

УДК 004.925.3

С.И. ВЯТКИН

Институт автоматизации и электрометрии СО РАН

А.Н. РОМАНЮК, Л.А. САВИЦКАЯ, Т.И. ТРОЯНОВСКАЯ

Винницкий национальный технический университет

МЕТОД ИЗЛУЧАТЕЛЬНОСТИ С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКИХ УСКОРИТЕЛЕЙ

Предложен метод излучательности, реализованный на параллельной архитектуре графических акселераторов. Предложено значения излучения вычислять непосредственно в текстурах и использовать для рендеринга. Показано, что расчет форм-фактора целесообразно выполнять на SIMD архитектуре GPU, потому что каждый форм-фактор независим и не имеет ветвей и условных переходов. С помощью вершинной программы преобразования вершин определяется видимость со стереографической проекцией. Для этого требуется всего один проход по геометрии. Приведены результаты тестирования.

Ключевые слова: излучательность, параллельные вычисления, форм-фактор, графические акселераторы.

С.І. ВЯТКІН

Институт автоматизації та електрометрії СВ РАН

О.Н. РОМАНЮК, Л.А. САВИЦЬКА, Т.І. ТРОЯНОВСЬКА

Вінницький національний технічний університет

МЕТОД ВИПРОМІНЮВАНОСТІ З ВИКОРИСТАННЯМ ГРАФІЧНИХ ПРИСКОРЮВАЧІВ

Запропоновано метод випромінюваності, реалізований на паралельній архітектурі графічних акселераторів. Запропоновано значення випромінювання обчислювати безпосередньо в структурах і використовувати для рендеринга. Показано, що розрахунок форм-фактора доцільно виконувати на SIMD архітектурі GPU, тому що кожний форм-фактор незалежний і не має розгалужень і умовних переходів. Наведено результати тестування. За допомогою вершинної програми перетворення вершин визначається видимість із стереографічною проекцією. Для цього потрібний всього один проход по геометрії.

Ключові слова: випромінюваність, паралельні обчислення, форм-фактор, графічні акселератори.

S.I. VYATKIN

Institute of Automation and Electrometry SB RAS

O.N. ROMANYUK, L.A. SAVYTSKA, T.I. TROIANOVSKA

Vinnytsia National Technical University

METHOD OF EMISSIVITY BASED ON GRAPHICS ACCELERATORS

When calculating diffuse illumination of surfaces, it is necessary to take into account not only direct light sources, but also secondary or reflected diffuse illumination. Diffuse reflection matrix is a classical technology for modeling light propagation in diffuse scenes. In the classical formulation, radiosity requires the solution of a set of linear equations called the equation form factor. These equations describe the transfer of energy between elements in a scene. Storing these equations requires space $O(N^2)$, which is very much for large scenes. In the classical approach, the surfaces of all objects in a three-dimensional scene are split into flat patches. The size of each patch should be such that the density of the light energy intensity distribution within its limits is a constant value. The assumption is that the light is scattered uniformly in all directions. This method is called the restricted elements method. Under its own surface energy refers to "own the issue" patch and under bring energy illumination, indirect diffuse all surrounding objects. Lambert's law calculates the calculation of energy from direct lighting. Assumptions are needed to account for the secondary diffuse reflection of the environment. The basic radiative equations calculate the radiosity value of this patch as the sum of the contributions of the radiosity value of all patches, that is, on the contrary, light is not collected, but emitted from this patch in the direction of the surrounding surfaces. This paper presents the method of emissivity based on graphics accelerators. The method of progressive refinement of estimates adapted to graphics processing units is proposed. There are several advantages to performing all calculations on graphics accelerators (GPUs). One of them is that the radiation values can be calculated directly in textures and used directly for rendering. Form factor calculation, which is a fundamental operation and is well parallelized. The form factor calculation is performed well on the SIMD architecture of the GPU fragment processor because each form factor is independent and has no branches and conditional transitions. Another advantage is that the performance of the graphics hardware is greater than the CPU; this allows increase the complexity of the model and keep the interactive rendering mode.

Keywords: radiosity, parallel computing, form-factor, graphic accelerators.

