture convolution

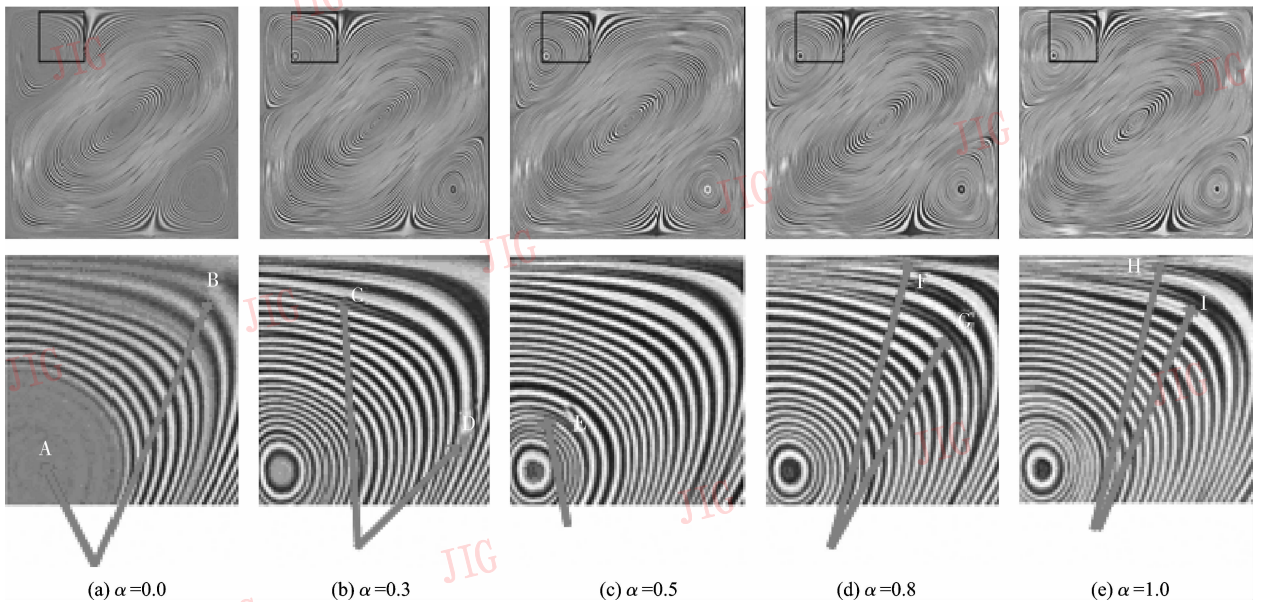


图 4 纹理卷积

Fig. 4 Texture convolution

当采样距离参数 α 值为 0.0 时,采样长度直接与速度成正比,中心区域 A 由于速度比较小,相应的采

样距离极短,因此该区域比较模糊,矢量线不突出;区域 B 也稍微模糊,矢量线总体有些暗淡。当 α 值为

0.3 时,图像总体质量较图 4(a)要好,矢量线清晰,而且矢量线之间对比明显,但也有局部区域不够理想,如区域 C 矢量线变的特别细,不够均匀;区域 D 存在部分模糊。当 α 值为 0.5 时,除了少部分区域 E 矢量线不够均匀外,图像质量进一步改善,整体矢量线清晰对比强。当 α 值为 0.8 时,图像质量开始下降,部分区域(如区域 F)开始变得模糊,部分区域(如区域 G)矢量线粒度不够均匀,甚至出现矢量线不连续的情况,图像质量开始下降。当 α 值为 1.0 时,图像质量继续下降,如区域 H 矢量线开始变得模糊,区域 I 的矢量线粒度更加不均匀。通常,对于不同的数据需

