

Синтез объемных динамических эффектов для имитационно-тренажерных комплексов

М.В. Михайлюк, М.А. Торгашев

Аннотация. Одной из важных подзадач при создании систем визуализации для имитационно-тренажерных комплексов является синтез объемных динамических эффектов. К таким эффектам относятся, в частности, такие процессы, как огонь и дым. В статье рассматривается технология моделирования объемных эффектов на основе шумовых функций, а также их визуализации в режиме реального времени с помощью шейдерной трассировки на GPU. Рассмотрены также такие аспекты, как описание эффектов в системе 3D моделирования и их интеграция в имитационно-тренажерный комплекс.

Ключевые слова: визуализация, объемные эффекты, трассировка, имитационно-тренажерные комплексы.

Введение

Система визуализации имитационно-тренажерного комплекса должна обеспечивать максимально возможную реалистичность синтеза изображений виртуальных трехмерных сцен, при этом обязательным требованием является соблюдение реального режима времени, что означает визуализацию со скоростью не менее 25 кадров в секунду. Одной из наиболее сложных задач визуализации при построении тренажерных комплексов является синтез объемных динамических эффектов. К их числу относятся такие природные явления и искусственные процессы, как горение, задымление, объемный туман и т.д. До недавнего времени при моделировании подобных эффектов в системах визуализации использовались упрощенные подходы, использующие, например, систему плоских частиц (спрайтов), что позволяло лишь в некоторой степени имитировать внешний вид эффектов. При этом плоская природа эффектов очевидно проявлялась при изменении ракурсов и близком приближении. С бурным

развитием аппаратных средств визуализации, шейдерных технологий и технологии высокопараллельных вычислений на GPU стала возможной и «честная» объемная визуализация подобных эффектов и явлений в режиме реального времени. Сам эффект в этом случае описывается в виде скалярного дискретного поля, которое задается с помощью трехмерной текстуры. В работе предложена новая технология для моделирования объемных динамических эффектов в режиме реального времени и ее реализации в системе визуализации для имитационно-тренажерных комплексов. Рассмотрим две составных подзадачи процесса визуализации объемных динамических эффектов в режиме реального времени – синтеза трехмерного поля и его визуализации на плоскость экрана.

1. Моделирование объемных динамических эффектов

Для синтеза объемных динамических эффектов в настоящее время существуют две основных технологии – физическое моделиро-

¹ Исследования выполнены при поддержке РФФИ (грант № 13-07-00708).

вание и моделирование с использованием шумовых функций. Первый подход состоит в моделировании физической природы эффекта на основе реальных физических уравнений, например, уравнений распространения газа или жидкости. Реализация этого подхода стала возможной лишь в последнее время с помощью высокопараллельной обработки на графических процессорах [1]. Идея реализации заключается в том, что физические параметры среды (скорость, плотность, температура и другие) описываются с помощью дискретных скалярных и векторных полей, которые хранятся в видеопамяти в виде трехмерных текстур, в процессе моделирования выполняется последовательный перерасчет этих полей с решением физических уравнений итерационными методами. Преимуществом технологии является возможность наиболее достоверно смоделировать эффекты с учетом их физической природы. Существенным же недостатком является то, что для получения качественного результата в режиме реального времени требуются высокие вычислительные затраты, что ограничивает детализацию эффектов и их общее допустимое количество в сцене.

Второй распространенной технологией синтеза объемных эффектов является метод шумовых функций. Шумовая функция используется для искажения базового профиля эффекта, синтез эффекта также осуществляется в трехмерном объеме [2]. Преимуществом подхода является достаточно высокая правдоподобность моделирования таких явлений, как огонь и дым, при этом современные аппаратные средства позволяют одновременно синтезировать несколько эффектов в режиме реального времени. Рассмотрим эту технологию более подробно.

