Модификация воксельного метода для построения единой триангуляционной оболочки объекта по множеству видов

В.А.Бобков, А.П.Кудряшов
Институт автоматики и процессов управления ДВО РАН
Владивосток, Россия
{bobkov, kudryashova}@dvo.ru

Аннотация

Описан алгоритм построения единой триангуляционной трехмерной текстурированной модели сцены по множественному набору пространственных данных, основанный на воксельном подходе. Предложена оптимизированная структура исходных данных, обеспечивающая повышение производительности алгоритма. Реализована вычислительная схема с организацией параллельных вычислений на графических процессорах средствами языка CUDA.
Ключевые слова:*воксельное пространство, трехмерная реконструкция, триангуляционная оболочка, текстурирование, марширующие кубики, CUDA, GPU.*

1. ВВЕДЕНИЕ

Один из подходов к созданию трехмерных компьютерных моделей реальных объектов и сцен основывается на использовании дальностных данных, получаемых с помощью лазерных дальномеров или методами компьютерного зрения по фотоизображениям. В литературе на такие данные ссылаются как на дальностные изображения (rangeimages) и карты глубин (depthmaps). В отличие от обычных изображений с каждым пикселем связывается расстояние до видимой точки объекта в пространстве сцены. Эти пространственные данные могут быть представлены в виде облака точек в трехмерном пространстве или в виде триангуляционной поверхности. Технологии получения дальностных данных и генерации цифровых пространственных моделей на практике применяются в робототехнике, геоинформатике, производстве, медицине и в др. приложениях. Нетривиальной задачей при построении трехмерных моделей по последовательности дальностных изображений, представленных в виде отдельных триангуляционных сеток (оболочек), является задача объединения этих триангуляционных сеток. Ее решение подразумевает построение единой связной оболочки, отвечающей исходным данным, без дублирования участков поверхности, видимых с нескольких видов. Важным требованием к алгоритмическому решению задачи является высокая скорость обработки данных, поскольку для насыщенных сцен со сложными объектами может потребоваться много видов с большим количеством описывающих треугольников (порядка 10^4-10^6).

Очевидный путь к решению указанной задачи состоит в сопряжении исходных сеток с устранением избыточного триангуляционного описания посредством совместного анализа данных в трехмерном пространстве сцены и в двумерном пространствах изображений видов. Подобная схема решения была реализована в [4]. Однако она, как отмечают авторы, полностью не исключает дублирования. Более интересным представляется воксельный подход, основанный на использовании воксельной структуры пространства сцены. Эффективная реализация такого подхода была предложена в работе [1]. Основным преимуществом предложенного решения является сведение исходной задачи к известной задаче построения изоповерхности в скалярном поле. Значительным недостатком данного метода можно его считать высокую ресурсоемкость, из-за чего реконструкция даже небольшой сцены по нескольким десяткам изображениям занимает много часов [2]. Однако, в последнее время, с появлением технологий использующих для вычислений графический процессор (GPU), таких как CUDA (*ComputeUnifiedDeviceArchitecture*) от компании NVIDIA, стало возможным увеличить скорость расчёта ресурсоемких алгоритмов в десятки и сотни раз. В работе [5] можно ознакомиться и с некоторым систематизированным обзором других методов реконструкции поверхности по плотному набору дальностных данных.

Представленный в настоящей статье метод построения единой сеточной модели основывается на подходе, предложенном в [1]. Работа выполнялась в контексте решения авторами более общей задачи по созданию трехмерных моделей на множественных последовательностях изображений с построением карт глубин. Вклад авторов состоит в предлагаемой оптимизации вычислений за счет эффективной структурной организации данных и реализации параллельных вычислений на графических процессорах, что в целом обеспечивает более высокую скорость обработки данных.

2. ОПИСАНИЕ МЕТОДА

Задача объединения поверхностей формулируется следующим образом. Имеются n видов, для каждого из которых на предварительном этапе была построена видимая поверхность в виде пространственной триангуляционной сетки. Триангуляционная сетка строится по плоской триангуляции на регулярной пиксельной решетке изображения с использованием карты глубин, которая вычисляется для точек, видимых на данном виде. Задача заключается в построении единой текстурированной триангуляционной оболочки, которая и будет являться трехмерной моделью сцены.

Предлагаемый метод работает в воксельном пространстве сцены. Используется непрерывная неявная функция $D(\mathbf{V}_i)$, представленная значениями в узлах воксельной решетки. Функция конструируется как взвешенная сумма получаемых для n видов расстояний $d_1(\mathbf{V}_i)$, $d_2(\mathbf{V}_i)$,… $d_n(\mathbf{V}_i)$ от точки $\mathbf{V}_i$ до ближайшей поверхности (рис. 1). Расстояние d_j берется на луче, направленном из центра проекций O_i в точку $\mathbf{V}_i$, и является величиной со знаком (положительное для точек, находящихся перед поверхностью, и отрицательное для точек за поверхностью).