要适当调整 α 值的合理范围,一般 α 在 0.3 ~ 0.5 之间,矢量线最为清晰,图像质量最好。

3.2 纹理卷积

考虑到不稳定矢量场中流线和迹线不重合,选取了与 ISA 算法一致的向后 LIC 卷积方式。卷积核通常有直方、三角或指数类型等,选用最适合 GPU 实现的直方型卷积核。另外选择上节所述的强化处理方法,选择的采样参数 α 为 0.5,通过实验来测试合理的卷积长度。图 5 显示了卷积结果,其卷积长度分别为 5、10、20、30 和 40。总体上,随着卷积次数的增加,矢量线更加平滑,图像质量更好。

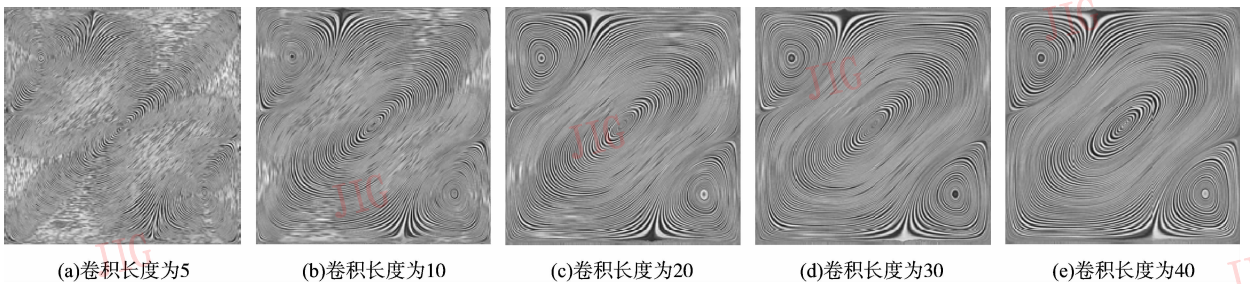


图 5 不同卷积长度的纹理卷积

Fig. 5 Texture convolution with different convolution length

具体来看,当卷积长度为 5 时,图像中黑白斑点区域比较多,卷积后的矢量线色彩强度存在突变,矢量线度不够。当卷积长度为 10 时,矢量线较光滑,且图像中黑白斑点区域减少,图像质量得到一定改善。当卷积长度为 20 时,矢量线更加光滑,图像中黑白斑点区域显著减少,图像质量极大地改善。在卷积长度为 30 和 40 时,矢量线更加光滑连续,图像中黑白斑点区域几乎消失,图像质量又有所提高,但改善的幅度明显趋缓。总的来说,随纹理卷积长度增加,矢量线趋于平滑,但这也导致流场运动方向不易观察,同时计算量大大增加,一般认为合理的卷积长度在 20 ~ 30 之间。

3.3 矢量属性映射纹理

在流场绘制中,单显示流场运动方向往往不够,有时候需要考察流场其他特征,如速率大小,拓扑结构和关键点区域。矢量大小映射纹理可以显示矢量场的速率分布特点;而矢量角度映射纹理不仅可以显示任意区域的矢量角度变化,而且有利于矢量场的关键点区域的识别;矢量场曲率映射纹理能有助于观察流场的拓扑分布和关键点区域,如涡流,但其本身没有任何方向信息。与 Risquet 的算法不同,本文算法中所有的属性映射计算都是在图形硬件中直接实现,并结合 FrameBuffer objece 技术直接将映射结果转成

相应的属性映射纹理,用于最后的合成显示。

3.3.1 基于矢量大小的映射纹理

基于矢量大小的色彩映射有助于观察矢量大小的分布及任意位置矢量的大小。图 6 (a)是矢量大小的映射色彩表,所有的矢量大小均被映射到 0.0 ~ 1.0 范围,蓝色表示矢量最小,黄色居中,黑色表示矢量最大。从图 7(a)中可以看出,数据 I 中区域 A 和 B 矢量速率最大,而区域 C、D 和 E 的矢量速率最小。数据 II 中区域 A、B、C 和 D 的矢量较小。

