第 58 卷 第 4 期/2021 年 2 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

基于改进光线投射算法的室内烟雾可视化研究

刘颖[1*]，陆后军[2]，苌道方[1]

[1] 上海海事大学物流科学与工程研究院，上海 201306；

[2] 上海海事大学物流工程学院，上海 201306

摘要 针对传统光线投射算法在三维场景中绘制大量烟雾数据时存在计算资源消耗大、绘制速度缓慢等一系列问题，提出一种基于改进光线投射算法的室内烟雾可视化方法。将三维数据场按照统一大小划分成均匀的数据块，求出光线穿越数据块时入射点和出射点的中点位置，利用视点和中点之间的距离比例来调整采样频率，从而获得重采样点的位置。再通过对光线上的重采样点进行分级分组操作，对处于不同级别的采样点采取不同的插值策略，最后使用图像合成算法完成光线上重采样点数据值的映射，得到室内烟雾的渲染效果。实验结果表明，该方法是可行且有效的，与现有的光线投射算法相比，在保证绘制图像真实性和稳定性的前提下，改进过后的光线投射算法极大地减少了渲染过程中重采样和线性插值时的计算量，同时帧率能够稳定保持在 75 frame·s^{-1} 左右，可满足不同室内场景下烟雾的实时绘制要求。

关键词 图像处理；光线投射；室内烟雾；采样频率；线性插值

中图分类号 TP391 **文献标志码** A **doi**：10.3788/LOP202158.0410005

Indoor Smoke Visualization Based on the Improved Ray-Casting Algorithm

Liu Ying[1*], Lu Houjun[2], Chang Daofang[1]

[1] *Logistics Science and Engineering Research Institute, Shanghai Maritime University, Shanghai* 201306, *China*;

[2] *School of Logistics Engineering, Shanghai Maritime University, Shanghai* 201306, *China*

Abstract The traditional ray-casting algorithm demonstrates a range of drawbacks, such as large consumption of computing resources and a low draw speed, when drawing vast smoke data in a three-dimensional (3D) scene. Thus, a visualization method of indoor smoke based on the improved ray-casting algorithm is proposed. First, the 3D data field is divided into uniform blocks of data according to the uniform size, neutral positions of the incident and emission points are calculated when the ray travels through the blocks, and the sampling frequency is adjusted with the help of the distance ratio between the point of sight and midpoint to spot the resampling point. For sampling points at different levels, different interpolation strategies are followed by classifying the resampling points in the rays. Finally, a picture-synthesis algorithm is adopted to complete the mapping of the sampling sight data in each ray, realizing the rendering effect of indoor smoke. Experimental results show that the method is workable and effective. Compared with the existing ray-casting algorithm, the improved one considerably reduces the computing effort of resampling and linear interpolation in the rendering process on the premise of guaranteeing the authenticity and stability of the images. Moreover, the frame rate can stably maintain 75 frame·s^{-1}, which can satisfy the real-time rendering requirements of smoke in different indoor scenes.

Key words image processing; ray-casting; indoor smoke; sampling frequency; linear interpolation

收稿日期：2020-06-29；**修回日期**：2020-07-16；**录用日期**：2020-08-03

基金项目：上海临港地区智能制造产业专项(ZN2018010105)

* **E-mail**：ahthly0830@163.com

OCIS codes 100.6890; 100.3010; 110.6880

1 引 言

光线成像技术[1]通过模拟光线的传播方式，实现了对图像的渲染，现已广泛用于三维重建[2]、视觉光学[3]、三维测量[4]等领域。光线投射算法是三维重建领域中的关键算法。利用光线投射算法，在三维场景中根据真实的烟雾扩散数据模拟室内烟雾效果，有助于研究人员清晰地把控室内火灾环境中烟雾的变化趋势，对火灾预防和火灾疏散等方面具有指导与促进作用。近年来，在保证渲染图形具有较强真实感的前提下，如何提高图形的渲染速率是光线投射算法亟待解决的关键问题。

一些学者对光线投射算法的改进进行了大量研究，例如在重采样阶段改变采样频率的计算方式[5]、对数据场中无效数据的处理方式[6]、基于图形处理器（GPU）的并行计算架构方法等[7-9]。Deakin 等[10]使用一种空白空间跳跃的方法并结合切比雪夫距离图来增大光线穿过空白区域的遍历速度，该方法获得了更高的帧速率，实现了用于头戴式显示设备的低延迟显示。袁显纬等[11]提出了一种基于八叉树自适应体归并的光线跟踪加速结构，尽可能地减少了光线与空白节点的求交次数，缩短了一半的相交测试时间，但由于对三维数据场的要求比较单一（需要包含大量的空体素），该方法应用范围较小。Kwon 等[12]将光线投射算法应用于非笛卡尔坐标系中，通过在将光线行进过程进行校正，降低了采样过程中的计算损耗。王奇等[13]在像素重采样阶段，结合深度学习的光场合成方式与基于几何结构的弥散园渲染方法，渲染出了单幅图像重聚焦的视觉效果。Ro 等[14]首次将光线深度信息引入光线投射算法中，根据用户与物体之间的距离补偿灵敏度来提高图像绘制的准确性。Binyahib 等[15]提出了一种基于并行架构的体绘制算法，它将经典对象顺序和图像顺序进行混合，用来处理一些非结构化数据。Jeong 等[16]针对不平坦的物体提出一种自身渲染方法，在光线投射算法的基础上增加了额外的阴影计算，改善了渲染物体的深度感知能力，较大地提升了渲染真实性，但是占用了较多的计算资源。

已有的研究方法主要是通过减少算法总的运算量来提高渲染效率，没有考虑算法计算量大量减少对渲染图像真实感的影响。为实现室内烟雾真实高效的模拟，本文对光线投射算法进行改进，并将改进算法应用于不同三维场景中，通过与现有算法进行比较，在确保生成烟雾真实性的前提下，验证了本文改进算法在渲染效率方面提升的有效性和可行性。