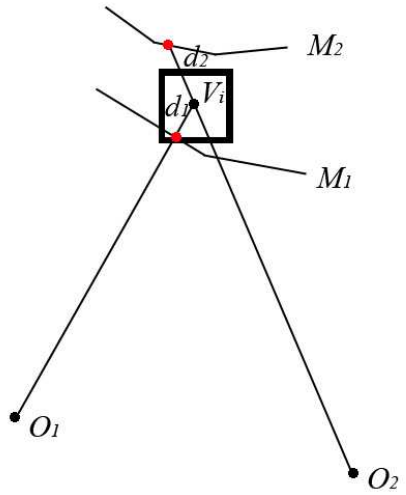

Рис. 1 Построение скалярной функции.

В качестве весовой функции W($\mathbf{V}_i$) берется скалярное произведение вектора нормали к поверхности и вектора направления на точку наблюдения. Такой выбор исходит из предположения, что при корреляционном подходе к определению расстояния до поверхности степень неопределенности для наклонных поверхностей выше, чем для поверхностей, наблюдаемых под прямым углом.

Весовая функция W($\mathbf{V}_i$) – константа для всех вокселов до видимой поверхности (для данного вида), а для вокселов за поверхностью линейно убывает до нуля в пределах ε-окрестности. Такой выбор области определения весовой функции направлен на предотвращение возникновения ложных поверхностей.

В результате последовательной обработки всех видов каждому вокселю присваивается значение интегральной функции расстояния D($\mathbf{V}_i$) (величина со знаком). Тогда построение изоповерхности D($\mathbf{V}_i$) =0 и является решением поставленной задачи. Построение изоповерхности скалярного поля, заданного на воксельной решетке, выполняется с помощью известного алгоритма марширующих кубиков [3].

3. МОДИФИКАЦИЯ МЕТОДА

Предполагается, что для каждого из видов определены параметры камеры и построены триангуляционные сетки. Вначале создается воксельное пространство сцены, размерность которого задается пользователем. Эта размерность определяет детализацию результирующей трехмерной сцены.

Наиболее ресурсоемкой частью представленного выше метода является определение расстояний от вокселов до ближайшей поверхности (рис.1.) при формировании скалярного поля. Вычислительная трудоемкость возникает из-за необходимости перебора всех треугольников с вычислением соответствующих расстояний. Чтобы избежать избыточных вычислений предлагается механизм с предварительной индексацией всех треугольников сетки каждого вида (рис. 2). Для этого используется целочисленный массив с размерностью равной размеру исходного изображения.

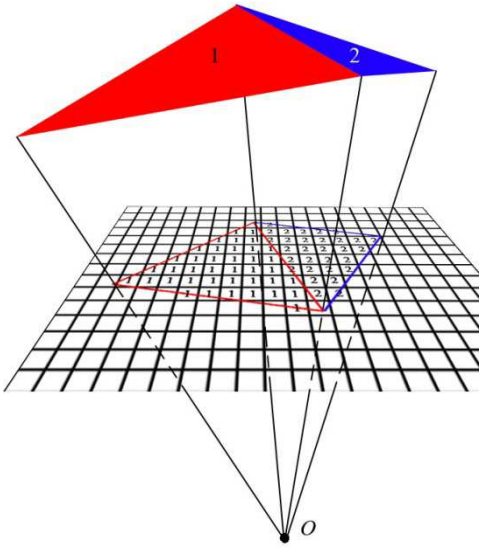

Рис. 2. Индексация треугольников.

Для каждого i-го вида строится проекционная матрица по формуле:

$$H_i = \begin{pmatrix} c_{11}^i f & c_{12}^i f & c_{13}^i \\ c_{21}^i f & c_{22}^i f & c_{23}^i \\ c_{31}^i f & c_{32}^i f & c_{33}^i \\ c_{41}^i f & c_{42}^i f & c_{43}^i \end{pmatrix}, \qquad (1)$$

где c_{kl}^i - элементы матрицы камеры $i-$го вида, а $f-$фокус.

Тогда можно спроецировать каждый треугольник исходной сетки i-го вида на i-е изображение по формуле:

$$P_{tex}^i = H_i \times P, \qquad (2)$$

где P - координаты точки в трехмерном пространстве.

В результате каждый элемент массива индекса приобретает индекс, указывающий на принадлежность к определенному треугольнику. Такая процедура возможна, поскольку исходная сетка для каждого вида является заведомо «видимой», т.е. не содержит невидимых на этом виде треугольников. Все непроиндексированные пиксели приравниваются к одному значению. Использование индексов при последующей работе алгоритма позволяет избежать полного перебора по всем треугольникам каждого вида и существенно уменьшает время работы алгоритма.