3.3.2 基于矢量角度的映射纹理

基于矢量角度的色彩映射有助于观察矢量角度的变化及任意位置矢量的角度。矢量角度的色彩映射是依据矢量的角度确定其对应色彩,图 6 (b)给出了矢量角度的色彩的映射表,其中 α 为矢量的角度,矢量到中心点距离与其矢量大小成正比。在矢量角度从 0°时,色彩为绿色,逆时针到 90°时,过度到白色,180°时,过度到粉红色,在 270°时,过度到黑色,最后转到 360°时,颜色回归到最初 0°时的绿色。

从图 7 (b)中可以观察到数据 I 有 3 个色彩汇聚中心 C、D 和 E,变化规律与图 6 (b)映射表中心点类似,中心点周围的矢量角度急剧变化,属于关键区域,对应拓扑结构可参照图 8(a)。图 7 (b)数据 II 有 4 个色彩汇聚中心 A、B、C 和 D 为关键点区域。

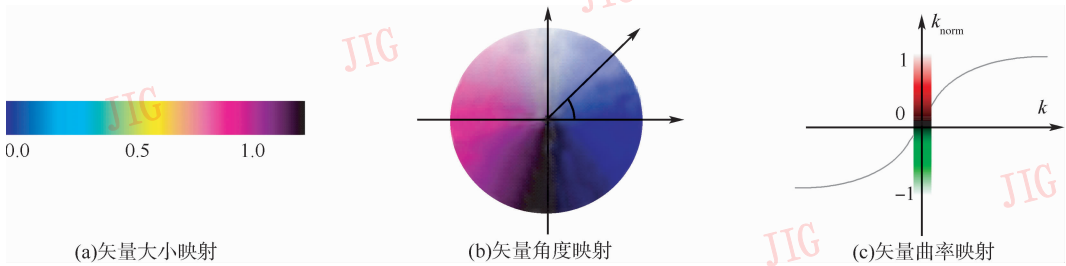


图 6 标量属性映射表

Fig. 6 Mapping of scalar properties

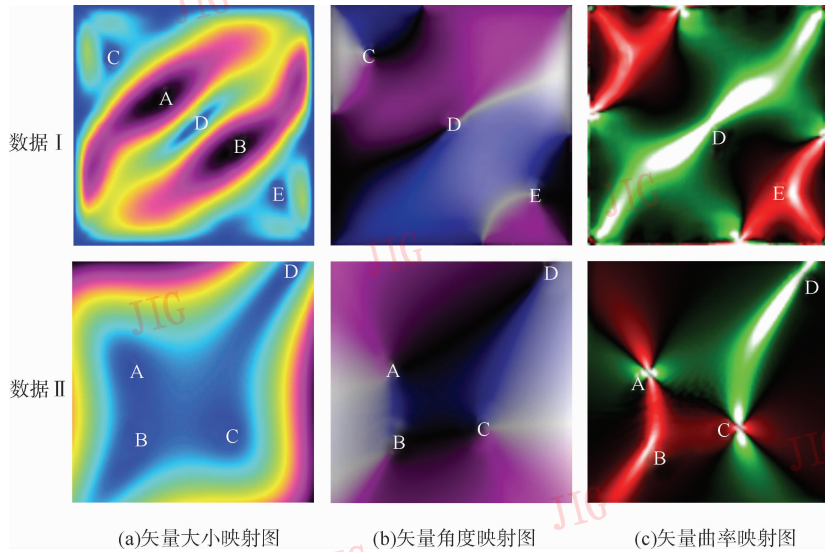


图 7 属性映射纹理

Fig. 7 Texture Mapping of Properties

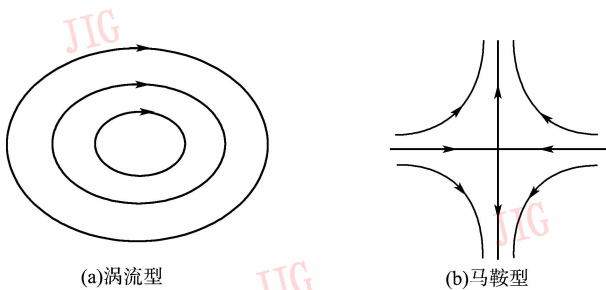


图 8 2 维矢量场的 2 种拓扑结构
Fig. 8 Topology of 2D vector fields

3.3.3 基于矢量曲率的特征纹理

矢量场的曲率图往往能直接反映出矢量场的主要拓扑结构和中心关键区域,有助于分析和观察矢量场的特征,Theisel 对矢量场的曲率和拓扑结构相关问题进行了很详细的分析。对于 2 维矢量场任意位置的曲率计算,无需计算该位置矢量线,即可按下式计算^[12]:

$$k(P) = \frac{(v_{x2} \cdot v_{yu} - v_{y2} \cdot v_{xu} + v_x \cdot v_y \cdot (v_{yv} - v_{xu})) (P)}{\|V(P)\|^3} \quad (3)$$

其中, $k(P)$ 表示 P 点的曲率, (v_x, v_y) 表示 P 点的矢量, $\|V(P)\|$ 表示 P 点处矢量的大小, (v_{xu}, v_{yu}) 和 (v_{xv}, v_{yv}) 分别表示矢量分别在 (u, v) 方向的偏导数。

曲率的范围是 $(-\infty, +\infty)$, 而曲率色彩映射表的范围在 $[-1, 1]$, 式 (4) 可将曲率归一化到 $[-1, 1]$ 区间的 k_{norm} 。

