

田丰林, 朱新升, 刘巍, 等. 基于传输函数的中尺度涡旋时空连续可视化[J]. 海洋学报, 2020, 42(9): 119–133, doi:10.3969/j.issn.0253-4193.2020.09.013

Tian Fenglin, Zhu Xinsheng, Liu Wei, et al. Time-space continuous visualization of mesoscale vortices based on transfer function[J]. Haiyang Xuebao, 2020, 42(9): 119–133, doi:10.3969/j.issn.0253-4193.2020.09.013

基于传输函数的中尺度涡旋时空连续可视化

田丰林^{1,2}, 朱新升^{1,2}, 刘巍³, 韩妍娇⁴, 陈戈^{1,2*}

(1. 中国海洋大学 信息科学与工程学院, 山东 青岛 266100; 2. 青岛海洋科学与技术试点国家实验室 区域海洋动力学与数值模拟功能实验室, 山东 青岛 266237; 3. 青岛市计量技术研究院, 山东 青岛 266100; 4. 哈尔滨市勘察测绘研究院, 黑龙江 哈尔滨 150010)

摘要: 本文结合二维流线可视化技术和中尺度涡旋识别技术, 提出了 3 种中尺度涡旋时空连续可视化的方法: 基于 OW 参数的涡旋可视化方法、基于栅格模板的涡旋可视化方法和基于矢量模板的涡旋可视化方法, 这 3 种方法分别基于 Okubo-Weiss 算法、Faghmous 的算法和 Liu 的算法来进行涡旋识别, 同时将流场可视化的结果填充到涡旋内部, 以获得更好的可视化效果。在可视化过程中本文引入了传输函数来对涡旋中的流线颜色和透明度进行实时交互, 能够在控制界面上通过设置 Key 值点的颜色和位置来控制速度、涡度和 OW 参数等信息的显示效果。本文在性能和显示效果方面比较了 3 种方法的优劣。从性能上来讲, 性能由高到低依次为: 基于 OW 参数的涡旋可视化方法、基于栅格模板的涡旋可视化方法和基于矢量模板的涡旋可视化方法。从显示效果上来讲, 基于 OW 参数的涡旋可视化方法在三者中最差, 效果中有较多的杂乱的短线, 同时涡旋边界较小, 局限于涡旋核心区; 基于栅格模板的涡旋可视化方法较第一种方法的显示效果有所提升, 杂乱的短线较少, 涡旋相对完整, 但由于数据分辨率不够高的原因, 在放大多倍后涡旋边界呈现锯齿状; 基于矢量模板的涡旋可视化方法显示效果最好, 涡旋完整、饱满。同时, 因为首先进行了涡旋边界的重构, 将涡旋边界矢量化, 涡旋边界更加平滑。相对于传统长时间序列的涡旋可视化的方法而言, 这 3 种方法提供了一个美观、动态和更富信息性的可视化方法, 同时由于传输函数的加入, 其可以成为科研人员研究涡旋的一个实用的工具。

关键词: 涡旋可视化; 中尺度涡; 传输函数

中图分类号: P731.16

文献标志码: A

文章编号: 0253-4193(2020)09-0119-15

1 引言

中尺度涡旋是旋转运动着的并伴随有温度异常的水团, 它的直径一般为数十至数百千米, 寿命可达数天到数月不等^[1]。中尺度涡旋广泛存在于全球海洋中, 根据旋转方向不同, 可将它们分为反气旋涡和气旋涡, 在北半球, 反气旋涡顺时针旋转, 气旋涡逆时针

旋转, 而南半球则相反^[2]。它们在动量、质量、热量、营养物质以及盐和其他海水元素的运输中发挥着重要的作用, 影响着海洋环流、水分布和海洋生物分布^[3-5]。中尺度涡旋的发现是人类理解海洋环流的一个突破, 改变了人们对洋流的传统看法。因此, 长时间序列全球范围的涡旋研究对于分析和理解海洋能量和质量输运以及全球涡旋的时空变异性具有重要意义并且

收稿日期: 2019-05-30; 修订日期: 2020-04-16。

基金项目: 山东省海洋科技基金(2018SDKJ0102-08); 国家重点研发项目(2016YFC1401008, 2017YFA0603203); 中央高校基础研究基金(201762005)。

作者简介: 田丰林(1978—), 男, 山东省青岛市人, 博士, 副教授, 主要从事海洋信息可视化与海洋遥感工作。E-mail: tianfenglin@ouc.edu.cn

* 通信作者: 陈戈(1965—), 男, 博士, 教授, 主要从事海洋遥感与大数据工作。E-mail: gecchen@ouc.edu.cn

成为近几年物理海洋学领域研究的重点。

Dong 等^[6]根据使用数据集的不同将涡旋识别方法分为两大类: 欧拉(Eulerian)方法和拉格朗日(Lagrangian)方法。欧拉数据是指时刻快照数据或空间场数据, 拉格朗日数据是指水团或物质粒子的轨迹数据^[7]。其中欧拉方法可以细分为 OW(Okubo-Weiss)参数法^[8-9]、缠绕角法(Winding-Angle, WA)^[10]和基于涡旋几何特征的涡旋检测方法(Vector-Geometry, VG)^[11]等方法。在欧拉方法的基础上, 发展出了一些涡旋追踪算法。例如 Liu 等^[12]运用此方法, 利用 1993–2010 年的高度计地转流异常数据, 对北太平洋海域的 6 000 多个涡旋进行了追踪研究; Faghmous 等^[13]也曾基于 1993–2014 年的卫星高度计数据, 共提取了 4 500 万个中尺度涡旋和约 330 万个涡旋轨迹, 并建立了 1993–2014 年的涡旋识别及追踪数据集。

在流场可视化方面, Jobard 和 Lefer^[14]于 1997 年提出了一种用于二维稳定场的均匀放置流线的算法, 并且在 2000 年将这种算法拓展到多分辨率不稳定场的可视化上^[15], 之后基于此种方法, 出现了多种改进算法^[16-17]; Weiskopf 等^[18]于 2005 年提出了一种结合了粒子和纹理的时空连续可视化框架, 用于时变流场可视化, 他们使用了 Pighin 等^[19]提出的径向基函数(Radial Basis Functions, RBFs)来维持线条的动态均匀分布; 何珏等^[20]和 Tian 等^[21]将 Weiskopf 等^[18]的可视化框架和径向基函数运用到海洋流场中。

在传输函数方面, 其被广泛应用在空间标量场可视化中, 例如 Kindlmann 和 Durkin^[22]利用二维传输函数进行空间标量场可视化, 他们通过计算三维标量场的一阶和二阶方向导数与标量值构造标量场的统计直方图来控制可视化效果; Sereda 等^[23]通过构建体素局部极大值和极小值的 LH 统计直方图来进行标量场数据边界的可视化。矢量场方面, 传输函数也有一些应用, 例如 Helgeland 和 Andreassen^[24]于 2004 年实现了基于纹理方法的三维流场可视化, 并且将可交互的传输函数应用其中。

传统的涡旋识别方法得到的识别结果多数是以统计图表或者图像的形式显示^[25], 这些方式显示的结果可能会造成部分原始信息的缺失, 并且不利于观察涡旋的时空连续运动情况。传统的涡旋识别与追踪都是分开进行的, 无法将二者有机的结合起来。同时其不提供交互方法, 无法按用户想法调整数据的显示方式和效果。在这种情况下, 能够恢复甚至增强数据的整体结构和具体细节的时空连续可视化就显得尤为重要。现有的研究中, 时空连续的中尺度涡旋可视

化方面的研究还很少。

为解决上述问题, 本文结合二维流线可视化技术与传输函数技术提出了 3 种方法来实现涡旋运动轨迹及其二维表面流动情况的时空连续可视化。这 3 种方法分别是: 基于 OW 参数的涡旋可视化方法、基于栅格模板的涡旋可视化方法和基于矢量模板的涡旋可视化方法。这 3 种方法的创新之处在于: (1) 实现了二维流线可视化技术与涡旋识别与追踪技术的结合; (2) 利用线性插值算法来解决不同时间数据之间连续可视化的平滑过渡的问题; (3) 结合可交互的传输函数实现流场数据的交互显示。

2 数据集和处理方法

2.1 MSLA 数据集

本文所使用的 MSLA-UV 速度场纹理数据集的类型是 Ssalto/Duacs 网格化多任务高度计产品, MSLA-UV 数据分为 two-sat-merged^[25]与 all-sat-merged^[26]两种类型, 本文所使用的是延时的 all-sat-merged 类型的数据, 其融合了多颗卫星的数据, 与 two-sat-merged 类型数据相比, 数据质量较高。数据格式为 NetCDF, 空间分辨率为 $0.25^{\circ} \times 0.25^{\circ}$, 时间分辨率为 1 d。使用前将数据进行格式转换, 将 NetCDF 格式转换为 TIFF(Tad-ded Image File Format)格式(图 1)。

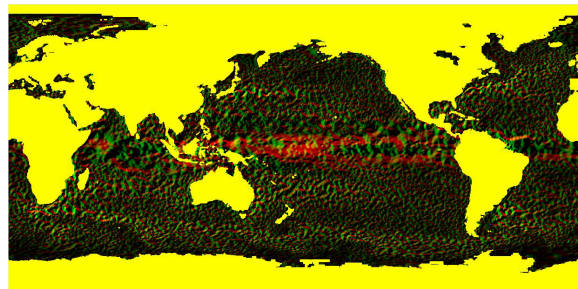


图 1 TIFF 格式的 MSLA-UV 数据(2016 年 1 月 1 日)

Fig. 1 MSLA-UV data in TIFF format(January 1, 2016)

黄色区域为无效值区域, 其他不同颜色表示流速差异

The yellow area is the invalid value area, and other different colors indicate the flow rate difference

2.2 涡旋栅格模板数据

本文利用 Faghmous 等^[13]提出的涡旋识别算法来生成涡旋栅格模板数据, 此方法首先对海面高度异常(Sea Level Anomaly, SLA)栅格数据进行 5×5 的邻域搜索, 寻找出邻域中的极值作为种子点从而找到涡心, 然后从涡心向外迭代 SLA 等高线, 直到包含另一个涡心时停止(其迭代 SLA 等高线的步长为 0.05 cm, 涡旋边界内包含的像素数目不小于 4), 从而识别出