При реализации метода был разработан дополнительный алгоритм с использованием альфа-канала, который необходим для отделения реконструируемого объекта от фона. Для каждого пикселя проиндексированного изображения проверяется его значение в альфа-канале исходного изображения. Если это значение равно нулю (т.е. пиксель является фоном), то этому пикселю присваивается то же значение, что и для непроиндексированных пикселей, что исключает построение поверхностей, относящихся к фону.

На последнем этапе, после формирования скалярного поля с помощью алгоритма марширующих кубиков выполняется построение изоповерхности D($\mathbf{V}_i$) = 0 , которая и является решением задачи построения общей триангуляционной сетки по множеству видов.

4. РЕАЛИЗАЦИЯ НА CUDA

Поскольку метод работает с большими объемами данных, была реализована схема центральный процессор + графические процессоры, где наиболее ресурсоемкие вычисления выполняются на графической плате. Самая ресурсозатратная часть метода – это проход по всем вокселям сцены с вычислением значений скалярной функции D(**V**). Поскольку вычисления этих значений для каждого вокселя являются независимыми, алгоритм хорошо распараллеливается с помощью технологии CUDA. Алгоритм марширующих кубиков также был оптимизирован для работы с помощью GPU. Основной сложностью в реализации этого метода является неизвестное количество получаемых треугольников. Некоторые реализации этого метода, например в OpenCV, предлагают использовать количество памяти, заведомо большее, чем необходимо для хранения полученных треугольников. В предлагаемом методе используется двукратный запуск алгоритма, первый из которых определяет количество треугольников и заводит под них память, а второй запуск уже заполняет ее. Данный подход позволяет избегать перезаполнения видеопамяти при обработки сцен больших размерностей.

5. ТЕКСТУРИРОВАНИЕ

Текстурирование полученной модели производится с использованием всех текстур исходных видов. Каждый треугольник полученной трехмерной модели проецируется на каждоеизображение исходных видов и принимается в качестве текстуры, если соответствует следующим критериям: видимость (треугольник виден на изображении), наблюдаемость (треугольник не заслоняется другими треугольниками модели), имеет максимальную площадь проекции.

6. РЕЗУЛЬТАТЫ

Для получения сравнительных оценок эффективности предложенного метода были проведены вычислительные эксперименты на модельных сценах 1 и 2 (рис. 3, 5) и реальной сцене 3 «Temple» (рис.7) (http://vision.middlebury.edu/mview/) [5]. Параметры используемого вычислительного оборудования: процессор IntelCoreDuo 2,6 Ггц, графическая плата NVidiaGeForce 9600GT. Во-первых, сравнивались производительности двух алгоритмических реализаций на центральном процессоре – предложенной в настоящей работе и реализации, аналогичной описанной в [1]. Во-вторых, оценивалось ускорение вычислений предложенного алгоритма за счет реализации параллельных вычислений на графической плате. Результаты вычислительных экспериментов приведены в табл.1 и на рис.4, 6, 8. Как видно из таблицы время выполнения практически не зависит от сложности сцены (количество треугольников), что подтверждает эффективность реализованного механизма индексации. Предложенный механизм индексации при реализации алгоритма на центральном процессоре ускоряет вычисления в 12-20 раз. Использование графических процессоров дает дополнительное ускорение в 7-15 раз. Таким образом, суммарное ускорение оценивается в 200-300 раз.

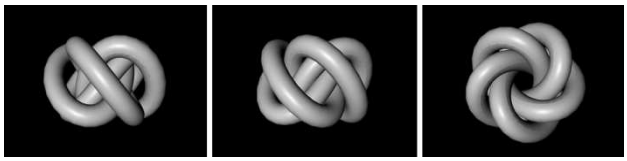

Рис. 3. Исходные виды (3 из 6) сцены 1.

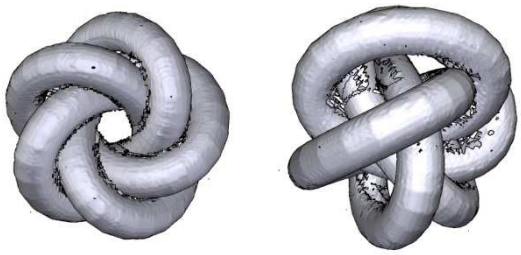

Рис. 4. Реконструкция сцены 1 по шести видам

Рис. 5. Исходные виды сцены 2 (2 из 11).

