



计算机科学

COMPUTER SCIENCE

基于气象数据与多噪声融合的体积云模拟研究

卢春海, 徐新海, 张帅, 李豪

引用本文

卢春海, 徐新海, 张帅, 李豪. 基于气象数据与多噪声融合的体积云模拟研究[J]. 计算机科学, 2023, 50(6): 236-242.

LU Chunhai, XU Xinhai, ZHANG Shuai, LI Hao. [Study on Volume Cloud Simulation Based on Weather Data and Multi-noise Fusion](#) [J]. Computer Science, 2023, 50(6): 236-242.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[结合绘画先验的线稿上色方法](#)

Sketch Colorization Method with Drawing Prior

计算机科学, 2022, 49(4): 195-202. <https://doi.org/10.11896/jsjcx.210300140>

[基于MPI的分布式并行Gazebo仿真优化与测试](#)

Simulation Optimization and Testing Based on Gazebo of MPI Distributed Parallelism

计算机科学, 2021, 48(11A): 672-677. <https://doi.org/10.11896/jsjcx.210100109>

[基于异步机制的Gazebo仿真优化研究](#)

Study on Simulation Optimization of Gazebo Based on Asynchronous Mechanism

计算机科学, 2020, 47(11A): 593-598. <https://doi.org/10.11896/jsjcx.200300131>

[大规模MIMO室外无线光通信系统中基于分段高斯近似的最大似然盲检测算法](#)

Maximum Likelihood Blind Detection Algorithm Based on Piecewise Gaussian Approximation for Massive MIMO Outdoor Wireless Optical Communication Systems

计算机科学, 2020, 47(3): 255-260. <https://doi.org/10.11896/jsjcx.190200310>

[基于投资者情感倾向的P2P市场成交量预测模型](#)

Prediction Model of P2P Trading Volume Based on Investor Sentiment

计算机科学, 2019, 46(6A): 60-65.

基于气象数据与多噪声融合的体积云模拟研究

卢春海 徐新海 张 帅 李 豪

军事科学院 北京 100091

(luchunhai01@163.com)

摘 要 为了在智能无人机集群仿真中构建逼真的仿真环境,需要考虑基于气象数据对云进行建模与渲染。而当前基于真实气象数据的云模拟大多采用物理建模方法,如求解 NS 方程和粒子系统方法,这些方法为繁重的微积分方程求解任务所累,存在因计算量大而无法在大规模场景下实现实时仿真的缺点。针对该问题,提出了一种使用气象数据生成纹理与多噪声融合的体积云建模方法,并将气象数据与高度相关函数相结合来定义云的形状和密度在高度上的变化,有效地将气象数据与非物理建模方法进行结合。渲染时采用光线步进算法从视线方向和朝向太阳两个方向累积云的密度,结合光的吸收和散射定律计算每个样本点的颜色与透明度,最终绘制成云。实验结果表明,模拟出的体积云与气象数据中的云层信息较为一致,效率高,且在形态和颜色上都接近真实的云。

关键词: 体积云;气象数据;高度相关函数;多噪声;光照模型

中图法分类号 TP391.9

Study on Volume Cloud Simulation Based on Weather Data and Multi-noise Fusion

LU Chunhai, XU Xinhai, ZHANG Shuai and LI Hao

The Academy of Military Science, Beijing 100091, China

Abstract In order to build a realistic simulation environment in the smart drone swarm simulation system, it is necessary to consider modeling and rendering clouds based on weather data. At present, cloud simulations based on real weather data generally adopt physical modeling methods, such as solving NS equations and particle system, which are burdened by heavy calculus equation solving tasks, so they have the disadvantages of large computational volume and inability to achieve real-time simulation in large-scale scenarios. Aiming at this problem, a method of modeling volumetric cloud is proposed. Firstly, weather data is used to generate a texture, then combined with the height dependent functions to define changes in the shape and density of clouds in height, and finally the cloud is modeled in combination with multi-noise. So the weather data is effectively combined with non-physical modeling methods. In the rendering, the color and transparency of each sample point are calculated by using raymarching algorithm to accumulate the density of the cloud from the line of sight direction and the sun, in combination with the law of light absorption and scattering, and finally the cloud is drawn. Experiments show that the simulated volumetric clouds are consistent with the cloud information in the weather data, highly efficient, and close to the real cloud in terms of shape and color.

Keywords Volume cloud, Weather data, Height dependent functions, Multi-noise, Light model

1 引言

智能无人机集群是正在发展中的热门技术,为支撑其自主感知、自主行动、集群智能等算法的训练,需要构建逼真的仿真环境。云作为一种常见的大气现象,对包括可见光在内的电磁波具有吸收和散射的作用,可以直接影响无人机的可见光感知能力,同时对雷达和红外装置的探测能力也有一定影响^[1],从而成为仿真中不可或缺的一环。

对云模拟的相关研究,从需求上讲大致可以分为满足视觉需求和具备实际意义两种,前者主要应用在影视游戏中,

用于增加场景的逼真度;后者可应用于气象分析、天气预报及军事仿真等领域,要求仿真结果与真实世界的时空背景相吻合,因此要以气象数据为依据进行云的模拟,这也是本文的研究思路。

目前针对满足视觉需求的云模拟,已经研究出了大量方法并广泛应用于工业界。其中常用的是采用非物理方法进行建模,不考虑云复杂的物理生成机制,而直接借助数学随机噪声等基于过程的方法来构建云的絮状外形,因此又被称为启发式方法。1985年 Gardner^[2]率先将启发式方法用于云模拟,通过把不同类型的纹理映射到椭球体上来生成三维云图像。

到稿日期:2022-05-08 返修日期:2022-10-27

基金项目:国家自然科学基金青年科学基金(62102446)

This work was supported by the Young Scientists Fund of the National Natural Science Foundation of China(62102446).