涡旋。最终结果以灰度图的形式呈现出来, 图 2 所示为 2016 年 1 月 1 日 $35.0^{\circ}\sim 47.8^{\circ}\text{N}$, $141.5^{\circ}\sim 169.5^{\circ}\text{E}$ 黑潮延伸体的一部分。

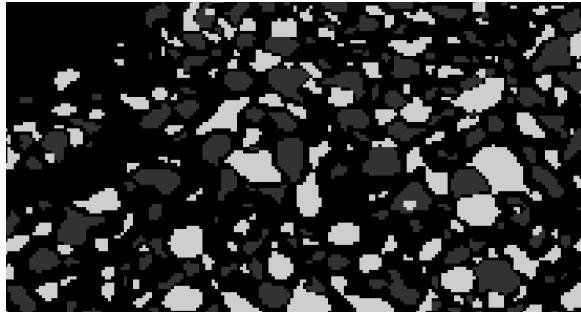


图 2 栅格模板数据 (2016 年 1 月 1 日)
Fig. 2 Grid template data (January 1, 2016)

黑色为无效值区域 (大陆与北极冰雪覆盖地或非涡海域), 浅灰色为冷涡, 深灰色为暖涡
Black represents the invalid value area (continent and Arctic snow-covered areas or non-vortex sea areas), light gray represents cold vortex, and dark gray represents warm vortex

2.3 涡旋矢量模板数据

2.3.1 识别数据

涡旋矢量模板识别数据集是基于 Liu 等^[12]提出的算法而得到的。该方法通过半径、振幅、涡核、封闭 SLA 等值线等来表征涡结构。其将全球海平面异常数据分块, 用并行计算方法对这些区域进行同时识别, 之后再对识别结果拼接成一张全球涡旋图。

在可视化过程中我们需要对两张数据纹理进行插值, 来平滑由于数据变化带来的视觉突变。但是

Liu 等^[12]的方法生成的涡旋边界上的节点数目会受到涡旋大小和形状的影响, 从而导致每个涡旋的节点数不同, 并且不均匀。这个特点并不利于我们对涡旋进行可视化, 因此, 我们对涡旋边缘的节点坐标进行了重构, 重构示意图如图 3 所示。

我们以涡心为中心建立笛卡尔坐标系, 从 X 轴方向开始顺时针将涡旋分成 36 个扇形, 即每个扇形的顶角均为 10° , 从而形成涡旋边界上的 37 个顶点, 其中第一个顶点与最后一个顶点重合。对所有的涡旋都进行这样的操作, 这样每个涡旋都会被重建, 并且在同一轨迹上相邻两天的涡旋可以建立时间连续的顶点到顶点的几何关系。从而保证了矢量化识别数据在插值过程中的时空连续性。

2.3.2 涡旋追踪数据集

本文中使用的涡旋追踪算法来自 Sun 等^[27]提出的混合追踪算法。这种混合算法兼顾了涡旋的物理属性, 如距离、面积和振幅大小, 同时又兼顾了涡旋的几何属性, 如涡旋传播过程中边界形状的变化。追踪的结果如图 4 所示, 图中的红色折线是涡心所经过的路径, 也就是涡旋运动所形成的轨迹, 该折线上的黑色圆点即为该条轨迹上每一天识别出的涡旋的涡心所在的位置, 而不同颜色的多边形边界是通过涡旋识别得到的涡旋每一天的边界。

3 二维流场可视化

3.1 时空框架

流场可视化是科学可视化领域的一个经典研究方向, 其方法主要可以分为 4 类: 图标法、几何法、纹

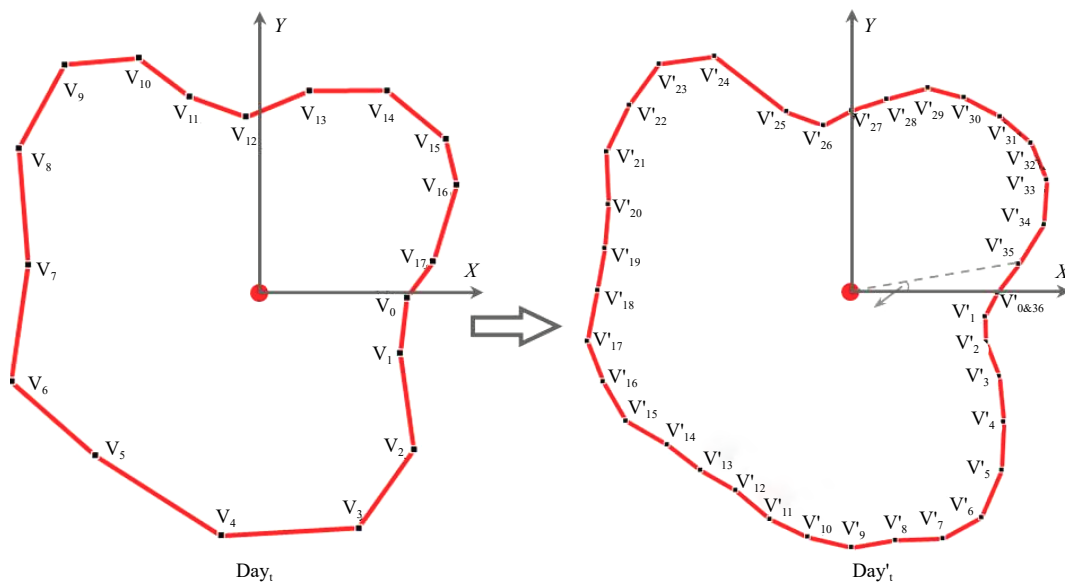


图 3 涡旋顶点坐标重构示意图
Fig. 3 Reconstruction of eddy vertex coordinates

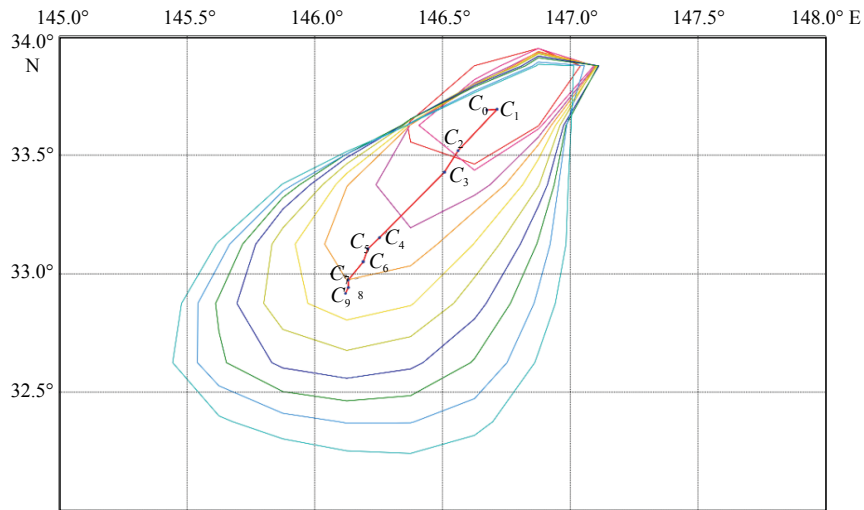


图 4 涡旋追踪结果示意图

Fig. 4 Schematic diagram of eddy tracking results

红色折线是涡心所经过的路径,折线上的黑色圆点为涡心位置

The red line is the path through the eddy center, and the black dots on the red line are the position of the eddy center

理法、拓扑法。其中几何法将离散几何对象置于海洋流场中,其特征反映海流的基本性质,适合我们实现时空连续的流场可视化,所以我们选择几何法来可视化流场。

在时空流场可视化方法中,迹线和流线是两个重要的概念。在时变流场中,迹线用于描述流场中一个粒子在某个时间段的流动轨迹,而流线描述时变流场在某一时刻任意一点处速度的切线方向,对于流场中的特定位置,某一时刻有且仅有一条流线通过该点(除奇点外)。该点处流线的切线方向即表示该点处速度场的方向^[28]。用流线表达的流场中,我们可以看到流场中每个点的速度信息,但是无法获得粒子在流场中的运动情况。而用迹线表达的流场中,我们能表达粒子的运动情况,但又不能看到流场的速度信息。Weiskopf等^[18]提出了一种不稳定场的时空连续可视化框架,参考其观点,我们构建的框架如图5所示。黑色实心圆点代表粒子,迹线 \mathbf{x}_{path} 用蓝色虚线表示,迹线与瞬时空间切片的交点用灰色圆点表示,红色实线表示流线 $\mathbf{x}_{\text{stream}}$, t_0 、 t 表示两个瞬时空间切片。在此框架中,我们将流场以数据的时间分辨率为间隔切分成若干个瞬时空间切片,每一个瞬时空间切片都可以看作一个二维稳定流场。在瞬时空间切片上进行流线积分来表示流场结构,并且通过粒子在不同切片中的轨迹移动来表现时间相关性。

在二维欧式空间中,不稳定场可以用映射 $u(x, t)$ 来表示,每点的值代表在 t 时刻、 x 位置上的速度矢量值,迹线 \mathbf{x}_{path} 由下面的常微分方程决定^[29]:

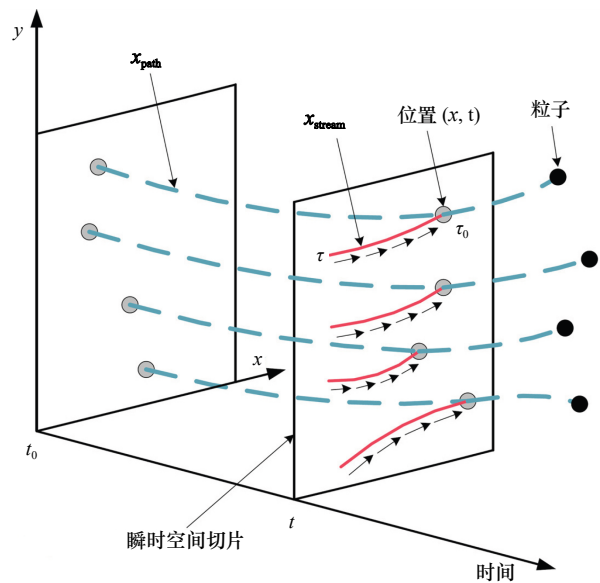


图 5 二维矢量场时空连续框架示意图(修改自文献^[20])