Постановка проблеми

На современном этапе развития компьютерной графики актуальным становится вопрос о повышении реалистичности формирования графических сцен [1–3], поскольку мощности видеокарт достаточно для поддержания сложных вычислений за приемлемое время.

Для повышения реалистичности часто используют метод излучательности (radiosity), согласно которого рассчитывают вторичные отражения лучей света от одних поверхностей объектов к другим, а также от окружающей среды к объектам.

Каждый объект рассматривается как источник излучения для освещения всех других объектов. Метод использует уравнения, заимствованные из области теплопередачи. Он требует значительно большей вычислительной мощности, чем альтернативный метод трассировки лучей.

Анализ последних исследований и публикаций

При расчете диффузного освещения [1] поверхностей необходимо учитывать не только освещение прямыми источниками света, но и вторичное, или отраженное диффузное освещение [4]. Матрица диффузного отражения – это классическая технология для моделирования распространения света в диффузных сценах [5].

В данной работе предложен метод излучательности на базе графических ускорителей. Существует несколько преимуществ для выполнения всех вычислений на графических акселераторах (GPU). Одно из них заключается в том, что значения излучения могут быть вычислены непосредственно в текстурах и использоваться непосредственно для рендеринга.

Расчет форм-фактора хорошо выполняется на SIMD архитектуре процессора фрагмента GPU, потому что каждый форм-фактор независим и не имеет ветвей и условных переходов.

Другое преимущество состоит в том, что производительность графического оборудования больше, чем центрального процессора (CPU). Это позволяет увеличить сложность модели и сохранить интерактивный режим рендеринга.

В классической формулировке, излучательность (radiosity) требует решения набора линейных уравнений, называемых форм-фактором уравнений. Эти уравнения описывают передачу энергии между элементами в сцене. Хранение этих уравнений требуют пространства $O(N^2)$, что очень много для больших сцен. Существует множество методов для решения этих уравнений.

Метод [4], вычисляет эти форм-факторы во время трассировки, и таким образом, избегает любого явного хранения. Исчерпывающее объяснение диффузного отражения можно найти в работе [3], где с помощью метода "hemі-cube" вычисляется видимость.

Расчет форм-факторов, который является фундаментальной операцией и хорошо распараллеливается описан в [7–9].

В классическом подходе поверхности всех объектов трехмерной сцены разбиваются на плоские патчи (patch) [3]. Размер каждого патча должен быть таким, чтобы плотность распределения интенсивности световой энергии в его пределах была постоянной величиной (константой).

Вводится допущение, что свет рассеивается равномерно по всем направлениям. Такой метод получил название метода ограниченных элементов. Под собственной энергией поверхности понимается "собственная эмиссия" патча, а под приносимой энергией – освещение, отраженное диффузно всеми окружающими объектами. Вычисление энергии от прямого освещения рассчитывается по закону Ламберта [1]. Чтобы учесть вторичное диффузное отражение окружения, необходимы допущения. Основное уравнение расчета излучательности имеет следующий вид:

$$B_i = E_i + \rho_i \sum_{j=1}^n B_j F_{ij},$$

где B_i – излучательность i -ого патча; E_i – диффузное отражение света от прямого источника; ρ_i – коэффициент, характеризующий свойства диффузного отражения поверхности, которой принадлежит патч; B_j – излучательность j -ого патча; F_{ij} – форм-фактор, определяющий часть излучательности j -ого патча, дошедшей до i -ого патча; n – количество всех патчей трехмерной сцены.

Основные уравнения излучательности рассчитывают излучательность данного патча как сумму вкладов излучательности всех патчей, то есть наоборот – не собирается свет, а излучается свет из данного патча в направлении окружающих поверхностей:

$$\Delta B_i = \frac{1}{A_i} \cdot \rho_i B_l F_{li} A_l,$$

где ΔB_i – вклад в излучательность i -ого патча от первого патча; A_l – площадь первого патча; A_i – площадь остальных патчей; ρ_i – коэффициент, характеризующий свойства диффузного отражения поверхности, которой принадлежит i -ый патч; B_l – излучательность l -ого патча; F_{ij} – форм-факторы первого патча в количестве n ; n – количество всех патчей трехмерной сцены.