Моделирование эффекта выполняется в пределах заданного габаритного контейнера, для хранения и передачи эффекта используется трехмерная текстура, описывающая дискретное скалярное поле. Форма эффекта на макроуровне задается текстурой (текстура профиля эффекта), для создания деталей на микро уровне используются шумовая функция. Текстура профиля эффекта задает двумерный срез эффекта плоскостью, проходящей через вертикальную ось Z в пространстве эффекта. Эта текстура создается

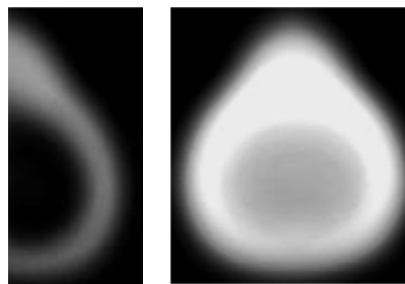


Рис. 1. Текстура профиля эффекта и ее развертка в пространстве

дизайнером и описывает базовую форму эффекта, его спектральный состав и интенсивность. На Рис. 1 приведена иллюстрация текстуры профиля эффекта и результирующий вид эффекта в пространстве, получаемого вращением текстуры профиля вокруг оси Z .

Развертка текстуры профиля выполняется по следующим формулам:

$$\begin{aligned} (u, v) &= (\sqrt{x^2 + y^2}, z) \\ Color &= EffectTexture(u, v) \end{aligned}$$

где (x, y, z) - положение точки в нормализованном объеме эффекта ($(x, y) \in [-1, 1]$, $z \in [0, 1]$), (u, v) - текстурные координаты, которые используются для выборки из текстуры профиля эффекта *EffectTexture*.

На Рис. 2 приведена иллюстрация различных текстур профиля, разработанных для различных эффектов – стандартного огня, огня газовой горелки и дыма.

Для генерации деталей эффекта на микро уровне используется процедурно генерируемая шумовая функция. Идеальным вариантом, ши-

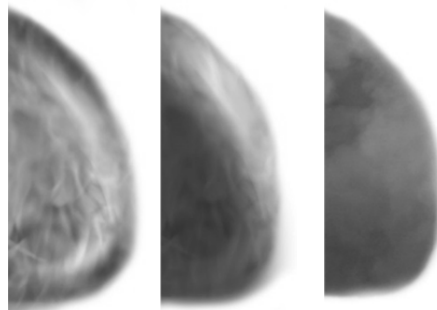


Рис. 2. Несколько вариантов текстур профиля для различных эффектов

роко используемым в компьютерной графике для генерации таких природных явлений, как дым, огонь, облачность и т.д., является классическая шумовая функция Перлина и ее развитие – симплекс шум Перлина [3], которая требует меньшего количества вычислений. Классический шум Перлина и симплекс-шум относятся к классу так называемых градиентных шумов. В ячейках пространства задаются псевдослучайные значения градиента, значение же в произвольной точке вычисляется с помощью интерполяции значений ближайших ячеек полиномиальной функцией, имеющей непрерывную первую и вторую производную на границах интервала $[0,1]$: $f(t) = 6t^5 - 15t^4 + 10t^3$ ($f'(0) = f'(1) = f''(0) = f''(1) = 0$). Для классического шума Перлина используется регулярное разбиение пространства (на квадраты - для 2D случая и кубы - для 3D случая), а для симплекса шума – разбиение на так называемые симплексы с наименьшим количеством вершин (на треугольники - для 2D и тетраэдры - для 3D). Преимущества функции симплекса шума, предложенного Перлином специально для прикладных задач компьютерной графики, состоят в том, что она легко масштабируется от двумерного случая на большую размерность пространства (3D, 4D и далее) и достаточно просто реализуется на современном графическом оборудовании. Иллюстрация внешнего вида симплекса шума размерности 2D и 3D приведена на Рис. 3.

Генерация симплекса шума Перлина может быть эффективно реализована на современных графических процессорах GPU, которые поддерживают так называемый шейдерный язык (GLSL для библиотеки OpenGL и HLSL для DirectX). Существует несколько реализаций для генерации симплекса шума (например [4]), наиболее эффективная из них требует порядка 40 инструкций для генерации одной гармоники шума.