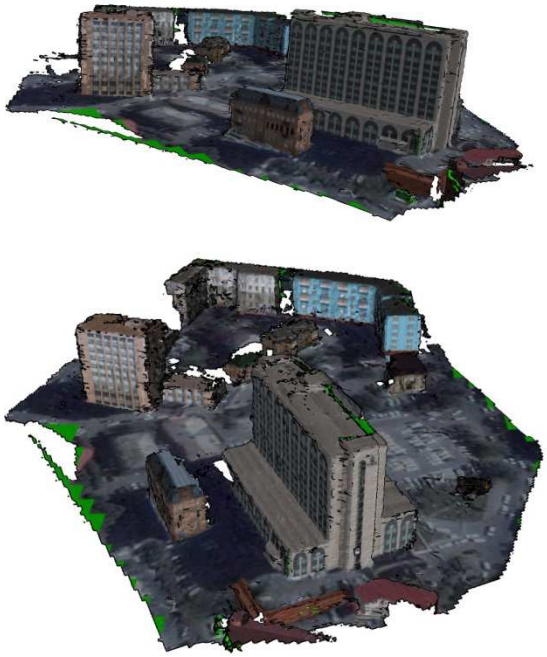

Рис. 6. Реконструкция сцены 2 по одиннадцати видам.

Рис. 7. Исходные виды сцены 3 «Temple».

Рис. 8. Реконструкция сцены 3 по трем (слева) и пяти (справа) видам.

	Размер-тьвокс. простр-ва	Количеств видов		
		2	6	60
Модельная сцена 1 (8т.треуг. на вид)	128^3	66; 4.3; 0.9	145; 9.3; 1.5	641; 42; 3.4
	256^3	413; 21.2; 3.4	1034; 49; 6.2	4082; 186; 15.4
Модельная сцена 2 (22т.треуг. на вид)	128^3	132; 8.5; 1.1	194; 12.7; 2.4	894; 59; 4.2
	256^3	619; 31.2; 3.9	1438; 61.5; 6.7	5865; 225; 18.7
Реальная сцена 3 «Temple» (16т.треуг. на вид)	128^3	114; 7.5; 0.8	161; 10.7; 1.4	764; 52; 3.3
	256^3	589; 28.4; 3.4	1308; 55.5; 5.9	5143; 205; 13.8

Таблица 1. Время построения трехмерной модели без текстурирования (сек.). Первое значение – время работы аналога алгоритма [1], второе значение – время работы предложенного алгоритма с индексацией треугольников на CPU, третье значение – время работы алгоритма с использованием CUDA.

7. ЗАКЛЮЧЕНИЕ

В данной работе представлен алгоритм построения единой триагуляционной трехмерной текстурированной модели сцены по множественному набору пространственных данных, основанный на воксельном подходе. Предложенная структура данных с использованием индексации треугольников позволила существенно повысить производительность алгоритма в сравнении с аналогами. Дополнительное ускорение получено за счет организации параллельных вычислений на графических процессорах средствами языка CUDA.

8. СПИСОК ЛИТЕРАТУРЫ

[1] Brian Curless and Marc Levoy.A Volumetric Method for Building Complex Models from Range Images // Computer Graphics (SIGGRAPH '96 Proceedings). 1996. – P. 303-312.

[2] M. Goesele, B. Curless, S. M. Seitz. Multi-View Stereo Revisited // Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, Vol. 2. 2006. - P. 2402-2409.

[3] W.E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm // In Computer Graphics (SIGGRAPH '87 Proceedings),volume 21. 1987. - P. 163–169.

[4] P. Mordohai, J.-M. Frahm, A. Akbarzadeh, B. Clipp, C. Engels, D. Gallup, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, H. Towles, G. Welch, R. Yang, M. Pollefeys, and D. Nister. Real-time video-based reconstruction of urban environments, - ISPRS Working Group V4 Workshop 3D-ARCH 2007: 3D Virtual Reconstruction and Visualization of Complex Architectures, (ETH Zurich, Switzerland), 2007. – P. 121-131.

[5] Steve Seitz, Brian Curless, James Diebel, Daniel Scharstein, Richard Szeliski. A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms /Vision and Pattern Recognition. // IEEE Computer Society Conference on, vol. 1. 2006. – P. 519-526.

Об авторах

Бобков Валерий Александрович - д.т.н, зав. лабораторией машинной графики Института автоматики и процессов управления ДВО РАН.
Адрес: г. Владивосток, 690090, ул. Радио,5, ИАПУ.
Телефон: (4232)313776
E-mail: bobkov@iacp.dvo.ru

Кудряшов Алексей Павлович – к.т.н, м.н.с. лаборатории машинной графики Института автоматики ипроцессов управления ДВО РАН.
Адрес: Владивосток, 690090, ул. Радио, 5, ИАПУ
E-mail: kudryashova@iacp.dvo.ru