$$k_{norm} = \text{sgn}(k) \cdot (1 - e^{-|k| \cdot con}) \quad (4)$$

式中, k_{norm} 为归一化曲率, k 是矢量场任意一点的曲率值, con 为正, 用来调节矢量曲率映射纹理的明暗度, con 越大, 矢量曲率映射纹理图像越明亮, 否则, 纹理图像越暗淡。

图 6 (c) 为矢量曲率的色彩映射表, 当曲率为 $\pm \infty$ 的时, 颜色为白色, 曲率由 $-\infty$ 变化到 0, 色彩从白色变换到绿色再到黑色; 当曲率由 $+\infty$ 变化到 0, 色彩从白色变红色再到黑色。从图 7 (c) 上中可以观察到 3 个主要的高亮区域 C、D 和 E, 中心均为白色, 表示曲率为无限大, 为关键点区域。区域 C 和 E 的周围为红色, 表示其为逆时针涡流, 而区域 D 周围为

绿色,表示其为顺时针涡流。这些涡流方向不同,对应拓扑结构可参照图 8 (a)。从图 9 对应合成显示区域也可以观察到该拓扑类型。从图 7 (c)下中可以观察到 3 个主要的高亮区域 A、B、C 和 D,其中心均为白色,表示曲率极大,属关键点区域。区域 B 和 D 分别是逆时针涡流和顺时针涡流;而区域 A 和 C 周围是红绿相间,属于马鞍型,其拓扑如图 8 (b)所示。也可从图 9 对应的合成显示区域观察到拓扑。

3.4 噪声注入与合成

在卷积纹理中通常会进行噪声注入混合的处理,主要是防止多次纹理卷积后导致属性纹理无变化或无显著变化。在本文算法中,预生成一张随机纹理作为所需注入的噪声纹理,其后在流场循环绘制中,按一定比例如 3% ~ 5%,随机进行噪声强度取反,最后按 10% 的比例与卷积纹理混合。

卷积纹理经噪声处理和强化处理后,可与矢量场的色彩映射纹理合成并最终显示,具体合成模式可以是比例混合或直接相乘。

比例混合公式为

$$T_b = (T_c \cdot \alpha + T_p \cdot (1 - \alpha)) \cdot L \quad (5)$$

直接相乘公式为

$$T_b = T_c \cdot T_p \cdot L \quad (6)$$

式中, T_c 是矢量线强化后的卷积纹理值; T_p 为基于

矢量场特性的映射纹理值,如矢量大小; α 为比例混合的参数一般为 0.3 ~ 0.7; L 为亮度调节系数,调节整体显示的明暗效果,一般设为 0.8 ~ 1.3; T_b 为最后显示的纹理合成值。