Fig. 5 Schematic diagram of space-time continuum frame of two-dimensional vector field (modified from reference^[20])

$$\frac{d\mathbf{x}_{\text{path}}(t; \mathbf{x}_0, t_0)}{dt} = \mathbf{u}[\mathbf{x}_{\text{path}}(s; \mathbf{x}_0, t_0), t]. \quad (1)$$

初始条件 $\mathbf{x}_{\text{path}}(t_0; \mathbf{x}_0, t_0) = \mathbf{x}_0$ 在 (\mathbf{x}_0, t_0) 处给出,通过积分求得常微分方程(1)的解,即迹线的方程为

$$\mathbf{x}_{\text{path}}(t; \mathbf{x}_0, t_0) = \mathbf{x}_0 + \int_{t_0}^t \mathbf{u}[\mathbf{x}_{\text{path}}(s; \mathbf{x}_0, t_0)] ds. \quad (2)$$

在每一时刻的空间切片中,流场都可以看成一个稳定场,流线可以更好地表达稳定场的结构,流线 $\mathbf{x}_{\text{stream}}$ 是下面方程的解:

$$\frac{d\mathbf{x}_{\text{stream}}(\tau; \mathbf{x}_0, t_0)}{d\tau} = \mathbf{u}[\mathbf{x}_{\text{stream}}(\tau; \mathbf{x}_0, t_0), t]. \quad (3)$$

τ 与 τ_0 和 t 不同,它只是控制曲线积分的一个参数,决定了曲线的精细程度,与真实的物理时间没有关系。积分可以求得方程的解:

$$\mathbf{x}_{\text{stream}}(\tau; \mathbf{x}_0, t_0) = \mathbf{x}_0 + \int_{\tau_0}^{\tau} \mathbf{u}[\mathbf{x}_{\text{stream}}(s; \mathbf{x}_0, t_0)] ds, \quad \tau < \tau_0. \quad (4)$$

$\tau < \tau_0$ 表示我们采用后向积分方式积分流线。这样能确保粒子位于流线的头部,获得较好的可视化效果。

3.2 自适应分辨率的流线可视化

在本文的可视化方法中,粒子除了具有位置信息之外,还具有年龄信息,其记录了粒子初始生成到当前时刻的时间。该年龄的初始值为0,如果初始化时粒子被播撒到数据范围以外(大陆或者北极冰盖上),年龄会被标记为“死亡”状态。粒子有一个设定的年龄上限,粒子年龄达到该上限时也会变成“死亡”状态。“死亡”状态的粒子不会参加流线积分,并会基于Simplex噪声算法^[30]以随机的方式重新播撒到视口中。每有一个粒子死亡,就会有一个新粒子生成,所以在视口中的粒子数量是一定的,流线的数量也是恒定的。如图6所示,当相机向地球拉近时,地球表面的数据区域单位面积内被更多的粒子可视化,可视化范围变小但分辨率提高;当相机拉远时,相同大小的数据范围被更少的粒子可视化,分辨率降低。这样就实现了基于视口的自适应分辨率的可视化表达。

3.3 密度控制

随着流场的作用,粒子可能汇聚到一起,导致某些地方非常稀疏而某些地方非常密集,这时就需要控制粒子密度来获得更好的显示效果,本文采用根据自适应径向基函数(RBF)^[18-19]计算出来的粒子影响域来控制粒子密度,即

$$I(\mathbf{x}) = \sum_{\|\mathbf{x}-\mathbf{x}_i\| < R} \lambda_i \varphi_i(\|\mathbf{x}-\mathbf{x}_i\|), \quad (5)$$

式中, λ_i 、 \mathbf{x}_i 、 \mathbf{x} 、 φ_i 分别为第 i 个粒子的权重、中心位置、所求点位置和径向基函数。

$$\varphi(r) = e^{-r^2/(2\sigma^2)}, \quad (6)$$

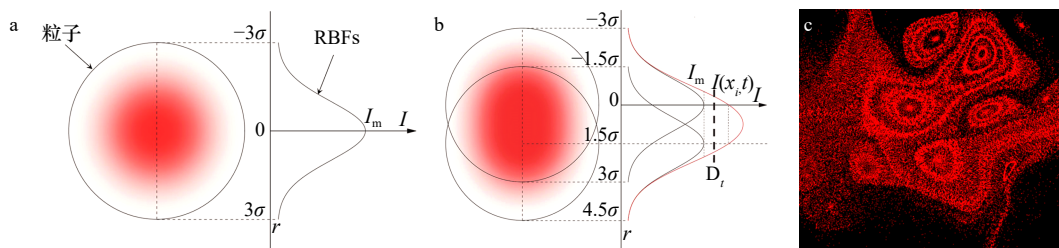


图7 单个粒子的影响域(a),两个粒子的影响域叠加(b)和局部的概率密度图(c)

Fig. 7 The influence domain of a single particle (a), the superposition of the influence domain of two particles (b), and the local probability density diagram (c)

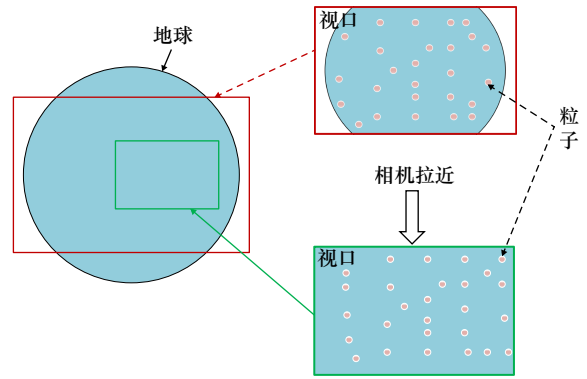


图6 粒子密度随相机变化示意图(视口内播撒的粒子数目相同)

Fig. 6 Schematic diagram of particle density changing with the camera (the number of particles in the viewport are the same)

使用式(6)作为影响域的径向基函数。对于一个瞬时空间切片,影响域 $I(\mathbf{x}, t)$ 为

$$I(\mathbf{x}, t) = \sum_i \sum_{\|\mathbf{x}-\mathbf{x}_{\text{path}}(t; \mathbf{x}_i, t_0)\| < R} \lambda_i \varphi_i(\|\mathbf{x}-\mathbf{x}_{\text{path}}(t; \mathbf{x}_i, t_0)\|). \quad (7)$$

如图7a和图7b所示,单个粒子的影响域在径向基函数的作用下从中心到边缘不断降低。粒子聚集时,聚集区某一位置会受到多个粒子的影响,影响域值相互叠加。根据 $3-\sigma$ 法则,将高斯函数的标准差 σ 设置为 $R/3$, R 是径向基函数的半径,设定为概率密度图中粒子直径的像素距离的一半, λ_i 设为1。将这些影响域为 R 的粒子渲染到纹理,就生成了一张概率密度图,如图7c所示为概率密度图的一个局部,在流场的作用下,一些地方粒子汇集产生高值区域,一些地方粒子稀疏,甚至产生了黑色的0值区域。

3.4 基于GPU的算法实现

一个PASS指的是一个渲染流水线,可以包含多种着色器和一些相关的状态设置,它是一个基本的功能单元。流场可视化的实现采用多PASS技术,步骤如图8所示。第一步是初始化粒子分布,在时空框架中,使用可以移动的粒子作为生成流线的种子点,在

