

С. И. Вяткин, Б. С. Долговесов

(Новосибирск)

**ВИЗУАЛИЗАЦИЯ ПОЛУПРОЗРАЧНЫХ ОБЪЕКТОВ
НА БАЗЕ ФУНКЦИЙ ВОЗМУЩЕНИЯ И ПРОЗРАЧНОСТИ**

Обсуждается проблема синтеза высокореалистичных изображений. Рассматривается способ задания трехмерных облаков, ограниченных поверхностями свободных форм. Предложен алгоритм быстрого отображения полупрозрачных объектов. Приведен метод моделирования неоднородности при визуализации атмосферных эффектов.

Введение. С появлением систем виртуальной реальности, в которых наблюдатель погружается в мир модели, возникает потребность визуализации виртуальной среды, максимально приближенной к естественным условиям. В настоящее время широкое распространение получили методы визуализации внутренних структур объектов. Эти методы в задачах компьютерной графики называются визуализацией объема. В методах, основанных на представлении объектов поверхностями, сначала создается промежуточная модель на базе плоских треугольников. Далее выполняется визуализация объектов. Другие методы, основанные на воксельном представлении объемов, создают трехмерное изображение объекта непосредственно из объемных данных. Объемно-ориентированная технология визуализации отличается от традиционной растровой полигональной графики кардинальным образом [1]. В полигональной растровой графике объекты задаются поверхностями, представленными полигонами, накладываемыми на проволочный каркас модели. Такое задание объектов достаточно для игровых приложений, анимационных и синтетических объектов, но совсем недостаточно для отображения внутренней структуры естественных объектов или явлений. Например, в симуляторах полета необходимо отображать объемные облака.

При сканировании двумерного пространства нельзя получить полноценное трехмерное изображение. Полигональная трехмерная графика со сканированием полигонов в плоскости изображения не является трехмерной в полном смысле этого слова. Информация, которая предоставляется пользователю в такой технологии, неполная. Отсутствует основная информация о глубине объекта: имеется в виду не отсутствие Z -координаты точки поверхности, а отсутствие информации о луче, проходящем сквозь объект.

Целью данной работы является визуализация объектов, образованных поверхностями свободных форм и заполненных трехмерной текстурой, с быстрым вычислением интегрального значения прозрачности на заданном отрезке. Для визуализации использовался рекурсивный алгоритм растривания объектного пространства с учетом перспективы [2].

Также приводятся результаты исследований по моделированию полупрозрачных функционально заданных объектов с применением вещественных функций возмущения. Функциональное задание объектов является перспективным подходом, который предполагает поверхности объектов и их внутреннюю структуру определять с помощью функций [2].

Визуализация трехмерных облаков. В работах [3, 4] трехмерные облака представлены в виде текстурированных полупрозрачных эллипсоидов. В работе [5] описан метод генерации облаков и тумана, который планировалось реализовать компанией “Evans & Sutherland”. В этом проекте облака заданы в виде слоев эллипсоидов, дана оценка необходимого количества эллипсоидов для более или менее реалистичного изображения. Предлагается модель, в которой атмосфера разбита на слои, в каждом слое есть свой цвет c_i и константа затухания a_i . Приведена формула, по которой вычисляется цвет объекта, наблюдаемого через облака:

$$\begin{aligned} \text{Apparent color} = & (\text{Actual color}) \exp[-0,5(a_0 + a_1)d] + \\ & + 0,5(c_0 + c_1)\{1 - \exp[-0,5(a_0 + a_1)d]\}, \end{aligned} \quad (1)$$

где Actual color – реальный цвет объекта, а d – расстояние между наблюдателем и объектом.

Согласно формуле (1) цвет и константа затухания линейно интерполируются между двумя слоями. Если объект перекрывают несколько слоев, применяется виртуальный цвет объекта: сначала Apparent color вычисляется для двух слоев, потом результат используется как Actual color для двух других слоев и т. д. При этом константа затухания должна увеличиваться по мере увеличения высоты слоя. Это необходимо для того, чтобы уменьшить количество вычислений при определении цвета объекта, находящегося на земле и наблюдаемого через N слоев, при условии, что все слои достаточно прозрачны. Иначе приходится проводить вычисления цвета объекта для каждого слоя. Если верхние слои плотны, то наблюдатель не видит ничего, кроме цвета данного слоя облаков. Аналогичные методики при визуализации атмосферных эффектов использует компания NVIDIA в своих графических картах, применяя следующие способы моделирования тумана.