Для моделирования таких явлений, как огонь и дым, используется фрактальная функция [5] как сумма нескольких гармоник шума с различными весами, так называемая функция турбулентности. Эта функция используется для искажения текстурных координат, которые применяются для выборки из базовой текстуры

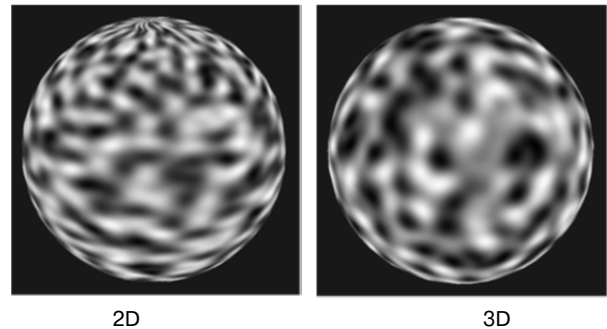


Рис. 3. Внешний вид симплекса шума Перлина

профиля. Формулы для функции турбулентности и ее использования для искажения текстурных координат имеют вид:

$$Turb(Pos_{noise}) = \sum_{i=0}^{N-1} w_i \cdot snoise3D(Pos_{noise} \cdot 2^i)$$

$$Pos_{noise} = (x, y, z - time) \cdot noiseScale$$

$$u = \sqrt{x^2 + y^2}$$

$$v = z + Roughness \cdot Turb(Pos_{noise}) \cdot \sqrt{z}$$

$$Color = EffectTexture(u, v)$$

где: $Turb$ – функция турбулентности, $snoise3D$ – функция симплекса шума размерности 3D, N – используемое количество гармоник, w_i – весовые коэффициенты для разных гармоник шума, $time$ – время, $noiseScale$ и $Roughness$ – пользовательские коэффициенты, регулирующие пространственную частоту эффекта и степень его турбулентности. Практика показывает, что для качественной генерации эффекта пламени оказывается достаточно от 2 до 4 гармоник симплекса шума.

В формулах для искажения вертикальной текстурной координаты вводится время для создания эффекта непрерывно поднимающегося потока. Регулировка пользовательских параметров позволяет настроить желаемый вид эффекта и его динамическое поведение.

Для моделирования процесса активации/деактивации эффекта предлагается выполнять анимацию таких пользовательских параметров, как размеры габаритного контейнера эффекта и степень прозрачности текстуры профиля. Для имитации освещения сцены от источников огня