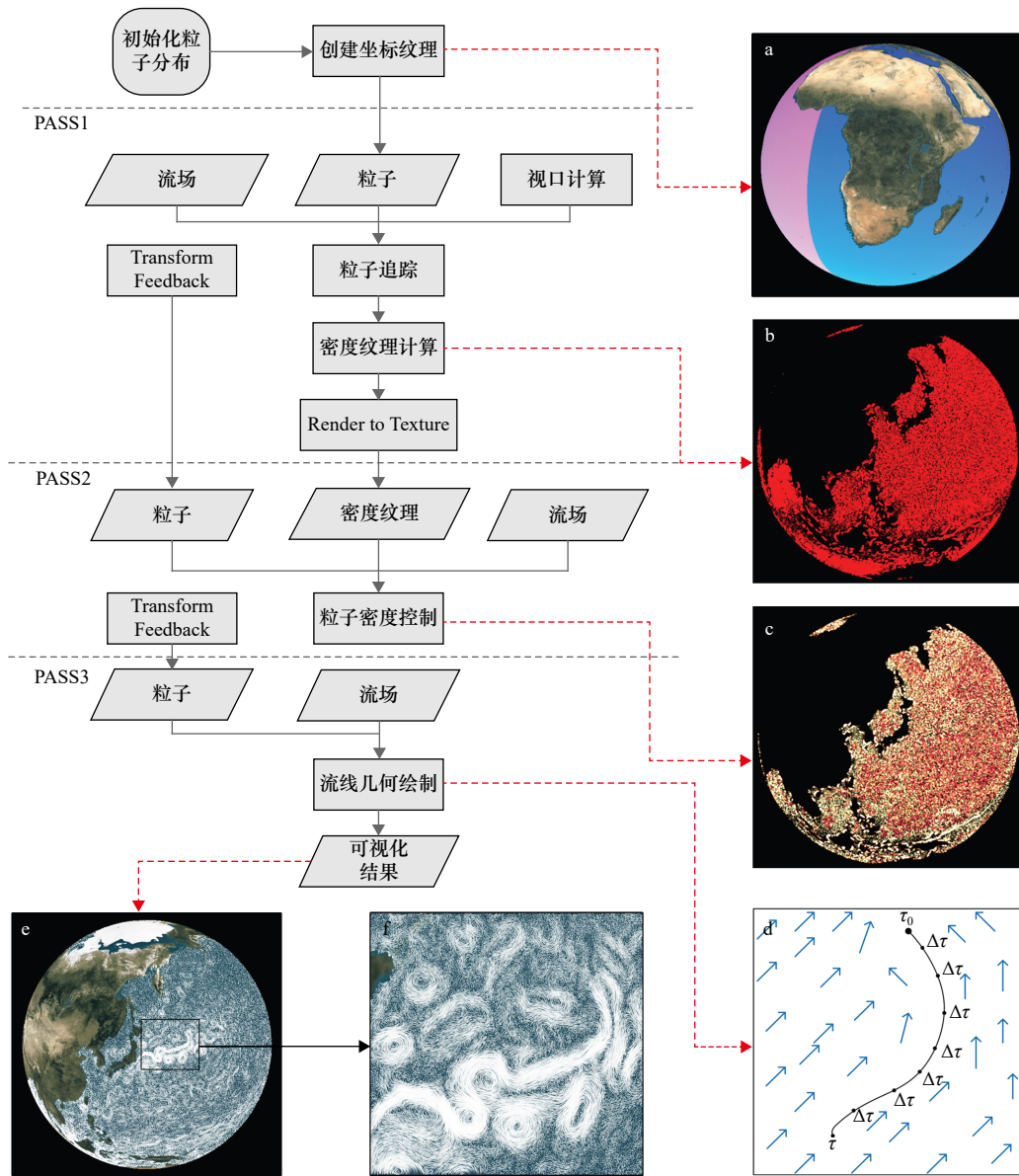


图 8 二维流线可视化流程图

Fig. 8 2D streamline visualization flowchart

- a. 坐标纹理; b. 概率密度纹理; c. 颜色表示年龄信息的密度纹理, 粒子年龄为 0 时, 颜色为白色, 年龄在 [0,1] 时为黄色, 年龄在 [2,3] 时颜色为红色; d. 逐步积分生成流线, τ_0 为起点, τ 为终点; e. 地球表面生成的流线; f. 地球表面生成流线的局部放大图
- a. coordinates texture; b. probability density texture; c. color represents the age information of density texture; when the particle's age is 0, the color is white; when the age is [0,1], it is yellow; when the age is [2,3], it is red. d. step-by-step generation of streamlines, τ_0 and τ are the beginning and end of integral time; e. streamlines on the earth's surface; f. local magnification of streamlines on the earth's surface

全球范围内均匀播撒, 在 $1^\circ \times 1^\circ$ 的格网内部播撒一颗固定位置的粒子和一颗随机位置的粒子(在格网内随机取位置), 总共播撒 $360 \times 180 \times 2$ 个粒子, 粒子随后被传入 VBO(Vertex Buffer Object)中, 进而传入到 PASS1 中的顶点着色器。

此过程的同时要渲染一张坐标纹理, 如图 8a 所示, 该纹理的 R 和 G 通道分别为归一化到 [0,1] 的经度和纬度。可以通过采样这张纹理的 R、G 通道的

值, 获得传入粒子的经纬坐标, 进而通过经纬坐标采样流场数据纹理中该粒子处的流场速度值。

之后在 PASS1 中的顶点着色器中进行粒子追踪: 通过 4 阶龙格库塔算法计算出粒子的位移, 赋予粒子新的位置并增加年龄, 追踪后的粒子信息(位置信息、年龄信息)有两个去向: (1) 传入 PASS1 的片元着色器中, 在此计算粒子影响域, 生成一张概率密度图; (2) 通过 Transform Feedback 技术传入 PASS2 中。

PASS1 中通过 OpenGL 的 Transform Feedback 技术传入的粒子信息在 PASS2 中继续进行粒子追踪, 不过我们同时利用 PASS1 中生成的概率密度图进行粒子年龄的控制, 进而控制粒子密度。根据粒子的位置, 对概率密度图进行采样, 得到概率密度值, 并根据此值更改年龄。高密度区域的粒子年龄迅速增加, 快速死亡, 低密度区域的粒子年龄缓慢增长, 存活时间较长。图 8c 表示年龄在自然增长和概率密度图的双重控制下的变化情况。新撒入的粒子年龄为 0, 我们将其编码为白色, 随着时间的推移, 年龄增长到 1, 颜色变为黄色, 年龄超过 2 时, 年龄为红色, 红色的粒子在不久后将会死亡。

最后一步就是根据式(4)采用龙格库塔四阶积分反向积分生成流线, 如图 8d 所示, 在 PASS3 中的几何着色器中绘制流线, 通过调整顶点数量 N 和积分步长 $\Delta\tau$, 我们可以修改流线的长度和精度。

还要说明的一点是: 在每个 PASS 中都会进行 MSLA-UV 速度场纹理数据集插值。因为数据的时间分辨率为 1 d, 每一天就会有一张数据纹理, 绘制是连续进行的, 所以如果要进行长时间序列的不稳定场可视化, 需要在每帧之间进行数据插值, 来使得流场平滑变化, 获得较好的可视化效果。利用下式对流场纹理进行线性插值:

$$\text{Speed}_{\text{inter}} = \text{Speed}_{\text{first}} \times (1 - \text{timeinter}) + \text{Speed}_{\text{sec}} \times \text{timeinter}, \quad (8)$$

式中, $\text{Speed}_{\text{inter}}$ 、 $\text{Speed}_{\text{first}}$ 、 $\text{Speed}_{\text{sec}}$ 分别为当前帧插值后

的中间纹理、前一天的纹理、后一天的纹理, timeinter 是线性插值参数。最终产生的绘制结果如图 8e 和图 8f 所示, 可以看到可视化效果很好, 粒子分布均匀, 海洋现象如海流、涡旋也能够较好的表达。

4 涡旋可视化

4.1 涡旋可视化算法流程

本文基于二维流线可视化方法实现了 3 种涡旋可视化方法: 基于 OW 参数的涡旋可视化方法、基于栅格模板的涡旋可视化方法和基于矢量模板的涡旋可视化方法, 以下简称这 3 种方法分别为 OW 方法、栅格方法和矢量方法。这 3 种方法的基本思想就是在二维流线可视化的基础上, 通过 3 种数据模板来控制流线的绘制范围, 绘制涡旋范围内的流线, 抛弃涡旋范围之外的流线, 进而实现涡旋的可视化。算法流程如图 9 所示。

4.1.1 基于 OW 算法的涡旋可视化方法

OW 算法是一种基于物理参数的算法, 被广泛应用到中尺度涡的研究中^[31-32], OW 算法参数 W 由该点的剪切变形率 (s_x)、正交变形率 (s_n) 和相对涡度 (ω) 定义, 即

$$W = s_n^2 + s_x^2 - \omega^2. \quad (9)$$

各分量的计算方法为

$$s_n = \frac{\delta U}{\delta x} - \frac{\delta V}{\delta y}, s_x = \frac{\delta V}{\delta x} + \frac{\delta U}{\delta y}, \omega = \frac{\delta V}{\delta x} - \frac{\delta U}{\delta y}. \quad (10)$$

U 、 V 分别代表纬向和经向上的分量, 可以通过地

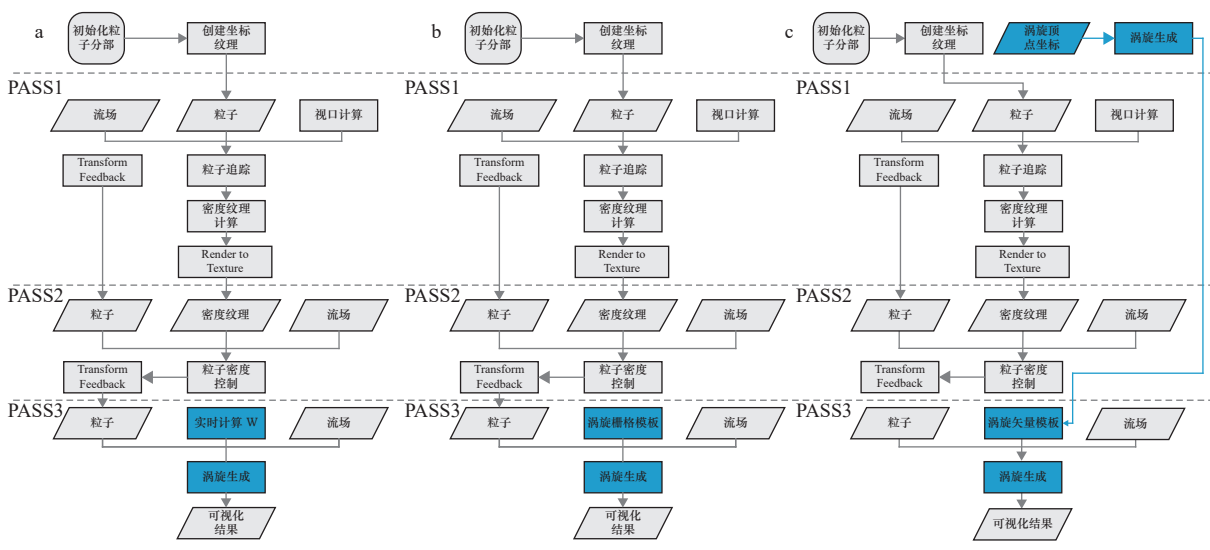


图 9 OW 方法(a)、栅格方法(b)和矢量方法(c)的可视化流程图

Fig. 9 Visual flow charts of OW method (a), grid method (b) and vector method (c)

与流线可视化主要区别在 PASS3 中的蓝色部分

The main difference from streamline visualization is in the blue section of PASS3

转关系由 SLA 梯度计算得到:

$$U = -\frac{g}{f} \frac{\delta(SLA)}{\delta y}, V = \frac{g}{f} \frac{\delta(SLA)}{\delta x}, \quad (11)$$

式中, g 是重力加速度; f 是科氏力参数。

涡旋一般存在于 W 值为负且以旋转为主的流场中^[33]。Chelton 等^[4] 和 Nieto 等^[34] 都曾使用固定的 W 值作为阈值来研究海洋涡旋, 本文基于 OW 的可视化中同样也指定固定的 W 阈值来实现涡旋的可视化效果。

OW 方法的具体流程如图 9a 所示, 前两个 PASS 和二维流线可视化算法的实现细节相同, 不同之处在第 3 个 PASS 中, 在几何着色器中实时计算 W 参数, 并将其作为判定涡旋的依据, 指定 W 阈值, 在几何着色器中绘制条件范围内的流线几何, 抛弃范围之外的流线几何, 进而实现涡旋的可视化。