2 基于改进光线投射算法的室内烟雾渲染

传统的光线投射算法沿着每条光线以预先设定好的等步长采样频率选择多个重采样点，然后对每个采样点进行线性插值操作，获得该采样点的颜色值和不透明度值，再进行数据的混合映射，最后得到图像平面上像素点的颜色，完成图像的合成，如图 1 所示。随着三维数据场的数据密集度日益增大，该方法无法满足海量烟雾数据的实时渲染，大量的冗余采样点参与了线性插值过程，使得算法的计算速度和计算效率受到很大影响，导致室内烟雾的渲染效果并不理想。

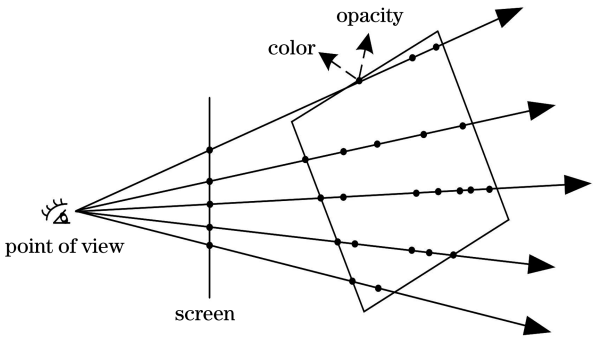

图 1 光线投射算法原理

Fig. 1 Principle of ray-casting algorithm

为解决这一问题，需对传统光线投射算法进行改进。在重采样阶段，充分考虑观察者在场景中的视点变化，定义视锥内数据的重要程度，使其与该数据距离视点的远近相关，即距离越近，数据的重要程度越高，然后再对不同重要程度的数据使用不同的采样频率；在线性插值阶段，对获取得到的重采样点位置进行分级分组操作，对处于不同级别的采样点采取不同的分组策略，之后插值计算每组首个采样点的数据值，并将其赋值给组内其他采样点，最后使用图像合成算法获得屏幕每个像素点的颜色值，完成室内烟雾效果的渲染。改进光线投射算法的伪代码如下：

Input：3D scene real smoke data

Output：Screen pixel color value

Begin

1：　Read real smoke data of 3D scene

2：　Vertex assigned opacity value O and color value C

3：　Divide uniform data blocks according to the uniform size

4：　Viewing point projecting light into 3D data field

5：　Calculate coordinate N that the ray is emitted from the data field

6：　**for** $i=1, i\leqslant n, i++$

(i is the current data block, and n is all data blocks that the current light passes through)

6.1：　Calculateentry point $A[i]$ and exit point $B[i]$

$S=1/2(A[i]+B[i])$

$k=\max[f'(S)]$

6.2：　Calculate distance $|MS|$ between viewpoints M and S and distance $|MN|$ between viewpoint and exit point

6.3：　Sampling frequency $F=k\times|MN|/|MS|$

7：　Get resampling point coordinate $S[m]$

8：　**for** $j=1, j<m, j++$

(j is the current sampling point, and m is all sampling points on the ray)

8.1：　**if** $0<S[j]<1/(3|MN|)$

Interpolate $S[j]$ with minimum distance of 8 vertices

return Color value C and opacity value O after interpolation

8.2：　**if** $1/(3|MN|)<S[j]<2/(3|MN|)$

Every 2 sampling points are divided into a group

Interpolate $S[j]$ with minimum distance of 8 vertices

Assign to the $S[j+1]$ in group

return Color value C and opacity value O after interpolation

8.3：　**if** $2/(3|MN|)<S[j]<|MN|$

Every 3 sampling points are divided into a group

Interpolate $S[j]$ with minimum distance of 8 vertices

Assign to $S[j+1]$ and $S[j+2]$ in group

return Color value C and opacity value O after interpolation

9：　Calculate mixed color value

10：　Map to screen pixels

11：　Get rendered image

End

2.1　自适应采样频率计算

传统的光线投射算法使用的是等步长采样方法，即光线通过图像平面上的像素点穿过三维数据场时，按照预先设定的等步长采样频率进行均匀采样，其中包含了大量的无用采样点，增加了算法的计算量，极大地降低了渲染速度。针对该问题，本研究在重采样计算阶段，将三维数据场按照统一大小划分成均匀的数据块，根据数据块在观察者视锥范围内的距离远近对采样点的采样频率进行约束，使采样频率可以随时变化，在不降低绘制质量的前提下，提高了算法绘制效率。图 2 为传统光线投射算法与本文改进算法在采样点个数上的比较示意图。

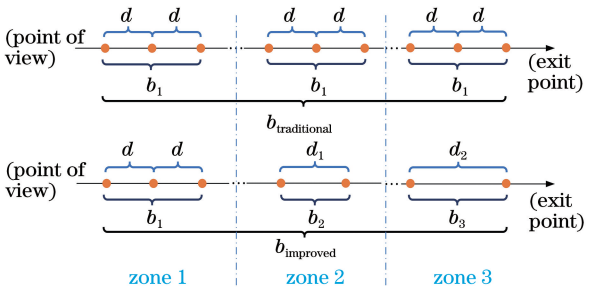

图 2　改进算法与传统算法采样方法比较

Fig. 2　Comparison of sampling method of improved algorithm and traditional algorithm

将一条光线上的所有采样点分为 3 个数据块，

传统算法采用均匀采样方法，各个采样点之间距离均为 d，且每个分区的采样点个数均为 b_1，因此传统算法的采样点总个数为 $b_{\text{traditional}}=b_1+b_1+b_1$；由于同一光线上离视点越远的重采样点对渲染图像的贡献越小，本文改进算法通过改变采样频率的大小可以降低采样点的个数，如图 2 所示，随着光线的射入，各个采样点之间的距离越大，即 $d\ll d_1<d_2$，且每个分区的采样点的个数逐渐减少，即 $b_1\gg b_2>b_3$，改进算法采样点总个数为 $b_{\text{improved}}=b_1+b_2+b_3$，故在保证视锥范围内的渲染画面并没有得到损失的前提下，采样点的个数得到了有效地减少，达到了提高计算效率的目的。