4 实验结果

实验环境为 Windows XP 普通 PC 机,CPU P4 3G、1 024 M 内存、128M Geforce 6800LE 图形卡,算法采用 OpenGL 和 Cg 渲染语言实现,显示窗口的分辨率为 512 × 512。目前本文算法主要是针对 2 维稳定矢量场,选择 2 个 2 维矢量数据来测试本文算法。选用的数据 I 的网格为 256 × 256,数据 II 为 40 × 40。具体实现时,纹理卷积和噪声的注入混合可在图形卡的一次渲染过程中完成。虽然数据 I 和 II 的网格不同,但因矢量场均直接映射为矢量纹理而不影响流场绘制的速度,所以对于这类矢量数据,绘制算法均能超过 30fps,完全满足实时绘制要求。

图 9 显示 2 种合成模式下绘制的部分结果,左边的合成模式是直接相乘,右边是比例混合,比例混合模式偏亮点。图中部分区域色彩偏暗,与色彩映射表中选择了黑色有关,亦可选择其他映射色彩来提高亮度。

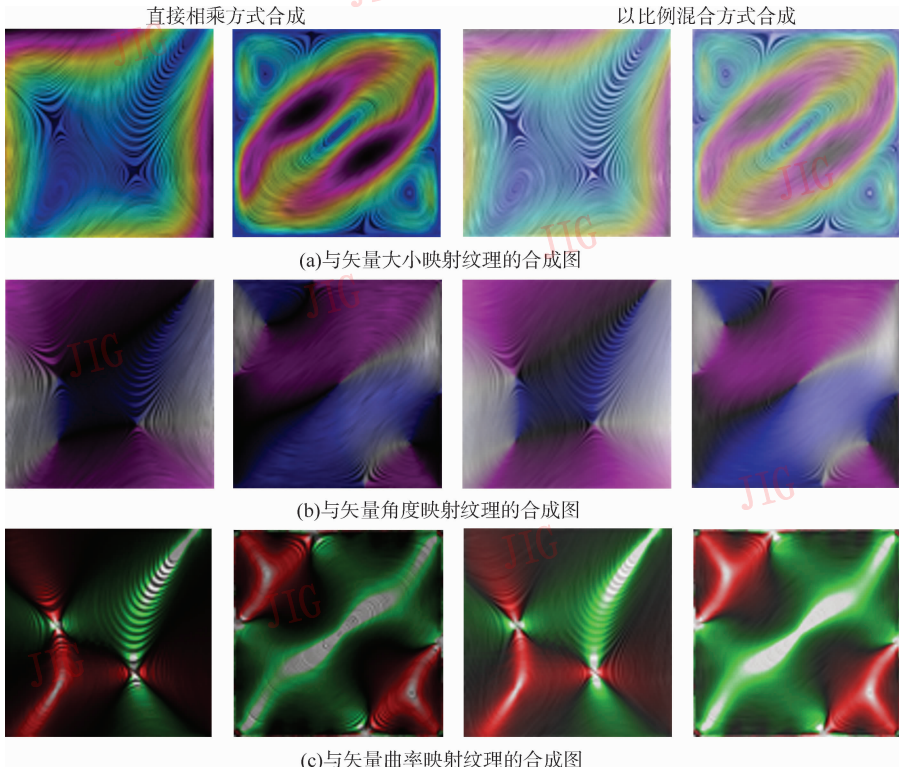


图 9 部分数据的流场绘制

Fig. 9 The rendering results

5 结 论

提出了一种有效的增强型2维流场实时绘制算法。该算法不仅可以清晰显示流场运动方向,而且能同时显示流场其他属性特征,有助于观察流场特征及主要拓扑结构。同时,算法采用矢量线强化策略,极大地增强矢量线间的强度对比,改善了绘制图像质量及其空间相关性。