1. Цвет в пикселе с учетом тумана вычисляется в вершинах треугольников, затем интерполируется для произвольного пикселя по данным значениям в вершинах.

2. Цвет в пикселе с учетом тумана берется из таблицы и не зависит от вершин. Такой способ вычисления называется табличным или пиксельным способом вычисления тумана.

Для моделирования тумана применяется несколько видов функций затухания, реализованных в основном в Direct3D.

Линейная функция затухания вычисляется как

$$F = (\text{fogend} - d) / (\text{fogend} - \text{fogstart}), \quad (2)$$

где fogstart, fogend – точки входа в туман и выхода из него соответственно.

Экспоненциальная функция затухания определяется в виде

$$F = 1 / (d \cdot \text{fogdensity}), \quad (3)$$

где fogdensity – плотность тумана.

Квадратичная экспоненциальная функция затухания находится как

$$F = 1 / (d \cdot \text{fogdensity})^2. \quad (4)$$

Кроме того, компанией NVIDIA предложена принципиально новая методика на базе карт высот, основная идея которой заключается в использовании альфа-канала для хранения некоторых значений (условно высот), применяемых при выборе нужного текстурного слоя. В рамках этой методики предложен новый способ отображения объемных атмосферных эффектов (облака, дым). К достоинствам данного метода следует отнести использование небольшого количества примитивов (например, треугольников) и объемной текстуры. Однако для данного метода существует ограничение на ориентацию объекта.

В тренажерной системе MaxVue канадской компании “CAE Electronics” облака реализованы в виде наборов таких примитивов, как светящиеся точки. В работах [6–9] аналогом светящихся точек являются элементарные частицы, а в [10, 11] – множество дискретных трехмерных выборок объекта, связанных с двумерными спрайтами. К недостаткам данных подходов следует отнести большое количество примитивов для моделирования реалистичного облака.

В работе [12] показано, что для вычисления функции плотности применяется текстура. В этом случае цвет на поверхности объекта берется из текстурной карты, а функция прозрачности вычисляется по этому цвету (или вообще берется из отдельной карты). Например, можно брать плотность пропорциональной или обратно пропорциональной цвету.

Помимо этого описывается метод визуализации атмосферных неоднородностей, которые можно получить в системе Pov-Ray с помощью алгоритма трассировки лучей.

Компанией NGRAIN разработаны алгоритмы, которые, будучи не слишком ресурсоемкими, позволяют достичь высокого уровня реалистичности трехмерных облаков. Эти алгоритмы базируются на методе, называемом сплаттингом (splat – звук шлепка), что означает вычисление формы «отпечатка» вокселя на экране. Этот метод напоминает швыряние снежками в стекло. Сами облака задаются трехмерными массивами вокселей. Объемный объект условно разделяется на воксели – трехмерные точки, и плоский отпечаток каждой из них на экране именуется «сплатом» [13].

Из изложенного выше следует, что при визуализации атмосферных эффектов используются два основных подхода:

- 1) применение объемных текстур с малым количеством примитивов;
- 2) применение объектов сложной формы с большим количеством примитивов и относительно небольших текстурных карт.

В данной работе предлагается использовать в качестве трехмерных облаков полупрозрачные объекты с различной текстурой [14], описанные функциями, задающими сложные формы (рис. 1, а) и неоднородную внутреннюю



Рис. 1. Функционально заданные формы: непрозрачная (a), полупрозрачная (b)



Рис. 2. Объемное облако: со средней плотностью (a) и малой плотностью (b)

структуру (рис. 1, b). На рис. 2, a, b показаны трехмерные облака с меньшей плотностью в сравнении с рис. 1, b.