本文改进算法在重采样阶段的示意图如图 3 所示，光线从坐标点 M（视点）出发射向三维数据场，分别穿越了数据块 I 和数据块 II，设光线射入时的坐标点分别为 A_{I} 和 A_{II}，射出时的坐标点分别为 B_{I} 和 B_{II}，光线射出整个三维数据场时的坐标点为 N。A_{I} 分别计算出光线在数据块 I 和数据块 II 上的中点 S_{I} 和 S_{II}，则三维数据场在射线上的中点位置分别存在偏导数。用 $k_i(i=1,2)$ 表示两个中点关于 x,y,z 的偏导数绝对值的最大值，则

$$k_i=\max\left(\left|f'_x(x_i,y_i,z_i)\right|,\left|f'_y(x_i,y_i,z_i)\right|,\left|f'_z(x_i,y_i,z_i)\right|\right)\leqslant 1,\tag{1}$$

式中：(x_i,y_i,z_i) 为中点坐标。

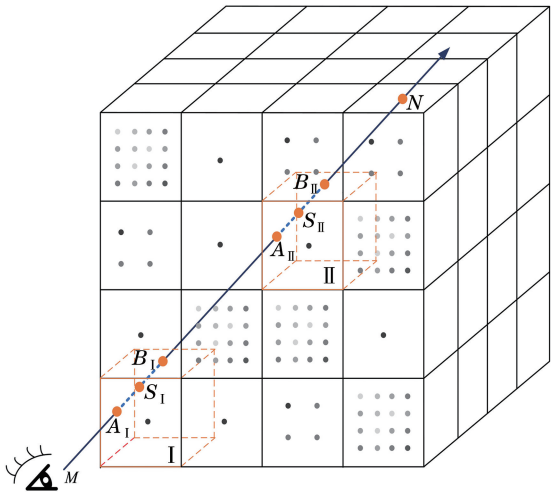

图 3　改进光线投射算法采样频率计算方法

Fig. 3　Sampling frequency calculation method of improved ray-casting algorithm

视点 M 到 S_{I} 和 S_{II} 之间的距离，以及 M 到 N 之间的距离公式分别为

$$|MS_{\text{I}}|=\sqrt{(x_{\text{I}}-x_M)^2+(y_{\text{I}}-y_M)^2+(z_{\text{I}}-z_M)^2},\tag{2}$$

$$|MS_{\text{II}}|=\sqrt{(x_{\text{II}}-x_M)^2+(y_{\text{II}}-y_M)^2+(z_{\text{II}}-z_M)^2},\tag{3}$$

$$|MN|=\sqrt{(x-x_M)^2+(y-y_M)^2+(z-z_M)^2},\tag{4}$$

式中：$(x_{\text{I}},y_{\text{I}},z_{\text{I}})$ 和 $(x_{\text{II}},y_{\text{II}},z_{\text{II}})$ 为数据块 I 和数据块 II 上中任意一点的坐标；(x_M,y_M,z_M) 为 M 点的坐标。则 S_{I} 和 S_{II} 点处数据块的采样频率可分别表示为 $F_{\text{I}},F_{\text{II}}$，即

$$F_{\text{I}}=\frac{k_1|MN|}{|MS_{\text{I}}|}=\frac{|f'|\cdot\sqrt{(x-x_M)^2+(y-y_M)^2+(z-z_M)^2}}{\sqrt{(x_{\text{I}}-x_M)^2+(y_{\text{I}}-y_M)^2+(z_{\text{I}}-z_M)^2}},\tag{5}$$

$$F_{\text{II}}=\frac{k_2|MN|}{|MS_{\text{II}}|}=\frac{|f'|\cdot\sqrt{(x-x_M)^2+(y-y_M)^2+(z-z_M)^2}}{\sqrt{(x_{\text{II}}-x_M)^2+(y_{\text{II}}-y_M)^2+(z_{\text{II}}-z_M)^2}}。\tag{6}$$

根据(5)式和(6)式，数据块与视点之间的距离和光线与三维数据场的切线斜率呈反比例的关系，$k_1>k_2$，又因为 $|MS_{\text{I}}|<|MS_{\text{II}}|$，故数据块 I 中 S_{I} 点的采样频率比数据块 II 中 S_{II} 点的采样频率高，$F_{\text{I}}>F_{\text{II}}$。通过上述重采样方法可以获得三维数据场中各个数据块的采样频率，采样频率会根据视点与数据块的距离变化而改变，离视点近的数据块在视锥范围内对渲染图像的真实感影响较大，采样频率较高，反之采样频率较低。该方法可以有效地减少重采样阶段的计算量，提高图像的渲染效率。

2.2　线性插值重采样

在光线投射过程中，视点发出的光线不能精确地穿过三维数据场中的顶点，需要对重采样点周围的顶点进行线性插值运算，求出该点的值。插值原理如图 4 所示，图中 V_0、V_1、V_2、V_3、V_4、V_5、V_6 和 V_7 表示在三维数据场中与重采样点距离最小的顶点，u_1、u_2、u_3、u_4、w_1、w_2 表示重采样点周围的其他 6 个点（位置见图 4），f 表示重采样点。由于各个顶点的数据值已知，分别对 x、y、z 三个轴方向作插值处理，即