图 10 显示了着色器中插值与 W 参数计算的过程。图 10a 和图 10b 分别代表当前帧前一天和后一天的不稳定场纹理, 每一个格网代表一个像素, 格网中箭头表示的是速度矢量。根据公式(8)对流场纹理进行插值, 利用公式(9)计算每个像素的 W 值, 并设置一个阈值 W_0 , 我们只保留在 $W < W_0$ 范围内的粒子, 在几何着色器中对这些粒子积分, 绘制流线几何, 就实现了涡旋的可视化。

4.1.2 基于栅格模板的涡旋可视化方法

相似地, 栅格方法也是基于二维流线的可视化方法进行的, 流程图如图 9b 所示, 此方法的输入数据是利用 Faghmous 等^[13] 提出的涡旋识别算法生成的涡旋栅格模板数据, 将 TIFF 格式的模板数据作为输入纹理引入到 PASS3 中, 判定依据不再是 W 阈值而是模

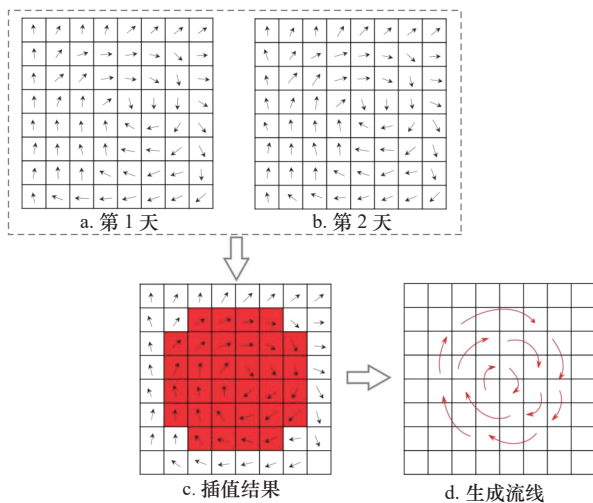


图 10 OW 方法插值过程示意图

Fig. 10 A schematic diagram of the interpolation process in OW method

板纹理值。图 11 显示了插值的过程, 格网表示模板纹理中的像素单元, 蓝色填充的地方表示涡旋存在的区域, 白色填充的地方表示不存在涡旋的区域。图 11a 和图 11b 分别表示第一天与第二天同一位置的两个涡旋, 由于涡旋的运动, 它的形态发生了改变, 插值公式与式(8)相似, 不过公式中的速度值替换成模板纹理值。然后得到如图 11c 中所示的插值结果, 在有效区域内绘制流线, 实现涡旋可视化。

4.1.3 基于矢量模板的涡旋可视化方法

矢量方法中, 首先要对数据进行预处理, 然后读取识别和追踪数据集, 通过索引号在它们之间建立连接。如图 12 所示, 涡旋识别算法将每天的涡旋识别出来并编码相应的日期 (Day) 和涡旋编号 (index), Day 指的是数据日期, 涡旋编号 index 是 Day 天第 index 号涡旋。追踪数据中存储了很多条涡旋轨迹, 每条轨迹都由一系列涡旋节点组成, 节点信息中存储着识别数据中的索引号和日期, 通过每个涡旋节点的信息去识别数据中寻找相应的涡旋数据, 从而获得涡旋的顶点坐标信息。上述这一过程在 CPU 中进行。GPU

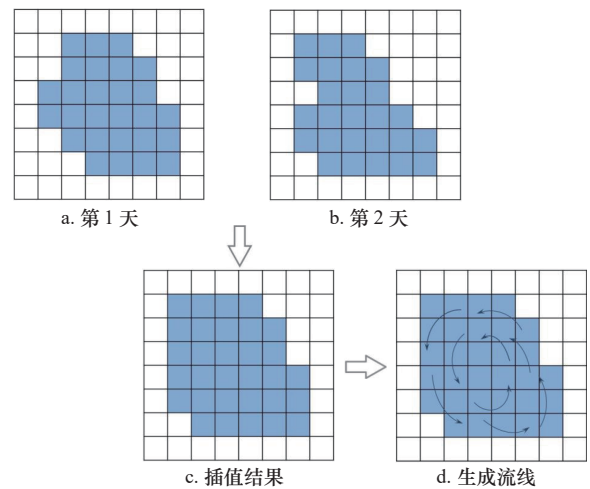


图 11 栅格方法插值过程示意图

Fig. 11 A schematic diagram of the interpolation process in grid method

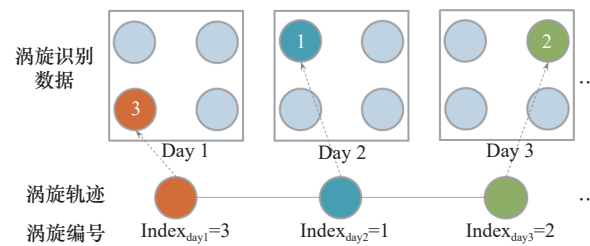


图 12 追踪数据与识别数据的关系示意图

Fig. 12 Schematic diagram of the relationship between tracking data and identifying data

中的处理过程如图 9c 所示,相较于前两种方法,矢量方法需要一个额外的绘制流程来绘制涡旋结构,在此流程中,通过 SSBO(Shader Storage Buffer Object)传入数据,在几何着色器中绘制涡旋几何,然后渲染到纹理,将纹理数据传入 PASS3 中作为判定依据,实现涡旋的可视化。

在绘制涡旋几何的过程中也需要对涡旋的顶点坐标进行插值,如图 13 所示,利用下式获得插值结果,

$$\text{coord}_{\text{inter}} = \text{coord}_{\text{prev}} \times (1 - \text{timeinter}) + \text{coord}_{\text{after}} \times \text{timeinter}, \quad (12)$$

式中, $\text{coord}_{\text{inter}}$ 、 $\text{coord}_{\text{prev}}$ 、 $\text{coord}_{\text{after}}$ 分别为插值后的顶点坐标、前一天数据顶点坐标、后一天数据顶点坐标; timeinter 为线性插值参数。

在实际处理过程中,因为数据量较大,CPU 数据预处理的过程和绘制涡旋几何的过程耗时较大,所以为了减少耗费的时间,对流程进行了修改:将数据预处理和涡旋几何纹理生成的流程预先进行,然后将生成的纹理图像保存下来,之后像栅格方法那样,将保存下来的纹理作为模板直接传入 PASS3,最后在涡旋区域内部绘制流线,实现涡旋可视化。

4.2 传输函数

传输函数是一组定义了数据值及其相关属性与颜色、不透明度等视觉元素之间映射关系的函数^[15]。不透明度决定了显示哪些特征,颜色定义了如何显示这些特征。

一维传输函数的数学表达式一般为: $T: x \rightarrow \{c, \alpha\}$, $x \in \mathbb{R}^n$ 。定义域 x 为自变量,代表数据的属性值;值域为颜色 c 和不透明度 α ,颜色具有 3 个通道,分别为 R、G、B,代表红色、绿色和蓝色。

用户可以通过设定传输函数,突出显示数据的某

种特定属性或者某种属性的某段值域进而突出显示流场特征,并且能够解决显示杂乱的问题,提高可视化的效果。本文分别用速度、涡度和 W 值作为传输函数自变量来进行可交互的可视化。

以流速为自变量的传输函数为例说明其对显示效果控制的基本原理:首先对将要进行涡旋可视化的流速数据进行统计,生成一张流速分布直方图。以这张直方图的流速分布作为指导,选取适合的属性范围来生成一张一维纹理,在 PASS3 的几何着色器中,以计算出的属性值采样这张一维纹理获得颜色值和不透明度,然后将获得的颜色值和不透明度值传入片元着色器中进行染色。

如图 14 所示,通过在界面中设置 Key 值点来调节生成的纹理和颜色。图 14a 中把速度为 0 m/s 的地方映射为淡蓝色,速度大于或等于 3 m/s 的地方映射为红色,中间的颜色值依据 Key 值点来设置,不透明度全部设置为 1,两个 Key 值点之间的颜色和透明度通过算法平滑过渡。图 14b 在图 14a 的基础上调节了 Key 点的不透明度,最终产生的一维纹理供几何着色器中纹理采样使用。

5 涡旋可视化效果及性能对比

5.1 涡旋可视化效果

本文的可视化效果是基于自研渲染引擎实现的,引擎代码使用 C++、Qt 和 OpenGL 实现,利用着色器语言 GLSL 控制显卡进行图形绘制。

图 15 分别展示了 3 种方法的可视化效果,第一行、第二行和第三行分别为使用 OW 方法、栅格方法和矢量方法实现的效果,其中 OW 方法使用的阈值 W_0 为 -2.0×10^{-10} 。第一列是使用 W 参数作为传输函数