Алгоритм состоит из следующих шагов:

1. Для всех патчей излучательности определяются освещенностью источниками света по закону Ламберта и собственным излучением. Далее производится упорядочивание патчей в порядке убывания.
2. Находится наиболее освещенный патч, и вычисляются вклады первого патча в излучательности всех остальных патчей. Вычисленные вклады суммируются с излучательностью патчей. Вклад первого патча в излучательность второго патча прибавляется к излучательности второго патча и суммируется со значением переменной, не излученной излучательности второго патча.
3. Далее выполняются итерации, пока значение максимального вклада патча не станет меньше заданной величины.

Цель исследования

Предложить модификацию метода излучательности, которая позволяет увеличить производительность формирования графических сцен и обеспечить интерактивный режим рендеринга.

Изложение основного материала исследования

В данной работе реализован модифицированный метод прогрессивной излучательности на графическом оборудовании.

Излучательность и остаточная энергия хранятся в виде текстур, отображаемых на полигонах сцены.

Вместо деления в геометрии, как и в классической излучательности, используются текстелы как элементы системы излучения, как описано в [10]. Для этого инвертируются вычисления с помощью итерации для входящих полигонов и тестируется каждый элемент на видимость. Такой подход эффективно использует вычислительные мощности параллельной архитектуры SIMD графических акселераторов.

SIMD-принцип компьютерных вычислений, позволяющий обеспечить параллелизм на уровне данных. SIMD-архитектура предполагает наличие одного командного процессора, и нескольких модулей обработки данных, называемых процессорными элементами. Управляющий модуль принимает, анализирует и выполняет команды. Если в команде встречаются данные, контроллер рассылает на все процессорные элементы команду, и эта команда выполняется на нескольких или на всех процессорных элементах.

Для определения видимости используется расширенная форма метода [10]. С помощью вершинной программы преобразования вершин, определяется видимость со стереографической проекцией. Для этого требуется только один проход по геометрии, по мере необходимости, а не пять как hemi-cube [10].

Стереографическая карта сцены отрисовывается и хранится в виде буфера элементов текстуры. Далее каждый полигон, который мог бы получить энергию, рендерится в буфер кадров с тем же разрешением, что и связанная с ним излучательная текстура. Это позволяет фрагменту программы работать непосредственно с диффузным отражением текстелей с фрагментами буфера кадра.

Использованная стереографическая проекция состоит из векторной нормализации для поиска в буфере элементов. Если идентификатор элемента буфера совпадает с идентификатором этого текстеля полигонов, то этот текстель объявляется видимым. Этот процесс напоминает проекционное отображение теней. Используются запросы на видимость для сокращения числа полученных полигонов, которые должны быть протестированы. В течение теста на видимость запрос инициируется для каждого полигона. Только для тех полигонов, которые проходят тест на видимость, делается выборка из текстуры.

Формула вычисления форм-фактора выглядит следующим образом [11]:

$$F_{ij} = \frac{1}{A_i} \cdot \int_{A_i} \int_{A_j} \frac{\cos \theta_i \cdot \cos \theta_j}{\pi r^2} HID_{ij} dA_j dA_i$$

где F_{ij} – форм-фактор i -ого патча; A_i – площадь i -ого патча; A_j – площадь j -ого патча; HID_{ij} – функция видимости патчей; r – расстояние между патчами; θ_i, θ_j – углы между нормальными патчей и отрезком, который их соединяет.

Для вычисления интеграла используется численный метод и упрощение [12, 13]:

$$dF_{A_j \rightarrow dA_i} = \frac{dA_i}{m} \cdot \sum_{i=1}^m \frac{\cos \theta_j \cos \theta_i}{\pi r_i^2 + \frac{A_j}{m}}$$

Это приближение вычисляет форм-фактор между конечной площадью элемента j и дифференциальной площадью элемента d_i , разделив на m ориентированных дисков.

Уравнения форм-фактора хорошо обрабатываются на архитектуре SIMD, таких как процессор фрагментов GPU, потому что нет зависимостей данных и ветвления.

На рис. 1 приведен пример реализации предложенного метода излучательности.