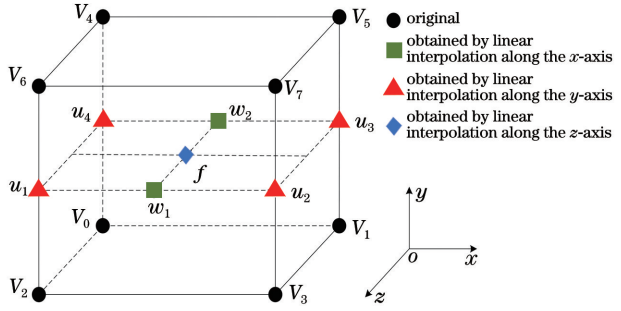

图 4　线性插值计算原理

Fig. 4　Calculation principle of linear interpolation

$$\begin{cases} w_1 = xu_1 + (1-x)u_2 \\ w_2 = xu_4 + (1-x)u_3 \end{cases}, \tag{7}$$

$$\begin{cases} u_1 = yV_2 + (1-y)V_6 \\ u_2 = yV_3 + (1-y)V_7 \\ u_3 = yV_1 + (1-y)V_5 \\ u_4 = yV_0 + (1-y)V_4 \end{cases}, \tag{8}$$

$$f = zw_2 + (1-z)w_1。 \tag{9}$$

经过三次线性插值计算后，重采样点 f 值可以表示为

$$\begin{aligned} V_f = {} & xyzV_0 + (1-x)yzV_1 + xy(1-z)V_2 + \\ & (1-x)y(1-z)V_3 + x(1-y)zV_4 + \\ & (1-x)(1-y)zV_5 + x(1-y)(1-z)V_7 + \\ & (1-x)(1-y)(1-z)V_7。 \end{aligned} \tag{10}$$

根据(10)式，传统光线投射算法的线性插值重采样阶段按照 x、y、z 三个轴被分解为 7 个线性插值操作，计算单个重采样点的值需要进行 19 次加法运算、24 次乘法运算，计算花费总时长为 $t = 19 \times t_{add} + 24 \times t_{mul}$（其中，$t_{add}$ 为一次加法计算时间，t_{mul} 为一次乘法计算时间）。假设该数据场中共有 n 个重采样点，则线性插值阶段花费的总时长为 $t_{total} = t \times n$，重采样点的数量 n 即使在自适应采样频率阶段进行了削减，其数量也是相当庞大的，其计算过程相当复杂，耗费的计算资源较多，这也是造成传统光线投射算法无法实时绘制的主要原因。

为解决上述问题，在插值计算阶段，将一条光线上的所有重采样点按照与视点的距离划分为 3 个级别，对每一级别采用不同线性插值的策略，算法原理如图 5 所示。

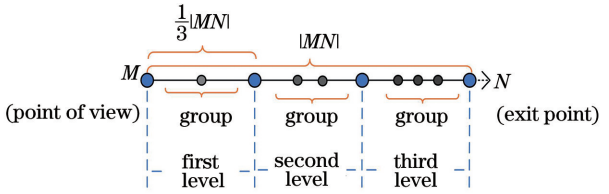

图 5　分级分组线性插值方法

Fig. 5　Linear interpolation method of hierarchical grouping

第一级：离视点最近的一组，当重采样点 S 在区间 $\left(0, \frac{1}{3}|MN|\right]$ 时，对该区间内的每一个采样点进行上述的三线性插值运算；

第二级：距离居中的一组，当重采样点 S 在区间 $\left(\frac{1}{3}|MN|, \frac{2}{3}|MN|\right]$ 时，将每两个重采样点划分为一组，对每组的第一个重采样点进行三线性插值操作，并用运算后的结果迭代同组中的另外一个重采样点；

第三级：距离视线最远的一组，当重采样点 S 在区间 $\left(\frac{2}{3}|MN|, |MN|\right]$ 时，将三个重采样点划分为一组，对每组的第一个采样点进行三线性插值操作，并用运算后的结果迭代同组的剩余两个重采样点。

上述分级分组线性插值方法相较于传统算法，在没有削减重采样点个数、没有降低图像真实性的前提下，根据重采样点在光线上的位置不同，从第一级到第三级逐渐减少每级重采样点的插值计算量，极大地缩短了插值计算总时间，达到了提高算法渲染效率的目的。插值计算总时间的差距如图 6 所示。

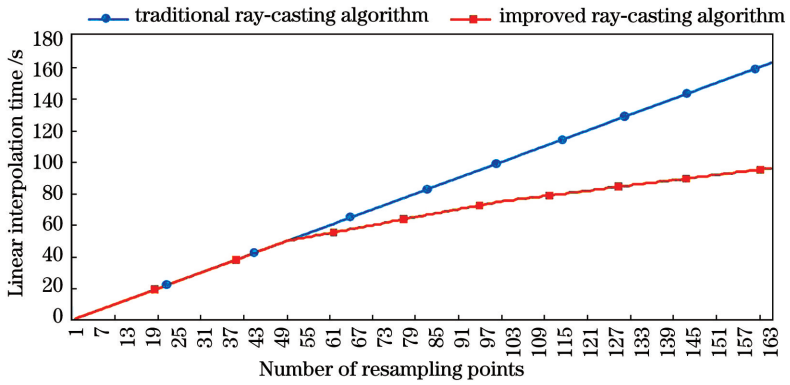

图 6　改进算法与传统算法插值计算时间

Fig. 6　Calculation time of interpolation for improved algorithm and traditional algorithm

2.3 图像合成

图像合成是室内烟雾渲染的最后一个步骤。根据光照原理，光线穿越空间中的三维物体时，会引起光线波长比例的变化，并且这种变化会随着穿越物体个数的增加而累积。烟雾本身是一种具有透光特性的物体，当光线穿过烟雾的三维数据场时，需要将光线入射时的初始颜色值与烟雾数据进行混合计算，最后得到光线射出后的颜色值，这种图像合成方法被称为 alpha 混合技术。该技术按照合成顺序分为两种算法：自后向前的合成算法和自前向后的合成算法。假设光线穿过三维数据场单个体素时的数据参照如表 1 所示。

表 1　图像合成数据参照

Table 1　Data reference of image synthesis

Data type	Current voxel	Before incidence	After incidence
Color	C_{cur}	C_{in}	C_{out}
Opacity	O_{cur}	O_{in}	O_{out}