通信作者:张帅(zhangshuai9999@outlook.com)

Lewi^[3]引入了程序噪声来处理云形状的变化。2001年 Perlin^[4]提出采用3D噪声对云纹理进行干扰以达到渐变的效果,其给出的Perlin噪声此后被广泛使用。2003年Joshua等^[5]直接对Perlin噪声进行插值、衰减等操作,从而干扰生成云的密度场并绘制成云。最新成果是Schneider等^[6]开发的云系统,该系统使用Perlin噪声和Worley噪声的组合来完成云的建模,效果逼近真实云景。

基于气象数据的云模拟,最直观且较为理想的方法是直接从雷达测量数据中获取云完整的三维浓度场,再对场数据恰当地可视化,但局部区域的场数据较少、分辨率低,在应用中过于粗糙。常用方法是基于云的物理生成机制进行建模,通过求解流体动力学和热力学方程来计算云的演化,也被称为数值模拟方法,其中较著名的是Harris等^[7]的研究成果,其效果较好但存在计算量大且效率低的问题。

2000年Dobashi等^[8]使用元胞自动机(CA)对原数值模拟方法做简化,通过物理计算得出每个元胞在每帧发生的转换。Miyazaki等^[9]对上述方法进行了改进,采用耦合映射网格CML模型替代元胞并更严格地遵循云的物理形成规律,核心是求解NS方程。2008年Nes^[10]对天气预报模式(WRF)数据进行了可视化,用气象数据中的大气参数来计算光照在云中的衰减,用体渲染的方法进行绘制,但数据分辨率低和绘制效率低的问题未得到有效解决。

此外还有基于粒子系统的建模方法,从物理过程出发结合数学运算来控制粒子的出现、运动以及消失。2017年Goswami等^[11]在拉格朗日模型中引入光滑粒子流体动力学方法(SPH)来求解热力学和流体动力学方程。Huang^[12]将云当作不可压缩流体,通过对NS方程进行离散化实现了GPU上的计算。到目前为止,基于物理的云模拟工作虽然随着计算机性能的提升不断发展,但依然无法在大规模场景中兼顾实时性与逼真度。

因此本文提出了一种结合气象数据和噪声来共同完成体积云模拟的方法,为了保证实时性,采取了非物理的建模方法,提取WRF气象数据中的云层信息描述云的基本外形,再结合三维Perlin和Worly噪声共同完成云的形状模拟,以解决云外形不够逼真的问题。应用光线步进算法进行采样,样本点的颜色与透明度通过比尔定律与HG相位函数简化计算,最终完成与气象数据中云层信息、形状、颜色、位置均匹配的体积云渲染。

2 气象数据与多噪声的处理

本文采用基于气象数据的2D云图纹理结合3D噪声纹理以及高度相关函数来共同完成云的建模。气象数据经筛选预处理后得到的云图决定了云的基础轮廓,将结果存放在一个2D纹理CloudMap中;高度相关函数和3D噪声用于改变云的形状和密度,使云的边缘更加逼真。

2.1 气象数据的预处理

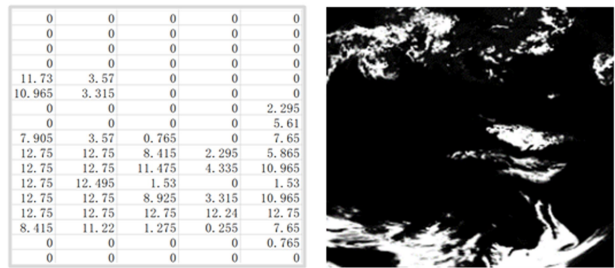
气象数据的来源一般包括气象模型预报数据和卫星观测数据等。预报数据相对更加规则和结构化,便于处理。本文采用美国国家环境预报中心(NCEP)发布的全球预报系统(GFS)数据,数据格式为grib2。由专用工具cfgrib读取指定

区域和时段的相对湿度(rh)数据,从而得到一张二维 rh 数值矩阵,图1(a)为2021年11月17日东亚地区(北纬 $0\sim 55^\circ$ 、东经 $39^\circ\sim 105^\circ$)上空400mb处的相对湿度数据。经样条插值后参考文献[13]对每个格点的数值经由式(1)判断该处云是否生成以及相应的覆盖率,结果存放在2D纹理CloudMap的红色R通道中(见图1(b)),图像分辨率为 6500×5500 。对该纹理数据直接可视化得到了天空中云覆盖的位置(见图1(c)和图1(d)),但其在形态上缺乏云的特征。