Визуализация полупрозрачных функционально заданных объектов. В работе [2] предложен способ задания свободных форм на базе квадрик с использованием вещественных функций возмущения в виде неравенства $F(X) \geq 0$, благодаря чему можно визуализировать не только поверхность, но и внутреннюю структуру объекта.

Для того чтобы повысить реалистичность изображения и расширить класс отображаемых объектов (полупрозрачные структуры с внутренним распределением плотности, трехмерные текстуры [14]), необходимо визуализировать внутреннюю полупрозрачную структуру объекта, т. е. сканировать сцену по Z -координате. Используя смешение цветов и моделируя прохождение света через материалы, можно получить эффект полупрозрачности и тем самым добиться большей реалистичности изображения.

Формула, по которой вычисляется цвет пикселя, имеет следующий вид [15]:

$$P_\lambda = \sum_{n=0}^N I_{\lambda n} \Omega_n \prod_{m=0}^{n-1} (1 - \Omega_m), \quad (5)$$

где P_λ – конечный цвет пикселя, а λ может быть r , g или b (т. е. красный, зеленый или синий соответственно); $I_{\lambda n}$ – интенсивность в n -м вокселе, вычисленная по модели освещения Фонга; Ω_n – непрозрачность n -го вокселя; $I_{\lambda 0}$ – отраженный свет от первой точки на луче сканирования; $I_{\lambda N}$ – цвет фона; $\Omega_N = 1$.

Преодоление порога непрозрачности можно отследить следующим образом: если на k -м шаге общая прозрачность $(1 - \Omega_0)(1 - \Omega_1) \dots (1 - \Omega_{k-1})$ становится меньше некоторого ε , то это значит, что вклад всех вокселей, следующих за k -м вокселем, будет малым, поэтому сканирование можно прекратить.

Если применяется перспективное преобразование, то вносится поправка в алгоритм накопления цвета в пикселе, потому что размеры вокселя в результате трансформации геометрических примитивов зависят от координаты Z : с увеличением Z -координаты размер вокселя увеличивается. Поэтому при пересчете цвета в пикселе непрозрачность вокселя следует также пересчитывать с поправкой на изменение его длины.

Если в общем случае сумму (5) заменить интегралом, то интенсивность можно вычислить следующим образом [16]:

$$I = \int_0^D \text{color}(x(s)) \exp \left\{ - \int_0^s \text{opacity}(x(t)) dt \right\} ds, \quad (6)$$

где D – расстояние по Z -координате, на котором происходит вычисление интенсивности; $x(s)$ – точка на отрезке вдоль луча наблюдения; color , opacity – цвет и значение непрозрачности в данной точке соответственно.

Формула (6) более точная по сравнению с формулой (5), так как в ней учитывается протяженность отрезка по Z -координате. Однако необходимо вычислять интеграл на каждом отрезке. Для этого приходится использовать постоянство значения прозрачности на протяжении отрезка. В этом случае формулу (5) для N элементов можно заменить выражением

$$P = 1 - (1 - \Omega)^N, \quad (7)$$

где Ω – непрозрачность в точке; P – суммарная непрозрачность для N элементов. Если рассматривать N не как количество элементарных отрезков на луче, а как его длину, то окончательно будем иметь

$$P = 1 - (1 - \Omega)^{\Delta Z}, \quad (8)$$

где ΔZ – длина отрезка.

Как уже отмечалось, вследствие проективных преобразований геометрических примитивов длина вокселя становится зависимой от Z -координаты. Проведя дробно-линейное преобразование для координат начала и конца отрезка, эту зависимость можно выразить следующим образом:

$$\Delta Z = M_{zz} \left(\frac{Z_{\max}}{M_{tz} Z_{\max} + M_{tt}} - \frac{Z_{\min}}{M_{tz} Z_{\min} + M_{tt}} \right), \quad (9)$$

где M_{zz} , M_{tz} , M_{tt} – элементы матрицы перспективного преобразования, а Z_{\max} , Z_{\min} – начало и конец отрезка в системе координат объекта.

Таким образом, функциональное задание объектов имеет более компактное описание в сравнении с полигональным заданием [2], упрощаются геометрические операции и повышается реализм отображаемых сцен.