两种算法由于合成顺序的不同，在进行数据映射时有明显的区别，下面对两种算法分别进行描述和比较。

1）自后向前的合成顺序为

$$C_{out}=C_{in}\cdot(1-O_{cur})+C_{cur}O_{cur}。\tag{11}$$

根据(11)式，该合成算法只需要将光线的颜色值与当前体素的颜色值进行混合即可完成数据的映射；当光线穿越整个三维数据场时，需要对数据场内 n 个体素进行混合计算，设最后一个体素的颜色值为 C_n，不透明度值为 O_n，则最终合成的颜色值 C 为

$$\begin{aligned}C=&C_0(1-O_1)(1-O_2)\cdots(1-O_n)+\\&C_1O_1(1-O_2)(1-O_3)\cdots(1-O_n)+\cdots+\\&C_{n-1}O_{n-1}(1-O_n)+C_nO_n=\\&C_0\prod_{i=0}^{n}(1-O_i)+\sum_{i=1}^{n}C_iO_i\prod_{j=j+1}^{n}(1-O_j)。\end{aligned}\tag{12}$$

2）自前向后的合成顺序为

$$\begin{cases}C_{out}O_{out}=C_{in}O_{in}+C_{cur}O_{cur}(1-O_{in})\\O_{out}=O_{in}+O_{cur}(1-O_{in})\end{cases}。\tag{13}$$

根据(11)式和(13)式，两种合成算法都采用了统一的光照模型(Phong 模型)，但自前向后的合成算法需要对 O 值进行混合计算。由不透明度的特性可知，随着重采样点的不断累积，不透明度值逐渐趋于 1，此时图像平面上的像素几乎不透明，其余的采样点对接下来的图像合成过程没有起到作用，可以忽略不计。综上所述，第二种合成算法在计算复杂度上比第一种更具有优势，为此本研究采用自前向后的图像合成方法，图形合成原理如图 7 所示。

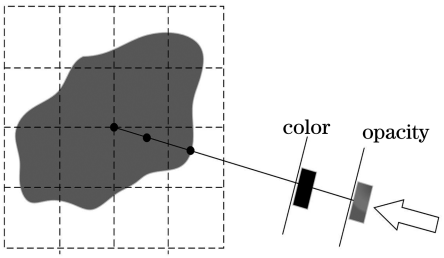

图 7　不透明度及颜色合成(自前向后计算顺序)

Fig. 7　Opacity and color synthesis (calculating order from front to back)

3　实验分析

本文实验采用的硬件环境配置为：Intel i7-8750H，2.21 GHz，8.0 G 内存，显卡为 NVidia GTX1060，操作系统为 Windows 10，编程环境为 Visual Studio 2017 和 Unity 3D，使用 C＃编程和 HLSL 着色语言。

首先测试算法在烟雾渲染方面的可行性。通常，烟雾的密度与不透明度值成正比，即烟雾的密度越大，浓度越高，不透明度值越大。基于此，首先根据流体物理模型计算固定空间内烟雾的三维密度场，进行烟雾模拟实验，用网格顶点中的密度值近似代替每个顶点颜色的透明度，每一个离散网格的颜色由相应点的密度值决定，即用不透明度值的变化来近似模拟烟雾的浓度变化。图 8(a)～(c)分别为固定空间场景下烟雾渲染实验结果，通过对第 100 帧、第 200 帧和第 300 帧的烟雾模拟效果图进行观察，可知初期烟雾由于受到热浮力的作用不断上升，当烟雾到达顶端边界时，逐渐聚集形成烟雾层，同时由于重力的作用，烟雾开始下降，烟雾在运动过程中遇到墙壁时，能够沿着墙壁向下运动，因此本文改进光线投射算法能够真实地模拟出烟雾在固定空间内的运动效果。

为了更加有效地验证模拟出的室内烟雾与复杂环境的交互情况，以上海中心大厦为原型建立了三维场景模型，渲染区域的网格数为 189×9×189，图 9、图 10 分别给出了烟雾在办公室和走廊室内场景中的渲染实验结果，当烟雾开始扩散后，随着烟雾浓度的增加，烟雾沿着墙壁逐渐弥漫整个室内场景，直至完全遮挡场景内座椅等物体。在模拟过程中，帧率稳定，保持在 75 frame·s^{-1}，由此证明实验采用的改进光线投射算法在室内扩散模拟过程中能够满足真实性和稳定性的要求。

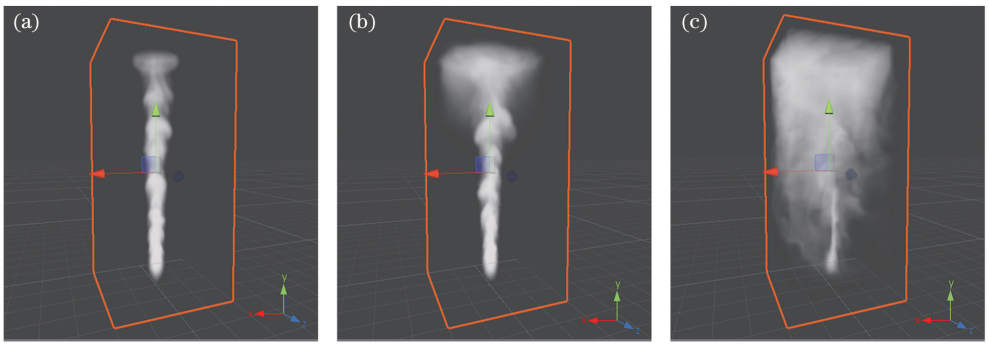

图 8 固定空间场景下烟雾渲染实验结果。(a)第 100 帧模拟效果图;(b)第 200 帧模拟效果图;(c)第 300 帧模拟效果图

Fig. 8 Experimental results of smoke rendering in a fixed space scene. (a) Simulation rendering for 100th frame; (b) simulation rendering for 200th frame; (c) simulation rendering for 300th frame