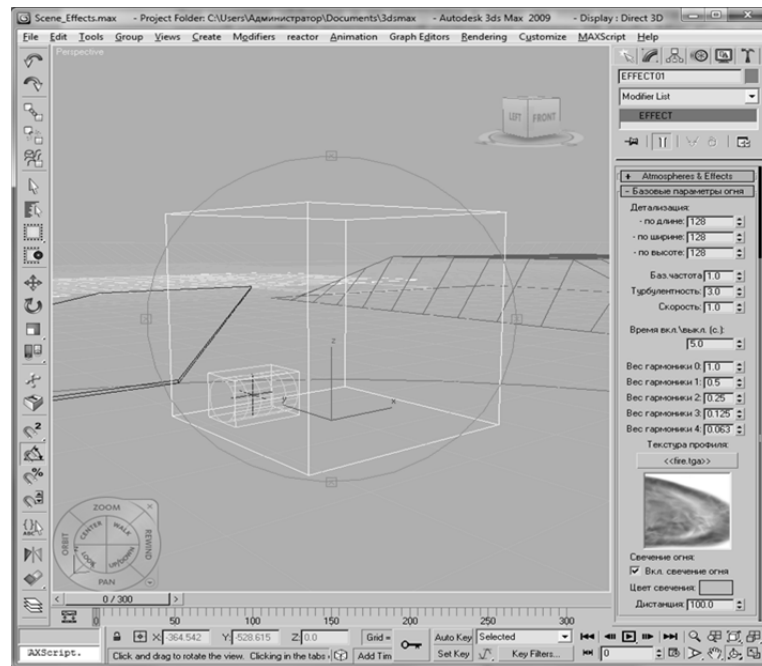


Рис. 4. Создание эффекта в 3D Studio Max

предлагается моделировать дополнительную подсветку сцены. Для этого в сцену процедурно добавляется один или несколько источников освещения, параметры которых меняются в соответствии с текущим состоянием эффекта, а также с добавлением случайно генерируемой вариации. Это позволяет дополнительно повысить реалистичность визуализации за счет имитации мерцающего освещения окружающей обстановки от пламени.

Для создания эффектов и их внедрения в сцену на этапе подготовки базы визуальных моделей разработан подключаемый модуль для системы 3D Studio Max. С его помощью дизайнер может создать габаритный контейнер эффекта, поместить его в сцене и настроить визуальные и динамические параметры. На Рис. 4 представлена иллюстрация пользовательского интерфейса для создания и настройки эффекта в системе 3D Studio Max.

2. Визуализация объемных динамических эффектов

В результате моделирования эффекта мы получаем трехмерную текстуру, для каждого элемента объема содержащую цвет эффекта и его прозрачность. На этапе визуализации полу-

ченная трехмерная текстура должна быть отображена в двумерное пространство экрана. Для выполнения этого шага существует два основных подхода. Первый метод состоит в последовательной визуализации сечений трехмерного объема в виде полигонов с наложенной текстурой соответствующего слоя. Недостатком этого подхода является его высокая вычислительная сложность, а также возникающий визуальный эффект дискретности получаемого изображения. В исследовании был взят за основу второй подход, имеющий название «ray marching». Этот метод заключается в трассировке лучей, исходящих от наблюдателя и проходящих через центр соответствующих пикселей.

Рассмотрим метод более подробно. На визуализацию отправляется габаритный контейнер эффекта (в базовом случае – прямоугольный параллелепипед). Для каждого пикселя, попадающего на экран, выполняется вычисление начальной точки и луча, направленного от наблюдателя в центр текущего пикселя. Далее в цикле выполняется сканирование объема по направлению луча с выбранным шагом и накопление результирующего цвета, до тех пор, пока луч не покинет объем эффекта либо не встретит некое препятствие. Иллюстрация метода приведена на Рис. 5.

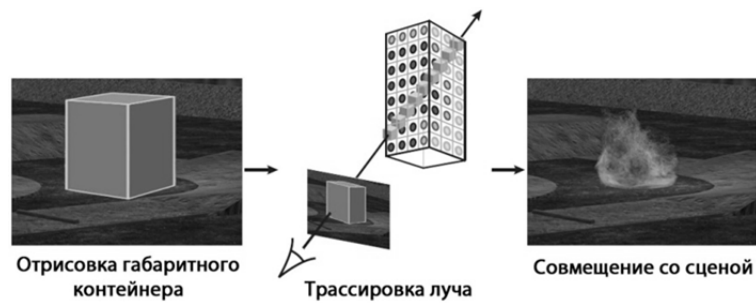


Рис. 5. Метод «ray marching»

Визуальное правдоподобие эффекта и скорость визуализации существенно зависят от величины шага и разрешения трехмерной текстуры эффекта. При больших значениях шага вследствие дискретности проявляется визуальный дефект, который выглядит как «полосатость» изображения, визуально различимые слои. Для устранения этого визуального дефекта применяется случайное смещение начального положения точки, найденной при первом пересечении луча с объемом. Для этого к найденному начальному положению применяется сдвиг по направлению луча на случайную величину в пределах длины одного шага.

Еще одна ситуация, которая может спровоцировать проблемы визуализации - случай, когда луч встречает препятствие до выхода из габаритного контейнера эффекта.