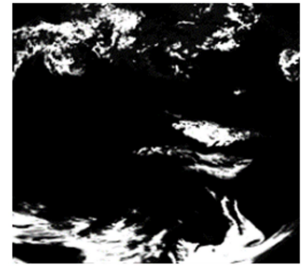
$$CM_c = \left(\frac{rh - rh_{cr}}{1 - rh_{cr}} \right)^2 \quad (1)$$

$$rh_{cr} = \begin{cases} 0.8, & \text{low cloud} \\ 0.65, & \text{medium cloud} \\ 0.8, & \text{high cloud} \end{cases} \quad (2)$$

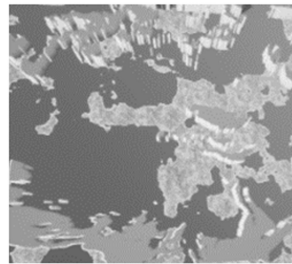
其中, CM_c 表示该格点的云覆盖比(即云量), rh, rh_{cr} 分别表示该处的相对湿度与成云的相对湿度临界值,不同高度云层的临界值不同。



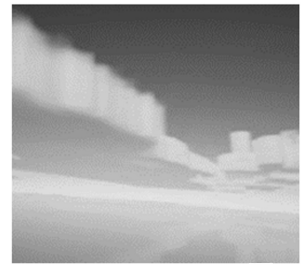
(a) Part of humidity data



(b) Cloud Map



(c) Visibility(planform)



(d) Visibility(side view)

图1 气象数据处理

Fig. 1 Weather data processing

2.2 引入高度相关函数

在引入高度相关函数前,对后文中反复使用的函数做如下声明:

重映射函数 R ,用于将一个值从一个范围重新映射到另一个范围,其中 v 是要重新映射的值, l_o 是原始范围的左端点, h_o 是原始范围的右端点, l_n 是新范围的左端点, h_n 是新范围的右端点。

$$R(v, l_o, h_o, l_n, h_n) = l_n + \frac{(v - l_o) \times (h_n - l_n)}{(h_o - l_o)} \quad (3)$$

标准化函数SAT,用于将一个值限定在 $[0, 1]$ 范围内。

$$SAT(v) = \begin{cases} 0, & \text{if } v < 0 \\ 1, & \text{if } v > 1 \\ v, & \text{otherwise} \end{cases} \quad (4)$$

本文方法中,云的形状实际上是由密度场决定的。由气象数据生成的CloudMap指定了云可以生成的区域范围,

采用 $Density=CM_c$ 的方式对其可视化会产生仅在水平方向密度不同而上下均匀平坦的云,没有体现在高度方向和边缘处的云形态特征,如图 1(d)所示。不同类型的云具有不同的形态特征。自然的云从基础海拔上可以分为 3 个种类,即低云、中云和高云,且各自包含多个亚种,以积云较为常见。本文根据积云在底部较平坦而顶部较圆润,以及随高度上升密度增大^[14]的形态特点,引入了两种不同的高度相关函数,使密度场在高度方向上产生变化,最终使其符合云形态的数据分布。在此基础上,通过 Perlin-Worly 噪声的组合来增加更加丰富的层次细节。

2.2.1 形状改变高度函数

如果不使用形状改变高度函数,云将具有上下平坦的边缘。函数(5)用于使云向底部变圆。其中, $ph \in [0, 1]$ 为当前采样点在云中的高度百分比。当 ph 从 0 变为 0.07 时,使用重映射函数 R 使 GS_b 从 0 变为 1。

$$GS_b = SAT(R(ph, 0, 0.07, 0, 1)) \quad (5)$$

函数(6)用于使云朝顶部变圆。当 ph 从 $CM_c \times 0.2$ 变为 CM_c 时,使用重映射函数(R)使 GS_t 从 1 变为 0。在函数中包含 CM_c 以确保使用 CloudMap 的 R 通道值可以改变云的高度,以反映相对湿度数据对云的影响。

$$GS_t = SAT(R(ph, CM_c \times 0.2, CM_c, 1, 0)) \quad (6)$$

最终的形状改变函数(7)是函数(5)和函数(6)的组合。这种组合能够确保云在底部略微圆润,而在顶部更有倾向性。相关参数选取为反复测试后的效果最优解。

$$GS = GS_b \times GS_t \quad (7)$$

2.2.2 密度改变高度函数

为了使云的底部更蓬松并且云顶具有更明确的形状,使用了与高度相关的密度函数。函数(8)用于降低趋向底部的密度。该函数随 ph 线性增加,并可进一步降低 ph 在区间 $[0, 0.15]$ 中云的密度。

$$GD_b = ph \times SAT(R(ph, 0, 0.15, 0, 1)) \quad (8)$$

使用函数(9)向顶部创建更柔和的过渡。当 ph 从 0.9 变为 1 时, GD_t 从 1 变为 0。

$$GD_t = SAT(R(ph, 0.9, 1.0, 1, 0)) \quad (9)$$

最终的函数(10)是函数(8)和函数(9)的组合,同时,函数(10)还引入了 CCloudMap 的云量信息(CM_c)。因此其能够确保云的底部看起来更蓬松,顶部具有更明确的形状,并且将 CM_c 乘以 2,使得当 $CM_c > 0.5$ 时会创建更高密度的云。效果如图 2 所示。

$$GD = GD_b \times GD_t \times CM_c \times 2 \quad (10)$$



图 2 组合 CloudMap 与高度相关函数创建的云

Fig. 2 Cloud created by combining CloudMap with height dependent functions

2.3 多噪声的处理

气象云图描述了云的基本轮廓,但生成的体积云上下平坦,没有类似云的特征。引入高度相关函数后效果有所改善,但云的边缘依旧平滑,缺乏随机性。为了弥补该缺陷,需要使用 3D 噪声干扰云的密度场,从而“雕刻”出云的细致纹理。常规方法是通过 FBM 组合不同频率的 Perlin 噪声来实现。该方法能创造出效果不错但十分程序化的云,在顶部缺少有动态感的波起,而真实云景常常在顶部呈现类似花椰菜的球状簇集,如图 3 所示,仅用 Perlin 噪声不能解决该问题。