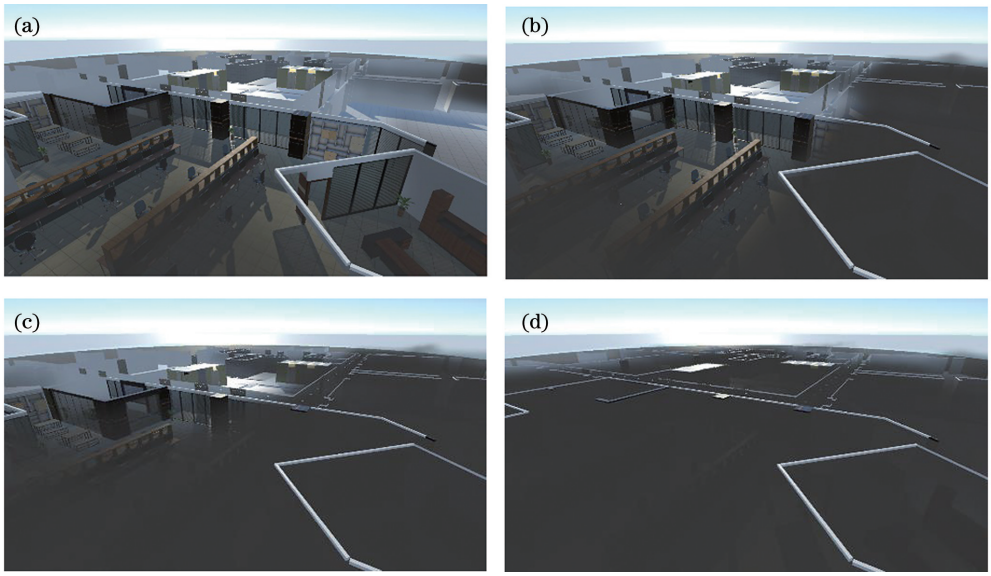

图 9 办公室场景下烟雾渲染实验结果。(a)第 50 帧渲染结果;(b)第 200 帧渲染结果;(c)第 500 帧渲染结果;(d)第 800 帧渲染结果

Fig. 9 Experimental results of smoke rendering in the office. (a) Rendering result for 50th frame; (b) rendering result for 200th frame; (c) rendering result for 500th frame; (d) rendering result for 800th frame

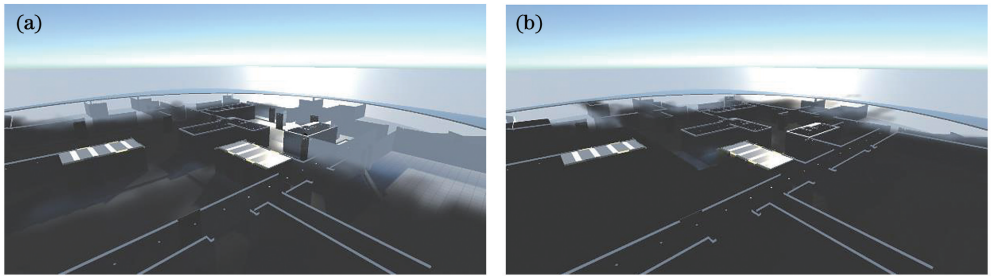

图 10 走廊场景下烟雾渲染实验结果。(a)第 300 帧渲染结果;(b)第 500 帧渲染结果

Fig. 10 Experimental results of smoke rendering in the corridor. (a) Rendering result for 300th frame; (b) rendering result for 500th frame

为了进一步测试改进光线投射算法相较于现有算法在计算效率和渲染真实性方面的提升效果,本研究将改进光线投射算法与传统光线投射算法、文献[17]中的光线投射算法在同一实验环境下进行仿真对比。室内烟雾渲染的场景分为简单场景和复杂场景。在简单场景中,在固定空间内放置若干几何体,在该场景中模拟烟雾的渲染效果,实验结果如图 11 所示,可以看出三种方法均能够实现室内场景

中的烟雾渲染效果，其中：图 11(a)是传统的光线投射算法的渲染实验结果，该方法能够实现遮挡空间物体的效果，但渲染出来的烟雾与场景内墙壁混合效果较差，明暗区分度不够，烟雾流动真实感不足；图 11(b)是采用设置阈值范围确定采样步长的光线投射算法的渲染实验结果，相较于传统光线投射算法，渲染出来的烟雾与空间内的物体融合较为柔和，没有出现墙壁灰白边界的问题，但该算法在重采样阶段没有考虑采样点对成像的重要程度，流动细节不足，渲染出来的烟雾真实感一般；图 11(c)是本文改进光线投射算法的实验结果，烟雾的渲染效果的真实性最佳，明暗区分较明显，烟雾流动等重要细节保存良好，符合人眼对渲染图形真实感的视觉要求。

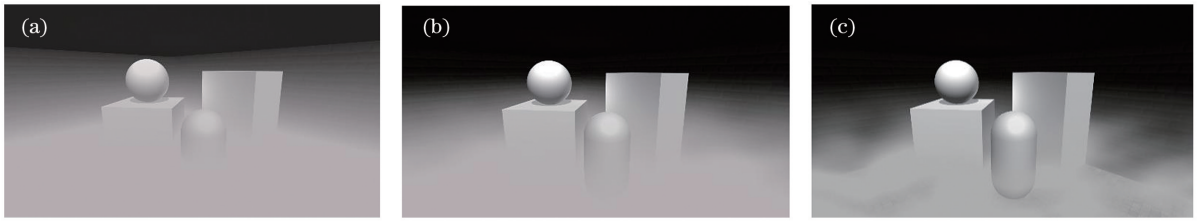

图 11　简单场景下烟雾渲染实验结果。(a)传统光线投射算法；(b)文献[17]光线投射算法；(c)本文改进光线投射算法

Fig. 11　Experimental results of smoke rendering in a simple scene. (a) Traditional ray-casting algorithm; (b) ray-casting algorithm in Ref. [17]; (c) improved ray-casting algorithm