В этой ситуации сканирование объема необходимо завершить, в противном случае результат синтеза будет неправильным, так как будут учитываться невидимые в реальности области пространства. Для обработки подобных ситуаций используется дополнительная текстура, которая содержит глубину (Z координата видового пространства) для всех пикселей изображения. Перед выполнением расчета текущего шага выполняется сравнение текущей глубины Z_c и глубины сцены Z_s . Если $Z_c > Z_s$, выполняется выход из цикла трассировки текущего луча.

Таким образом, мы получаем следующий алгоритм объемной визуализации (Рис.6).

Описанный метод был реализован в подсистеме визуализации [6] с помощью шейдерной

1. Визуализация сцены без эффектов
2. Копирование глубины сцены из целевого буфера в текстуру глубины *DepthTexture*
3. Цикл по эффектам
 - 3.1. Отправка на визуализацию габаритного контейнера эффекта
 - 3.2. Для каждого пикселя в рамках фрагментного шейдера:
 - 3.2.1. Вычисление начальной точки и направления луча в нормализованном пространстве эффекта
 - 3.2.2. Применение случайного сдвига начального положения
 - 3.2.3. Цикл сканирования объема эффекта:
 - 3.2.3.1. Выборка цвета и прозрачности эффекта из 3D текстуры эффекта
 - 3.2.3.2. Выборка глубины сцены из текстуры *DepthScene* и ее преобразование в нормализованное пространство эффекта Z_s
 - 3.2.3.3. Если глубина для текущего фрагмента $Z_c > Z_s$ или текущее положение фрагмента - вне габаритного контейнера эффекта - Выход из цикла
 - 3.2.3.4. Накопление цвета и прозрачности эффекта
 - 3.2.3.5. Перемещение по направлению луча на следующий шаг
 - 3.2.4. Конец цикла
 - 3.2.5. Смешивание вычисленного цвета и прозрачности эффекта с содержимым целевого буфера визуализации