(a) Cloud created by Perlin noise

(b) Photo of cloud

图 3 Perlin 噪声创建的云与真实云比较

Fig. 3 Comparison of cloud created by Perlin noise and real cloud

本文借鉴 Schneider 等^[6]的方法使用 Perlin 和 Worly 两种噪声对云进行更为精细的建模。两种噪声的组合形成两种 3D 纹理,即形状噪声纹理和细节噪声纹理,表 1 列出了两种纹理的细节。

表 1 两种噪声纹理

Table 1 Two noise textures

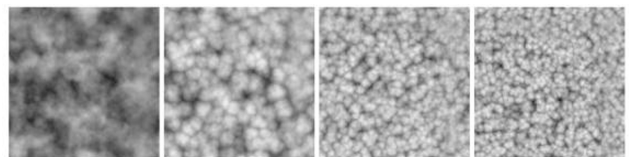
Texture	R	G	B	A
Shape noise	Low frequency	Medium frequency	High frequency	Highest frequency
	Perlin-Worly	Worly	Worly	Worly
Detail noise	Low frequency	Medium frequency	High frequency	
	Worly	Worly	Worly	

2.3.1 形状噪声

噪声有着和正弦波相似的基本形式,由波的叠加可知,不同倍频的噪声可以很容易地制造出随机形态。通过在一定循环内叠加频率逐次升高、振幅逐次下降的噪声,能够得到更加丰富的细节,这种方法被称为分形布朗运动(Fractional Brownian Motion, FBM)^[15]。本文使用简化的 FBM,如式(11)所示:

$$fbm(noise_i) = \sum_{i=1}^n w_i * noise_i \quad (11)$$

图 4 所示的 3D 噪声可视化了云形状噪声纹理的一个切片。该图像具有 $128 \times 128 \times 128$ 像素的分辨率,并在 4 个颜色通道(RGBA)中包含不同的噪声。



(a) 低频 Perlin-Worly 噪声

(b) 中频 Worly 噪声

(c) 高频 Worly 噪声

(d) 最高频 Worly 噪声

图 4 形状噪声纹理的组成

Fig. 4 Composition of shape noise texture

绿色、蓝色和 alpha 颜色通道中的噪声使用 FBM 进行组合,用于为红色通道中的主要形状噪声添加一些细节。其公式如下:

$$NS_{\text{tex}} = R(\text{snr}, \text{fbm}(\text{sng}, \text{snb}, \text{sna}) - 1, 1, 0, 1) \quad (12)$$

其中 snr , sng , snb 和 sna 分别是来自不同通道的形状噪声。 fbm 函数参数分别为 0.625, 0.25, 0.125。使用重映射函数而不是将所有颜色通道组合到 FBM 中,可确保云中心密度不会过于集中。

组合云量信息 (CM_c)、高度相关函数(函数(7)和函数(10))以及形状噪声纹理(函数(12))得到函数(13),可视化得到的云效果如图 5 所示。在重新映射之前执行 $NS_{\text{tex}} \times GS$ 的乘法以改变云的形状,并在重新映射之后执行与 DA 的乘法以改变密度。

$$NS = \text{SAT}(R(NS_{\text{tex}} \times GS, 1 - CM_c, 1, 0, 1)) \times GD \quad (13)$$



图 5 组合 CloudMap、高度相关函数以及形状噪声创建的云
Fig. 5 Cloud created by combining CloudMap, height dependent functions and shape noise

2.3.2 细节噪声

细节噪声用于为云的边缘增加更加紧密堆积的波涛形状。图 6 所示的 3D 噪声可视化了细节噪声纹理的一个切片。该图像具有 $32 \times 32 \times 32$ 像素的分辨率,并在 4 个颜色通道(RGBA)中包含不同的噪声。



(a) 低频 Worly 噪声 (b) 中频 Worly 噪声 (c) 高频 Worly 噪声

图 6 细节噪声纹理的组成

Fig. 6 Composition of detail noise texture

RGB 三通道的细节噪声使用 FBM 进行组合,并通过函数(15)进行修饰。随着 CM_c 值的减小,细节噪声的影响达到最大值 0.35,线性插值确保云向底部更蓬松,向顶部更波涛汹涌。

$$ND_{\text{fbm}} = \text{fbm}(\text{dnr}, \text{dng}, \text{dnb}) \quad (14)$$

$$ND_{\text{mod}} = 0.35 \times e^{-CM_c \times 0.75} \times Li(ND_{\text{fbm}}, 1 - ND_{\text{fbm}}, \text{SAT}(ph \times 5)) \quad (15)$$