在复杂场景下，以上海中心大厦为模拟场景，首先渲染视点较远时的烟雾效果(俯视场景)。从图 12(a)实验结果中可以看出传统方法渲染的烟雾效果准确性较差，出现了明显的伪影，部分区域的烟雾边界生硬，锯齿感较严重，明暗区分不明显，渲染出来的烟雾真实感较差；图 12(b)是文献[17]算法渲染的实验结果，与传统光线投射算法相比，边界生硬有所改善，没有出现伪影等情况，但明暗效果不佳，没有体现流动的真实感；而采用改进光线投射算法绘制出来的效果如图 12(c)所示，烟雾边界弥漫效果较为明显，明暗效果符合真实烟雾流动的视觉特征，同时，烟雾与场景内物体的交互等细节纹理也被完整地绘制出来，与现实生活中烟雾的流动形态特征相吻合。当渲染视点在室内复杂场景内部时，实验结果如图 13 所示，三种方法渲染效果差距不大，但本文采用的改进光线投射算法渲染出来的烟雾，流动性细节保留程度最佳，画面明暗程度柔和，局部信息表现突出，有较为真实的视觉效果。

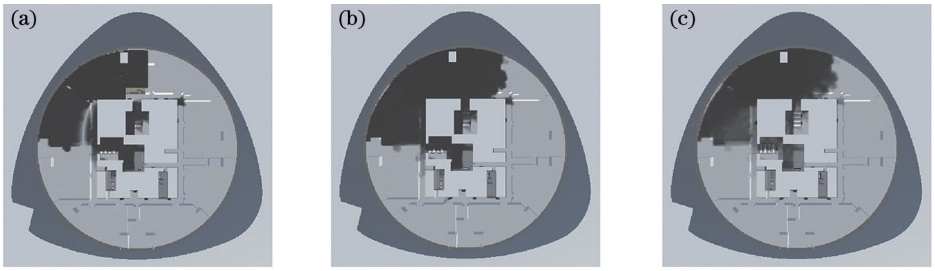

图 12　复杂场景下视点较远时烟雾渲染实验结果。(a)传统光线投射算法；(b)文献[17]光线投射算法；(c)本文改进光线投射算法

Fig. 12　Experimental results of smoke rendering in a complex scene with a farther viewpoint. (a) Traditional ray-casting algorithm; (b) ray-casting algorithm in Ref. [17]; (c) improved ray-casting algorithm

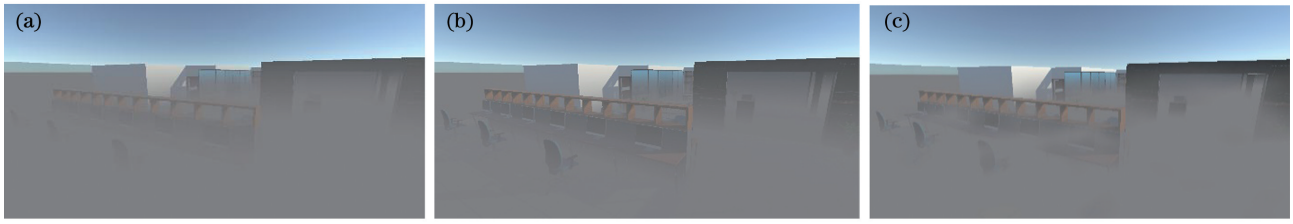

图 13　复杂场景下视点较近时烟雾渲染实验结果。(a)传统光线投射算法；(b)文献[17]算法；(c)本文改进光线投射算法

Fig. 13　Experimental results of smoke rendering in a complex scene with a closer viewpoint. (a) Traditional ray-casting algorithm; (b) ray-casting algorithm in Ref. [17]; (c) improved ray-casting algorithm

在算法计算效率以及渲染稳定性方面，从重采样次数、绘制时间、渲染帧率三个方面进行对比分析。根据表 2、表 3 的实验数据，可知：与传统光线投射算法的等步长采样方法相比，改进光线投影算法的重采样次数减少了约 39%，渲染时间减少了约 30%；与文献[17]算法的加速度步长采样方法相比，改进光线投影算法的重采样次数减少了约 13%，绘制时间减少了约 21%；在渲染帧率方面，如表 3 和图 14 所示，传统光线投射算法在 40 frame·s^{-1} 上下波动，但随着渲染过程的进行，帧率逐渐降至 38 frame·s^{-1} 左右；而文献[17]算法的帧率较传统方法有所提升，但随着实验的进行，帧率有所下降，波动较大；改进光线投射算法在烟雾渲染过程中帧率保持最佳，帧率能够始终稳定在 75 frame·s^{-1} 左右，波动幅度较小。因此，利用改进光线投射算法渲染的烟雾效果的实时性和稳定性都显著优于传统的光线投射算法。

表 2　烟雾渲染重采样次数实验结果

Table 2　Experimental results of smoke rendering resampling times

Method	Sampling method	Number of samples	Rendering time /s
Traditional ray-casting algorithm	Uniform sampling	321489	18.50
Ray-casting algorithm of Ref. [17]	Threshold range	224350	16.32
Improved ray-casting algorithm	Adaptive sampling	196389	12.95

表 3　烟雾渲染帧率实验结果

Table 3　Experimental results of smoke rendering frame rate

Frame number	Rendering frame rate /(frame·s^{-1})		
	Traditional ray casting algorithm	Ray-casting algorithm of Ref. [17]	Improved ray casting algorithm
135	45.32	70.34	73.92
178	44.24	68.50	74.24
220	47.43	67.33	73.35
252	43.44	68.72	74.93
351	37.72	54.32	75.14
384	38.26	56.34	74.92
423	40.53	52.00	75.15
485	38.34	55.46	75.23