Рис. 6. Алгоритм объемной визуализации

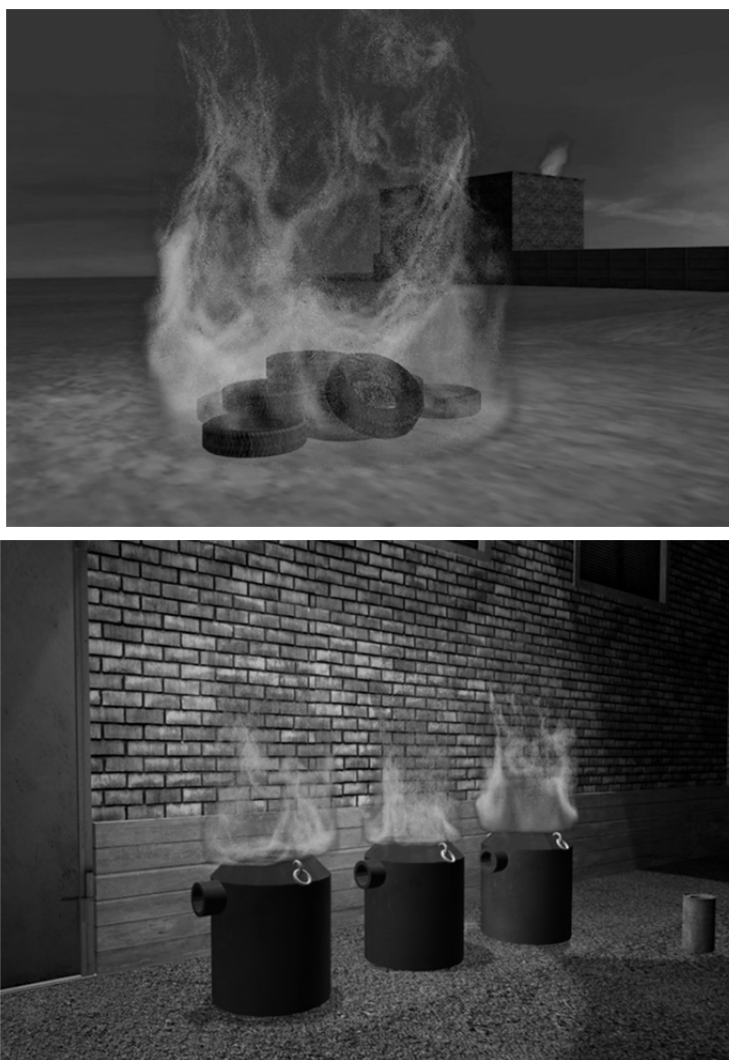


Рис. 7. Иллюстрации синтеза эффектов нескольких типов

программы на языке GLSL. Система позволяет синтезировать несколько эффектов одновременно в режиме реального времени. Иллюстрации результатов визуализации эффектов нескольких типов приведены на Рис. 7.

Заключение

В статье рассмотрена задача моделирования и визуализации объемных динамических эффектов для имитационно-тренажерных комплексов. Предложена технология моделирования объемных эффектов на основе шумовых функций и их визуализации в режиме реального времени на основе шейдерной трассировки. Рассмотренные методы и подходы были интегрированы в про-

граммный комплекс визуализации для имитационно-тренажерных комплексов [6, 7]. Визуальные и динамические характеристики эффектов регулируются с помощью пользовательских параметров, позволяющих настроить желаемый вид эффекта и его динамику. Смоделированы процессы распространения и затухания эффекта, а также дополнительное мерцающее освещение сцены от источников огня. Предложенные методы и алгоритмы основаны на использовании параллельных вычислений, выполняемых на графических процессорах, и обеспечивают одновременный синтез нескольких пространственных эффектов одновременно в режиме реального времени.

Литература

1. Nguyen, D.Q., Fedkiw R. and Jensen H.W. 2002. Physically Based Modeling and Animation of Fire. In Proceedings of SIGGRAPH 2002, ACM Press, – p. 721-728.
2. Alfred R. Fuller, Hari Krishnan, Karim Mahrous, Bernd Hamann, Kenneth I. Joy. Real-time Procedural Volumetric Fire. Proceedings of the 2007 symposium on Interactive 3D graphics and games Pages, – p. 175-180.
3. Perlin K. Noise hardware. In Real-Time Shading SIGGRAPH Course Notes (2001), Olano M., (Ed.).
4. Gustavson S. Simplex noise demystified, March 2005. <http://www.itn.liu.se/~stegu/simplexnoise>.
5. MANDELBROT B.B. 1983. The Fractal Geometry of Nature, revised and enlarged ed. W.H. Freeman and Co., New York, NY, USA.
6. М.В.Михайлюк, М.А.Торгашев. Система «GLVIEW» визуализации для моделирующих комплексов и систем виртуальной реальности. Вестник Российской академии естественных наук, №2, 2011, – стр. 20-28.
7. Mikhaylyuk M.V., Torgashev M.A. E-training system for operators of robots and manipulators. Proceedings of the International Conference «Southeast Asian Open and Distant Learning in the 21st Century», 26-28 October, 2012, Danang, Vietnam, – p. 70-77.

Михайлюк Михаил Васильевич. Зав. отделом Центра визуализации и спутниковых информационных технологий НИИСИ РАН. Окончил факультет вычислительной математики и кибернетики МГУ им. М.В. Ломоносова в 1975 году. Доктор физико-математических наук, профессор. Автор более 100 научных публикаций. Область научных интересов: компьютерная графика, системы виртуальной реальности, информационные технологии. E-mail: mix@niisi.ras.ru

Торгашев Михаил Александрович. Зав. сектором Центра визуализации и спутниковых информационных технологий НИИСИ РАН. Окончил Самарский государственный аэрокосмический университет им. С.П. Королева в 1997 году. Кандидат физико-математических наук. Автор 47 научных публикаций. Область научных интересов: компьютерная графика, системы виртуальной реальности, информационные технологии. E-mail: mtorg@mail.ru