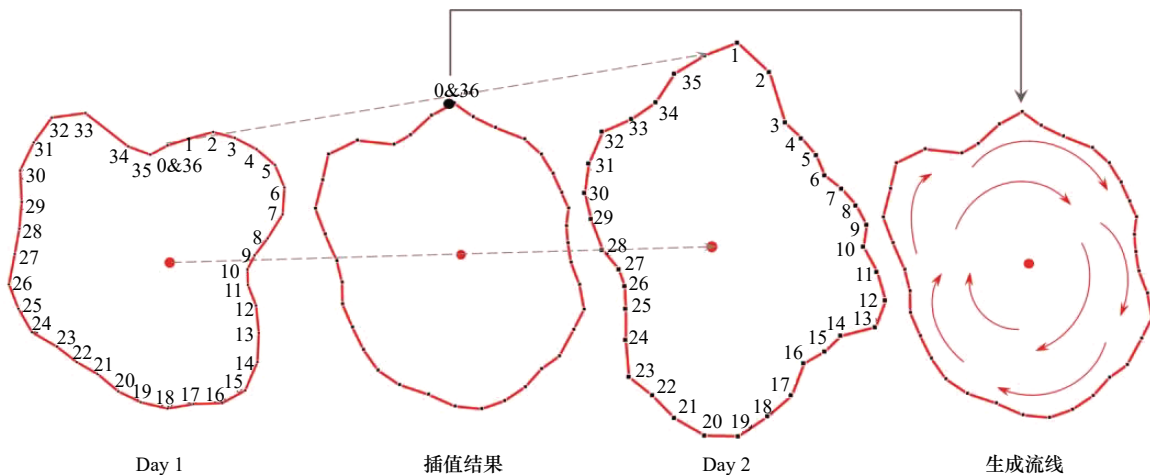


图 13 重构数据的顶点坐标插值示意图

Fig. 13 Interpolation diagram of vertex coordinates of reconstructed data

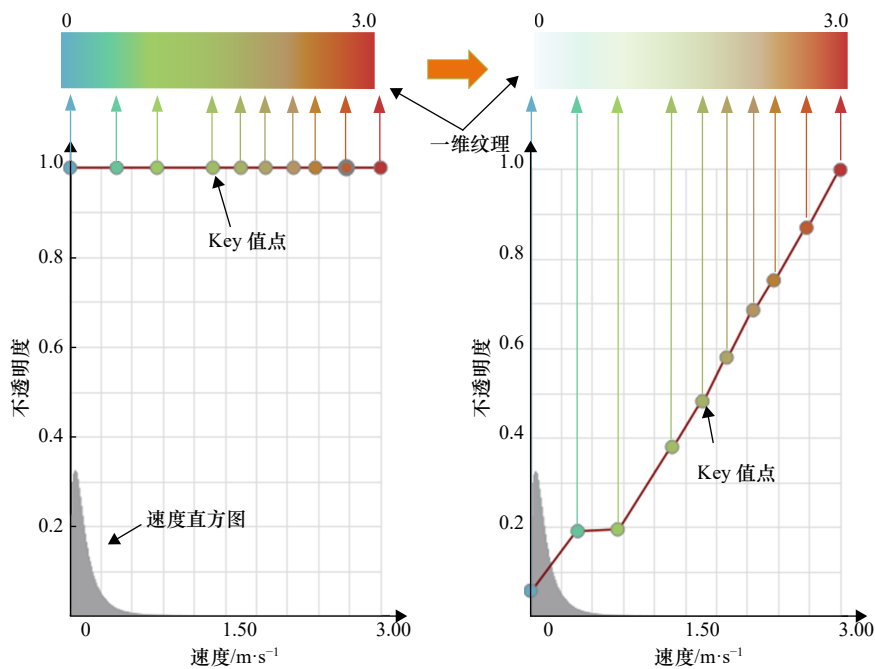


图 14 传输函数交互实现原理示意图

Fig. 14 Transfer function interaction principle diagram

自变量的效果图,第二列是使用涡度作为传输函数自变量的效果图,第三列的传输函数自变量为流速。从算法角度上来讲,OW方法属于基于物理参数的识别方法,其优点在于理论基础明确,并且参数求解简单^[35],其弊端在于严重依赖阈值控制识别效果,并且不同区域的最优阈值可能不同^[34]。同时此方法对噪声比较敏感^[36],并且局限于涡旋核心区^[37],可能对涡旋面积和半径有所低估。另外,此方法有涡旋探测过量的趋势^[11]。从图 15 的 OW 方法效果图可以看出,OW 方法存在一些没有形成涡旋的短线,这可能是数据中存在的噪声,由于其附近像素的 W 值在所设定的阈值之内,所以进行了后向积分,被绘制了出来,但这些短线可能并不属于任何一个涡旋。图 15 中对比其他两种方法,OW 方法得到的涡旋边界更小,局限于涡旋核心区。在图 16 中展示了局部涡旋的放大效果图,可以看到 OW 方法绘制的涡旋并不完整,这可能是由于涡旋内部 W 值分布不均匀和阈值设定不适当导致的。同时其识别出来的涡的数量较其他两种方法多,但是结合流场来看,有些可能不是涡旋,而是识别出的某些洋流转弯处的残片;栅格方法和矢量方法所用的涡旋识别数据是分别用 Faghmous 等^[13]和 Liu 等^[12]的识别算法得到的。这两种算法都基于 SLA 等值线数据,通过从涡心向外迭代等值线来确定涡旋边界,所以这两种方法识别出的涡旋更加饱满,但是其可能导致涡旋的错误分类(涡旋极性与相对涡度的对应符

号不一致)或者过高估计涡旋的大小^[2]。从可视化角度来讲,栅格方法由于使用的数据为栅格模板,数据分辨率为 $0.25^\circ \times 0.25^\circ$,在放大多倍后,数据的像素点相应的被放大,所以涡旋边缘呈锯齿状。而且栅格方法在进行栅格插值的时候只是简单的栅格像素线性插值,没有考虑相邻两张模板中的涡旋与涡旋之间的相关性。同时 Faghmous 等^[13]的算法限制最小涡的大小为 4 个像素,如果前一天的某个涡旋比 4 像素大(小),后一天比 4 像素小(大),这就会导致有些涡旋突然消失(出现)。矢量方法由于我们对其识别出的涡旋进行了重构,所以其显示精度要比栅格模板方法高,同时 Liu 等^[12]的算法将最小涡的大小限制在 8 个像素,所以其绘制的涡旋完整、饱满,相对于前两种方法效果最好,但同时可能漏掉一些小涡。从图 15d、图 15e、图 15g、图 15h 可以看出,通过自变量为 W 值和涡度值的传输函数产生的可视化效果并不是非常令人满意,这主要是因为栅格方法和矢量方法采用的涡旋识别算法主要依据 SLA 等值线形成的闭合几何来确定涡旋边界,与物理参数并不是十分匹配;但是在图 15c、图 15f、图 15i 中,速度的传输函数都能获得很好的显示效果。

图 17 展示了两张基于传输函数的交互效果图,图 17a 中,把传输函数设置成流速在 $0 \sim 0.5$ m/s 时透明度快速增加,0.5 m/s 之后透明度缓慢增加,颜色从橘黄色缓慢过渡到深蓝色。在图 17b 中,淡化了流速

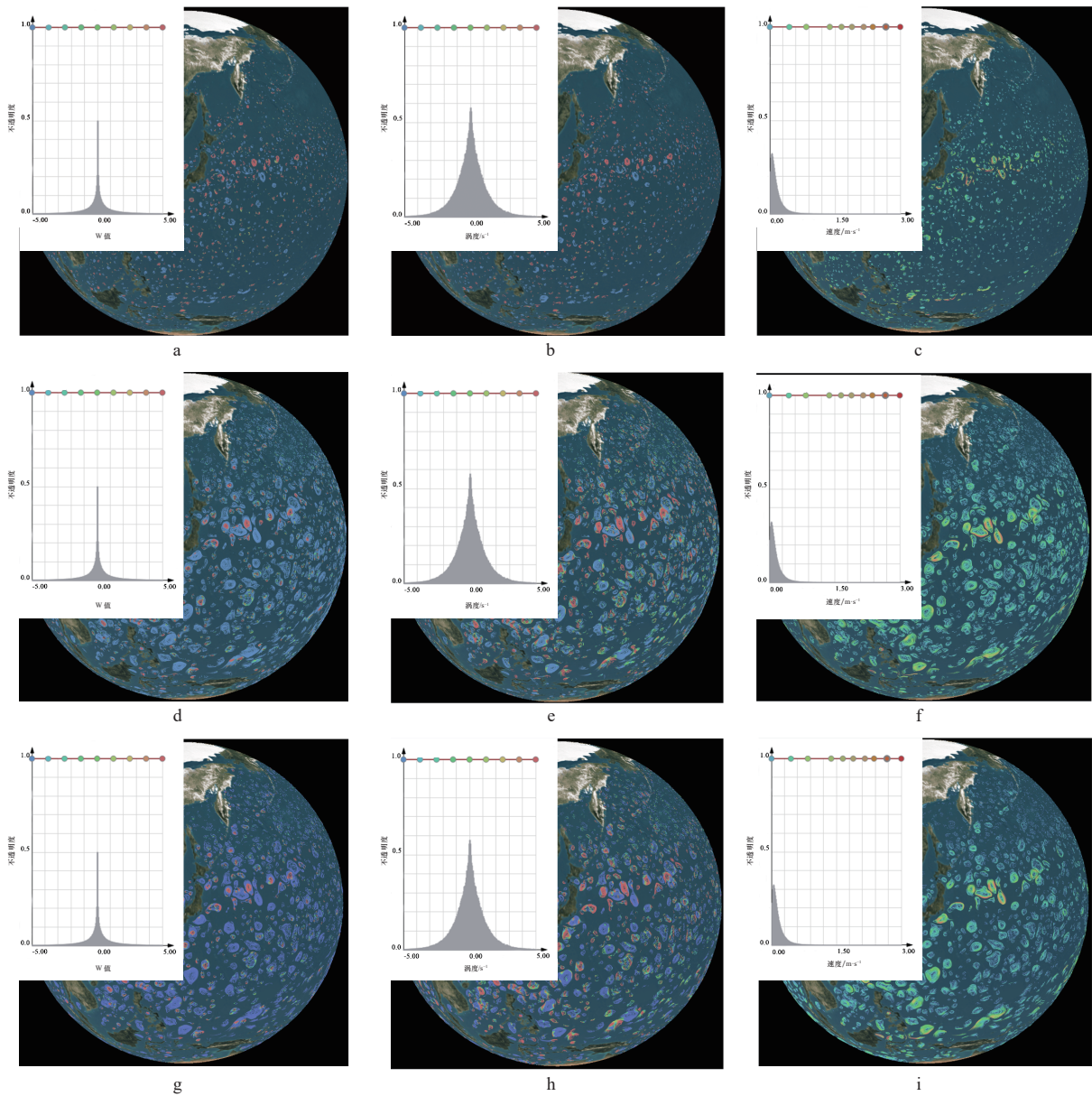


图 15 3 种方法绘制的涡旋可视化结果(2016 年 1 月 1 日)

Fig. 15 Visualization renderings of the three methods (January 1, 2016)

第一、二、三行分别为 OW 方法、栅格方法、矢量方法绘制的结果,第一、二、三列的传输函数分别为 W 值、涡度和速度。传输函数中 W 参数和涡度的横坐标值都被放大了 10^{10} 倍。涡度值的正号表示反气旋涡,负号表示气旋涡; W 参数值大于 0 的部分被移除,将气旋涡的 W 参数的绝对值取代了 W 参数大于 0 的部分