其中, fbm 函数参数分别为 0.625, 0.25, 0.125。线性插值函数 Li 如式(16)所示,表示在 v_0 和 v_1 之间线性插值 $ival$ 。当 $ival=0$ 时,函数将返回 v_0 ; 当 $ival=1$ 时,函数将返回 v_1 。中间的值被插值,超出范围 $[0, 1]$ 的值被外插。

$$Li(v_0, v_1, ival) = (1 - ival) \times v_0 + ival \times v_1 \quad (16)$$

为了将形状噪声与细节噪声一起使用,用函数(17)替换

函数(13),与 GD 的乘法将在最后执行。

$$NS_{\text{final}} = \text{SAT}(R(NS_{\text{tex}} \times GS, 1 - CM_c, 1, 0, 1)) \quad (17)$$

最终的云密度 d 使用来自函数(14)的归一化细节噪声以及形状噪声函数(17)来计算,组合为函数(18)。函数 NS_{final} 使用修改后的细节噪声 ND_{mod} 重新映射。与函数(13)相同,在重新映射后执行与 GD 的乘法以改变密度。效果如图 7 所示。

$$d = \text{SAT}(R(NS_{\text{final}}, ND_{\text{mod}}, 1, 0, 1)) \times GD \quad (18)$$



图 7 组合 CloudMap、高度相关函数、形状噪声及细节噪声创建的云
Fig. 7 Cloud created by combining CloudMap, height dependent functions, shape noise and detail noise

3 光照模型的选择与求解

当光线照射在不同材质的物体上时,可能发生吸收、散射及反射。云的主要成分是水,且绝大部分以小液滴的形式存在,能使各个波长的可见光发生强烈的散射,并随着云厚度的增加产生一定程度的吸收。经过散射之后的可见光拥有多重波长,从而使云呈现出最常见的颜色:白色。

3.1 计算光的吸收

比尔定律用于计算与介质属性相关的光衰减。可以用如下公式表示:

$$I_{\text{out}} = I_{\text{in}} \cdot e^{-\tau(d)} \quad (19)$$

其中, e 是自然数, I_{in} 是入射光, I_{out} 是出射光, d 为光线方向的深度值。 $\tau(d)$ 为光程,可以由衰减系数 β 沿光的传播路径积分求出,它们的关系可表示为:

$$\tau(S) = \int_0^S \beta \rho(s) ds \quad (20)$$

将衰减系数视为常数 b ,用比尔定律计算光吸收的部分 $e^{-\tau(d)}$ 即可简化为函数(21),其中 ds 是朝向太阳的累积密度。

$$E(b, ds) = e^{-b \times ds} \quad (21)$$

但在样本点向太阳步进的步长较大时,使用比尔定律可能产生另一个问题:明暗区域对比过于明显。这个问题可以通过函数(22)来钳制最大衰减,其中 $ac \in [0, 1]$ 是用于钳制衰减的值。

$$Ec = \max(E(b, ds), E(b, ac)) \quad (22)$$

3.2 计算光的散射

云中的水滴比空气中的分子大得多,因此云中的散射主要是 Mie 散射,且集中在光线的前进方向。由于 Mie 散射理论复杂,相位函数难以计算。忽略后向散射,使用 Henyey-Greenstein 相位函数可以有效地模拟 Mie 散射的前向散射的特性,既能有效降低计算难度,也能很好地模拟一些视觉相关的特性,如银边效果。

为兼顾实时性与逼真性,本文选择使用 HG 相位函数计算散射,如式(23)所示。其中 g 对应着不同的内部散射角度, θ 是太阳光方向和相机光线方向之间的夹角。

$$HG(\theta) = \frac{1}{4\pi} \frac{1-g^2}{[1+g^2-2g\cos(\theta)]^{3/2}} \quad (23)$$

3.3 光照模型的简化

光线在云中的传输方程是已知的,但考虑到多种光源和多次各项异性散射的光照计算,如图8所示,光源包括太阳光、天空光、地面反射光,并经由云中粒子的散射共同作用于采样点 p ,计算十分复杂。折中真实感和计算性能,本文仅考虑太阳光与天空环境光,并忽略后向散射,将散射作用近似成集中在光线传输方向的多次散射。

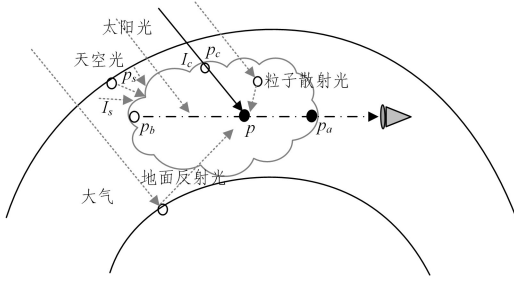


图8 云的光照

Fig. 8 Lighting of clouds

本文分3步实现最终的光照效果。首先考虑云中 p 点处的入射光强 $I_{in}(p, \omega)$ 。

$$I_{in}(p, \omega) = I_c E_c(pp_c) + \int_0^{D_s} I_s(p_s, \omega) E_c(pp_s) ds \quad (24)$$