Рис. 1. Реализация теней методом излучательности

Тестирование производилось на компьютере с процессором Intel Core2 CPU E8400 3.0 GHz, и графическим акселератором GTX 470. Тестирование показало, что время формирования графических сцен достаточно для реализации интерактивного режима. Так, для сцен, состоящих из миллиона текстелей, на проецирование потребовалась 1 миллисекунда, на вычисление форм-фактора – 2 миллисекунды.

Выводы

При использовании метода излучательности можно находить диффузную освещенность, тени, переносить и смешивать отраженный свет с одних поверхностей на другие. При этом вычисление освещенности не зависит от положения наблюдателя.

Учитывая большую вычислительную сложность метода излучательности и возможность распараллеливания вычислительного процесса, предложено использовать SIMD архитектуру графического процессора.

Предложено значения излучения вычислять непосредственно в текстурах и использовать для рендеринга. Показано, что расчет форм-фактора целесообразно выполнять на SIMD архитектуре GPU, потому что каждый форм-фактор независим и не имеет ветвей и условных переходов. С помощью вершинной программы преобразования вершин определяется видимость со стереографической проекцией. Для этого требуется всего один проход по геометрии

Реализация метода GPU позволила увеличить производительность формирования графических сцен и обеспечить интерактивный режим рендеринга.

Список использованной литературы

1. Романюк О. Н. Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів. Монографія / О.Н. Романюк, А.В. Чорний. – Вінниця : УНІВЕСУМ-Вінниця, 2006. – 190 с.
2. Романюк О.Н. Класифікація графічних відеоадаптерів / О.Н. Романюк, Р.Ю. Довгалюк, С.В. Олійник // Наукові праці Донецького національного технічного університету. Сер.: Інформатика, кібернетика та обчислювальна техніка. – 2011. – Вип. 14. – С. 211-215.

3. Маслов В.П. Сучасні тенденції розвитку тримірного рендерингу / В.П. Маслов, П.М. Кузьменко, Є.В. Риндич // Вісник Чернігівського державного технологічного університету. Серія: Технічні науки. – 2012. – № 3 (59). – С. 155-158.
4. Cohen M.F. Progressive Refinement Approach to Fast Radiosity Image Generation / M.F. Cohen , S.E. Chen., J.R. Wallace., D.P. Greenberg // In Proc. of SIGGRAPH'88, 1988. – P. 45-84.
5. Goral C.M. Modelling the Interaction of Light between Diffuse Surfaces / C.M. Goral, K.E. Torrance, D.P. Greenberg // In Proc. of SIGGRAPH'84, 1984. P. 213-222.
6. Cohen M.F. The Hemi-Cube: A Radiosity Solution for Complex Environments / M.F. Cohen, D. P. Greenberg // In Proc. of SIGGRAPH'85, 1985. – Vol. 19. – № 3. – P. 31-40,
7. Baum D.R. Real Time Radiosity through Parallel Processing and Hardware Acceleration / D.R. Baum, J.M. Winget // In Proc. of Symposium on Interactive 3D Graphics, 1990. – P. 67-75.
8. Varshney A. An Environment Projection Approach to Radiosity for Mesh-Connected Computers / A. Varshney, J.F. Prins // In Eurographics Workshop on Rendering, 1992. P. 271-281.
9. Funkhouser T.A. Coarse-Grained Parallelism for Hierarchical Radiosity using Group Iterative Methods / T.A. Funkhouser // In Proc. of SIGGRAPH'96, 1996. – P. 343-352.
10. Heckbert P.S. Adaptive Radiosity Textures for Bidirectional Ray Tracing / P.S. Heckbert // In Proc. of SIGGRAPH'90, 1990. – P. 145-154.
11. Chapter 5: Radiosity P. [Електронний ресурс]. – Режим доступу: <https://webcourse.cs.technion.ac.il/236373/Winter2013-2014/ho/WCFiles/05-Radiosity.pdf>
12. John R Wallace. A ray tracing algorithm for progressive radiosity / Wallace John R, Elmquist Kells A., Haines A. Eric // ACM SIGGRAPH Computer Graphics. 1989. – Vol. 23. – № 3. – P. 315-324.
13. Cohen Michael F. Radiosity and Realistic Image Synthesis / Michael F. Cohen and R. Wallace John. – Cambridge: Academic Press, 1993. – 383 p.