Методы отображения трехмерных облаков, представленные в работе, реализованы программно и адаптированы к аппаратной реализации (см. рис. 1, 2).

Заключение. Переход от растривания в плоскости изображения к растриванию в объеме в сочетании с предлагаемыми методами задания объектов повышает реализм отображаемых сцен. Разработан алгоритм визуализации полупрозрачных объектов, ограниченных поверхностями свободных форм. Предложен метод быстрого отображения внутренней неоднородной структуры объекта с применением функции затухания, при этом легко вычисляется интегральное значение прозрачности (β) для отрезков, протяженных по Z -координате. Реализован метод моделирования неоднородности при визуализации атмосферных эффектов с использованием текстуры для вычисления функций цвета и плотности. В этом случае цвет на поверхности объекта берется из текстурной карты, а функция прозрачности вычисляется по этому цвету пропорционально или обратно пропорционально.

К достоинствам предлагаемых методов следует отнести компактность описания баз данных на основе функций и возможность обработки неоднородных полупрозрачных структур, ограниченных поверхностями свободных форм.

СПИСОК ЛИТЕРАТУРЫ

1. **Vyatkin S. I., Dolgovosov B. S., Yesin A. V. et al.** Voxel Volumes volume-oriented visualization system // IEEE Comput. Soc. 1999. P. 234.
2. **Вяткин С. И., Долговесов Б. С., Есин А. В. и др.** Геометрическое моделирование и визуализация функционально заданных объектов // Автометрия. 1999. № 6. С. 84.
3. **Gardner G. Y.** Simulation of natural scenes using textured quadric surfaces // Comput. Graph. 1984. **18**, N 3. P. 11.
4. **Gardner G. Y.** Visual simulations of clouds // Comput. Graph. 1985. **19**, N 3. P. 297.
5. **Schachter B.** Computer Image Generation /Ed. B. J. Schachter. N. Y.: John Willey & Sons, 1980.
6. **Reeves W. T.** Particle systems – a technique for modeling a class of fuzzy objects // Comput. Graph. 1983. **17**, N 3. P. 14.
7. **Miller G., Pearce A.** Globular dynamics: A connected particle system for animating viscous fluids // Proc. conf. Siggraph'89. Boston, Massachusetts. 1989. P. 1.
8. **Szeliski R., Tonnesen D.** Surfase modeling with oriented particle systems // Proc. conf. Siggraph'92. Chicago, Illinois. 1992. P. 185.
9. **Sims K.** Particle animation and rendering using data parallel computation // Proc. conf. Siggraph'90. Dallas, Texas, 1990. P. 405.
10. **Pfister H., Zwicker M., Van Baar J., Gross M.** Surfels: Surface elements as rendering primitives // Proc. conf. Siggraph'2000. New Orleans, Louisiana. 2000. P. 335.
11. **Tobor I., Schlik C., Grisoni L.** Rendering by surfels // Ibid. P. 91.

12. **Сурип А. И.** Трассировка лучей с помощью Pov-Ray // Открытые системы. 1997. № 6. С. 38.
13. **Mueller K., Shareef N., Huang J., Crawfis R.** High-quality splatting on rectilinear grids with efficient culling of occluded Voxels // IEEE Trans. Visual. and Comput. Graphics. 1999. 5, N 2. P. 116.
14. **Вяткин С. И., Долговесов Б. С., Каипов Н. Р.** Отображение текстуры на плоские и криволинейные поверхности, свободные формы и объемы // Автометрия. 2002. № 1. С. 17.
15. **Knittel G.** Voxel Engine for real-time visualization and examination // Proc. conf. Eurographics'93. Barcelona. Spain. 1993. 12, N 3. P. 37.
16. **Max N., Craw R., Becker B.** Application of texture mapping to volume and flow visualization // Proc. conf. Graphicon'95. St-Petersburg: GRAFO, 1995. P. 108.

*Институт автоматизации и электрометрии СО РАН,
E-mail: sivser@mail.ru*

*Поступила в редакцию
28 мая 2004 г.*