其中, I_{in} 是云中采样点 p 处在入射光 ω 方向的光强, I_c 为云表面的人射光强, pp_c 表示 p 点到 p_c 点的距离, I_s 为点 p_s 处散射到 ω 方向上的入射光,考虑了任意方向天空光到点 p_s 处的散射光强。方程第一部分描述了太阳直射光沿光线方向到点 p 时未消散的部分,第二部分为天空环境光散射至点 p 的光强。

有了点 p 处的入射光强,再计算由点 p 出射到视点的光强为:

$$I_{out}(p, v) = I_{in}(p, \omega) \beta \rho(p) HG(\theta) E_c(pp_a) \quad (25)$$

其中, I_{out} 为点 p 沿视线 v 方向出射的光强, β 为衰减系数, ρ 为点 p 处的云密度, $HG(\theta)$ 是散射相位函数,用于描述散射的方向特性。

最终沿观察方向对出射光强进行累加计算即可。

$$I_v = \int_{p_a}^{p_b} I_{out}(p, v) dp \quad (26)$$

最终的光照计算是太阳光与天空光在入射方向和视线方向上衰减与散射的组合。下文将通过光线步进算法结合上述光照模型和体积场密度采样对体积云进行渲染。

4 与气象数据匹配的体积云渲染

本文使用光线步进(Raymarching)的方法对体积云进行渲染。

4.1 计算像素点的颜色和透明度

对于云这样的部分透明介质进行Raymarching时,采样点沿观察光线以一定步长递增到场景中,进入介质后会在每一步中累积其密度^[16]。在每个具有非零密度的样本点处,还会启动类似的朝向太阳的光线行进,以累积样本点和太阳之间的密度值,如图9所示。

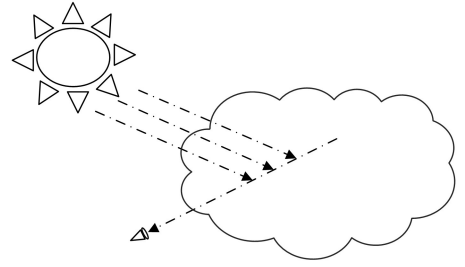


图9 光线行进

Fig. 9 Raymarching

当前渲染的像素点的最终不透明度(alpha)值仅取决于沿观察光线的密度累积,而颜色(RGB)值则取决于太阳光的颜色以及太阳与采样点之间的密度累积。

4.2 区域匹配与步长改进

Raymarching的步长、步数和行进间隔具有如式(27)所示的关系;行进间隔的长度除以步长近似等于步数 n 。

$$n = \text{interval} / \text{stepsize} \quad (27)$$

在云层处于大气中的情况下,行进间隔可以通过求取观察光线与云层(内外层球体)的交点来确定,如图10所示。

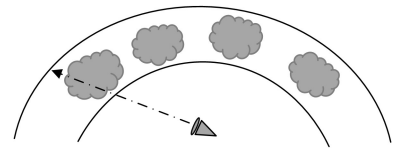


图10 渲染的起止位置

Fig. 10 Starting and ending position of rendering

在得到观察光线与云层的交点位置后,通过判断该点是否位于气象数据的关联区域,即可实现仅在指定区域(北纬 $0 \sim 55^\circ$ 、东经 $39 \sim 105^\circ$ 地区)渲染,从而减少了不必要的工作。类似地,当光线行进中的样本点达到完全不透明度($\alpha = 1$)时,在该路线的剩余路径上已经完全不透光,此时跳出采样,也能达到提升性能的目的。

光线行进时,较短的步长使得图像更精细,较少的步数则有性能优势。但在行进间隔确定的情况下,两者无法兼具。本文通过使步长随行进距离线性增加的方式来确保近处的云非常详细,而对于更远的云,也不需要花费很长的时间来计算。

5 实验结果与分析对比

本文在 Intel(R) Core(TM) i7-12700F CPU 2.9 GHz、16GB RAM、显卡 NVIDIA GeForce RTX3060、操作系统 Ubuntu18.04 的机器上,基于 UnrealEngine4 平台使用 C++ 结合 HLSL(着色器语言)实现上述方法。

实验的具体实现可以归纳为以下步骤:

(1)构建地球云层包围盒,用于计算光线步进的开始位置和结束位置;

(2)将第2节中气象数据与 Perlin-Worly 噪声共同建模得到的云密度场(融合过程在 UE 材质系统中实现)生成体积材质,传入引擎着色器,以便进行下一步的光照计算与体绘制算法;

(3)启动光线步进前,先传入视点位置、光源的光照强度和方向等基本参数,其中,通过改变光源的位置与强度即可

实现场景的昼夜变化;

(4)使用 *RayIntersectSphereCalculation* 函数计算光线与云层包围盒的交点,即光线步进的起始点和终止点,初始化 *StepNum,Stepsize*(式(27))和云材质位置、密度等参数,其中密度参数由云材质直接传入;

(5)执行 *Raymarching* 函数,当采样点进入云层中,根据第3节的光照计算方法累积云的密度,结合步骤(3)中的光照强度和光源颜色计算采样点的透明度和颜色值;

(6)最终屏幕上像素点的颜色由云渲染结果结合大气透射计算得到,效果在下面的实验分析对比中进行展示。

图11为从高空俯视云的形态与由气象数据生成的云图对比。图11(a)是将GFS数据中的相对湿度信息进行可视化得到的局部灰度图(CloudMap),图11(b)是根据CloudMap与噪声融合生成的体积云,可以看出两者形态基本一致,分为上、中、下3个部分,分别呈岛状、点状、弧状。