本文详细比较矢量强化的采样距离对卷积纹理的图像质量和矢量线的对比度的影响,认为采样距离参数 α 在0.3~0.5范围内可以达到较好的效果,针对不同数据该范围可以适当调整。合适的卷积长度在20~30之间,卷积长度太小容易导致图像质量降低,黑白斑点增多从而影响流场矢量线的连续性;卷积长度太大,虽然矢量场更加平滑,但矢量线本身强度变化偏小,不易流场实际运动方向的观察,并增加了图形硬件系统负担。

今后将研究改变矢量场的属性映射表及相关属性色彩映射方式。考察其对流场绘制质量的影响,同时,将本文算法拓展到非稳定流场绘制和3维曲面矢量场绘制也将是下一步研究的重点。

参考文献 (References)

- 1 Heidrich W, Westermann R, Seidel H-P, *et al.* Applications of pixel textures in visualization and realistic image synthesis [A]. In: Proceedings of ACM Symposium on Interactive 3D Graphics 1999 [C], Atlanta, Georgia, United States, 1999: 127~134.
- 2 Jobard B, Erlebacher G, Hussaini M Y. Hardware-assisted texture advection for unsteady flow visualization [A]. In: Proceedings of IEEE Visualization 2000 [C], Salt Lake City, Utah, United States, 2000: 155~162.
- 3 van Wijk J J. Image based flow visualization [A]. In: Proceedings of

- ACM SIGGRAPH 2002 [C], San Antonio, Texas, United States, 2002: 745~754.
- 4 Jobard B, Erlebacher G, Hussaini M Y. Lagrangian-eulerian advection of noise and dye textures for unsteady flow visualization [J]. IEEE Transactions on Visualization and Computer Graphics, 2002, 8(3): 211~222.
- 5 Laramée R S, Jobard B, Hauser H. Image space based visualization of unsteady flow on surfaces [A]. In: Proceedings of IEEE Visualization 2003 [C], Seattle, Washington, United States, 2003: 131~138.
- 6 Grabner M, Laramée R S. Image space advection on graphics hardware [A]. In: Proceedings of the 21st Spring Conference on Computer Graphics and its Applications 2005 [C], Budmerice, Slovakia, 2005: 75~82.
- 7 Weiskopf D, Erlebacher G, Ertl T. A texture-based framework for spacetime-coherent visualization of time-dependent vector fields [A]. In: Proceedings of IEEE Visualization 2003 [C], Seattle, Washington, United States, 2003: 107~114.
- 8 Shen H W, Kao D L. A new line integral convolution algorithm for visualizing time-varying flow fields [J]. IEEE Transactions on Visualization and Computer Graphics, 1998, 4(2): 98~108.
- 9 Liu Z P, Moorhead R J. Accelerated unsteady flow line integral convolution [J]. IEEE Transactions on Visualization and Computer Graphics, 2005, 11(2): 113~125.
- 10 Li G S, Tricoche X, Hansen C. GPUFLIC: interactive and accurate dense visualization of unsteady flows [A]. In: Proceedings of Eurographics/IEEE-VGTC Symposium on Visualizations 2006 [C], Lisbon, Portugal, 2006: 29~33.
- 11 Okada A, Lane D. Enhanced line integral convolution with flow feature detection [R], NAS-96-007, Moffett Field, California, United States: NASA Ames Research Center, 1996.
- 12 Theisel H. Vector field curvature and applications [D]. Rostock, German: University of Rostock, 1996.
- 13 Risquet C P, Theisel H. Enhanced Integrate and draw with feature visualization [EB/OL], <http://www.icg.informatik.uni-rostock.de/Projekte/FlowVisualization/EIDraw/PaperSource/EIDraw.doc>, 1998-12-20-2008-5-20.