The first, second and third rows are drawn by OW method, grid method and vector method, respectively. The transmission functions of the first, second and third columns are W value, vorticity and velocity, respectively. The x-coordinate values of W parameter and vorticity in the transfer function are magnified 10^{10} times. The positive sign of vorticity value means anti-cyclonic and negative represents cyclonic. The part where the value of the W parameter is greater than zero is removed, and the part where the W parameter is greater than zero is replaced by the absolute value of the W parameter of cyclonic

小于 0.75 m/s 的流线,流速大于 0.75 m/s 的部分透明度快速增加。通过交互可以把传输函数调节成任何形式来表现涡旋内部流场,为涡旋研究提供便利。

5.2 性能测试

性能测试基于以下计算机硬件配置: Windows 10 64 位操作系统, 12 G 内存, Intel Core i5-4430 处理器

(3.00 GHz), NVIDIA GeForce GTX1050Ti 显卡 (4 G 显存), 视口大小为 1 200×600, 概率密度图、栅格模板数据、矢量模板数据的分辨率都为 4 096×2 160, 单位为像素。利用每秒绘制的帧数 (Frames Per Second, FPS) 作为性能测试指标。使用 2016 年 1 月 1 日的 MSLA-UV 数据、栅格模板数据、矢量模板数据作为测试数

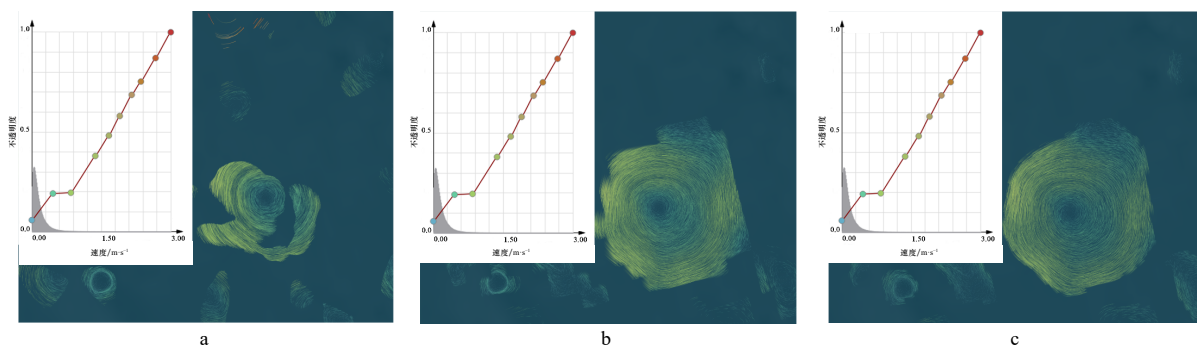


图 16 OW方法(a)、栅格方法(b)、矢量方法(c)绘制的涡旋可视化效果局部放大图
 Fig. 16 Local visualization renderings of OW method (a), grid method (b) and vector method (c)

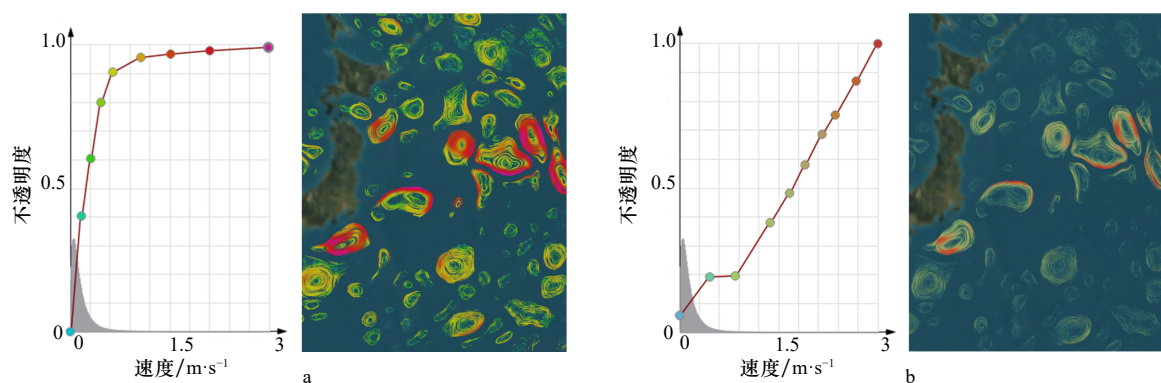


图 17 传输函数交互效果图
 Fig. 17 Transfer function interaction diagram

据, OW方法阈值设定为 -2×10^{-10} 。

图 18 表示当 $\Delta\tau$ 为 1 时, 每条流线上顶点数目 (PointNum) 对性能的影响, 从图中可以看出随着顶点数目的增加, 3 种方法的绘制效率都有较大幅度的降低。其主要原因是由于流线在几何着色器中逐点生成, 随着点数增加, 计算量会变大, 绘制效率就会逐渐降低。

图 19 表示积分步长 ($\Delta\tau$) 对性能的影响, 顶点数目为 0 时, 随着积分步长的增大, 绘制效率也有一定的降低, 但是同顶点数目对性能的影响相比, 积分步长对性能的影响更小。

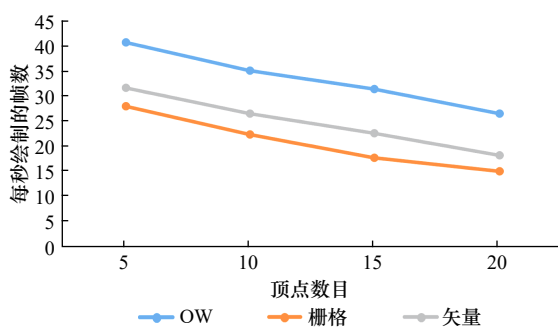


图 18 顶点数目对绘制效率的影响
 Fig. 18 The effect of the number of vertices on rendering efficiency

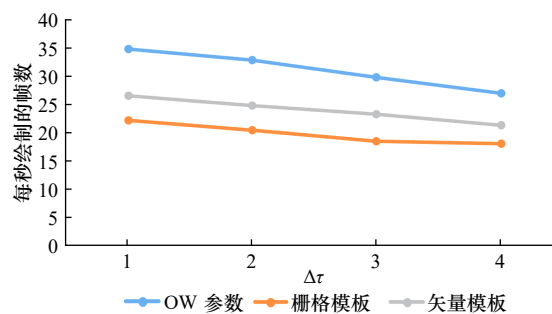


图 19 积分步长对绘制效率的影响
 Fig. 19 The influence of integral step on drawing efficiency

总体来讲, OW方法具有最好的绘制效率, 在 3 种方法平均 FPS 最高。主要的原因可能为设定的阈值使得涡旋范围较小, 需要绘制的涡旋范围内的流线较少, 计算量较小, 所以绘制效率会较高。栅格方法和矢量方法绘制效率差距大概在 4~5 帧每秒, 但是由图 18 和图 19 可见两种方法在顶点数目和 $\Delta\tau$ 不断增加的情况下, 有趋近于相同的绘制效率的趋势。

6 结论

为了解决涡旋连续可视化的时空连续性问题, 本文在前人的基础上提出了 3 种策略实现时空连续可

可视化, 分别采用了3种不同的涡旋识别手段: OW参数、栅格涡旋模板数据和矢量涡旋模板数据来实现涡旋的判定, 并结合二维流场可视化技术实现长周期的中尺度涡旋连续可视化。总体来讲, 基于OW参数的涡旋可视化方法具有最高的绘制效率, 但是其显示结果不尽如人意, 阈值的设置需要经验判定, 并且很多涡旋无法完整显示, 某些局部还存在杂乱的短线扰乱视觉。基于栅格模板的涡旋可视化方法中规中矩, 绘制效率低于OW方法, 但高于矢量方法, 显示效果上相对于OW方法, 涡旋显示更加饱满, 杂乱少, 但是相对于矢量方法显示精度不高, 放大后涡旋边缘会出现锯齿状, 同时由于最小涡旋限定为4个像素, 所以面对小涡时可能出现视觉上的突然出现或消失现象。

基于矢量模板的涡旋可视化方法有最好的绘制效果, 涡旋完整, 由于对涡旋进行了矢量化, 所以边界更加平滑, 其缺点就是绘制前我们需要大量时间来进行数据预处理。通过引入传输函数, 使得用户可以通过与界面交互来控制流场属性信息的显示, 使得研究者可以直观的看到涡旋周围的流场属性信息, 为研究涡旋提供一种新的工具。

当前的工作中还有许多不尽如人意的地方, 比如显示效果和绘制性能不能兼得的问题。我们在未来的工作中将不断提升涡旋的绘制性能和显示效果, 并且当前方法是基于二维数据进行的, 在未来的工作中, 三维数据的涡旋可视化是一个巨大的挑战。

参考文献:

- [1] Williams S, Petersen M, Hecht M, et al. Interface exchange as an indicator for eddy heat transport[J]. *Computer Graphics Forum*, 2012, 31(3pt3): 1125–1134.
- [2] Faghmous J H, Le M, Uluyol M, et al. A parameter-free spatio-temporal pattern mining model to catalog global ocean dynamics[C]//2013 IEEE 13th International Conference on Data Mining. Dallas, TX, USA: IEEE, 2013.
- [3] Chaigneau A, Le Texier M, Eldin G, et al. Vertical structure of mesoscale eddies in the eastern South Pacific Ocean: A composite analysis from altimetry and Argo profiling floats[J]. *Journal of Geophysical Research: Oceans*, 2011, 116(C11): C11025.
- [4] Chelton D B, Gaube P, Schlax M G, et al. The influence of nonlinear mesoscale eddies on near-surface oceanic chlorophyll[J]. *Science*, 2011, 334(6054): 328–332.
- [5] Chen Gengxin, Hou Yijun, Chu Xiaoping. Mesoscale eddies in the South China Sea: Mean properties, spatiotemporal variability, and impact on thermohaline structure[J]. *Journal of Geophysical Research: Oceans*, 2011, 116(C6): C06018.
- [6] Dong Changming, Liu Yu, Lumpkin R, et al. A scheme to identify loops from trajectories of oceanic surface drifters: an application in the Kuroshio extension region[J]. *Journal of Atmospheric and Oceanic Technology*, 2011, 28(9): 1167–1176.
- [7] 董昌明, 蒋星亮, 徐广珺, 等. 海洋涡旋自动探测几何方法、涡旋数据库及其应用[J]. *海洋科学进展*, 2017, 35(4): 439–453.
Dong Changming, Jiang Xingliang, Xu Guangjun, et al. Automated eddy detection using geometric approach, eddy datasets and their application[J]. *Advances in Marine Science*, 2017, 35(4): 439–453.
- [8] Okubo A. Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences[J]. *Deep Sea Research and Oceanographic Abstracts*, 1970, 17(3): 445–454.
- [9] Weiss J. The dynamics of enstrophy transfer in two-dimensional hydrodynamics[J]. *Physica D: Nonlinear Phenomena*, 1991, 48(2/3): 273–294.
- [10] Sadarjoen I A, Post F H. Detection, quantification, and tracking of vortices using streamline geometry[J]. *Computers & Graphics*, 2000, 24(3): 333–341.
- [11] Nencioli F, Dong Changming, Dickey T, et al. A vector geometry-based eddy detection algorithm and its application to a high-resolution numerical model product and high-frequency radar surface velocities in the southern California bight[J]. *Journal of Atmospheric and Oceanic Technology*, 2010, 27(3): 564–579.
- [12] Liu Yingjie, Chen Ge, Sun Miao, et al. A parallel SLA-based algorithm for global Mesoscale eddy identification[J]. *Journal of Atmospheric and Oceanic Technology*, 2016, 33(12): 2743–2754.
- [13] Faghmous J H, Frenger I, Yao Yuanshun, et al. A daily global mesoscale ocean eddy dataset from satellite altimetry[J]. *Scientific Data*, 2015, 2: 150028.
- [14] Jobard B, Lefler W. Creating evenly-spaced streamlines of arbitrary density[M]//Lefler W, Grave M. *Visualization in Scientific Computing'97*. Vienna: Springer, 1997: 43–55.
- [15] Jobard B, Lefler W. Unsteady flow visualization by animating evenly-spaced streamlines[J]. *Computer Graphics Forum*, 2000, 19(3): 31–39.
- [16] Liu Zhanping, Moorhead R, Groner J. An advanced evenly-spaced streamline placement algorithm[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2006, 12(5): 965–972.
- [17] Liu Zhanping, Moorhead II R J. Interactive view-driven evenly spaced streamline placement[C]//Proceedings of SPIE 6809, Visualization and Data Analysis 2008. San Jose, California, United States: SPIE, 2008: 68090A.

- [18] Weiskopf D, Schramm F, Erlebacher G, et al. Particle and texture based spatiotemporal visualization of time-dependent vector fields[C]//VIS 05. IEEE Visualization, 2005. Minneapolis, MN, USA: IEEE, 2005: 639–646.
- [19] Pighin F, Cohen J M, Shah M. Modeling and editing flows using advected radial basis functions[C]//Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Goslar, Germany: Eurographics Association, 2004.
- [20] 何珏, 田丰林, 张昉, 等. 基于实时几何流线生成的二维海流数据交互可视化方法[J]. 海洋技术学报, 2015, 34(3): 91–96.
He Jue, Tian Fenglin, Zhang Fang, et al. Study on the interactive visualization method of 2D ocean current data based on real-time geometric streamline generation[J]. Journal of Ocean Technology, 2015, 34(3): 91–96.
- [21] Tian Fenglin, Cheng Lingqi, Chen Ge. Transfer function-based 2D/3D interactive spatiotemporal visualizations of mesoscale eddies[J]. International Journal of Digital Earth, 2018, 13(5): 546–566.
- [22] Kindlmann G, Durkin J W. Semi-automatic generation of transfer functions for direct volume rendering[C]//IEEE Symposium on Volume Visualization (Cat. No. 989EX300). Research Triangle Park, NC, USA: IEEE, 1998.
- [23] Sereda P, Bartoli A V, Serlie I W O, et al. Visualization of boundaries in volumetric data sets using LH histograms[J]. IEEE Transactions on Visualization and Computer Graphics, 2006, 12(2): 208–218.
- [24] Helgeland A, Andreassen O. Visualization of vector fields using seed LIC and volume rendering[J]. IEEE Transactions on Visualization and Computer Graphics, 2004, 10(6): 673–682.
- [25] Copernicus Programme. Global ocean gridded 14 sea surface heights and derived variables reprocessed (1993-ongoing)[R/OL]. (2016-01-01) [2019-05-30]. http://marine.copernicus.eu/services-portfolio/access-to-products/?option=com_csw&view=details&product_id=SEALEVEL_GLO_PHY_L4_REP_OBSERVATIONS_008_047.
- [26] Copernicus Programme. Global ocean gridded 14 sea surface heights and derived variables reprocessed (copernicus climate service)[R/OL]. (2016-01-01) [2019-05-30]. http://marine.copernicus.eu/services-portfolio/access-to-products/?option=com_csw&view=details&product_id=SEALEVEL_GLO_PHY_CLIMATE_L4_REP_OBSERVATIONS_008_057.
- [27] Sun Miao, Tian Fenglin, Liu Yingjie, et al. An improved automatic algorithm for global eddy tracking using satellite altimeter data[J]. Remote Sensing, 2017, 9(3): 206.
- [28] 陈为, 沈则潜, 陶煜波. 数据可视化[M]. 北京: 电子工业出版社, 2013: 268–270.
Chen Wei, Shen Zeqian, Tao Yubo. Data Visualization[M]. Beijing: Electronic Industry Press, 2013: 268–270.
- [29] Hansen C D, Johnson C R. Visualization Handbook[M]. Cambridge: Academic Press, 2011.
- [30] Ken Perlin. Improving noise[C]//Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'02). New York, NY, USA: Association for Computing Machinery, 2002: 681–682.
- [31] Williams S, Hecht M, Petersen M, et al. Visualization and analysis of eddies in a global ocean simulation[J]. Computer Graphics Forum, 2011, 30(3): 991–1000.
- [32] Chen Changheng, Kamenkovich I, Berloff P. Eddy trains and striations in quasigeostrophic simulations and the ocean[J]. Journal of Physical Oceanography, 2016, 46(9): 2807–2825.
- [33] 杨光. 西北太平洋中尺度涡旋研究[D]. 青岛: 中国科学院海洋研究所, 2013.
Yang Guang. A study on the mesoscale eddies in the northwestern Pacific Ocean[D]. Qingdao: The Institute of Oceanology, Chinese Academy of Science, 2013.
- [34] Nieto K, McClatchie S, Weber E D, et al. Effect of mesoscale eddies and streamers on sardine spawning habitat and recruitment success off southern and central California[J]. Journal of Geophysical Research: Oceans, 2014, 119(9): 6330–6339.
- [35] 崔凤娟. 南海中尺度涡的识别及统计特征分析[D]. 青岛: 中国海洋大学, 2015.
Cui Fengjuan. Mesoscale eddies in the South China Sea: identification and statistical characteristics analysis[D]. Qingdao: Ocean University of China, 2015.
- [36] Souza J M A C, De Boyer Montégut C, Le Traon P Y. Comparison between three implementations of automatic identification algorithms for the quantification and characterization of mesoscale eddies in the South Atlantic Ocean[J]. Ocean Science Discussions, 2011, 8(2): 483–531.
- [37] Shivamoggi B K, Van Heijst G J F, Kamp L P J. The Okubo-Weiss criteria in two-dimensional hydrodynamic and magnetohydrodynamic flows[J]. Physics, 2015, arXiv: 110.6190.

Time-space continuous visualization of mesoscale vortices based on transfer function

Tian Fenglin^{1,2}, Zhu Xinsheng^{1,2}, Liu Wei³, Han Yanjiao⁴, Chen Ge^{1,2}

(1. *College of Information Science and Engineering, Ocean University of China, Qingdao 266100, China*; 2. *Laboratory for Regional Oceanography and Numerical Modeling, Pilot National Laboratory for Marine Science and Technology (Qingdao), Qingdao 266237, China*; 3. *Qingdao Institute of Measurement Technology, Qingdao 266100, China*; 4. *Harbin Institute of Investigation and Surveying, Harbin 150010, China*)

Abstract: In this paper, three methods for continuous visualization of mesoscale eddies are proposed, which are based on the technique of 2D streamline visualization and technique of mesoscale eddies identification: the method of eddy visualization based on OW parameters, the method of eddy visualization based on grid template and the method of eddy visualization method based on vector template. These three methods are respectively based on Okubo-Weiss algorithm, Faghmous algorithm and Liu's algorithm for eddy recognition, and the visualization results of the flow field are filled into the eddy to obtain better visualization effect. In the process of visualization, we introduce the transfer function to conduct real-time interaction between the color and transparency of the streamline in the eddy, which can control the display effect of setting the velocity, vorticity, OW parameters and other information by setting the color and position of the Key point on the control interface. In addition, we also compared the advantages and disadvantages of the three methods in terms of performance and display effect. In terms of performance, the performance is from high to low: the method of eddy visualization based on OW parameters, the method of eddy visualization based on grid template and the method of eddy visualization method based on vector template. In terms of display effect, the method of eddy visualization based on OW parameters is the worst among the three, with more chaotic short lines and smaller eddy boundary, which is limited to the core region of the eddy. The method of eddy visualization based on grid template has better display effect than the first method, with fewer messy short lines and relatively complete eddy. However, due to the lack of high resolution of data, the eddy boundary appears jagged after being put up for more than one time. The method of eddy visualization method based on vector template has the best display effect. The eddy is complete and full. At the same time, since the eddy boundary is reconstructed and vectorized, the eddy boundary is smoother. Compared with the traditional method of continuous visualization of eddies with long time series, these three methods provide a beautiful, dynamic and more informative visualization method. At the same time, they can become a practical tool for researchers to study eddies due to the addition of transfer function.

Key words: eddy visualization; mesoscale eddy; transfer function