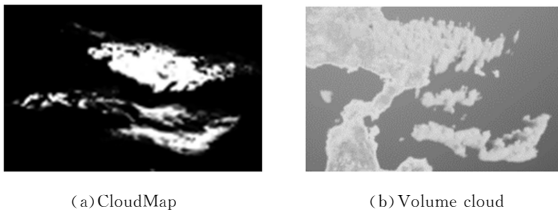


图11 气象数据与体积云
Fig. 11 Weather data and volume cloud

为进行进一步的定量分析,采用Dice相似系数(Dice Similarity Coefficient, DSC)进行体积云模拟拟合程度的客观检验。Dice相似系数是一种集合的相似度度量函数,其简单易计算,在图像处理中常用来计算两幅图像的相似度,取值为[0,1],通常高于0.8时即说明相似性较好。针对图11,提取VolumeCloud图像中的云轮廓,背景置黑,与CloudMap同构为1020*656大小的图像;DSC值通过式(28)计算,CloudMap与VolumeCloud求交表示气象云图和体积云图重合的像素点数。最终DSC值结果为0.827,表明本文方法可以较好地完成模拟云与气象云图的形状匹配。

$$DSC = \frac{2|CloudMap \cap VolumeCloud|}{|CloudMap| + |VolumeCloud|} \quad (28)$$

图12给出了不同时刻模拟的体积云视觉效果。图12(a)为正午,太阳在最高点,云的表现较为通透;图12(b)为傍晚,太阳在侧面,云体呈现为黄昏效果;图12(c)为夜晚,月光呈现了云的轮廓而其他部分较为灰暗。可以看出,本文选用的光照模型可以使体积云取得与现实较为一致的视觉效果。

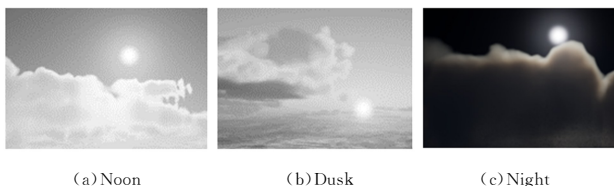


图12 不同时刻的云

Fig. 12 Clouds at different moments

图13给出了本文建模方法与其他方法的对比。图13(a)为文献[17]中基于粒子系统对卫星云图进行三维仿真得到的

云,虽然做到了在形态上与卫星云图较为一致,但缺乏层次细节,云的密度均匀无变化。图13(b)为本文场景中一朵云的实现效果,可以看出,云的形态和密度更加连续、自然。

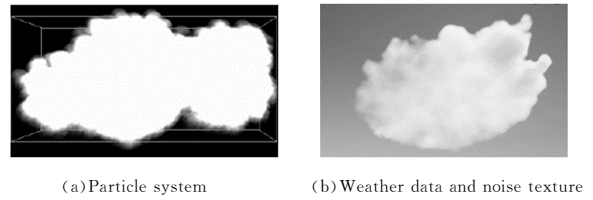


图13 与粒子系统建模比较

Fig. 13 Comparison of the proposed method and particle system modeling method

图14给出了本文方法与近年来的最新方法在视觉效果上的对比结果。图14(a)为Schneider等^[6]发表于SIGGRAPH2015并在《地平线:零之黎明》游戏中使用的体积云,其采用了基于噪声生成的建模方法,具有较强的真实感,但云的覆盖区域是随机无规则的。图14(b)为Tang等^[18]于2019年发表的使用多噪声混合模拟出的体积云,其采用Simplex噪声、Worley噪声以及Curl噪声共同建模,云的底部较为平坦,且缺少对光衰减的钳制,明暗对比明显。图14(c)为本文方法创建的体积云效果,在保留了多噪声纹理建模的优点基础上,引入气象数据来表现云的轮廓和覆盖范围,模拟出的云在形态和颜色上都接近真实的云。

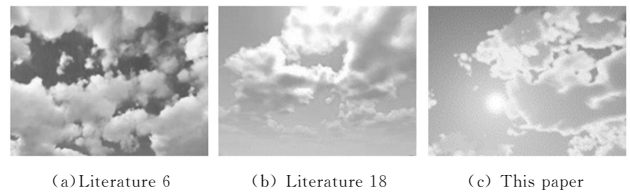


图14 与其他方法视觉效果上的对比

Fig. 14 Visual effects comparison of the proposed method and other methods

表2列出了本文方法与其他方法渲染云的帧率对比结果。传统的方法是采用物理方法如粒子系统对气象数据进行三维云仿真,绘制速率和使用的粒子数息息相关,难以达到实时性;文献[17]通过简化算法大大减少了粒子数,但渲染帧速率也仅在10~55fps之间;采用启发式方法可以有效提高渲染效率,满足实时性要求,Schneider等的方法每帧需要花费约20s来渲染整个场景,渲染速率为50fps;文献[18]也满足了实时性要求但与气象数据不匹配;本文在渲染时结合气象数据和体积云的特点对光线步进算法做了进一步优化,如可变步长和提前跳出,渲染整个场景的帧速率达到了60fps以上,每帧渲染体积云部分的时间开销仅在2ms以内。