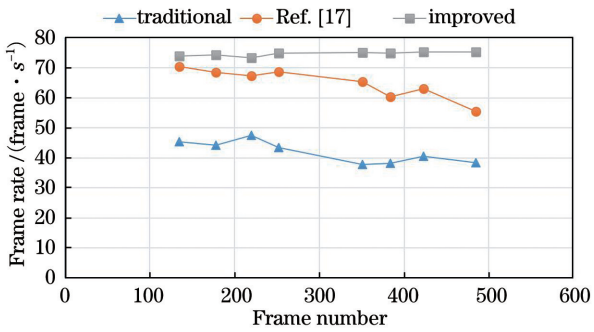

图 14　烟雾渲染帧率实验结果

Fig. 14　Experimental results of smoke rendering frame rate

4　结　　论

为解决传统的光线投射算法中存在的计算速度缓慢、渲染图像真实感不佳等问题，针对室内烟雾提出一种改进的光线投射算法。该方法在重采样计算阶段，将视锥范围内能够影响渲染图像特征的一部分重要数据纳入考虑范围，通过自适应采样频率计算方法获得每条光线上重采样点的位置，在不损失渲染图像质量的前提下，对重要数据进行采样，然后在线性插值过程中，对采样得到的重采样点进行分级分组操作，最终实现烟雾实时渲染。实验结果证明，该算法在简单或复杂三维场景下均是可行、有效的。同时在渲染图像真实感、渲染效率和稳定性方面，均优于传统的光线投射算法。流体的模拟一直是计算图形学领域的热点和难点，测试本文改进算法在渲染不同流体时的有效性是未来可开展的研究工作。

参　考　文　献

[1]　Shao X P, Liu F, Li W, et al. Latest progress in

computational imaging technology and application [J]. Laser & Optoelectronics Progress, 2020, 57(2): 020001.
邵晓鹏, 刘飞, 李伟, 等. 计算成像技术及应用最新进展[J]. 激光与光电子学进展, 2020, 57(2): 020001.

[2] Song X L, Li S, Gu M T, et al. Three-dimensional reconstruction of micro-scale flow field based on light field microscopic imaging[J]. Acta Optica Sinica, 2019, 39(10): 1011002.
宋祥磊, 李舒, 顾梦涛, 等. 光场显微成像微尺度流场三维重建方法研究[J]. 光学学报, 2019, 39(10): 1011002.

[3] Fu L, Hong H B, Wang X, et al. Non-Lambertian photometric stereo vision based on inverse reflectance model[J]. Acta Optica Sinica, 2020, 40(5): 0520001.
付琳, 洪海波, 王晰, 等. 基于逆向反射模型的非朗伯光度立体视觉[J]. 光学学报, 2020, 40(5): 0520001.

[4] Feng W, Tang S J, Zhao X D, et al. Three-dimensional shape measurement method of high-reflective surfaces based on adaptive fringe-pattern [J]. Acta Optica Sinica, 2020, 40(5): 0512003.
冯维, 汤少靖, 赵晓冬, 等. 基于自适应条纹的高反光表面三维面形测量方法[J]. 光学学报, 2020, 40(5): 0512003.

[5] Xie Y H, Ji Y. Ray casting algorithm based on viewpoint correlation for 3D cloud visualization[J]. Semiconductor Optoelectronics, 2019, 40(5): 694-699.
谢永华, 姬瑜. 视点相关光线投射算法在三维云可视化方面的应用[J]. 半导体光电, 2019, 40(5): 694-699.

[6] Meng X X, Liu L, Lang J Y, et al. Region space guided transfer function design for nonlinear neural network augmented image visualization[J]. Advances in Multimedia, 2018, 2018: 1-8.

[7] Ma T C, Li P, Ma T B. A three-dimensional Cartesian mesh generation algorithm based on the GPU parallel ray casting method [J]. Applied Sciences, 2019, 10(1): 58.

[8] Jiang L, Huang Z C, Wang H X, et al. OCT-based 3D visualization of the subcutaneous tissue of the finger [J]. Journal of Zhejiang University of Technology, 2019, 47(5): 560-566.
蒋莉, 黄佐昌, 王海霞, 等. 基于 OCT 的手指皮下组织的三维可视化[J]. 浙江工业大学学报, 2019, 47(5): 560-566.

[9] Wu Y G, Han B B. Rapid simulation and volume rendering of electromagnetic environment based on geographic coordinate system[J]. Journal of System Simulation, 2020, 32(3): 362-370.
武玉国, 韩贝贝. 基于地理坐标系的电磁环境快速仿真与体绘制[J]. 系统仿真学报, 2020, 32(3): 362-370.

[10] Deakin L J, Knackstedt M A. Efficient ray casting of volumetric images using distance maps for empty space skipping [J]. Computational Visual Media, 2020, 6(1): 53-63.

[11] Yuan Y W, Quan J C, Wu C, et al. Ray tracing acceleration structure based on octree adaptive volume merging[J]. Acta Optica Sinica, 2017, 37(1): 0120001.
袁昱纬, 全吉成, 吴晨, 等. 基于八叉树自适应体归并的光线跟踪加速结构[J]. 光学学报, 2017, 37(1): 0120001.

[12] Kwon K, Lee B J, Shin B S. Reliable subsurface scattering for volume rendering in three-dimensional ultrasound imaging[J]. Computers in Biology and Medicine, 2020, 117: 103608.

[13] Wang Q, Fu Y T. Single-image refocusing using light field synthesis and circle of confusion rendering [J]. Acta Optica Sinica, 2020, 40(1): 0111021.
王奇, 傅雨田. 利用光场合成与弥散圆渲染的单幅图像重聚焦[J]. 光学学报, 2020, 40(1): 0111021.

[14] Ro H, Chae S, Kim I, et al. A dynamic depth-variable ray-casting interface for object manipulation inar environments [C] // 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), October 5-8, 2017, Banff, AB, Canada. New York: IEEE Press, 2017: 2873-2878.

[15] Binyahib R, Peterka T, Larsen M, et al. A scalable hybrid scheme for ray-casting of unstructured volume data[J]. IEEE Transactions on Visualization and Computer Graphics, 2019, 25(7): 2349-2361.

[16] Jeong H, Kim H J, Hyeon M G, et al. Shadow extension for ray casting enhances volumetric visualization in real-time 4D-OCT [J]. Optics Communications, 2020, 460: 125237.

[17] Zeng Y Y, Pei Q Q, Li B K. Ray projection algorithm based on adaptive compound interpolation [J]. Journal of System Simulation, 2018, 30(11): 164-171.
曾艳阳, 裴庆庆, 李保锟. 基于自适应复合插值的光线投射算法[J]. 系统仿真学报, 2018, 30(11): 164-171.