表2 与其他方法的帧率对比

Table 2 Frame rate comparison of the proposed method and other methods

Methods	Scene resolution	Simulation scale	fps
文献[17]	1280 * 1024	A single cloud	10~55
文献[6]	1920 * 1080	regional clouds	50~55
文献[18]	1280 * 549	regional clouds	62~81
Our Method	1920 * 1080	regional clouds	60~72

结束语 本文围绕基于气象数据的体积云模拟问题,从气象数据与多噪声融合来建模的角度入手,主要研究了建模与优化、光照模型和有效的体绘制等问题。将气象预报数据中的云量信息与多噪声相结合,避免了物理建模方法计算量大的问题;使用FBM组合Perlin与Worley噪声得到的形状噪声纹理与细节噪声纹理,为云的形态添加细节,保证了云的真实感。绘制时采用前向多次散射光照模型绘制云密度的体数据,同时用HG相位函数简化计算,既满足了实时性,又保证了绘制出的云的颜色接近真实的云。

本文提出的基于气象数据与噪声融合的云模拟技术可以快速地实现气象数据中云层信息的可视化,能够为智能无人机集群仿真环境提供一种有效的气象条件。但本文只是对单一时间点的云层数据进行了可视化仿真,之后我们将对有时间变化的云层数据进行仿真实验,并实现更丰富的云形态模拟效果。另外,本文云模拟只是实现了与可见光的交互,尚未引入云对雷达及红外电磁波,这些将在今后的工作中进一步研究。

参 考 文 献

- [1] ZHANG X Y. Construction of simulated battlefield test environment based on Unity3D and VR technology[D]. Xi'an: Xidian University, 2020.
- [2] GARDNER G Y. Visual Simulation of Clouds[J]. SIGGRAPH Computer Graphics, 1985, 19(3): 297-304.
- [3] LEWIS J P. Algorithms for Solid Noise Synthesis[C]//Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques. ACM, 1989: 263-270.
- [4] PERLIN K. Noise hardware [C]//Real-Time Shading SIGGRAPH Course Notes, 2001: 1-24.
- [5] JOSHUA S, JOSEPH S, DAVID S E, et al. A real-time cloud modeling, rendering, and animation system[C]//Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Eurographics Association, Switzerland, 2003: 160-166.
- [6] SCHNEIDER A, VOS N. The real-time volumetric cloudscapes of horizon: Zero dawn[EB/OL]. <http://advances.realtimerendering.com/s2015/The%20Real-time%20Volumetric%20Cloudscapes%20of%20Horizon%20-%20Zero%20Dawn%20-%20ARTR.pdf>.
- [7] HARRIS M J, BAXTER W V, SCHEUERMANN T, et al. Simulation of Cloud Dynamics on Graphics Hardware[C]//Proceedings of the ACM SIGGRAPH/Eurographics Conference on Graphics Hardware. Eurographics Association, 2003: 92-101.
- [8] DOBASHI Y, KANEDA K, YAMASHITA H, et al. A Simple, Efficient Method for Realistic Animation of Clouds[C]//Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. ACM Press/Addison-Wesley Publishing Co., 2000: 19-28.
- [9] MIYAZAKI R, YOSHIDA S, DOBASHI Y, et al. A Method for Modeling Clouds Based on Atmospheric Fluid Dynamics[C]//Proceedings of the Ninth Pacific Conference on Computer Graphics and Applications. IEEE, 2001: 363-372.
- [10] NES G. Physically plausible weather visualization[D]. Bergen: University of Bergen, 2008.
- [11] GOSWAMI P, NEYRET F. Real-time Landscape-size Convective Clouds Simulation and Rendering[C]//Workshop on Virtual Reality Interaction and Physical Simulation(VRIPHYS), Lyon. Eurographics-European Association for Computer Graphics, 2017: 1-8.
- [12] HUANG W. Dynamic cloud real-time simulation based on GPU and particle system[D]. Nanjing: Software Engineering Discipline of Southeast University, 2017: 1-51.
- [13] MARRA F, GRAGNANIELLO D, COZZOLINO D, et al. Detection of gan-generated fake images over social networks [C] B2018 IEEE Conference on Multimedia Information Processing and Retrieval(MIPR), 2018: 384-389.
- [14] HAMBLYN R. The Cloud Book[M]. Chicago: David&Charles, 2008.
- [15] MANDELBROT B B, VAN NESS J W. Fractional Brownian motions, fractional noises and applications [J]. SIAM Review, 1968, 10(4): 422-437.
- [16] MUÑOZ A. Higher order ray marching [C]//Computer Graphics Forum, 2014: 167-176.
- [17] BI S B, ZENG X W, PAN Q Y, et al. 3D Simulation and Simplified Algorithm of Satellite Cloud Map Based on Particle System [J]. Journal of System Simulation, 2014, 26(11): 2630-2635.
- [18] TANG Y, LIU Z X, GUO H L, et al. Real-time Simulation of Multi-noise Volumetric Cloud [J]. Small Microcomputer Systems, 2019, 40(11): 2461-2465.



LU Chunhai, born in 1998, master. His main research interests include environmental simulation and graphics rendering.



ZHANG Shuai, born in 1986, Ph.D. His main research interests include environmental simulation and space-time subdivision.

(责任编辑:何杨)